

AD-A115 695

AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/8 12/2

APPLICATION OF NETWORK AND DECISION THEORY TO ROUTING PROBLEMS.(U)

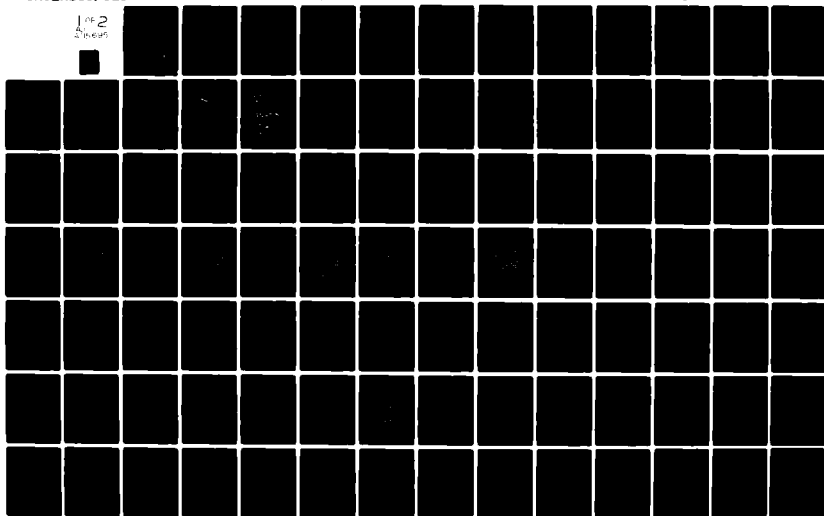
MAR 82 R J BAKER, J E CARTER

AFIT/OST/OS/82H-3

UNCLASSIFIED

NL

1 of 2
2 hours



AD A115695

AFIT/GST/OS/82M-3



APPLICATION OF NETWORK AND
DECISION THEORY TO ROUTING PROBLEMS

THESIS

Richard J. Baker, Captain, USAF
James E. Carter, Major, USAF

AFIT/GST/OS/82M-3

DTIC
ELECTE
JUN 18 1982
S D D

Approved for Public release; distribution unlimited

DTIC COPY

82 06 16 068

APPLICATION OF NETWORK AND DECISION THEORY
TO ROUTING PROBLEMS

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

Richard J. Baker, A.A., B.S., M.B.A.
Captain, USAF

James E. Carter, B.S., M.S.S.M.
Major, USAF

Graduate Strategic and Tactical Science

March 1982

Approved for public release; distribution unlimited

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	



Acknowledgements

The authors thank their thesis committee (Major Gerry Armstrong, Captain Aaron DeWispelare, and Mr. Dan Reynolds). Their expertise in the areas the research effort encompassed and their guidance, motivating words, and patience made this effort possible. A special thanks to Major Hal Carter, faculty member, for his help in getting the authors to understand one of the underlying algorithms in the methodology. Certainly not least, the authors thank Phyllis Reynolds for her expertise and hard work that made this thesis a professional product. Future readers of this thesis will surely continue to thank her.

Major Jim Carter's wife, Joan, and children, Jena, Joshua, and Jaci; and Captain Rick Baker's lady friend, Joyce Botka, were instrumental in completing this thesis. Their support, understanding, and love during many trying times will always be appreciated.

Contents

	Page
Acknowledgements	ii
List of Figures	v
List of Tables	viii
Abstract	ix
Glossary	x
I. Introduction	1
Background	1
General Situation	2
The Deployment Area and Construction Constraints	4
Design Studies	7
The Problem	9
Objectives	10
Scope	12
Overview	12
II. Methodology	14
Example Problem	15
Assumptions and Limitations	19
Finding Path Connections	21
Finding the Least Cost Network	40
Cost and Mean Time Optimization	47
Computation of the Cost Vector	52
Finding Shortest Time Paths	53
Methodology Results	56
III. Computer Implementation	62
Finding Path Connections	63
Finding the Minimum Steiner Tree	73

	Page
Construction of Desired Cost Vectors	81
Finding Shortest Time Paths	88
IV Use of Results	92
V. Conclusion and Recommendations	100
Recommendations for Use	102
Recommendations for Further Research	103
Bibliography	107
Vitae.	109

List of Figures

Figure		Page
1.	General Situation: Designated Transportation Network	5
2.	Proposed Designated Transportation Network (DTN)	6
3.	Example MX Layout with MX Terminology	16
4.	Example MX Layout with Network Terminology	17
5.	Grid Superimposed on MX Layout	23
6.	Expanding Source Cell: First Adjacent Cell	25
7.	Expanding Source Cell: All Adjacent Cells	25
8.	Expanding Right Frontier Cell	27
9.	Expanded Right Frontier Cell	27
10.	Cell Expansion Completed (Cost Run 1): All Targets are Frontier Cells	28
11.	Finding the Path Back	29
12.	Path From All Targets to Source (Cost Run 1)	31
13.	Least Cost Paths Between Target 1 and Targets 2, 3, and 4 (Cost Run 2)	32
14.	Least Cost Paths Between Target 2 and Targets 3 and 4 (Cost Run 3)	33
15.	Least Cost Path Between Target 3 and Target 4 (Cost Run 4 [N-1])	35
16.	Least Cost Paths Between Source and Target Nodes	36
17.	Minimum Time to Travel Paths from Source to All Targets (Time Run)	38

Figure		Page
18.	Example MX Layout with Least Cost Paths . . .	41
19.	Steiner Point Example	44
20.	Cost Vectors	50
21.	Efficient Frontier	51
22.	Non-Dominated Solution Set--Four Target Problem	57
23.	Option 1--Maxcost Steiner Tree	59
24.	Option 2--Intermediate Steiner Tree	60
25.	Option 3--Mincost Steiner Tree	61
26.	General Methodology Flowchart	64
27.	Least Cost/Time Path Algorithm Flowchart . .	65
28.	Possible Redundant Arc of Time Run	70
29.	Redundant Arc Deleted	70
30.	Illustration of Improvement by Diagonal Path	72
31.	A 5 Node/9 Steiner Point Sample Network . . .	75
32.	Minimum Spanning Tree of All Nodes and Steiner Points	76
33.	Minimum Spanning Tree with Steiner Points 6 and 7 Removed	78
34.	Minimum Spanning Tree with Steiner Points 6 and 9 Removed	79
35.	Minimum Spanning Tree with Steiner Points 6 and 10 Removed	80
36.	3 Node/1 Steiner Point Network and Arc List	83
37.	Binary Search for Desired Cost 6	84

Figure		Page
38.	Binary Search for Reciprocal Desired Cost 2	86
39.	Binary Search for Reciprocal Desired Cost 2 with Target Leaf Excluded	87
40.	Minimum Mean Time for Desired Cost Subset Flowchart	89
41.	Non-Dominated Solution Set (NDSS)	94
42.	Value Functions of Time and Cost	96
43.	Indifference Curves	97
44.	Reiterated Non-Dominated Solution Set with Indifference Curves	99

List of Tables

Table		Page
1.	Example MX Layout Cost/Time Data	18
2.	Limitations on Arc Subset Search	20
3.	Time and Cost Information of the Minimum Mean Time Network	39
4.	Cost Path Data	42
5.	Cost Vectors Optimized for Mean Travel Time	57
6.	Preference by Marginal Return	94

Abstract

This thesis presents a methodology for finding the routing of paths in a network. The methodology establishes the least construction cost paths and minimum time to travel paths between known points. Since the minimum time to travel network may not be the least construction cost network, Multiple Objective Optimization Theory is used to provide a decision maker with a solution set of networks that have been optimized for their least construction cost and minimum travel time. The methodology may be applied to any class of problems where conflicting objectives exist in determining a network routing.

Glossary

<u>AFRCE</u>	Air Force Regional Corps of Engineers
<u>ALV</u>	Assembly and Checkout Launcher Vehicle
<u>ASC</u>	Area Support Center
<u>BMO</u>	Ballistic Missile Office
<u>COE</u>	Corps of Engineers
<u>CMF</u>	Cluster Maintenance Facility
<u>CMV</u>	Canisterized Missile Vehicle
<u>DAA</u>	Designated Assembly Area
<u>DDA</u>	Designated Deployment Area
<u>DTN</u>	Designated Transportation Network
<u>HSS</u>	Horizontal Shelter Site
<u>LCC</u>	Life Cycle Costs
<u>MCDT</u>	Multiple Criteria Decision Theory
<u>MOOT</u>	Multiple Objective Optimization Theory
<u>MST</u>	Minimum Steiner Tree
<u>NDSS</u>	Non-Dominated Solution Set
<u>OB</u>	Operational Base
<u>SAC</u>	Strategic Air Command
<u>SAL</u>	Strategic Arms Limitations

APPLICATION OF NETWORK AND DECISION THEORY TO ROUTING PROBLEMS

I. Introduction

This thesis presents a methodology to find a combination of least cost to build path and minimum time to travel path connections between a source point and a number of end points. Once the methodology finds the combinations of the least cost and minimum time paths, these values are evaluated in their best combinations with each other so that a decision maker can select the paths he or she wishes to construct between the points.

This thesis will present the problem which started the research effort, the objectives, scope, and overview of the thesis effort, the methodology, a description of the computer programs to implement and verify the methodology, the use of the results for decision making, and the recommendations for further study as a result of this effort.

Background

The MX missile basing plan created much controversy. Of no small concern was the massive construction effort needed to bed down and service the missile field. This construction included 4,600 missile shelters,

"clustered" 23 for each of 200 missiles, support facilities, an Operational Base (OB), and a dedicated road system called the Designated Transportation Network (DTN).

The DTN design stressed minimum road length to minimize construction cost (Ref 5). A review of these designs by the operational planners at Headquarters Strategic Air Command (SAC) (Ref 9) raised doubts as to the benefits of a minimum construction cost DTN design when compared to the cost of MX operations on that DTN. In answer to these doubts, this thesis reviews the planned DTN use and DTN design effort and formulates the methodology to find the best DTN construction around the issues of acquisition of the DTN and operations on the DTN.

General Situation. The MX basing plan, as of 6 May 1980, placed the Designated Deployment Area (DDA) in the great basin of Nevada-Utah, where 20-35 valleys deemed "geotechnically feasible" would share the deployment of 200 clusters of Horizontal Shelter Sites (HSSs). Each cluster was designed to accept 23 HSSs for initial deployment.

Within each cluster, a Cluster Maintenance Facility (CMF) served as both a missile maintenance and checkout station and a storage point for the missile during Strategic Arms Limitation (SAL) verification (Ref 4:18). Each CMF was paired to a parent Area Support Center (ASC)

responsible for organizational maintenance and security of all assigned CMFs (Ref 4:21). Finally, the entire DDA would have been managed from an Operational Base (OB) which housed the Designated Assembly Area (DAA) as a point of origin for all missiles transported to the DAA. The dedicated DTN linked all these facilities to provide a transportation surface strong enough to support the special purpose vehicles used to move the missile overland and to insure a high degree of enroute security.

Initial use of the DTN would have been to transport an MX missile to each cluster by an Assembly and Checkout Launcher Vehicle (ALV). The ALV would have departed the OB in convoy with security vehicles and airborne security teams in HH-53 helicopters. The ALV would have traveled at an average speed of 25 miles per hour (Ref 6). This slow, methodical convoy movement was further restricted to daylight hours as a security precaution. The ALV could only travel on the DTN because of its limited maneuverability and extreme weight. Once the MX missile arrived at the CMF, it and a dedicated transporter vehicle would be sealed into the cluster by an earthen barrier across the single DTN entry point, this preventing movement of that missile to another cluster for SAL compliance (Ref 10).

After initial emplacement operations, a missile removal and replacement for maintenance would have been accomplished with a third vehicle, a Canisterized Missile

Vehicle (CMV). The CMV would travel on the DTN between the Operational Base and the Cluster Maintenance Facility. Missile removal and replacement operations would probably occur at a rate of 40 missiles per year (Ref 14). Figure 1 illustrates the general situation.

The Deployment Area and Construction Constraints.

The nomenclature *network* is a true descriptor of the DTN design characteristics. The DAA served as a source of missiles to the entire deployment area. The DTN was a road network connecting the 200 CMFs to the DAA. Actual routing was greatly constrained, however, by environmental, political, geographical, and design limitations.

The Nevada-Utah great basin contains two major contrasting geographical phenomena. These are broad, flat valleys: (1) oriented north and south, and (2) located between relatively sheer mountain ranges. Travel from valley to valley is restricted to a limited number, approximately 100, of accessible mountain passes (Ref 4:12). Figure 2 depicts the proposed layout of the DTN to connect clusters in the Nevada-Utah Designated Deployment Area. The Martin Marietta Corporation DTN specification document (Ref 6) outlined guidance and constraints on the DTN routing to include:

1. A twenty-year service life (the life of the MX system).

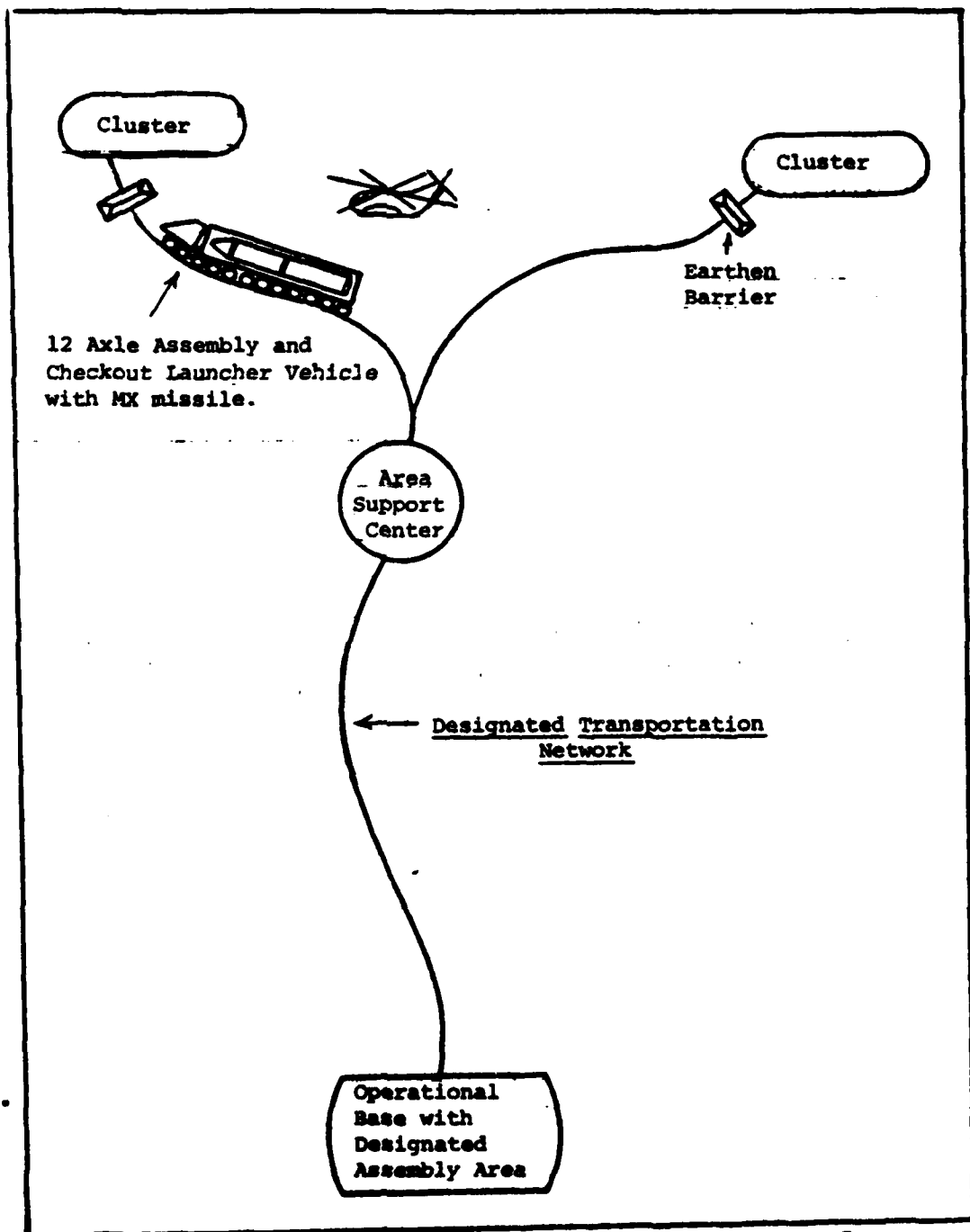


Fig. 1. General Situation: Designated Transportation Network

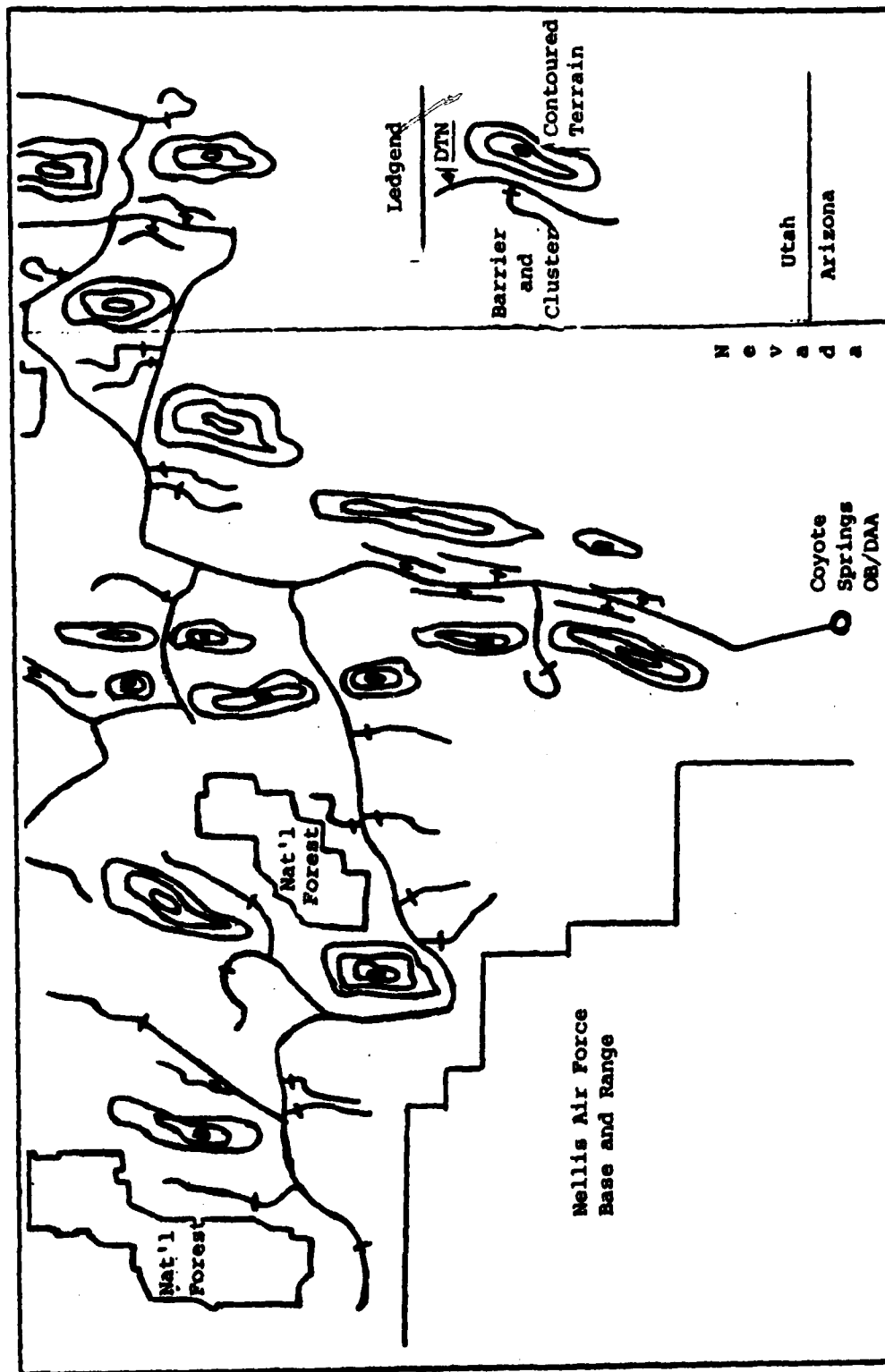


Fig. 2. Proposed Designated Transportation Network (DTN)

2. Compliance with federal, state, and local environmental quality regulations. This included avoidance of known archaeological sites and other environmentally sensitive areas.

3. Configuration to preclude special use vehicles (ALV, CMV, and Transporter) access to public roads.

4. A minimum horizontal curve radius of 500 feet.

5. A maximum road grade of 7 percent.

6. A non-special use vehicle design speed of 55 miles per hour.

7. Avoidance of co-location, joint use, with state or federal highways. Where joint use is unavoidable, as through narrow mountain passes, existing highways would have been improved to meet DTN specifications.

8. The DTN would not cross cluster roads except at designated single access points. The DTN would be terminated at this point by an earthen barrier to prevent movement of the transporter into or out of the cluster.

Design Studies. Four primary contractors, ERTEC, Inc.; TRW, Inc.; Henningson, Durham, and Richardson Sciences (HDR); and Martin Marietta Corporation were involved with the MX OB/ASC/DTN siting problems. These contractors worked under the control of the Air Force Regional Corps of Engineers (AFRCE), MX Siting Group, Norton AFB, California. ERTEC, as the geotechnical contractor,

solicited specialized information from the other contractors. A simplified information flow would start at ERTEC. Using field investigations and map studies of the area, ERTEC established suitable valleys for cluster deployment and identified mountain passes that satisfied DTN specifications. Clusters were then fitted into each valley, connected by the shortest possible route, generally following the path of least construction resistance to minimize the cost. HDR provided ERTEC with topographical maps depicting all known environmentally excluded or sensitive areas. Martin Marietta accomplished cost-benefit analyses of alternative routing, where deemed necessary by environmental conflicts or by a desire for more than an intuitive choice for the shortest route.

TRW ran the DTN revision through a conceptual cost model to insure DTN changes did not exceed a baseline budget figure. The total process was an iterative process that attacked the DTN problem a section at a time to resolve predictable construction, environmental, cultural, and political conflicts on a case-by-case basis. Research into each contractor's activities verified only Martin Marietta was using any optimization technique to design the DTN to consider construction cost and operations cost. However, at best, DTN designs and analyses were manual, "best judgement" models (Ref 14).

The Problem

The problem facing the planner of the Designated Transportation Network is to find the road network that provides the lowest total cost of acquisition and "cost" of operations. The DTN cost is minimized by the least expensive connecting routes between the operational base and the clusters because of the DTN investment cost and the size of the deployment area. The "cost" of operations may be lowered by building some additional routes in the minimum construction cost DTN. These operations "costs" should consider the cost to operate the large vehicles to move the MX missile and the possible adverse effect on security of the missile and on personnel safety and morale by extremely long and slow driving conditions. Because the assignment of specific operations "costs" to these latter operational concerns are difficult to judge, the SAC operations planner believes that the amount of time spent on the road between the OB and each cluster should be at a minimum for missile transportation and day-to-day traffic (Ref 10).

Consequently, the goal of this thesis effort was to find a methodology that would provide the DTN planner the road networks that provide the lowest total cost of acquisition and the minimum amount of travel time between the OB and clusters for operation on the DTN.

Objectives

The first objective of the thesis effort is to determine possible roads or paths, in terms of least construction cost and minimum time to travel. This was necessary because:

1. The DTN designers did not have a procedure to do so.
2. The various network algorithms to optimize paths for least cost or time routing require knowing these possible least cost or time paths. It was possible for the designer to estimate the cost to build or time to travel on a road in specific land areas, such as desert valleys or mountain passes.

The second objective was to find the road network that would enable the time of operations from the operational base to each cluster to be at a minimum. This network will consist of the paths found in the first objective for their minimum travel time. The time attribute was assumed measurable by the minimum mean time to travel from the operational base to the clusters. In other words, choosing a road network that was a minimization over the average time to travel from the OB to each cluster, would satisfy the operations planner. This road network would be the more expensive to build because of the addition of roads to effect the more direct routing from the OB to each cluster.

The third objective was to find the least acquisition cost road network, regardless of operational considerations. This least cost or minimum cost to build road network would represent the minimum cost the decision maker would have to spend in building the road network.

The fourth objective was to provide road networks that would satisfy both objectives two and three, minimum mean time and minimum cost to build, in varying degrees. This is because the minimum mean time to travel network would be costly compared to the minimum cost to build network. The minimum cost to build network may not be the network to minimize travel time to each cluster from the OB. The situation causes conflicting objectives. However, by finding various road networks that optimize the two objectives in varying degrees or combinations, the planner or decision maker is presented with a set of solutions (of road networks) that can be selected based on the decision maker's preference for least construction cost and minimum mean time to travel between the OB and clusters.

The methodology sequentially meets these four objectives. After meeting the objectives, the planner or decision maker has a set of networks, optimized in combination for the two planning objectives, from which the decision maker can select the best network to construct based on feelings for the importance of the attributes of least

construction cost and minimum travel time between the operational base and clusters.

Scope

The scope of this thesis is limited to providing a methodology to satisfy the objectives of the thesis. Because of the limitation of gathering MX DTN construction cost and time to travel data, the recent change of MX basing plans, and the cost of computer time, this thesis provides only the methodology to solve the problem; it does not find the set of networks applicable to the Nevada-Utah MX Designated Transportation Network routing.

Overview

The remaining chapters will explain the methodology, the computer implementation of the methodology, the use of the results from the methodology for the decision maker, and the conclusions and recommendations resulting from the research effort.

The next two chapters present the methodology in the order the objectives are met. This will allow the reader to relate the algorithms and methods of the methodology to the objectives of the research effort. Since the methodology finds a solution set of networks found for their combinations of least construction cost and minimum mean travel time for the decision maker's use, a brief description of the solution set is contained in the

methodology chapter before proceeding to find the solution set.

Following the methodology and computer implementation chapters, the use of results chapter describes how the decision maker can use the results of the methodology to decide on the preferred network to construct. The last chapter ends the thesis by recommending various applications for the methodology and various research efforts that could be accomplished based on this effort.

II. Methodology

The thesis methodology uses four techniques linked together to find different sets of paths found for their best combination of least construction cost and minimum mean time to travel. Lee's path connection algorithm starts the methodology by finding a set of best path connections for least cost and then for minimum time to travel between a source point (the Operational Base) and each end point (the clusters). Prim's algorithm uses the least cost paths and finds the minimum cost to connect all the points. A sum of subsets algorithm and the Bellman-Ford shortest path method then find the subsets of the set of paths from Lee's algorithm that represent the minimum mean time paths between the source point and the end points for desired cost levels. Using the minimum cost and minimum mean time information, the decision maker can choose the paths to construct.

Because the algorithms use general network theory, the remainder of this thesis will use mostly general network terminology instead of the MX basing terminology.

The Designated Transportation Network will consist of paths or arcs between nodes. The Designated Assembly Area at the Operational Base will be the *source* node.

The cluster sites will be termed *target nodes* or *end points*. Intersections in the road network will be called *steiner points*. The example problem in the next section will illustrate the conversion of the MX DTN problem into the general network terminology. The remainder of this chapter will use the example problem to illustrate the methodology.

The assumptions and limitations placed on the problem and the methodology will be discussed after the example problem. The remainder of the chapter will detail the algorithms and methods used to provide a solution set of networks.

Example Problem

Figure 3 shows a hypothetical MX Designated Deployment Area with valleys of clusters separated by mountains. The figure also details environmental areas to avoid and passes that may be used. The operational base is located in the lower center of the figure. The object is to build the least construction cost and minimum travel time road network to link the OB to each of the clusters. Table 1 lists the construction cost and average time to travel through the areas in Figures 3 and 4. Since the costs and times are scaled to one (1) unit in the predominantly flat land area, Table 1 also shows the scaling of the cost and time units of the other areas to the cost of the flat land area. Figure 4 converts Figure 3 to

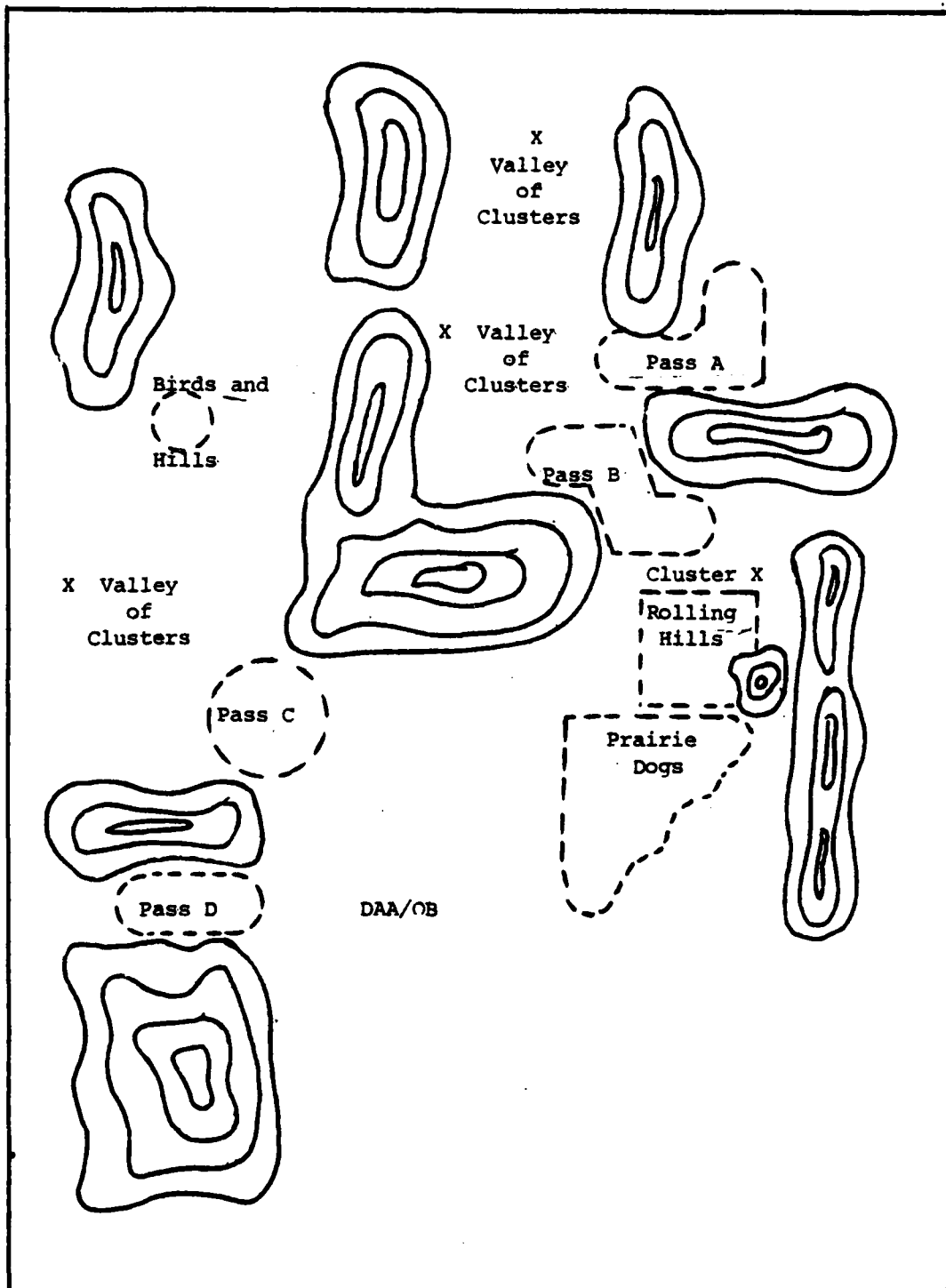


Fig. 3. Example MX Layout with MX Terminology

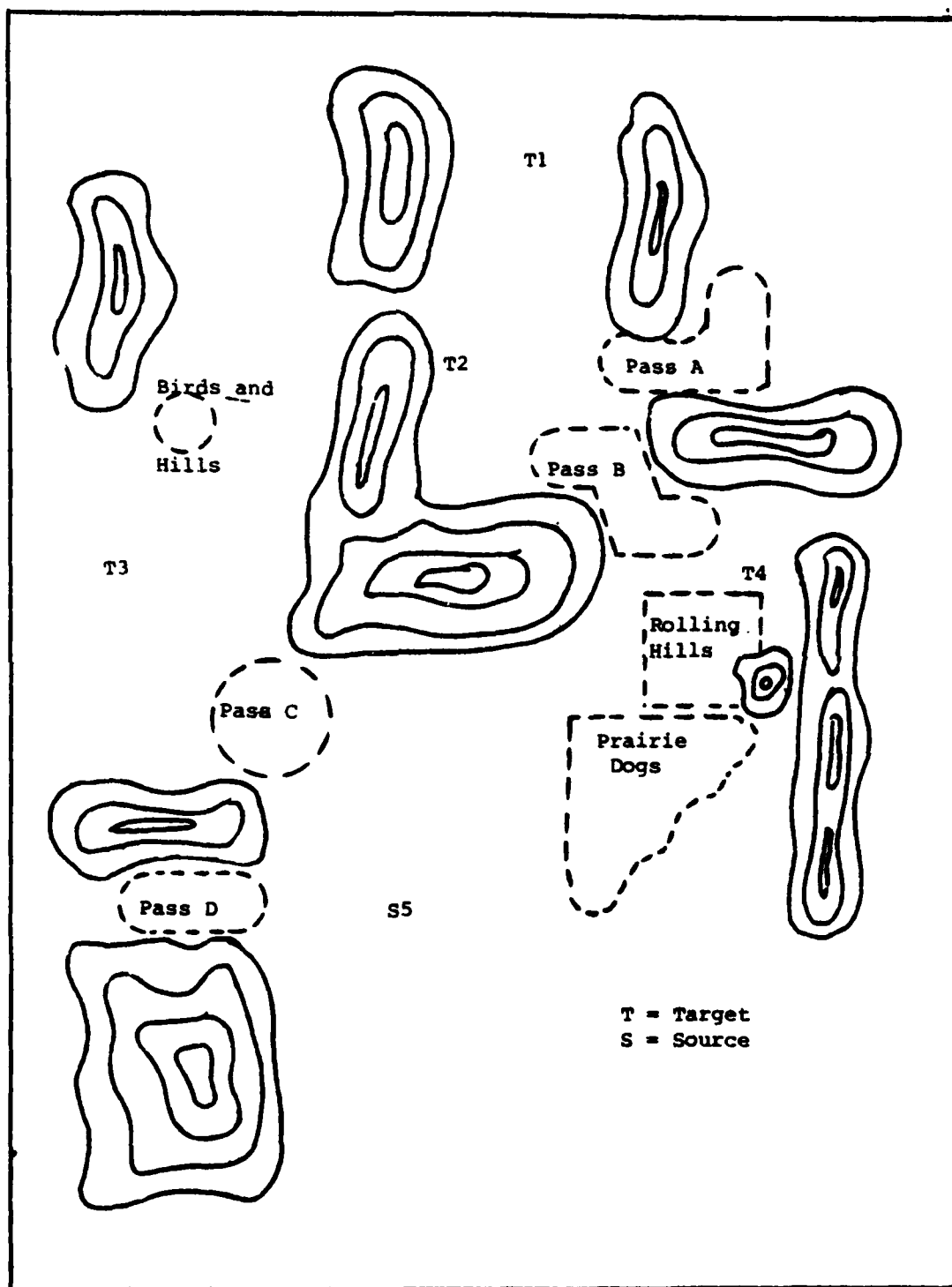


Fig. 4. Example MX Layout with Network Terminology

TABLE 1

EXAMPLE MX LAYOUT COST/TIME DATA

(1)	(2)	(3)	(4)	(5)
Flat Lands	.45	1	3	1
Pass A	.9	2	9	3
Pass B	.9	2	9	3
Pass C	.9	2	9	3
Pass D	.9	2	6	2
Rolling Hills	1.35	3	12	4
Birds and Hills	.9	2	9	3
Prairie Dogs	1.8	4	3	1
(1) -- Type Land Area				
(2) -- Average Mitigation and Construction Cost/Cell (Millions of \$)				
(3) -- Scaled Cost/Cell Integer Multiple of Flat Land Cost				
(4) -- Average Time to Travel/Cell (Minutes)				
(5) -- Scaled Time/Cell Integer Multiple of Flat Land Time				

network terminology. Each major valley containing one or more clusters is termed a target node. Target nodes are located in the center of the valleys. The OB becomes the source node. In both Figures 3 and 4, the mountains are obstacles that the road cannot be built through or on because of the design constraints of the DTN. Environmental areas are not obstacles in this example because procedures at a cost to the builder can be taken to alleviate the environmental impact of building the DTN through the environmental areas. These procedures, mitigation procedures, range from re-locating prairie dog towns to establishing museums for archaeological finds. This particular designated employment area will be used to illustrate the

methodology to find the set of networks to connect the source to the targets.

Assumptions and Limitations

The following assumptions are made on the general problem:

1. Areas requiring paths between them can be modeled as point sources or targets. This is so the algorithms of the methodology can operate on points or nodes. For example, design of some cluster sites may prevent DTN passage completely through a valley. The cluster designer should be able to lay out the cluster in the valley to permit the best DTN routing.
2. Large homogeneous areas containing more than one end point can be represented as one target. This limits the number of arcs that must be considered in the methodology to keep computer time within reasonable bounds. For example, valleys containing more than one cluster can be modeled as a one-target node in the valley. This assumption greatly reduces the number of paths that must be analyzed.
3. Problems containing a large number of targets can be partitioned into smaller problems. Again, this assumption limits the number of arcs to be considered in the methodology. Table 2 lists computer run times based on different numbers of arcs in the problem and illustrates

TABLE 2
LIMITATIONS ON ARC SUBSET SEARCH

No. of Targets	No. of Steiner Points	No. of Arcs	Computer Time (Seconds)
4	6	13	.96
5	7	16	13.32
5	7	17	32.59
6	8	19	94.73
6	8	21	~403.20

the reason for this and the previous assumption. For example, since the Area Support Centers were to support an equal number of cluster sites, the problem can be partitioned as follows:

- a. the routing of the road from the DAA to each ASC; and
- b. the routing of the road from each ASC to its cluster sites.

4. The mean time to travel between the source and targets is a good attribute to measure operational cost considerations. This measure has been discussed previously. Finding the best routing from the operations point of view needs to consider not only cost to operate, but the problems associated with response time to all sites and the effect of long, slow driving on the security of the missile in transportation and the safety and morale of the people

using the roads to transport the missile. It is difficult to place a cost on these other considerations.

5. Construction cost is the dominant consideration for building the network. It is reasonable to believe the planner primarily concerned with operational efficiency is also driven by construction cost limitations imposed by decision makers and budgets. This assumption supports approaching the problem by finding the minimum mean time to travel for different network costs rather than finding the cost for different mean time to travel networks.

The following limitation is a result of the methodology: The methodology finds rectilinear distances (non-diagonal) between nodes. However, the analyst has the opportunity to calculate any desired measure of distance before proceeding on with the methodology.

Finding Path Connections

The first objective of the thesis effort was to find the paths between the source and end points in terms of least cost to build and minimum time to travel. An infinite number of paths could be identified. However, only those identified by Lee's algorithm are of interest.

Lee's path connection algorithm finds the best path (least cost or minimum time) from one node to another using a numerical search technique to find the path of least resistance between the nodes. The path is the best

rectilinear or non-diagonal path (sometimes referred to as manhattan distance in network literature). The path is the best because of the successive minimizations of the numerical search technique the algorithm uses (Ref 13:346-353).

The algorithm has found its use primarily to find the best routing of wires on a printed circuit board (Ref 1:312-322). The printed circuit board problem usually considers only two points that must have a wire routed between them. On the printed circuit board, there are obstacles such as other points and wires which the routing of the new wire must avoid. The electrical engineering community uses Lee's algorithm to find the routing for the minimum amount of wire or similar conductive material. Because network algorithms and the algorithms used in the remainder of the methodology require knowing the possible paths between nodes, Lee's algorithm is used to find these paths. This use of Lee's algorithm makes the solution to the problem possible and starts the methodology.

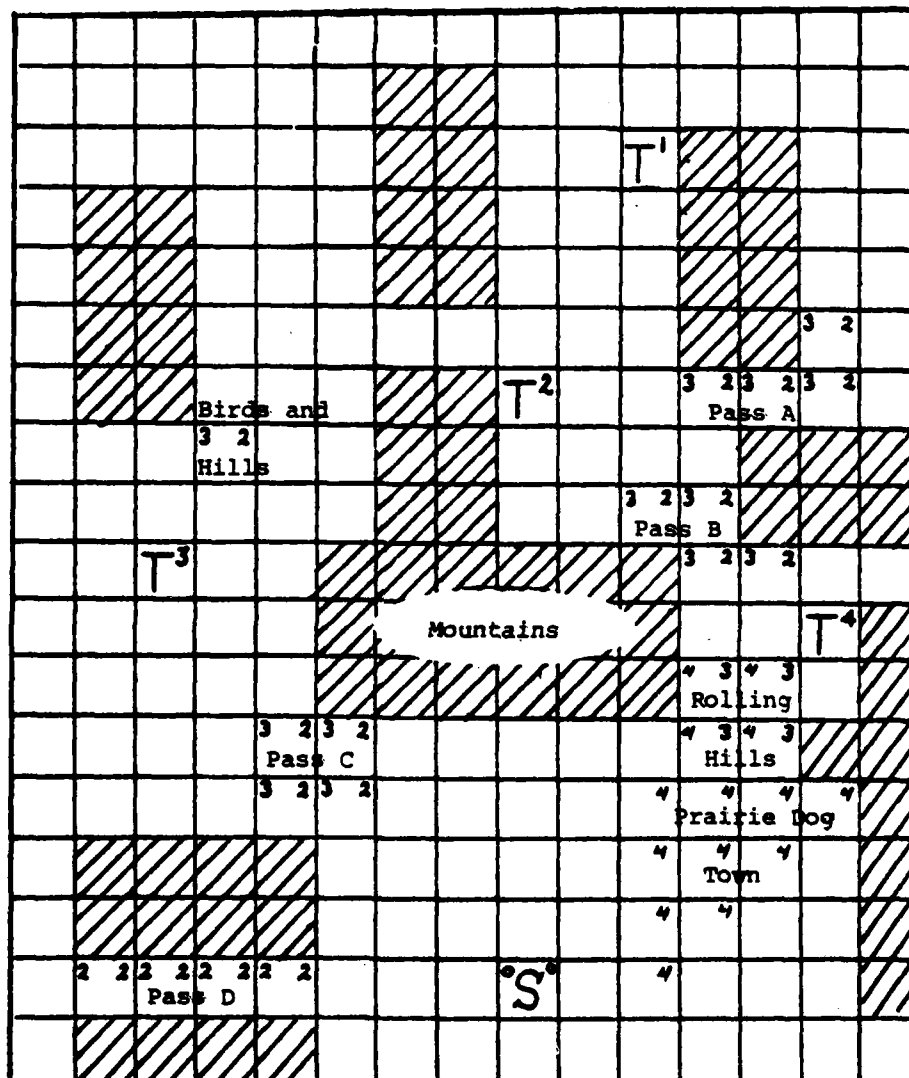
In order to use Lee's algorithm, a grid is superimposed on a map of the land surrounding the source and targets. The size of the grid should be such that the land features and different costs and items can be represented by the square cells of the grid. Figure 4 shows the ungrided map and Figure 5 shows the grided map. Then, one must establish the cost to build the road and the average time to travel on the road non-diagonally through each cell

Time to travel →

2	2
---	---

 ← Cost to Build
through cell through cell

Cells without corner numbers have nominal
cost/time of one unit.



S = Source
T = Target. Number inside
Target Cell is Target
number.

Fig. 5. Grid Superimposed on MX Layout

in all feasible land areas. This has been done and listed in Table 1.

The cells over obstacles, such as mountains, are hashed to represent cells the road or path cannot traverse. In Figure 5, the blank cells represent the large areas of flat land where the cost and time units have been scaled to one (1) unit. These units are not shown. The small number in the upper right corner of the cells represents the cost to construct a road through the cell, at cost other than one (1). The small number in the upper left corner of the cells represents the average time to travel on a road that traverses through the cell, at time other than one (1).

Now the numbering technique starts.

1. Referring to Figure 6, one adds the cost of the adjacent (non-diagonal) cell to the source cell's cost of zero. The accumulated cost is written in that adjacent cell. One accomplishes this four times for the source cell, once for each of the adjacent cells, as shown in Figure 7. These four adjacent cells are now called *frontier* cells, and the source cell is now called an *expanded* cell.

2. The frontier cells must now be expanded. One finds the frontier cell with the minimum accumulated cost. Ties are broken arbitrarily. Choosing an adjacent cell to the frontier cell that has not been expanded or that is not a frontier cell, one adds its cost to the frontier

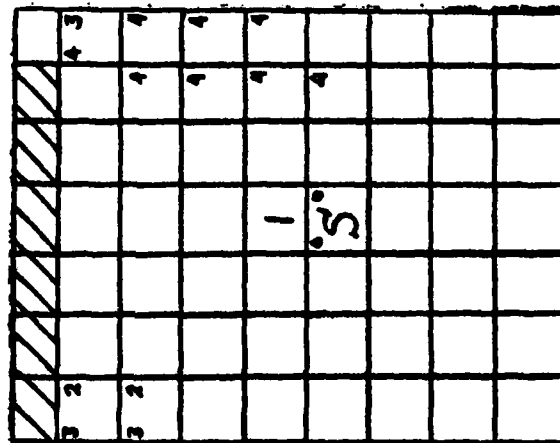


Fig. 6. Expanding Source Cell:
First Adjacent Cell

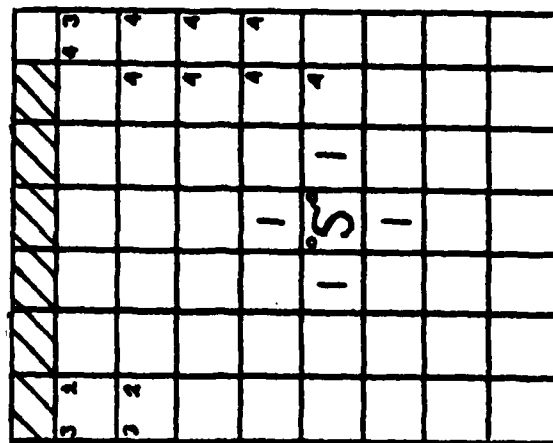


Fig. 7. Expanding Source Cell:
All Adjacent Cells

cell's accumulated cost and places that accumulated cost in the adjacent cell. Figure 8 shows the expansion of a frontier cell.

3. One continues finding adjacent cells and calculating their accumulated costs until no more cells are adjacent to the frontier cell being expanded. Figure 9 shows that the right frontier cell has been expanded as much as possible. The frontier cell is now an expanded cell, and the adjacent cells with the newly calculated costs are now more frontier cells.

4. The procedure continues by finding the next frontier cell with the minimum accumulated cost, and then expanding it (steps 2 and 3).

5. The procedure stops when all target cells become frontier cells, as illustrated in Figure 10. (In Figure 10, the highlighted cells are the source and target cells.) At this point, the accumulated cost entered in the target cell is the cost of a minimum cost path from the source to the target.

The following procedure is used to trace the paths back from the targets to the source:

1. The cost of the target cell is subtracted from its accumulated cost.

2. The difference is compared to the accumulated costs in the target cell's adjacent cells. The adjacent cell with the accumulated cost equal to the difference is

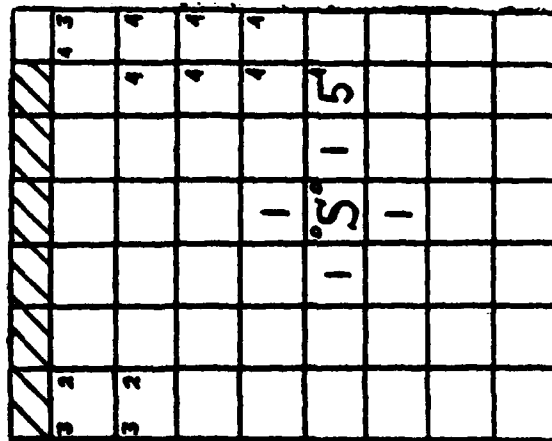


Fig. 8. Expanding Right Frontier Cell

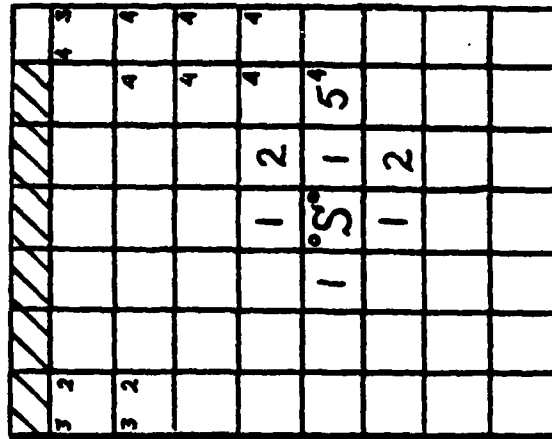
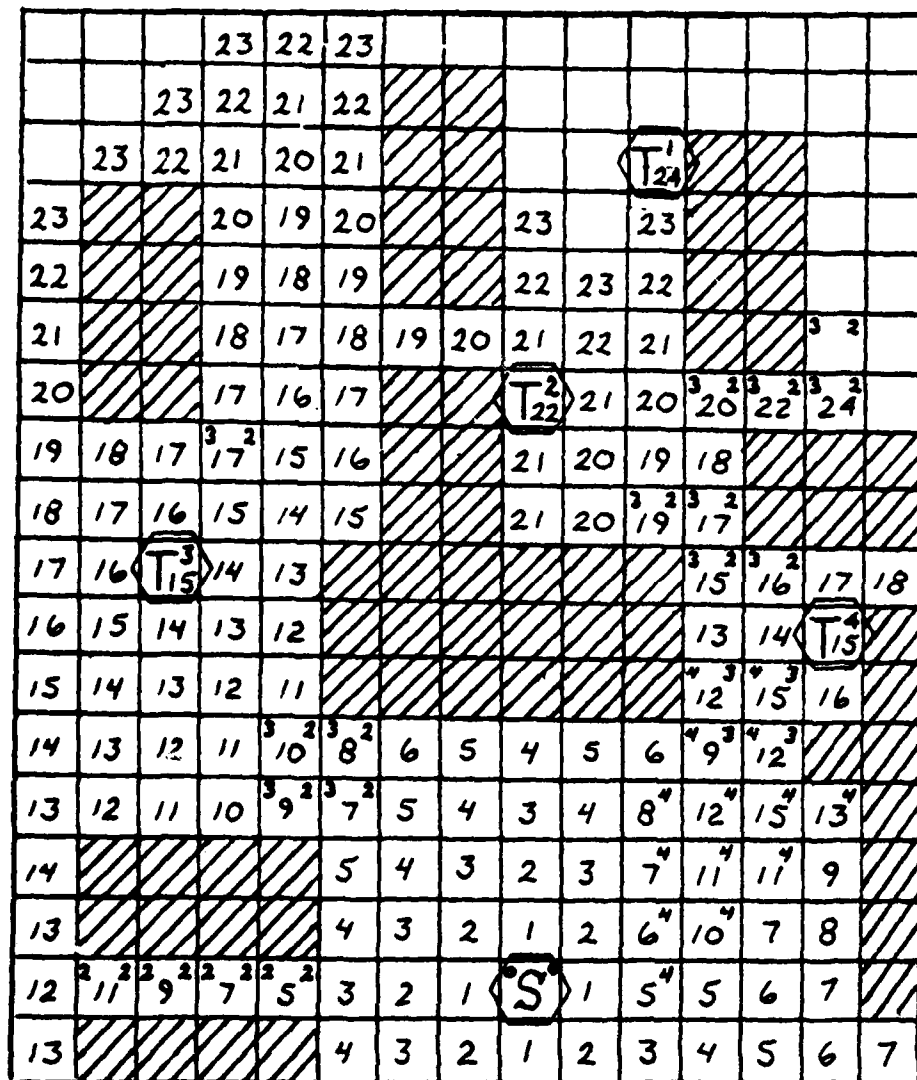


Fig. 9. Expanded Right Frontier Cell



Used to highlight the Source
and Target Cells for the reader.

Fig. 10. Cell Expansion Completed (Cost Run 1):
All Targets are Frontier Cells

the next cell in the path back to the source. Figure 11 points out these path cells in this procedure for the right most target cell in the example problem. Ties are broken in favor of the adjacent cell in the direction toward the source. This tie-breaking rule is not necessary, but speeds the process of reaching the source. All ties in the algorithm may be broken arbitrarily.

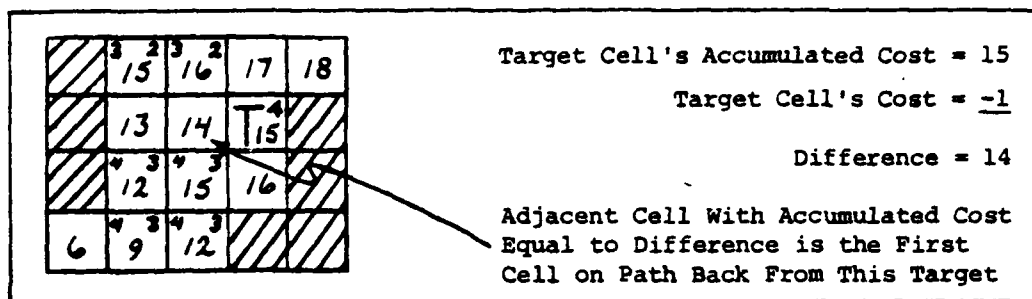


Fig. 11. Finding the Path Back

3. The cost of the new cell in the path is subtracted from its accumulated cost and compared to its adjacent cells.

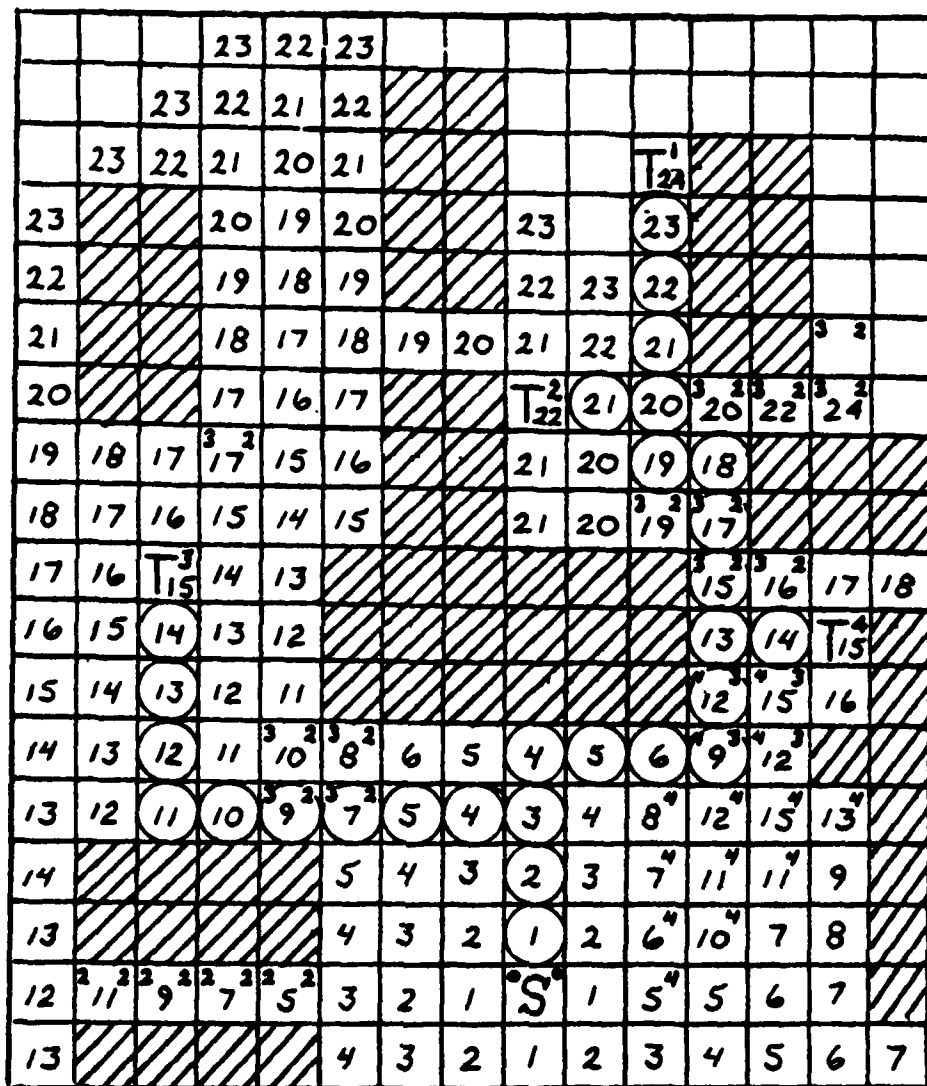
2. The adjacent cell with the accumulated cost equal to the difference is the next cell in the path back to the source. Ties are broken in favor of first, continuing in the same direction or, secondly, in the direction toward the source cell.

5. The trace back to the source ends when the source is reached. Steps 1 through 5 are repeated for

each remaining target. The problem of finding the least cost paths between all targets and the source ends when the trace back is completed for the last target, as illustrated in Figure 12 with the highlighted cells.

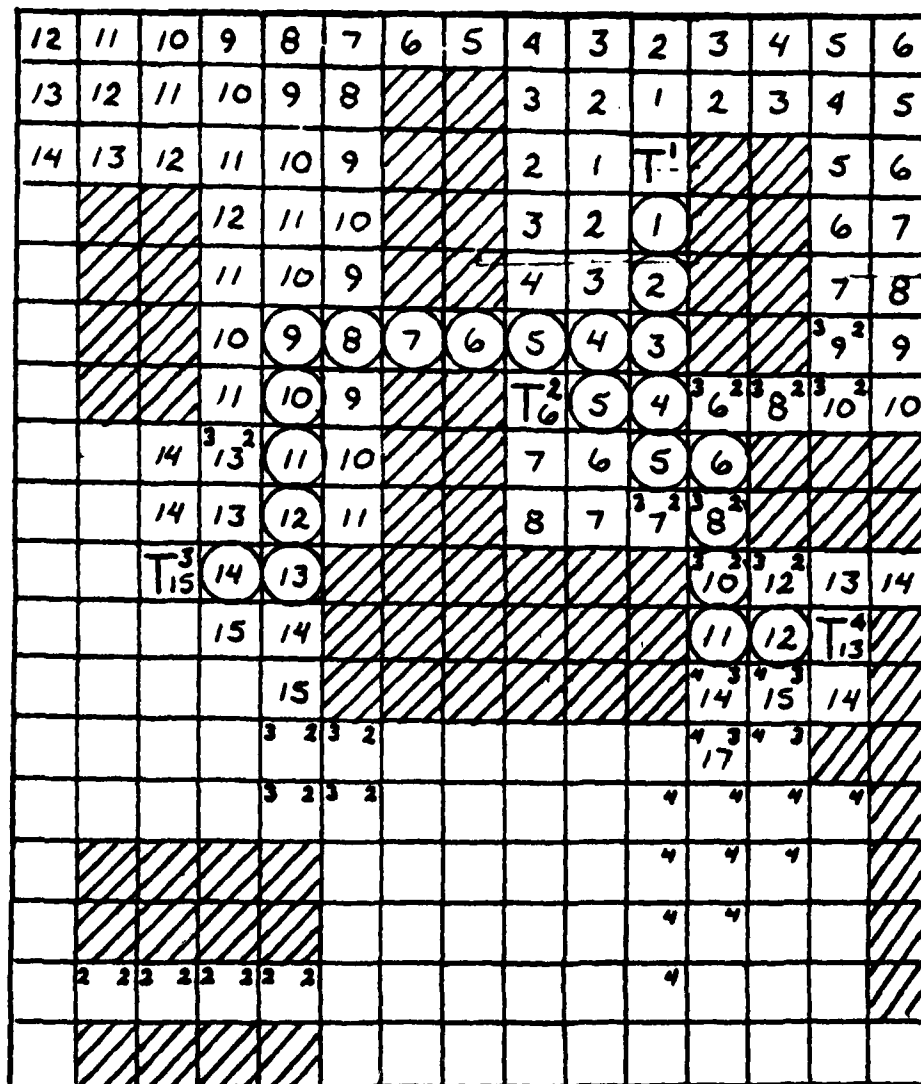
However, so far, the algorithm has only given the minimum cost paths from the source to the targets. By additionally knowing the least cost paths between all targets, then one could find the least cost network to link the source to all the targets without redundant roads. Consequently, the methodology repeats Lee's algorithm on the cost data $N-1$ times, where N is equal to the total number of nodes (source plus number of targets), to find the least cost path between all targets.

When repeating the algorithm the second time, the source node is deleted from the problem because the least cost paths between the source and all targets are already known. One of the target nodes becomes the source node. Least cost paths are found for the new system of nodes. Figure 13 highlights the paths found during the second iteration. When repeating the algorithm the third time, the source node of the second run is deleted. One of the remaining target nodes becomes the source. Least cost paths are found for the new system of nodes, as shown in Figure 14. The last repeat of the algorithm occurs when only two nodes, one source and one target, are left in the system.



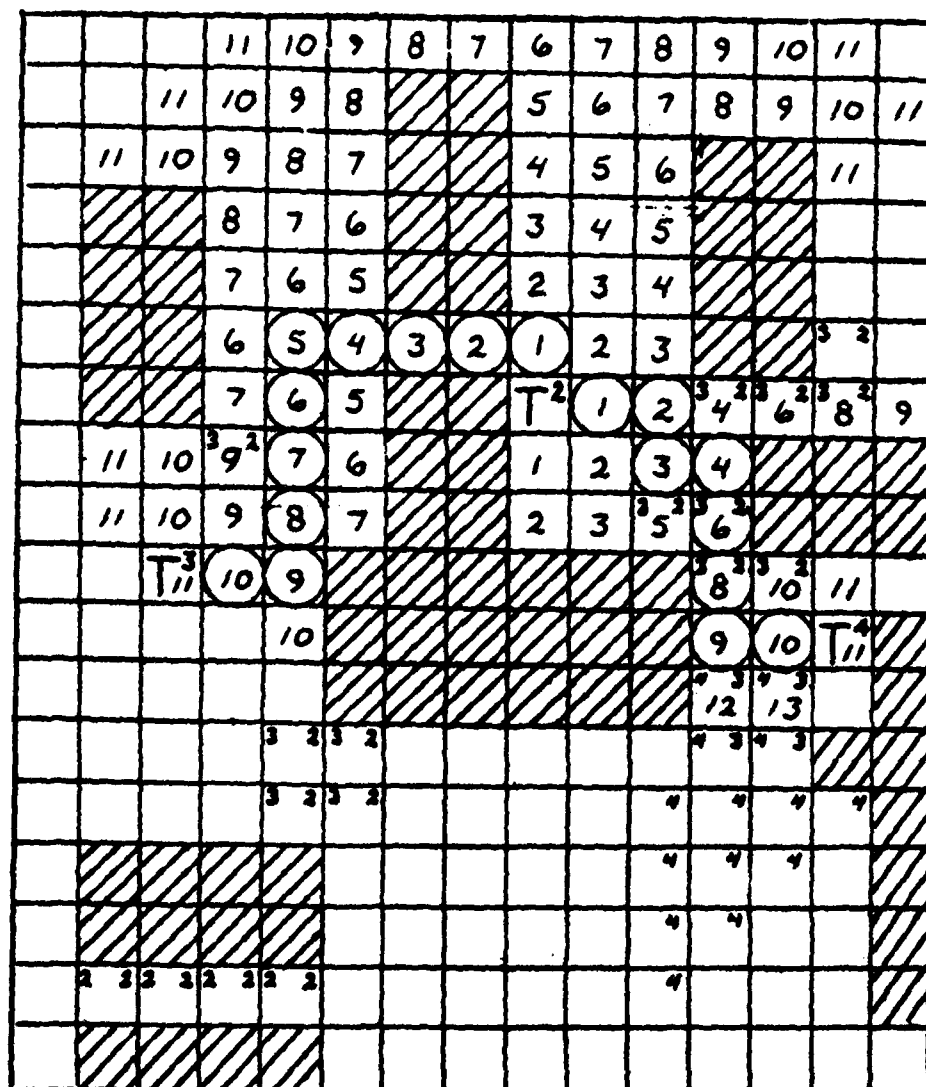
Indicates a Cell on the Path

Fig. 12. Path From All Targets to Source (Cost Run 1)



Target 2 is the new Source.

Fig. 13. Least Cost Paths Between Target 1 and Targets 2, 3, and 4 (Cost Run 2).



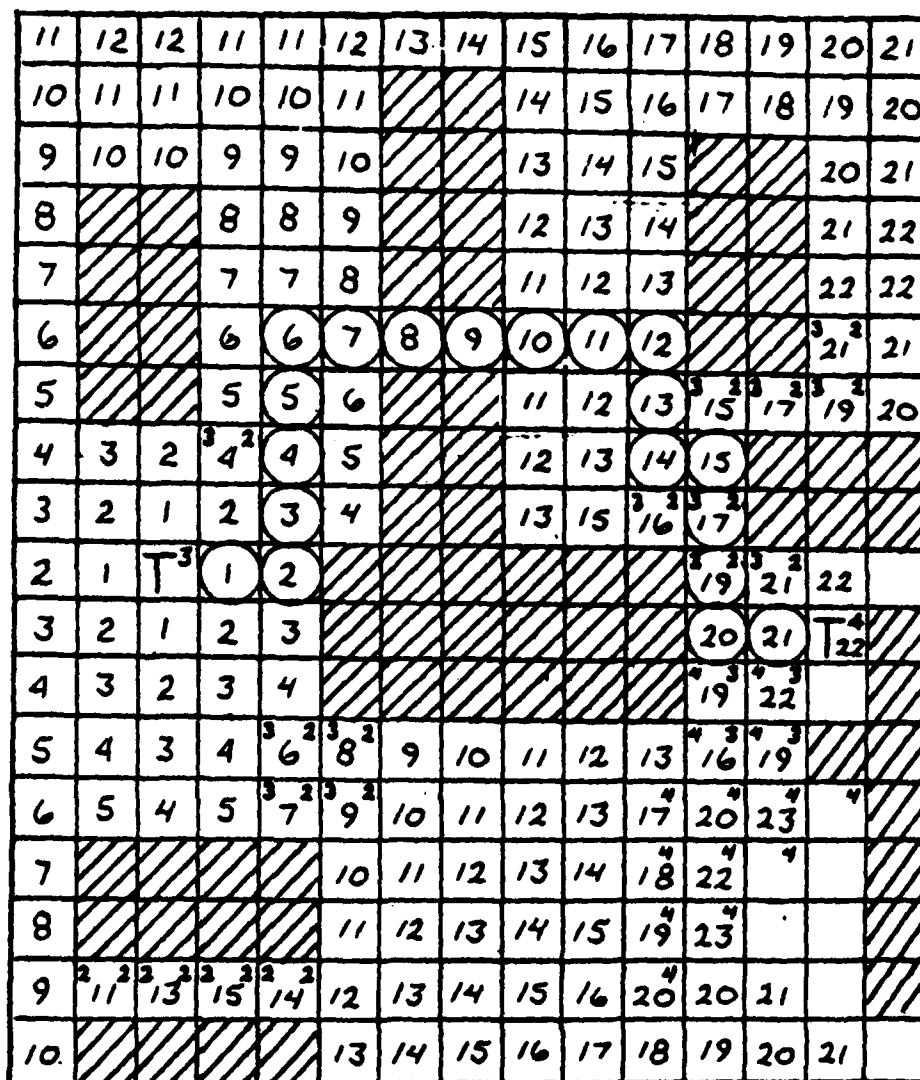
Target 2 is the new Source.

Fig. 14. Least Cost Paths Between Target 2 and Targets 3 and 4 (Cost Run 3)

Figure 15 shows the least cost path between the two remaining nodes.

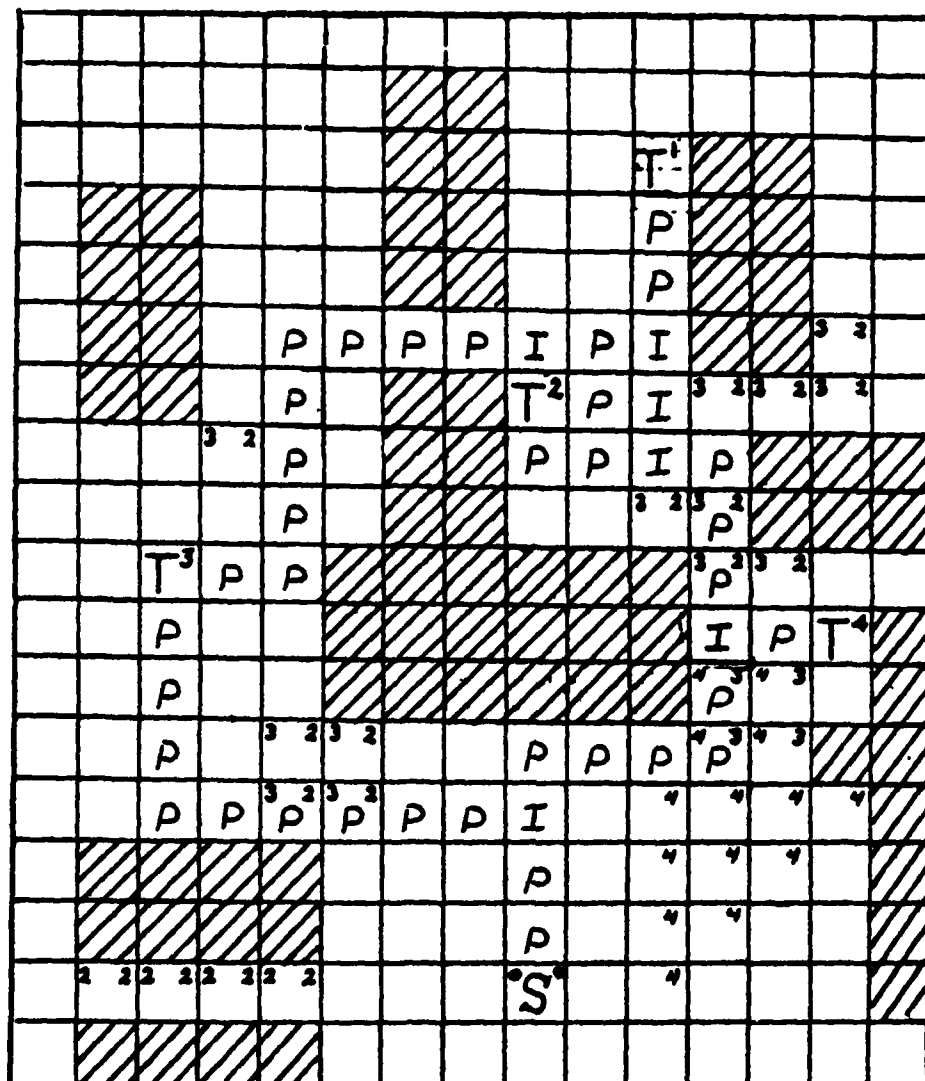
After accomplishing the algorithm N-1 times, least cost paths between all nodes are produced. Figure 16 represents all these path cells with "P's" in the cells. Paths will cross each other in certain cells, indicated by "I's" in Figure 16, and these cells become new nodes in the network. These new nodes, intersection nodes, may be needed as intermediate nodes in the network that connects the source to each target with the least cost without redundant paths. According to Lawler, "such [a network] a minimum tree, called a *steiner tree*, may contain nodes other than points which are to be spanned. These are called *steiner points* [Ref 12:290]." The use and significance of these steiner points will be discussed in the next section dealing with finding the least cost network of paths to connect the source to the targets.

The methodology accomplishes Lee's algorithm one more time to find minimum time to travel paths from the source to each target. This is needed in order to meet the second objective of the thesis (to completely satisfy the operational considerations for a minimum time network). The average time to traverse the cell, equal to one time unit or found in the upper left corner of the cell, instead of the cost to build through the cell, is used when accomplishing Lee's algorithm this time. Accomplishing Lee's algorithm one



Target 3 is the new Source.

Fig. 15. Least Cost Path Between Target 3 and Target 4 (Cost Run 4 [N-1])



T = Target Cell
 S = Source Cell
 P = Path Cell
 I = Intersection Cell

Fig. 16. Least Cost Paths Between Source and Target Nodes

time for time to travel, produces minimum time to travel paths to each target from the source. Figure 17 highlights the paths found for the time iteration. These paths represent the network found for minimum time between the source and the targets. The total time of each path from source to target is added and then divided by the total number of targets to give the minimum mean time to travel from the source to all targets. One should note, there are no better mean time paths. This is based on the successive minimizations Lee's algorithm used to find the paths (Ref 13:346-348). The cost of building the minimum time paths is found by adding the cost of the cells that correspond to the cell paths in the minimum mean time network. Table 3 lists the time and cost information of the minimum mean time network.

At this point in the methodology, the following information is available to the remainder of the algorithms and methods of the methodology:

1. The least cost paths between all nodes.
2. The value of the minimum mean time to travel network. The minimum mean time network is also an upper bound on the cost to connect the source to each target, as spending more money to build any other paths would not gain any operational efficiency, measured by time to travel.

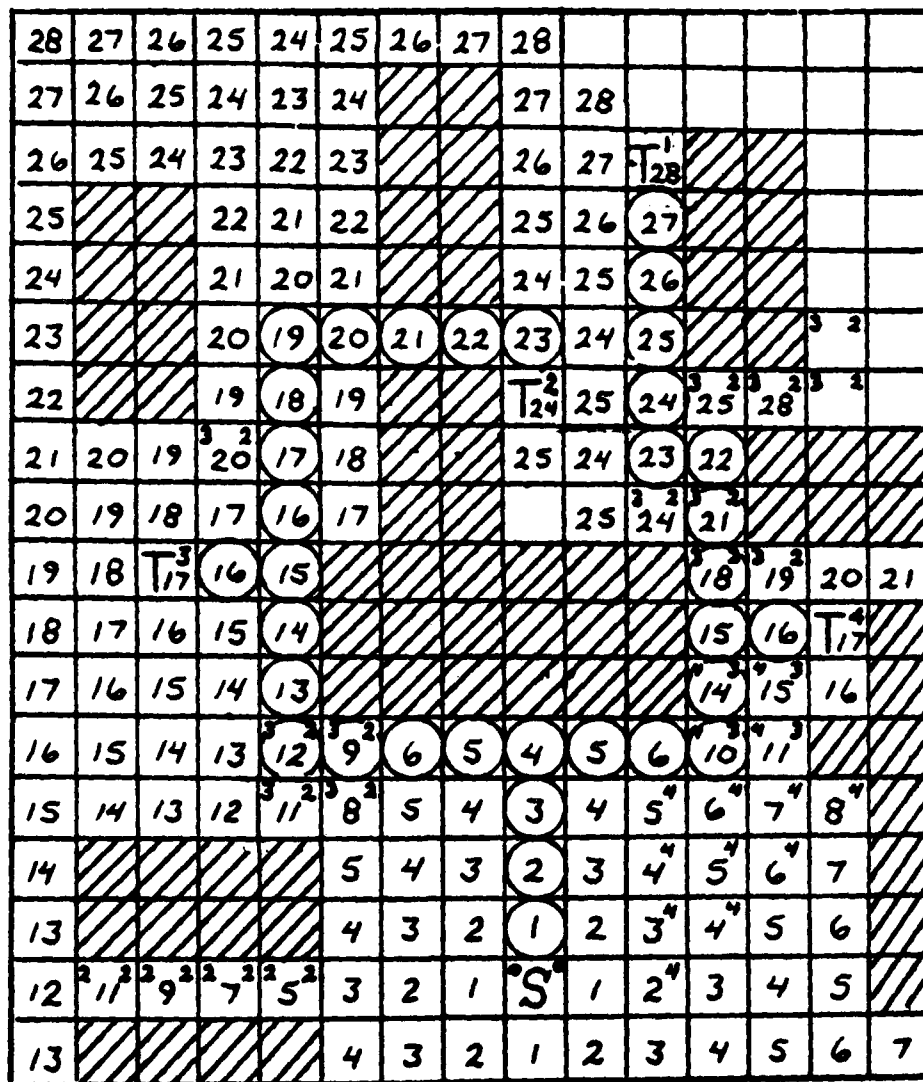


Fig. 17. Minimum Time to Travel Paths from Source to All Targets (Time Run)

TABLE 3
TIME AND COST INFORMATION OF THE MINIMUM
MEAN TIME NETWORK

Minimum Time to Travel from Source to	
Target:	T1 = 28 time units
	T2 = 24 time units
	T3 = 17 time units
	T4 = 17 time units
Total = 86 time units	
Minimum Mean Time to Travel from	
Source to All Targets:	
= 86/4	
= 21.50 time units*	
Cost to Construct Minimum	
Mean Time Network:	
= 46 cost units**	

Notes

Data for Table 3 taken from Figure 17.

*Minimum mean time lower bound

**Cost upper bound

3. The least construction cost paths between nodes and their associated construction cost and time to travel.

Figure 18 shows the least cost paths between the source and targets of the example problem. The nodes labeled "SP" are steiner points or intersections of the paths. Obstacles and mountain passes were avoided, if possible. Table 4 lists the network information (nodes and the cost and time units associated with the paths between the nodes) of the network in Figure 18.

The next objective of the thesis is to find the most inexpensive network cost to build the road from the source to each end point. The procedure described in the next section uses the information found thus far to accomplish that objective.

Finding the Least Cost Network

The third objective of the thesis was to find the least acquisition cost road network. As with any weighted network problem, the combination of arcs that connect all nodes for the least weight (cost) is defined as the minimum spanning tree. If this tree is allowed to contain nodes or points other than those of the original network, the added points are called *steiner points*, and the resultant minimum spanning tree is defined as the "minimum steiner tree (MST)." For the remainder of the thesis, the terms *steiner point* and *steiner node* are considered synonymous.

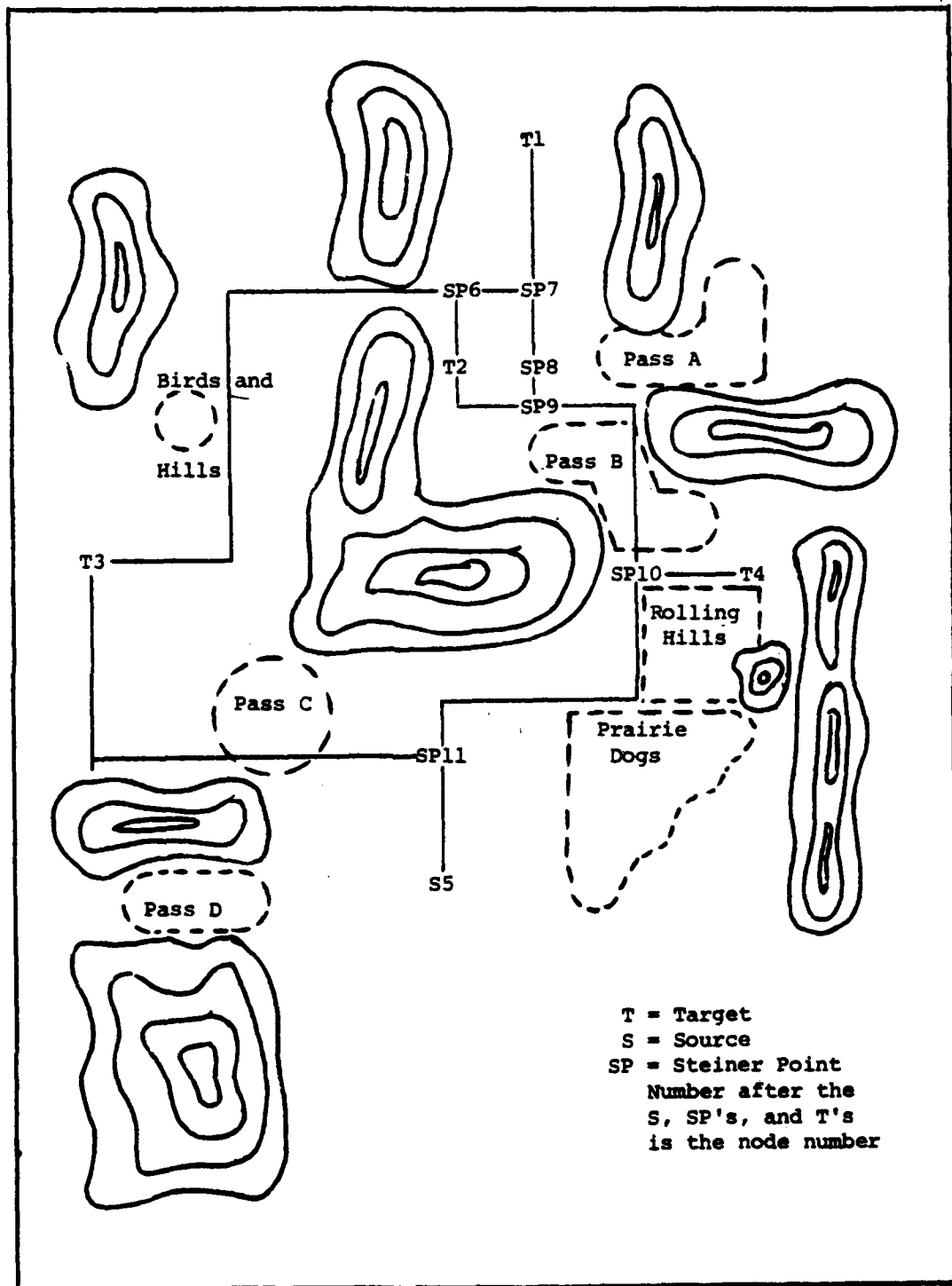


Fig. 18. . Example MX Layout with Least Cost Paths

TABLE 4
COST PATH DATA

Node	to	Node	Cost	Time
1		7	3	3
2		6	1	1
2		8	2	2
2		9	3	3
3		6	10	10
3		11	12	14
4		10	2	2
5		11	3	3
6		7	2	2
7		8	1	1
8		9	1	1
9		10	6	8
10		11	10	12

Note: Data taken from Figure 18.

The intersections of the least cost paths identified by Lee's algorithm will be steiner points in the least cost network solution. Therefore, the least cost network will be by definition a minimum steiner tree. The method used for solution of the minimum steiner tree is an adaptation of the steiner network algorithm from Lawler's text (Ref 12:292). Given a weighted network of two node sets, N and S, such that N does not exist in S, the Lawler algorithm selects the proper subset of S to span the N specified nodes (source and targets) in a tree of minimum weight. The tree may include any number of the S nodes as steiner points. A steiner point can be generally described as any point added to a graph to reduce the size of the minimum spanning tree of that graph. This is best depicted in a three-node graph as shown in Figure 19. This is a specific example of a steiner point where arcs must follow rectilinear paths. The number of arcs incident to a node defines the degree of that node. Notice a steiner node would always be at least degree three. To actually use the algorithm, it is necessary that arc weights are non-negative and satisfy the metric requirement that for any arc A_{ij}

$$A_{ij} \leq A_{ik} + A_{kj}$$

for all i, j, and k. In simple terms, this means if a direct path exists between two nodes, its length cannot

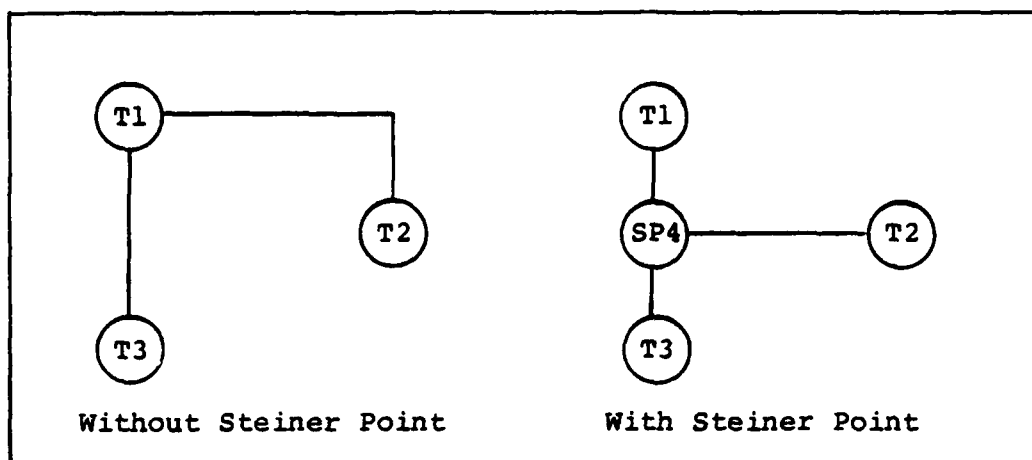


Fig. 19. Steiner Point Example

exceed the length of any other possible path between the nodes. To limit his search for the proper set of steiner points, Lawler verifies the number of steiner points in the minimum steiner tree can never be more than $N-2$, where N is the number of nodes that are not steiner points (Ref 12:291).

Lawler's steiner network algorithm is a three-step process:

1. Given a node set N , a steiner node set S , and sufficient arc lengths between them to form a path between all nodes, one computes the shortest path between all pairs of nodes. For the general problem, this removes any arc lengths that do not satisfy the metric requirements.

2. Using the lengths of these arcs from the shortest paths, one computes the minimum spanning tree for each subset of $N-2$ or less steiner points.

3. One selects the shortest minimum spanning tree to determine the correct steiner points for the minimum steiner tree solution. The tree is reconstructed using all arcs on the shortest paths between the nodes and steiner points selected.

Applying this procedure to the methodology, the least cost path determination performed by Lee's algorithm provides the steiner node set and arc weights necessary for step 1. The nature of this output allows the first departure from Lawler's process. Since only least cost paths are included in the arc set, the metric requirement will never be violated. This has two benefits: (1) computation of the shortest path between all pairs of nodes is unnecessary, and (2) minimum spanning tree computations in step 2 of Lawler's process can include all arcs in each path rather than just the pseudo arcs representing the shortest path. Under these conditions the least cost minimum spanning tree selected in step 3 will be the minimum steiner tree solution.

The second and more dramatic departure from Lawler's algorithm is in selection of the steiner node subsets. Lawler evaluates each subset of $N-2$ or fewer steiner nodes to find the least cost MST. This is an exponential process in the number of steiner nodes, S . This process is replaced by a numerical elimination of what are effectively unnecessary steiner nodes as explained

in detail in the computerization section of this thesis. The result is a heuristic development of the MST in at most $2^S - 1$ evaluations compared to all $2^S - 1$ subsets of steiner points minus N and $N - 1$ combinations, or

$$2^S - \left(\frac{S!}{N!(S-N)!} \right) - \left(\frac{S!}{(N-1)!(S-N+1)!} \right) - 1$$

evaluations by Lawler. In our example four target problem, this equates to 11 evaluations instead of 172.

The emphasis on this stage of the thesis methodology is in finding the lower bound on construction cost of the road network, or the cost of the minimum steiner tree. The procedure cannot guarantee this to be the only network at this cost, only that a network of less cost does not exist within the heuristic decision rules used to find the minimum steiner tree. Empirically, these heuristic decision rules seem to find the value of a least cost network.

Thus far, the operational efficiency of the least cost network has not been addressed. In fact, the least cost network may prove to be the least efficient or have the greatest mean time to travel to each end point. Since these two objectives conflict and there is a desire to optimize each, the next section of this thesis will define the method chosen to accomplish the optimization.

Cost and Mean Time Optimization

Thus far the routing problem has been reduced to a network of feasible least cost paths from the source node to all targets and from target to target. Each path can be described as a series of arcs connecting the source, targets, or steiner points on that path. The methodology further defines the arcs as having two performance indices: cost to construct and time to travel. Our objective is to minimize both the construction cost of the network and the mean time to arrive at a target node from the source. This problem can be best approached using the Multiple Criteria Decision Theory (MCDT) technique known as Multiple Objective Optimization Theory (MOOT). In fact, the problem contains some of the classical characteristics of MOOT.

The network design has:

1. Multiple objectives to minimize: cost and time.
2. Objectives which are conflicting in nature.
3. Decision variables (arcs) measured in noncommensurable units: cost units and time units (Ref 3:16).

The operational effectiveness index of the time could be reduced to an operational cost and the network optimized to a single objective, minimum cost. This approach was avoided for two reasons. First, operational cost of a specific arc is dependent on the number of targets it serves. Therefore, an arc cost capturing construction and operations expenditures would vary with network

design. Second, and most important, the index of time considers many unquantifiable aspects of operations that a pure cost model would not, as discussed in Chapter I.

"[MOOT] can be used for generating optimum solutions for the alternative actions which extremize the components of a vector of performance indices [Ref 2:218]."

In other words, this vector can be compared to an objective function that is maximized or minimized with respect to each of its variables. In the road network problem, the optimization vector $\bar{Z}$, consists of Z_1 , cost and Z_2 , mean time. Once the extremes of these elements are identified, they are checked for dominance to form a non-dominated solution set (NDSS) from which a decision maker can choose a course of action. The ultimate outcome of this methodology is to produce a NDSS of road networks. Analytically, let X be the set of all subsets of arcs, A , that span each target node and the source node. Define $\underline{X}$ as a single subset in X . The objective functions are to minimize Z_1 and Z_2 where for all $\underline{X}$ in X ,

$$Z_1(\underline{X}) = \sum_i K_i C_i,$$

and

$$Z_2(\underline{X}) = \frac{1}{\text{TGTS}} \sum_i K_i N_i T_i$$

where C_i = cost of arc a_i , $i=1,2,3,\dots$, total number of arcs;

$K_i = 1$, if a_i exists in $\underline{X}$;

$K_i = 0$, otherwise;

T_i = time to traverse a_i ;
 N_i = number of targets served by a_i ; and
 TGTS = total number of targets.

This assumes time to traverse an arc is independent of the direction of travel. This assumption is reasonable for a heavily loaded vehicle required to move up and down equal grades at the same average speed.

In the four target problem, all subsets of arcs for incremental costs 46 through 39 will be identified by a binary search of the arc set costs. This categorizes the subsets by their Z_1 , cost, index. Next, all subsets for each Z_1 value are evaluated for their Z_2 index, mean time, and the least mean time subset selected. This is graphically presented in Figure 20. The resultant subsets are candidates for the NDSS but must be checked for dominance to define the efficient frontier. In a two-dimension solution space, $\underline{X}$ is a subset in X on the frontier if and only if there does not exist another subset $\underline{X}'$ of less cost with the same or smaller mean time to travel. Mathematically, for any candidate subset, $\underline{X}$, in X , $\underline{X}$ is in the NDSS, if and only if no $\underline{X}'$, which is an element of X , such that

$$Z_1(\underline{X}') < Z_1(\underline{X})$$

and

$$Z_2(\underline{X}') \leq Z_2(\underline{X}).$$

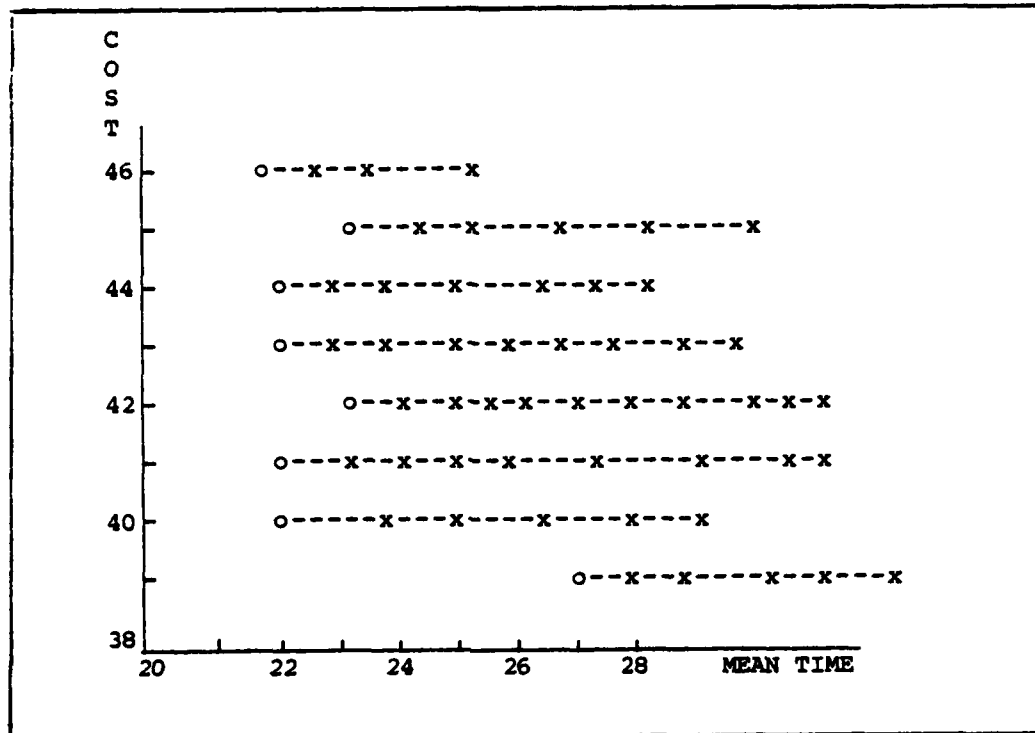


Fig. 20. Cost Vectors

Let X^* be the set of $\underline{X}$ that comprise the NDSS. The X^* are plotted as a discrete efficient frontier. Applying this dominance test to the selected subsets in Figure 20, first, let $\underline{X}$ be the subset of least mean time for the highest cost, 46. Since all other alternatives have higher mean times, this subset is in the NDSS. Now let $\underline{X}$ equal the least mean time subset for costs 45 and 41. These subsets are all dominated by the lower mean time of the subset at

cost 40. This leaves the subsets for cost 46, 40, and 39 on the efficient frontier, as plotted in Figure 21. The preferred solution is determined by comparison of the shape of the frontier to a decision maker's preference structure of cost and time. The "Use of Results" chapter explains more fully the decision maker's use of the efficient frontier.

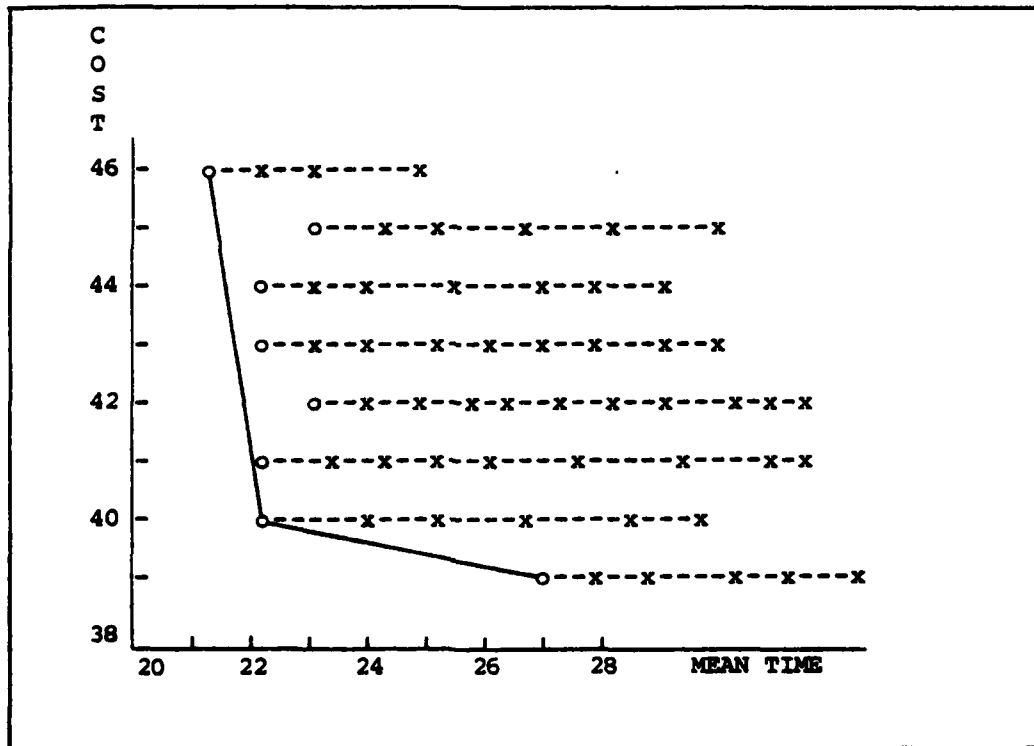


Fig. 21. Efficient Frontier

The remaining discussion of the methodology is concerned with the numerical processes required to obtain this multiple optimization, that is to select all arc sets for an incremental cost and calculate the mean travel time of each arc set selected.

Computation of the Cost Vector. The first step in the MOOT technique is to identify the elements of each cost vector or all networks that can be constructed from the arc set such that the sum of the included arcs equals a desired cost. The concept of identifying all subsets of arcs for a possible construction cost suggests complete enumeration of all subsets or a method to evaluate $2^E - 1$ arc sets where E is the number of arcs. Horowitz and Sahni have published an algorithm called SUMOFSUBS that uses binary branch and bound techniques to solve this problem without complete enumeration (Ref 8:342). The bounding rules of their algorithm are extremely efficient and readily adaptable to the road network problem. However, evaluation of the SUMOFSUBS process did result in modifications to improve computational time. These modifications and the basic algorithm will be discussed in the computerization chapter. Even after the improvements, this binary search is the single most limiting process in the methodology. Table 2 illustrated this limitation. The computational time still remains exponential in E , the number of

arcs. As with any exponential algorithm, the primary concern is to limit its required use. This emphasizes the importance of the previous methods that tightly bound the range of values used by this algorithm. Once bounded and modified, the Horowitz and Sahni algorithm, SUBOFSUBS, is used to select the elements of each desired cost vector needed for determination of the non-dominated solution set.

Finding Shortest Time Paths. At this point in the methodology, the minimum mean time of the networks contained within the possible path subsets is not known; however, each path's associated arc construction cost and time to travel are known. One wishes to know the time to travel between the source and each target for each possible subset so that a mean time to travel can be calculated. Then the minimum mean time subset can be selected for the construction cost being evaluated.

To find each subset's time to travel between the source and each target, the methodology finishes by using the Bellman-Ford method. This technique finds the shortest path between the source and any node. The shortest path is in terms of the minimum time to travel between the source and each target.

To accomplish the Bellman-Ford method, the following is accomplished (Ref 12:74-75):

1. From the subset being evaluated, the method forms a node adjacency matrix, $\bar{T}$, where the element t_{kj} is the time to travel between node k and node j.

2. The method establishes the vector $\bar{U}$, where u_j represents the time value of the j^{th} element in $\bar{U}$. The vector $\bar{U}$ is initially set to selected elements of the $\bar{T}$ matrix by the following:

$$u_j^{(0)} = t_{sj}, \text{ for all } j \neq s$$

and

$$u_s^{(0)} = 0,$$

where s is equal to the number of the source node. The superscript denotes the iteration number that calculates the $\bar{U}$ vector.

3. One calculates the new $\bar{U}$ based on the old $\bar{U}$ according to the following equation:

$$u_j^{(m+1)} = \min \{u_j^{(m)}, \min_{k \neq j} \{u_k^{(m)} + t_{kj}\}\},$$

for all j and k, where m is the iteration number.

4. When $u_j^{(m+1)}$ and $u_j^{(m)}$ are equal for all j, then the method stops, as iteration m+2 will not find a shorter path from the source to each target. No shorter paths exist (Ref 12:74-75). Each element of $\bar{U}$ now represents the shortest time from the source node to each node number. The node number is identified by the subscript, i.e., u_4 represents shortest time route from source to node 4.

5. The sum of the $\bar{U}$ elements associated with the target node numbers is divided by the total number of targets. The resultant represents the mean time to travel between the source and all targets.

6. The process repeats for the next subset of paths associated with the cost being evaluated.

The subsets from the set of subsets associated with the desired cost that have the minimum mean time to travel are the networks that can be built with the minimum mean time for the desired cost being evaluated. Since the shortest path algorithms are well known in network theory, an example of the procedure is unnecessary.

The costs associated with all the paths in the networks have been found for their least cost to build. The mean times to travel in the networks have been minimized by the last method, within the assumptions and limitations. Consequently, the two objectives, least construction cost and minimum mean time to travel between the source and end points, have been optimized in combination. In combination they yield a solution set of alternatives which now can form the non-dominated solution set. This set will be discussed in the next section, and its use will be discussed in the use of results section. However, at this point, the methodology has accomplished all of the stated objectives of the thesis effort. Now the planner or decision maker has a decision tool.

Methodology Results

This review of the methodology would be incomplete without an example of the form of the final output. Therefore, these concluding remarks will overview the methodology and illustrate the results of the four target problem.

As mentioned in the discussion of cost and time optimization, the ultimate goal of this methodology is to produce the set of non-dominated solutions that form the efficient frontier of feasible road networks. The construction cost, travel time, and geographical/environmental data base is converted into a weighted network of all possible paths for an efficient--least cost--road design. Various network analysis techniques are then applied to this general road structure to first, bound the feasible solutions; second, construct all solutions within the feasible range; and third, select from these solutions the road designs that best satisfy the chosen measures of effectiveness, least cost and minimum mean time to travel. Relating this output to the four target problem chosen to exemplify the methodology, Figure 22 and Table 5 depict the results of each cost/time optimization, the non-dominated solutions, and their position on the discrete efficient frontier.

The three points on the frontier represent the networks that have the minimum mean time for the desired

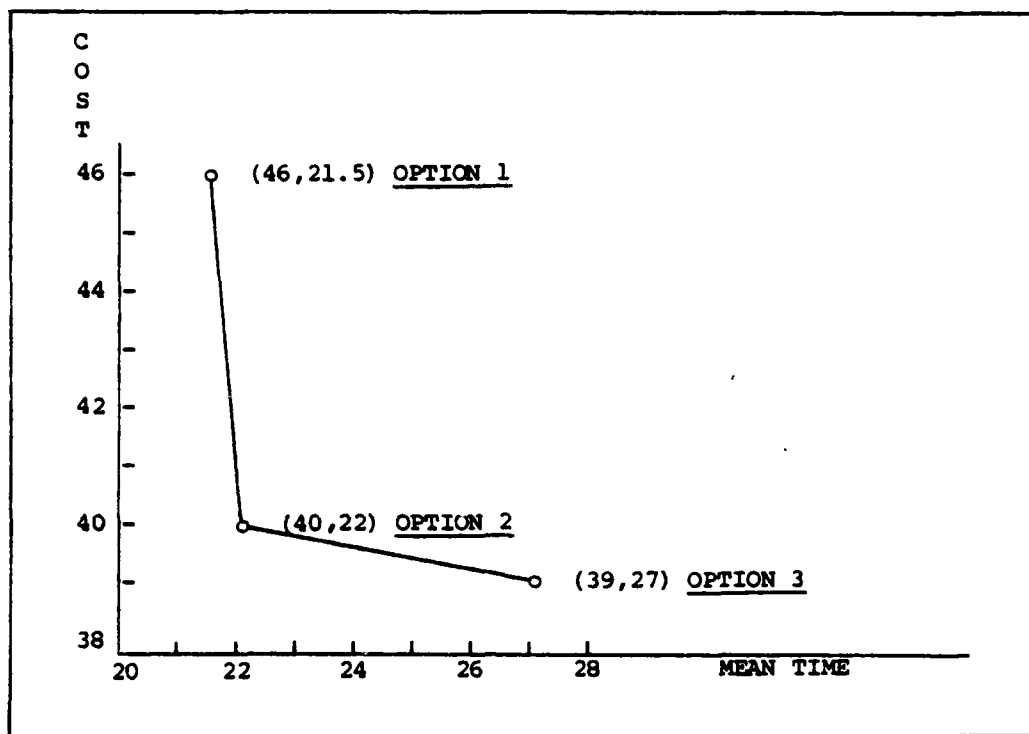


Fig. 22. Non-Dominated Solution Set--Four Target Problem

TABLE 5

COST VECTORS OPTIMIZED FOR MEAN TRAVEL TIME

Cost	Mean Time	No. of Networks Evaluated
46	21.50	4
45	23.00	6
44	22.00	7
43	22.00	9
42	23.00	11
41	22.00	9
40	22.00	6
39	27.00	6

cost evaluated. The actual networks are shown in Figures 23, 24, and 25 with cost and time annotated on each arc; i.e., (cost, time). Steiner points of less than degree 3 have been omitted from each path since they are unnecessary. Of the options evaluated, the decision maker is now presented these three options for his preference. Intuitively, option 2 is the best compromise of acquisition cost and operational efficiency. An important feature of the identification of the frontier in this procedure is an optimization over all options offered. The authors feel this ability to maintain a system perspective across a wide range of options is a major contribution their solution method offers the decision maker. How the decision maker could use the non-dominated solution set will be explained in the "Use of Results" chapter.

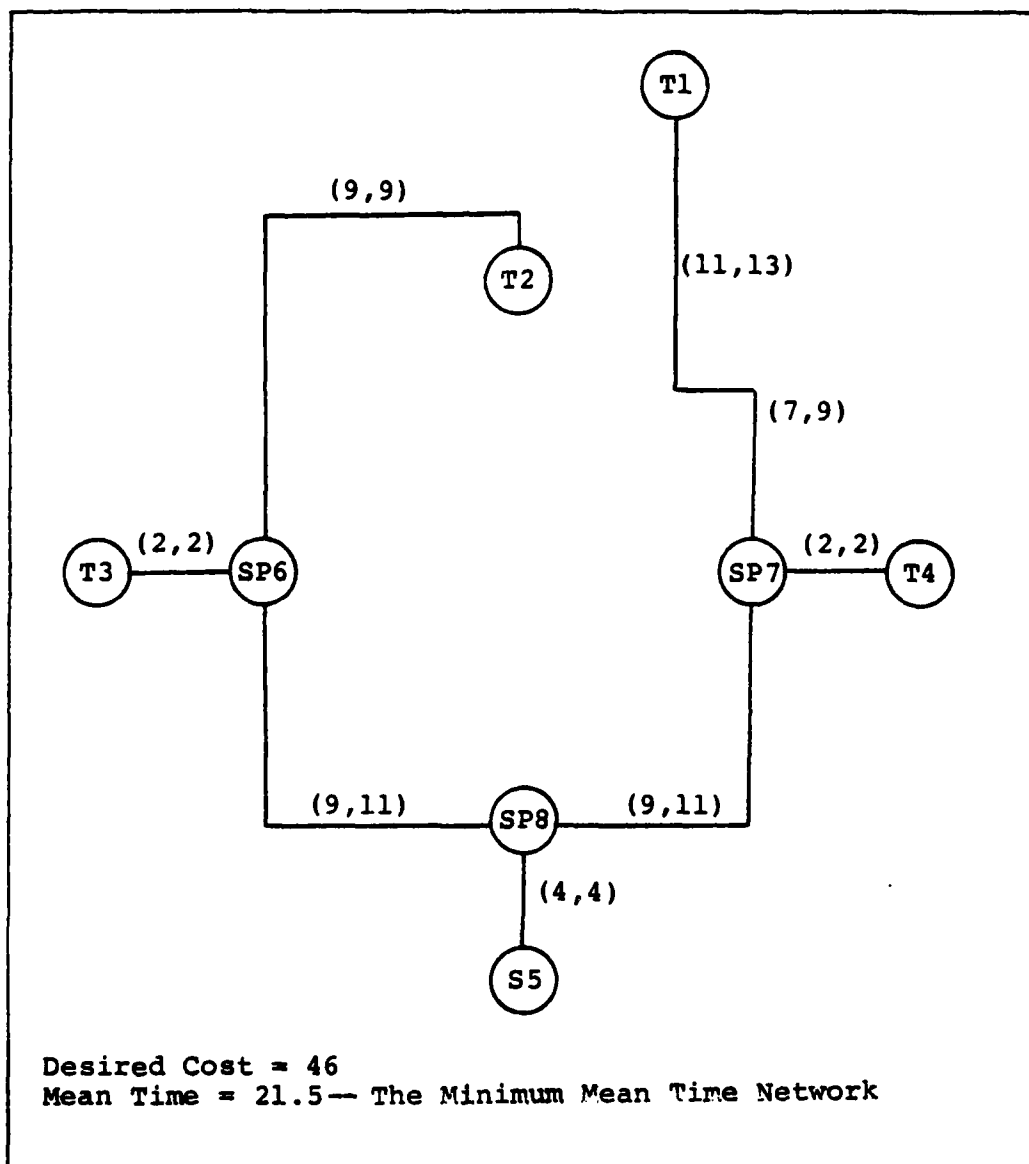


Fig. 23. Option 1--Maxcost Steiner Tree

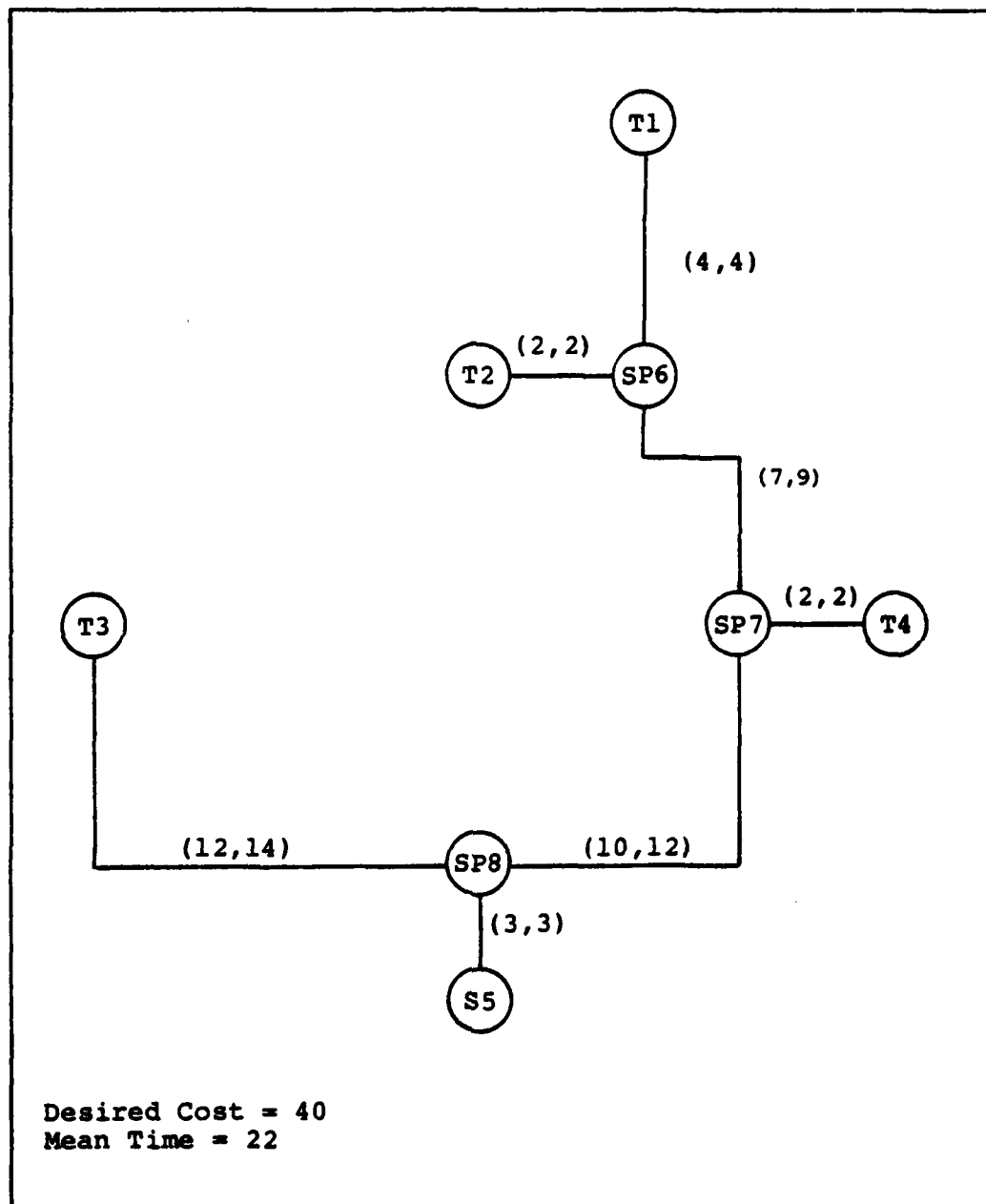


Fig. 24. Option 2--Intermediate Steiner Tree

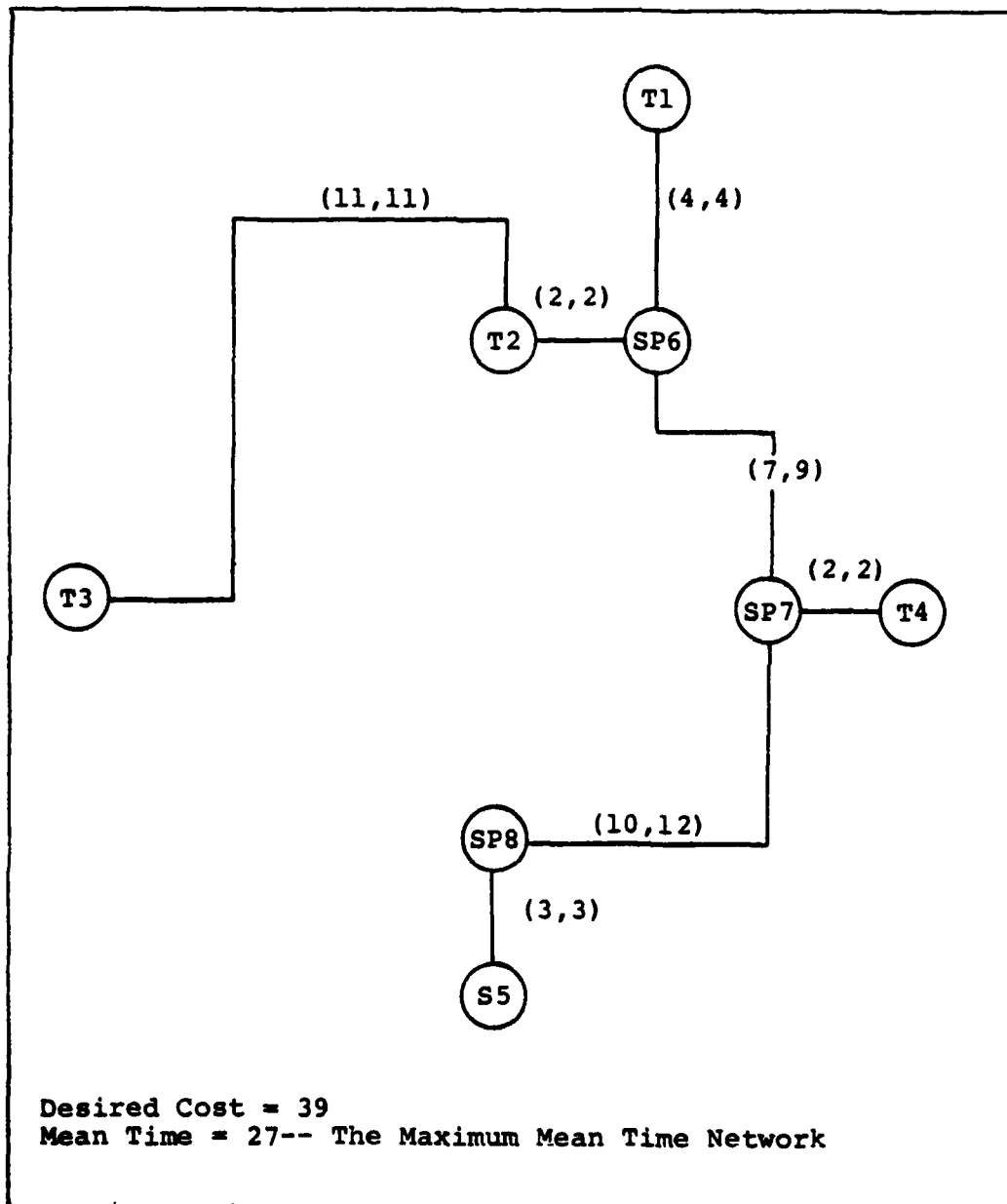


Fig. 25. Option 3--Mincost Steiner Tree

III. Computer Implementation

Because of the number of computations and iterations that the individual algorithms and methods of the methodology require, the research effort used computer programs to implement the algorithms and methods. The programs served to verify the methodology and make possible the use of the methodology on more elaborate problems. The nature of the methodology led to the modular and interactive design of the computer implementation. This chapter will review each computer program for their special features and use.

The implementation is modular because as each program is executed, its output data became the next program's input data. This modular concept also facilitates the ease of troubleshooting and the ability to analyze and modify the output data between each program. This latter feature will be discussed in the explanation of the programs, which is where it is important. However, in general, the ability to analyze the output helped in the verification of each program. The verification consisted of working examples by hand and verifying program output.

Each program was designed for use at the interactive level on a CDC Cyber 750. The programs are coded in standard CDC FORTRAN 77.

The flowchart in Figure 26 shows the general use of the programs to implement the methodology and illustrates the modular and interactive nature of the computerization. Additionally, the flowchart serves as a brief outline of the programs that will be discussed in the remainder of this chapter.

In general, the programs implement the methodology. First, the paths of least cost and minimum time must be found, given the basic cost, time, and land feature data. Second, a lower bound of total cost to build these paths must be found. Third, in order to find a solution space consisting of the combinations of networks that are best in terms of least cost and least time, all subsets of the network with these characteristics must be determined. This latter part of the methodology first searches for the subsets based on cost and then the next program determines the minimum times from the subset information.

Finding Path Connections

As stated in Chapter II, to find the best path connections between the source node and target nodes, the methodology uses Lee's algorithm. Best path connections means best in terms of all least cost paths to build between the nodes and the minimum time to travel from the source to each target. The structure of Lee's algorithm as programmed for the methodology is illustrated in the flowchart given in Figure 27. Before using the program,

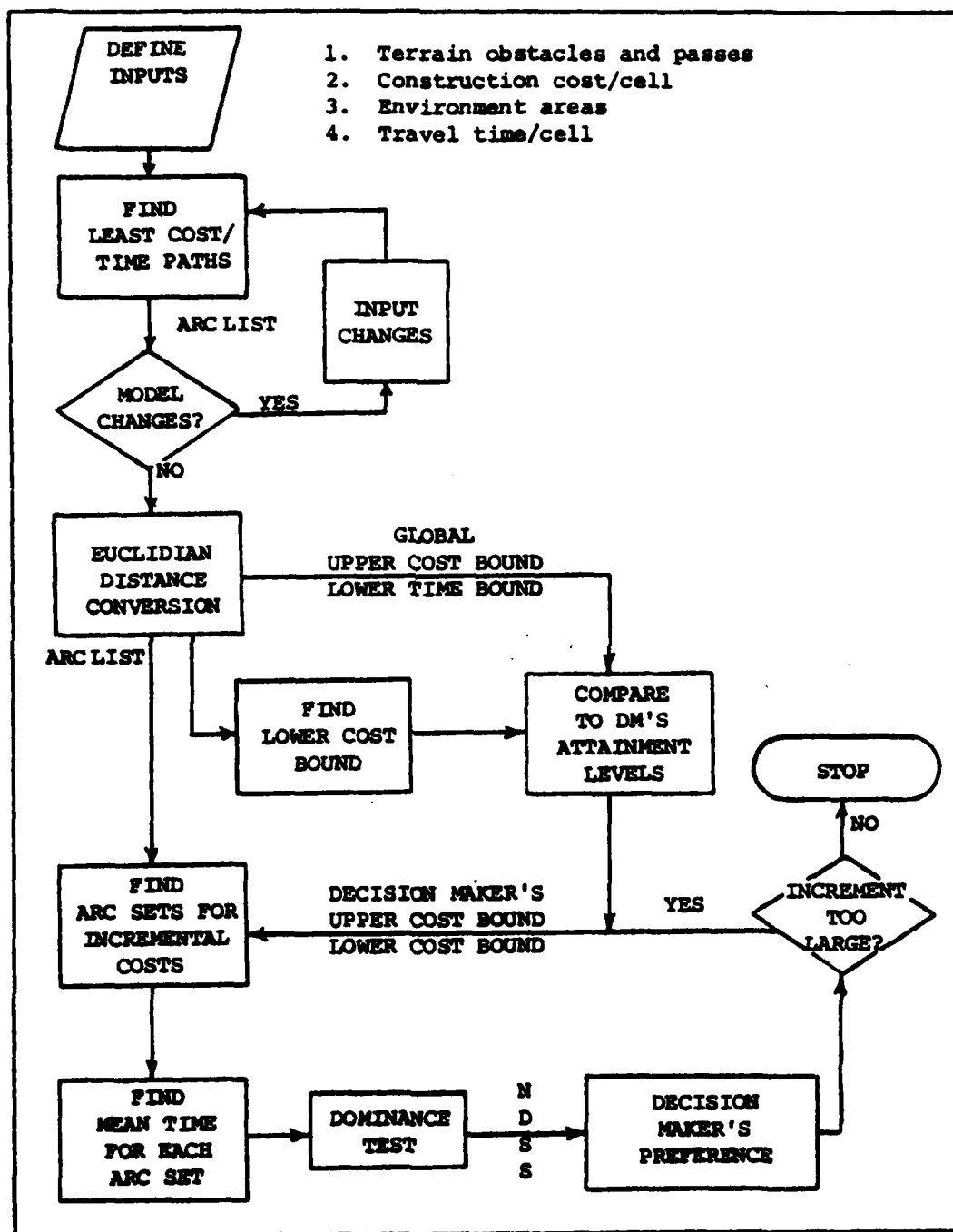


Fig. 26. General Methodology Flowchart

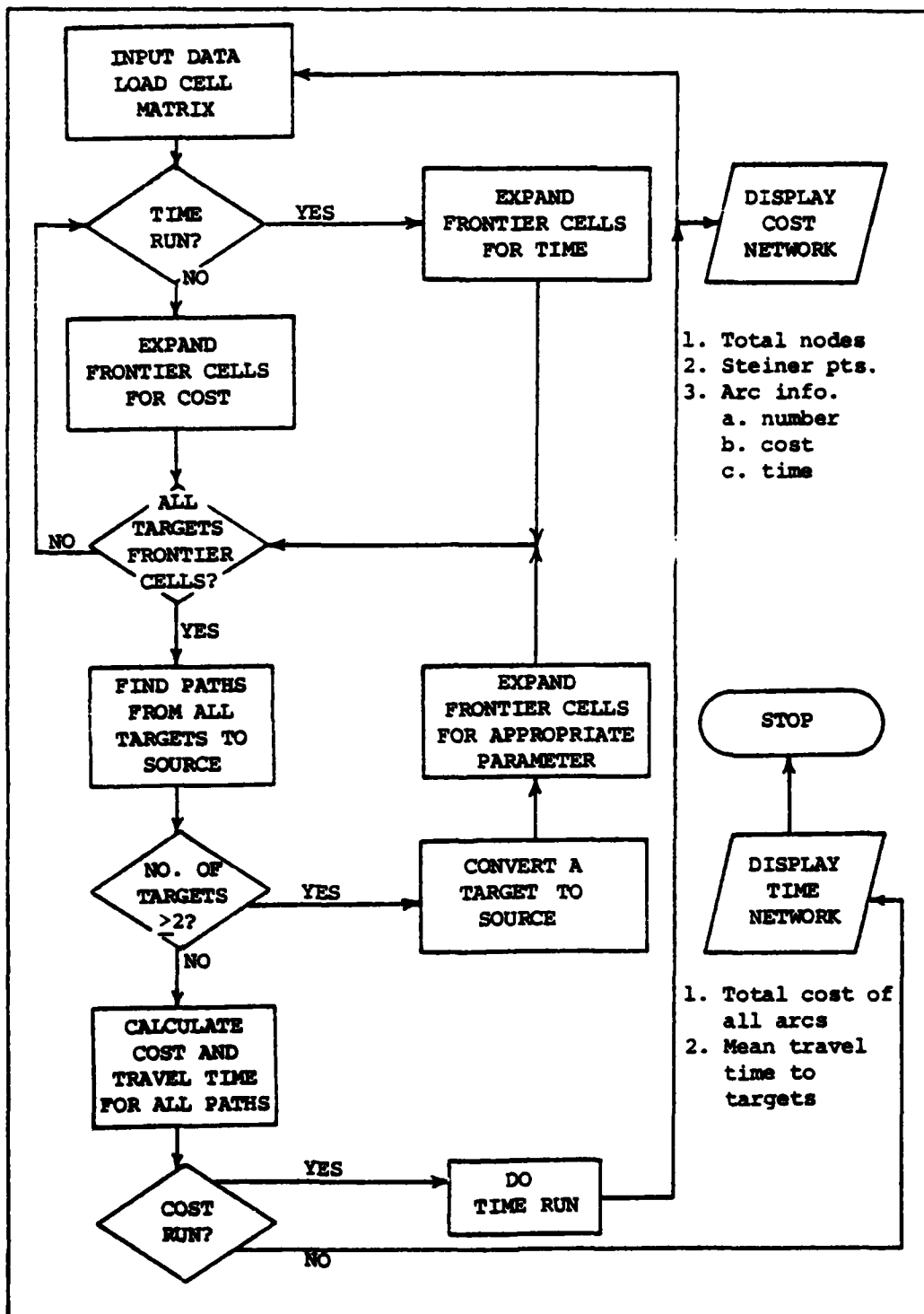


Fig. 27. Least Cost/Time Path Algorithm Flowchart

the user must first grid the map of the area in question with cells of equal dimension, determine the cost to build and average time to travel across each cell, non-diagonally. The cell size should be small enough to capture the detail of such things like the routing through mountain passes, significant but small obstacles like lakes and environmentally sensitive areas. The example problem shown in Figure 5 illustrated the gridding procedure. Cells are identified by their vertical and horizontal position in the grid.

Referring to the flowchart in Figure 27, the program loads a three-dimensional matrix with the specific data the user supplied. This matrix is called the *cell matrix* since the data describes the characteristics of the cells in the grid. Two dimensions designate the vertical and horizontal positions of the cells. The third dimension is used for storing calculations and various book-keeping operations. All frontier cell expansion calculations are stored in the matrix. The designation of cells as path cells between nodes or of cells as intersection cells (steiner points) are also stored in the cell matrix.

The program executes Lee's algorithm, given the input data. First, it expands the frontier cells. When all the targets have become frontier cells, the expansion stops. The program traces the path back to the source from each target. Because the grid structure of the area

can be designed so that the source cell is at the bottom center of the grid, a matter of orienting the grid, the program tries to initiate the trace back direction from the target to the source; first, from the top to the bottom of the grid; second, right to left; third, left to right; and fourth, from the bottom to the top of the grid. Once the program starts the trace back in a direction, it tries to continue that direction. Referring to Lee's original work (Ref 13:346-353), the random selection of which cell to first check as a possible cell in the path back to the source from the target does not change the algorithm's solution for a least cost path. Consequently, the incorporation of these rules into the trace back procedure, because of the lower center position of the source in the grid, tends to speed this process of finding the paths back from all targets to the source. During subsequent cost iterations of the algorithm, when the targets successively become sources, these decision rules do not appear to significantly decrease the computer run time of the algorithm. This is because when the targets become sources for subsequent cost iterations, they are not oriented in the bottom center of the grid. Additionally, during the program's solution for the best time paths, the decision rules may cause possible redundant arcs. The procedure for the user to delete the redundant arcs will be discussed later in this section.

After trace back, all path and intersection cells are stored in the cell matrix. When finding all least cost paths, the frontier cell expansion and trace back procedures are accomplished $N-1$ times, where N is the total number of source and target nodes. Because the source of the previous run is deleted and a target becomes the source for the next run, the amount of time to do the trace back procedure decreases with each cost iteration. After all the cost iterations, the cells representing the various cell types (the path, intersection, target, and the source) have been indicated (flagged) in the cell matrix.

With the cell types in the cell matrix, the program calculates the cost to build each arc between all nodes (target, source, and intersection) and the time to travel on each arc. This information is stored external to the program on tape or disc storage.

At the end of the $N-1$ cost iterations and arc calculations, the program reinitializes the cell matrix with the input data to prepare for program execution to find the minimum time to travel paths from the source to each target. According to the methodology, the program executes Lee's algorithm to find these paths using the average time to travel across each cell that the user input before starting the program. The total cost of the arcs associated with the minimum time to travel and the mean time of

these arcs or paths to each target are stored external to the program on tape or disc.

The user may select graphic display (terminal or paper output) of the input grid and of the cost and time networks found by the program. Because the program finds the best time to travel paths to each target from the source and, as mentioned earlier, because of the program decision rules for finding the path back from each target to the source, the time network may include redundant arcs. Referring to Figure 28, the circled part of the time arcs connecting the two targets to the source may be redundant. If they are redundant, then the cost of this network may not be the lowest for which the minimum time to travel network can be built. Since this time network also establishes an upper cost bound for later use in the methodology, the user should determine if elimination of these possible redundant arcs will lower the cost of the time network, while not changing the value of the minimum time to travel to each target. The user must make this determination.

To analyze this situation, the program directs the user to insert a dummy obstacle in the path of one of the redundant arcs to force the time network to use only one of the arcs. Figure 29 shows the resulting network caused by inserting a dummy obstacle in the path of the left arc of Figure 28. If the resulting minimum time between each target and the source in the two networks, Figures 28 and 29,

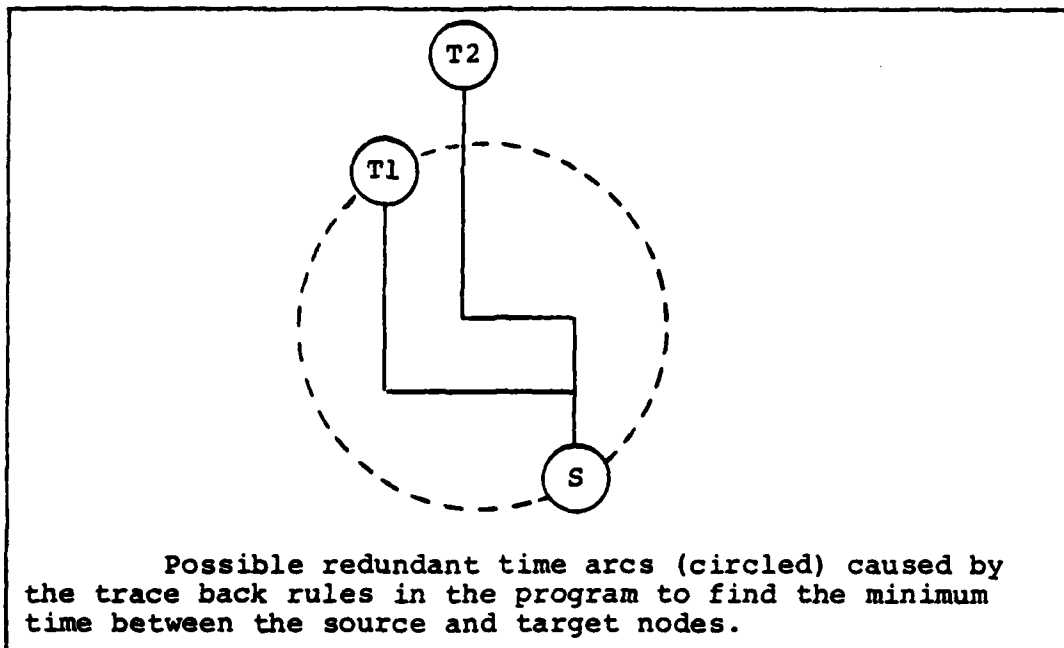


Fig. 28. Possible Redundant Arc of Time Run

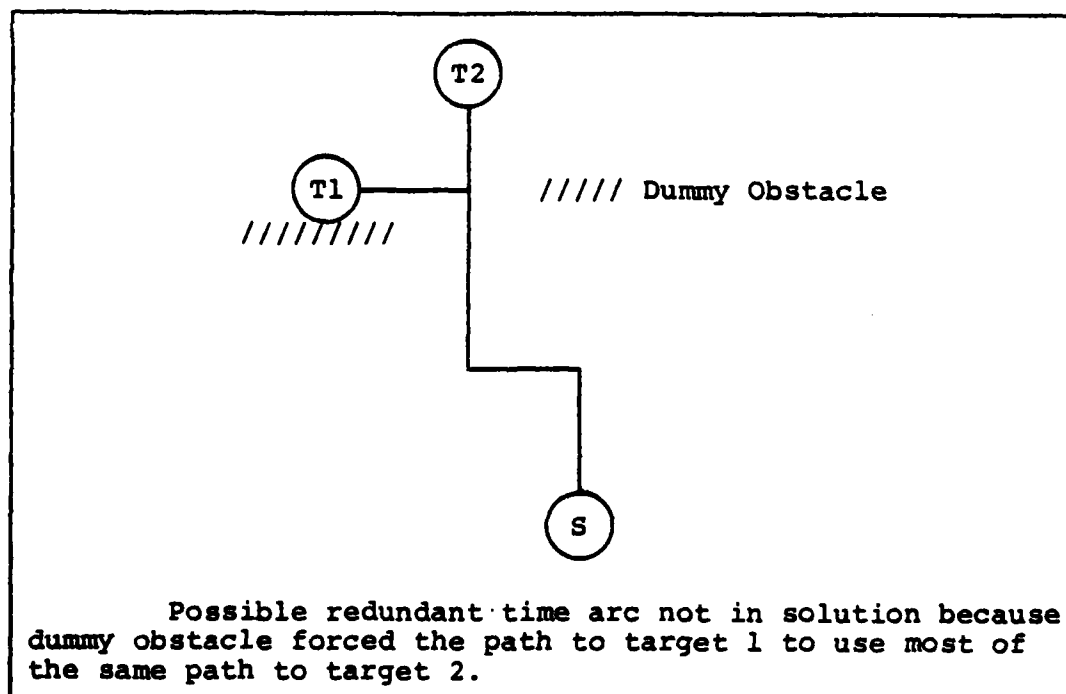


Fig. 29. Redundant Arc Deleted

are the same, then the network with lowest total cost to build the paths establishes the upper cost bound for later use in the methodology. Additionally, the feature of graphically depicting the networks and listing the network data (cost to build the arcs, time to travel on the arcs, and the nodes that the arcs connect) allows the user to reference this information during the remainder of the methodology.

Since Lee's algorithm finds rectilinear paths, the user may use the graphic display and grid cost and time information to determine if certain arcs may be represented by diagonal or euclidean distances. Since the arc information is stored external to the program, these changes can be made on the stored output without affecting the methodology. For example, in Figure 30, the rectilinear arc (solid line) from the target to the source may be improved (less cost and less time to travel between the two points) by the euclidean arc (dashed line). In the example, the total cost of the solid arc is 20 with time to travel of 20. The total cost of the dashed arc is 15 with time to travel of 15. To determine the cost and time of the euclidean arc, the user must consider the cost and time associated with crossing the land area diagonally. The user would then change the value of the cost and time of the arc in the external storage from 20 units to 15 units.

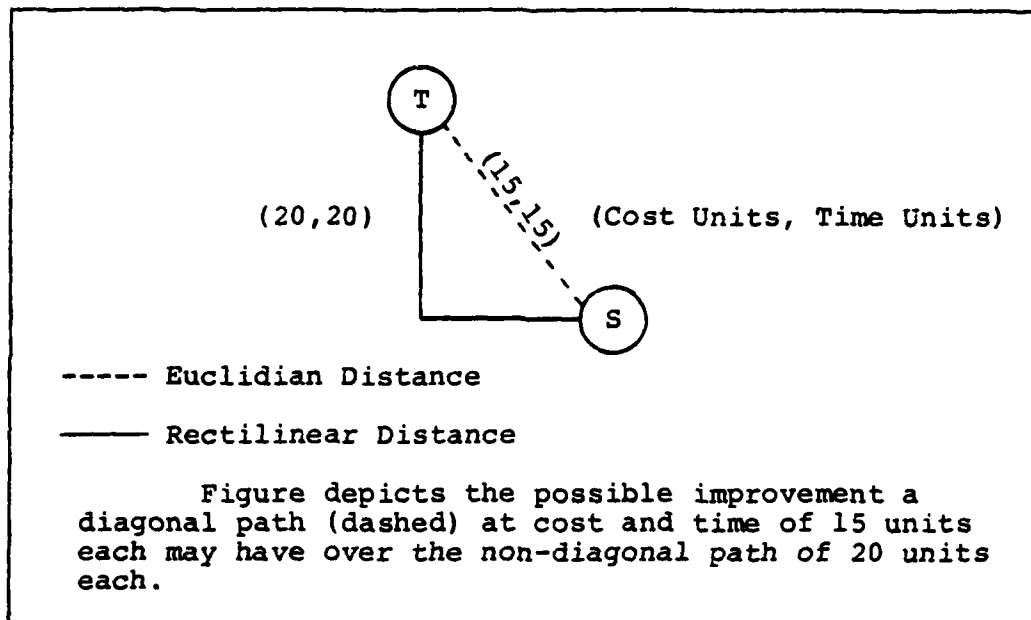


Fig. 30. Illustration of Improvement by Diagonal Path

At this point, the computer program module that executes Lee's algorithm has given the following information on external storage for the remainder of the methodology's algorithms and methods to use:

1. The total number of nodes in the network of least cost paths.
2. The total number of steiner points in the network.
3. The cost to build and time to travel on the arcs between the nodes in the network.
4. The minimum mean time possible between the source and targets.

Once the user is satisfied with the output data from this program, this output becomes the input to the next program which finds the lower bound on the cost to build the network to connect the source and the targets. This is done by finding the minimum steiner tree.

Finding the Minimum Steiner Tree

The minimum steiner tree (MST) is a network that spans all nodes for the least cost, adding steiner points to decrease cost, if necessary. These nodes are the source and target nodes. The methodology uses the greedy approach of Prim's method, as presented by Horowitz and Sahni, to evaluate the network created by the previous program to find a minimum steiner tree (8:176). Other techniques could have been used. The cost associated with the MST will be the lower cost bound of all possible networks that connect the source to each target. The methodology's adaptation of Prim's method builds this tree starting with an arbitrary least cost arc in the network. The selection rule for the least cost arc considers all arcs in the network including those connected to steiner points. Arcs are added to the tree by spanning the nodes of the starting arc to its nearest, neighbor node. The nearest neighbor node is defined as the node which can be connected to the existing tree with the least cost. As nodes are added to the tree, the spanning process continues to add unconnected

nodes until all nodes are included in the tree. Since all connections were made at the least cost, this constitutes the minimum cost tree to connect all nodes, or a minimum spanning tree. For an example of this process, one considers a modified case of the example problem of the last chapter that creates more redundant arcs. The modified example is illustrated in Figure 31.

Using the costs annotated, the adapted Prim method starts with the arc between steiner points 7 and 8 and spans outward to each least cost neighbor. The results are shown in Figure 32. In Figure 32, arcs are annotated by cost and the order they are added; i.e., (cost=3, order=4). This network is the minimum spanning tree of all nodes and steiner points, but not necessarily the minimum steiner tree of the nodes.

Steiner point 6 has only one arc adjacent to the tree and adds unnecessary cost to the MST. Steiner points of this type are defined in the methodology as *steiner leaves*. Also unnecessary to the MST solution are all redundant paths between any two nodes. These are not as obvious or as easy to remedy. The addition of steiner points was meant to decrease cost. If a node can be connected to the MST by a less cost set of steiner points, any higher cost paths are redundant. This means all unrequired steiner points must be removed. The program checks for redundant arcs by removing each steiner point

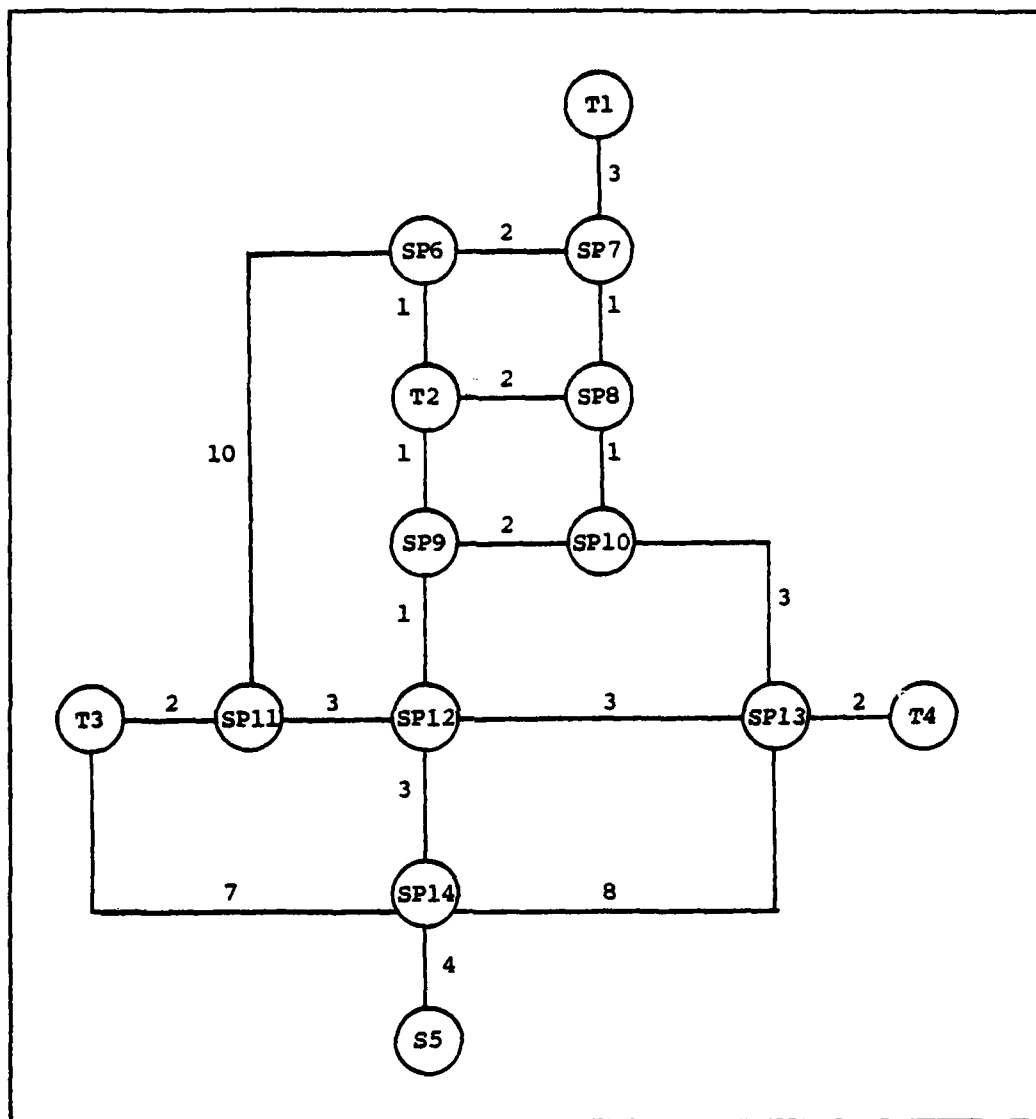


Fig. 31. A 5 Node/9 Steiner Point Sample Network

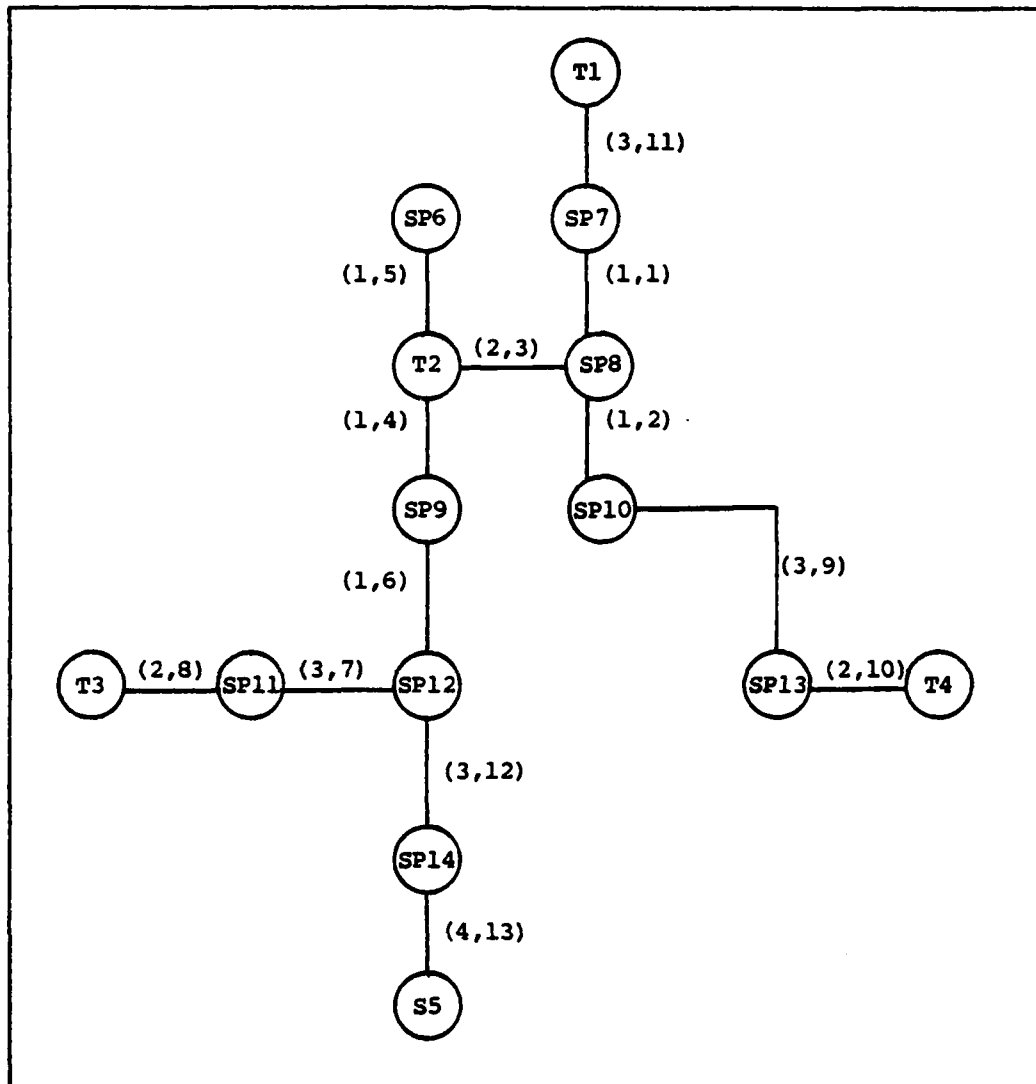


Fig. 32. Minimum Spanning Tree of All Nodes and Steiner Points

in turn and recomputes the minimum spanning tree. Each new tree must be checked for steiner leaves and that all nodes are spanned in case deletion of the steiner node divides the original tree into two networks not connected to each other or a forest. Examples of each of these cases can be viewed by an exercise of this procedure on the example network:

1. Remove steiner leaf 6 (Figure 33).
2. Removal of the next sequential steiner point 7 results in less cost, 17, but it creates a forest with target 1 disconnected (Figure 33). Steiner point 7 is put back into the network.
3. Removal of steiner point 8 would still create a forest.
4. Removal of steiner point 9 results in a tree that spans all nodes without new steiner leaves, but does not reduce the minimum cost (Figure 34). Steiner point 9 is replaced into the network.
5. Removal of steiner point 10 still satisfies both conditional tests of no new steiner leaves and all nodes spanned and does so at a less cost, 25. Therefore, steiner point 10 is deleted from the network (Figure 35). This is the best solution, as removal of steiner points 11, 13, and 14 create forests, and the removal of steiner point 12 increases cost.

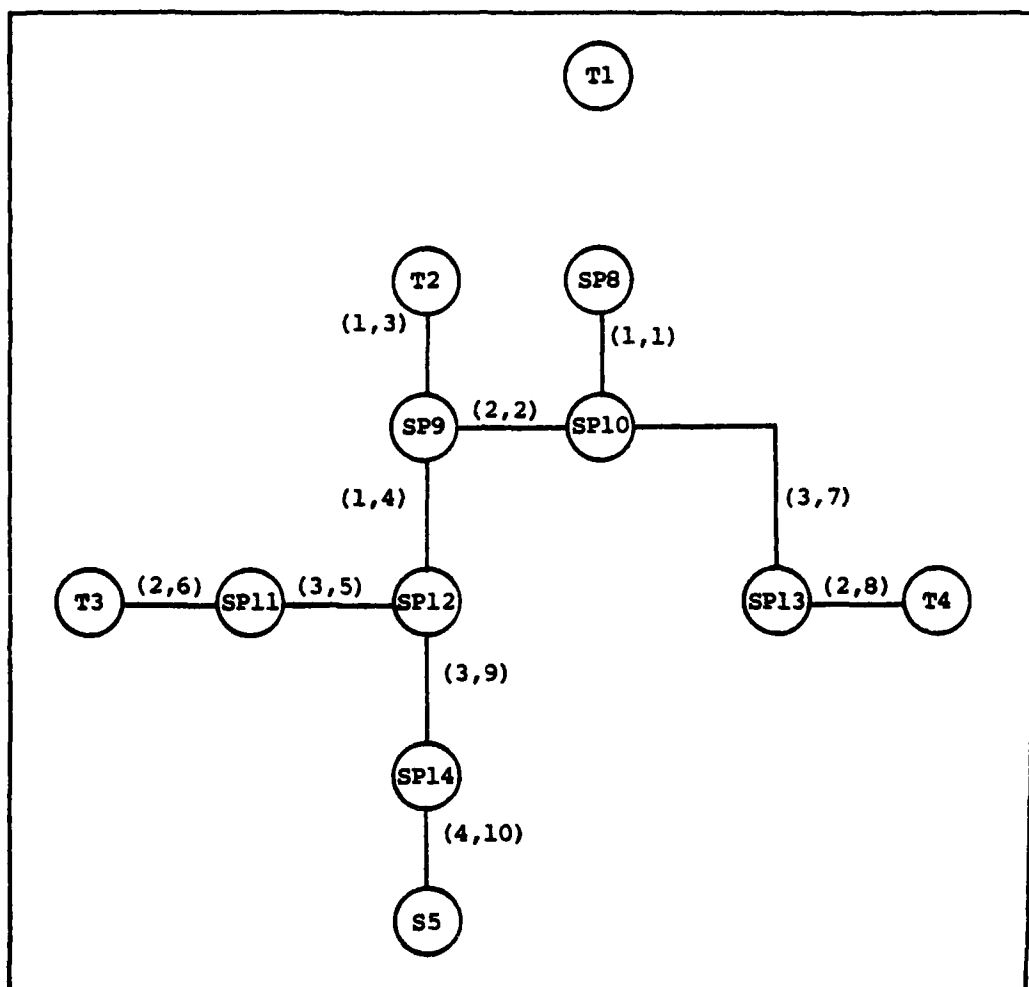


Fig. 33. Minimum Spanning Tree with Steiner Points 6 and 7 Removed

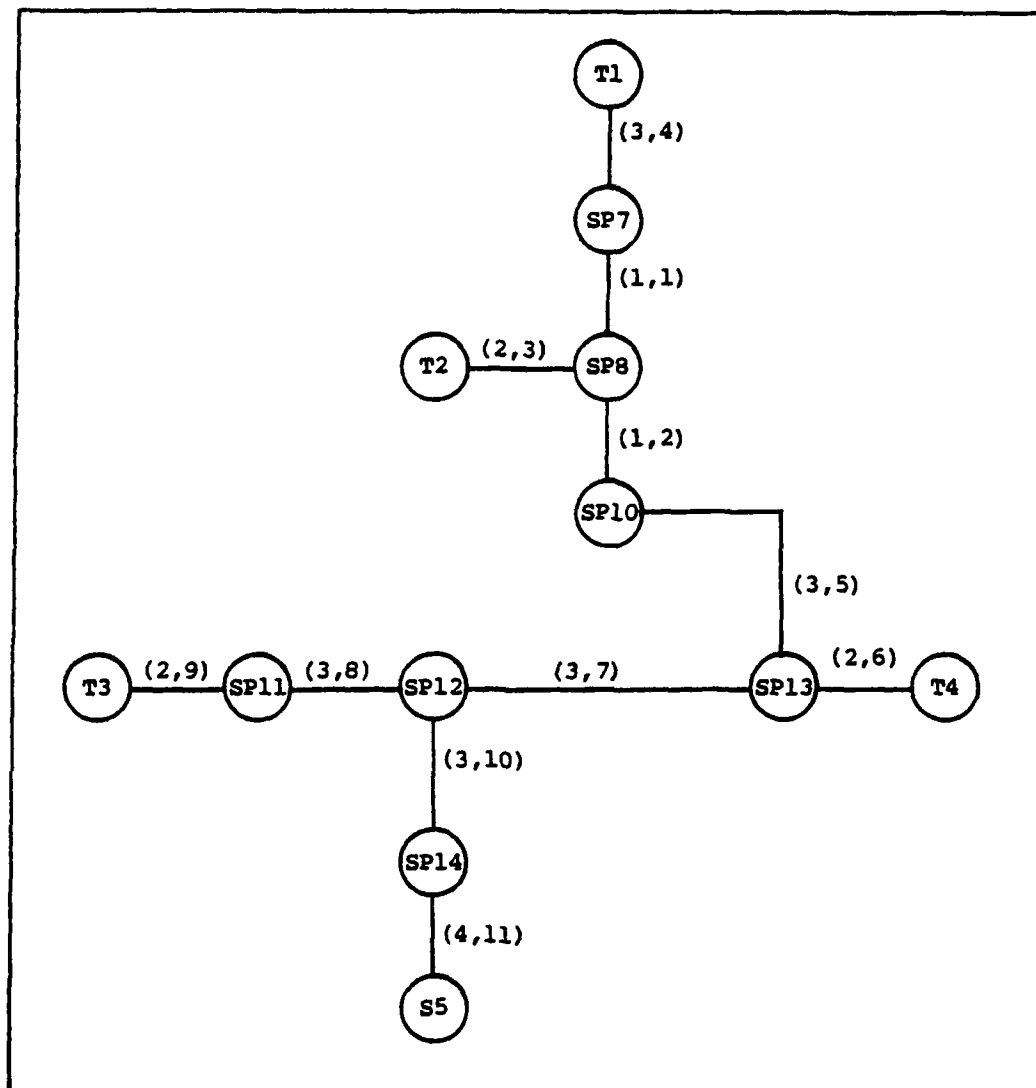


Fig. 34. Minimum Spanning Tree with Steiner Points 6 and 9 Removed

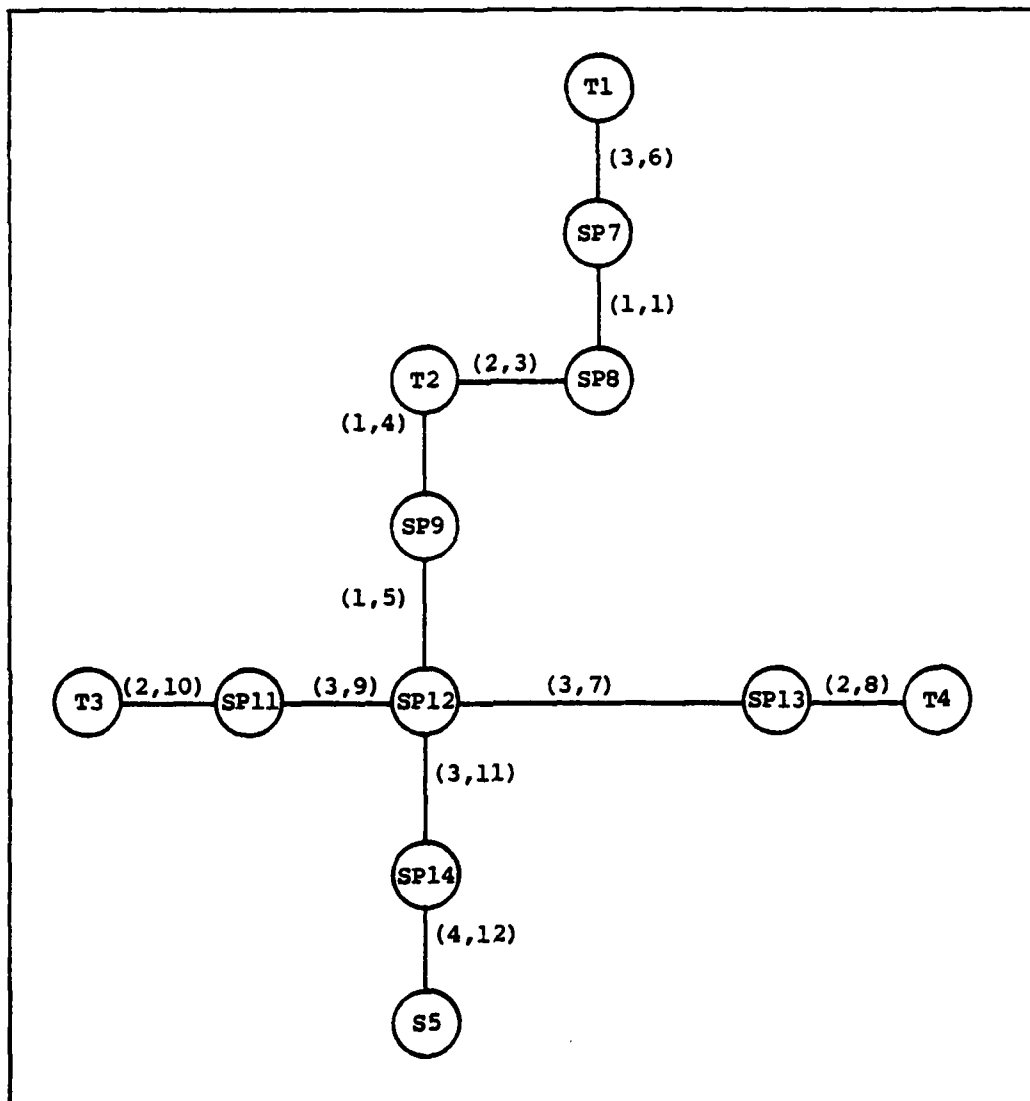


Fig. 35. Minimum Spanning Tree with Steiner Points 6 and 10 Removed

Each evaluation with steiner points deleted uses all arcs in the network except those connecting previously deleted nodes. The cost of the minimum steiner tree is the lower cost bound needed to calculate the incremental cost for the desired cost vectors, as defined below. This cost is stored on tape or disc for use in the next program of the methodology which calculates the desired cost vectors.

Construction of Desired Cost Vectors

Cost vectors are those sets of arcs that sum to the desired cost. These cost vectors are found by an exhaustive search of all combinations of arc costs. The Horowitz and Sahni algorithm, SUMOFSUBS, accomplishes this task. Since SUMOFSUBS requires arc cost to be in non-decreasing order, the arc list from Lee's algorithm must be sorted.

Since the algorithm chosen to sort the arc costs will only sort to increasing order, then arcs of equal cost need to be differentiated. Arcs of equal cost are differentiated by converting the arc costs to eight digit numbers with the arc cost in the first three digits and the sequence or index number in the last two digits; i.e., if arc 1 has cost 4, the arc cost equals 00400001. The index number is considered an increase in the cost of the arc, but at a level insignificant to the solution. Sorting

is accomplished by QUICKSORT, another Horowitz and Sahni algorithm (8:126). QUICKSORT was made an integral part of this module to make this part of the methodology self-contained.

After the arc list is sorted, this program module increments the cost range between the upper and lower cost bounds into ten or less desired costs. The size of the increment, at this point, limits the search. However, as described in the next chapter, the user may come back to this point in the methodology to search smaller increments. The increment between desired costs is the nearest integer to one-tenth of the cost range. Given the sorted arc list and the desired cost, the program does a binary search of arcs using depth first branch and bound techniques. The search starts with the left child (the left branch of the binary tree) and branches according to four exclusive decision rules:

1. If the sum of all included arcs, SS , and the next arc, $W(k)$, equals the desired cost, record the subset. $W(k)$ is the cost of arc k .
2. If adding the next arc does not yield the desired cost and if $SS + W(k) + W(k+1)$ is less than or equal to desired cost, then one continues search on left child. The procedure returns to step 1.
3. Failing 1 and 2, then

a. if $SS - W(k)$ plus the sum of the cost of all remaining arcs, RR , is equal to or greater than the desired cost; and

b. if $SS + W(k+1)$ is less than or equal to the desired cost, one continues search on the right child.

4. Failing all above, one backtracks to last added left child and continues searching on the right child.

For an example of this process, one can consider a network of a source, two targets, and one steiner point. Figure 36 shows a graph of the network and the sorted arc list.

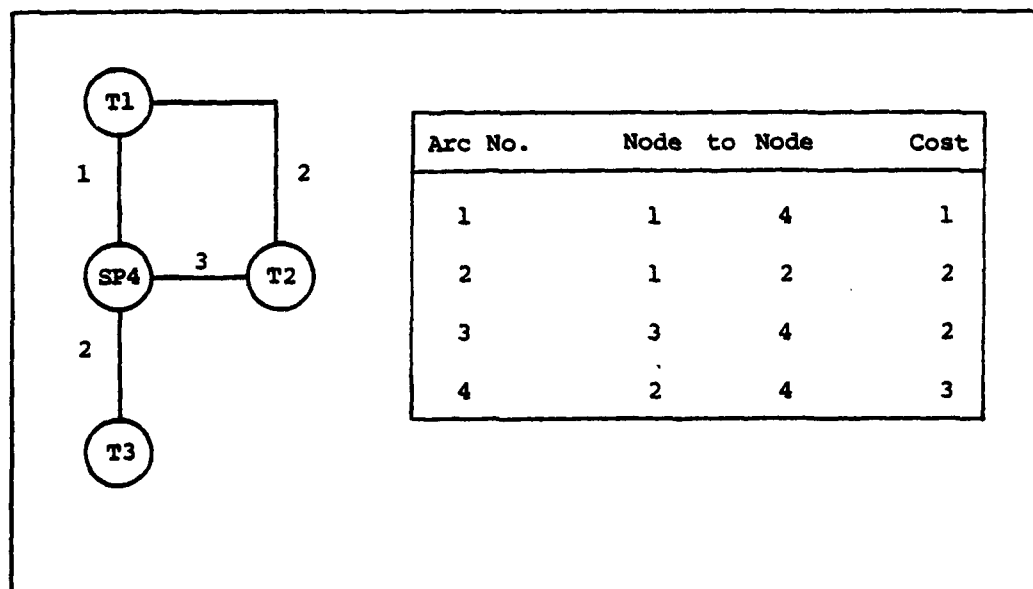


Fig. 36. 3 Node/1 Steiner Point Network and Arc List

Figure 37 is the portion of the binary tree searched by this program to find a desired cost of 6. Each level of the tree represents one arc. The nodes of the tree are annotated with the sum of all arcs included (left children) on that path. The tree arcs are numbered by the order in which they were included.

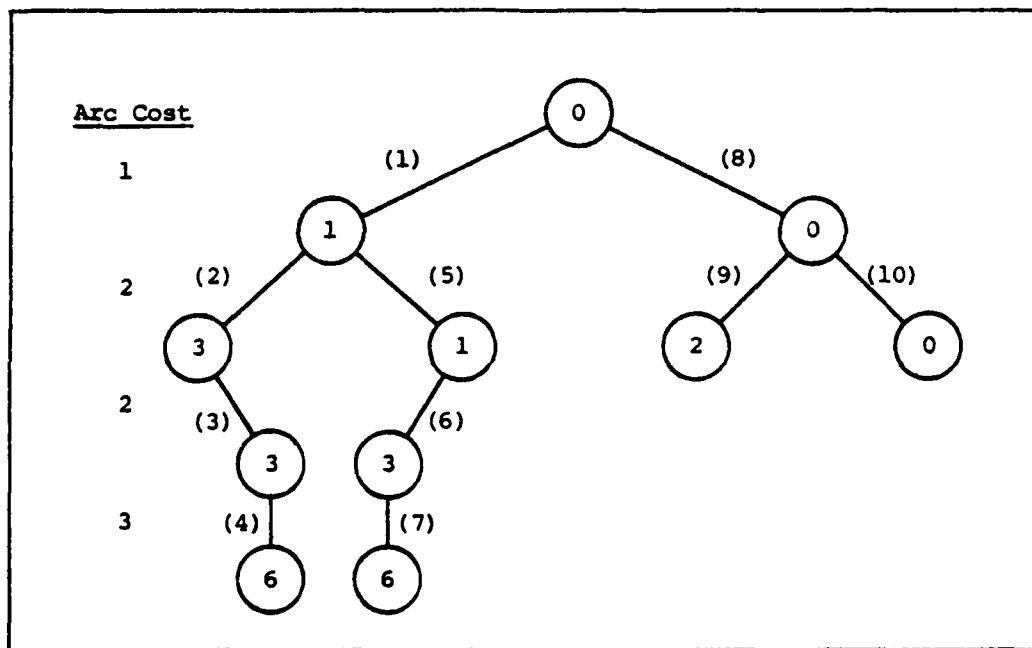


Fig. 37. Binary Search for Desired Cost 6

The output of this part of the program can be thought of as two binary vectors, 1101 and 1011, for the desired cost, representing the arc sets (1, 2, and 4) and (1, 3, and 4). Decisions are made at every node except the bottom, leaf nodes in the binary tree. As a simple measure of efficiency, the program used 10 arcs of the

tree to identify the two solutions. In a more formal sense, each decision point represents a subset evaluated. Therefore, 9 out of 15 total subsets were needed for the program module to find these subsets for a desired cost. The program uses two modifications to improve this efficiency.

The first modification could apply to all binary depth first searches, not just as used in this network evaluation. It was observed that as the desired cost increased, branch and bound fathoming (the process of excluding that node and all subsequent children from all further consideration, because considering them will not yield a better answer) occurred deeper in the tree. Consequently, more subsets were evaluated. If the desired cost is higher than the average network value, a search for the cost of the difference between the desired cost and the total network value would yield reciprocal binary vectors, 0010 and 0100, representing the complementary arc sets. The reciprocal desired cost is 2, total value (8) - desired cost (6). Since the reciprocal desired cost is small, fathoming occurs sooner. Figure 38 is the portion of the binary tree used to find a desired cost of 2. As suspected, only 6 subsets were needed to identify the two desired arc sets. The actual improvement factor of using this technique will vary from 0 to 50 percent, depending on how far the desired cost is from the average cost value of the network being evaluated. For the four target

AD-A115 695

AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL--ETC F/8 12/2
APPLICATION OF NETWORK AND DECISION THEORY TO ROUTING PROBLEMS. (U)

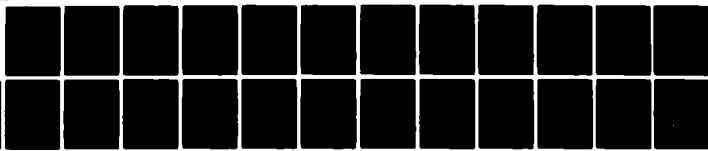
MAR 82 R J BAKER, J E CARTER

AFIT/OST/OS/82M-3

UNCLASSIFIED

NL

2nd
Page



END

DATE

FILED

7-82

DTIC

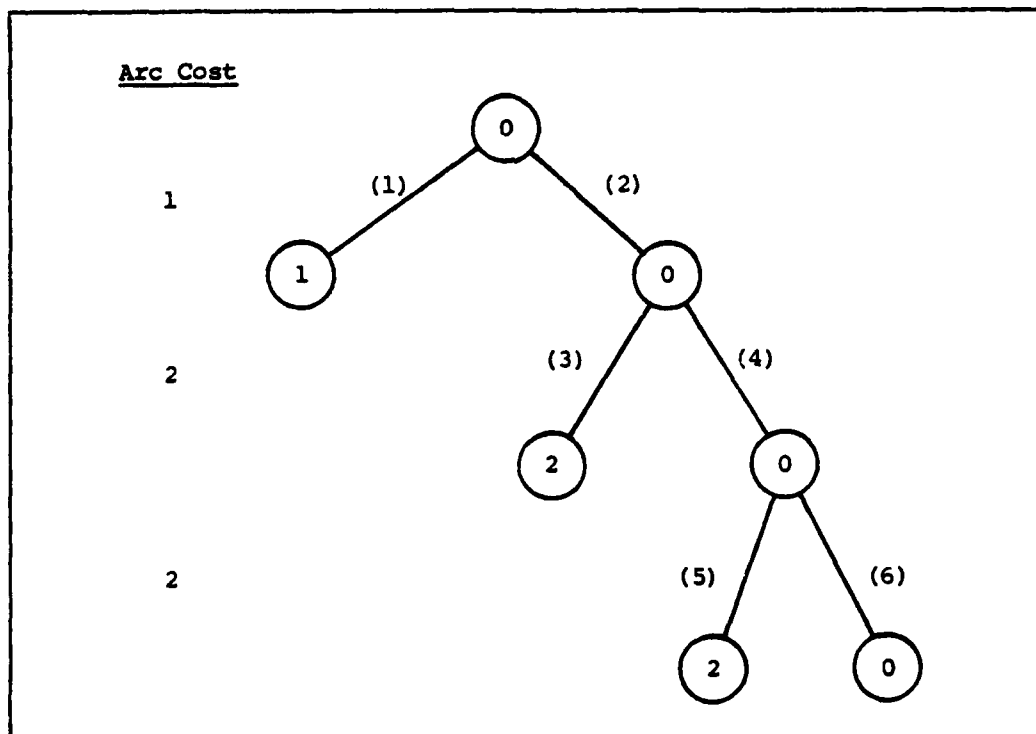


Fig. 38. Binary Search for Reciprocal Desired Cost 2

example problem used in the "Methodology" chapter, the computation time improved 35 percent.

The second modification used by this program module applies only to the network arc search problem. In Figure 36, any subset of arcs that excluded arc 3 would fail to span all nodes, therefore creating a forest of the source node. The modification is simply to always include these arcs in the search. Conversely, one always excludes them when using a reciprocal desired cost. This is achieved in the program by omitting decision rule 2 when searching

for reciprocal costs. In Figure 39 is the resultant binary search tree for desired cost 2.

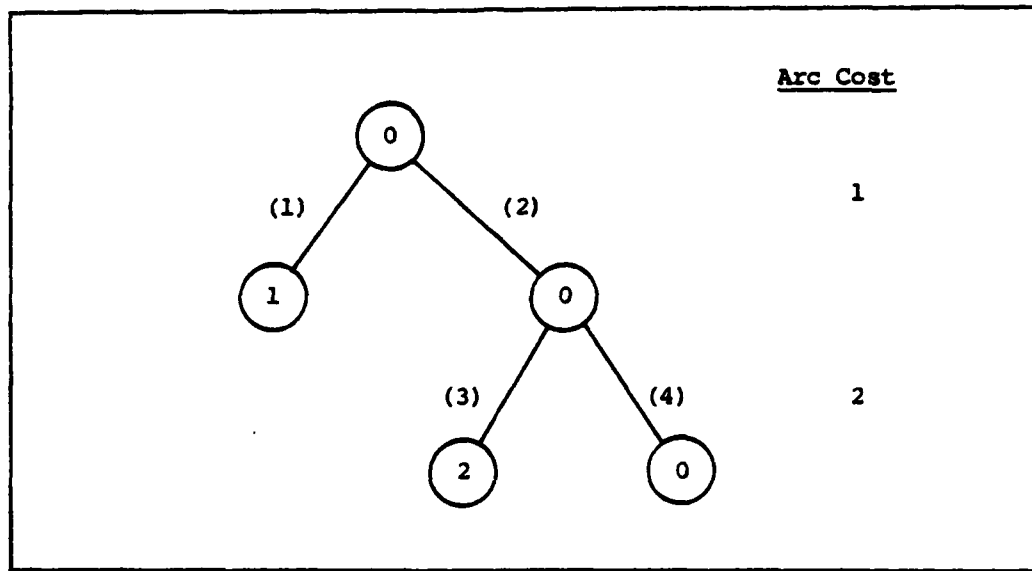


Fig. 39. Binary Search for Reciprocal Desired Cost 2 with Target Leaf Excluded

The program module is modified to apply this technique properly even if reciprocal costs are not advantageous. Again, the improvement factor is completely dependent on the number of nodes which are *target leaves*. For the four target example, all targets and the source were target leaves so the improvement was dramatic. The computation time was decreased by an additional 64 percent for a total computational improvement of 76.5 percent.

As with the minimum steiner tree, a feasible arc set should not contain steiner leaves and must span all

nodes. These conditions are put on each arc set before it is included in the desired cost vector. If only steiner leaves exist, the arc set may be computed at a lower desired cost. The conditions are identified in the same manner as during MST computation. Even when both tests are passed, the arc set may still define a forest with all nodes spanned. If this forest splits the node set, the arc set will be eliminated when the cost vector is optimized for time in the next program module of the methodology, as having an infinite time path to any forested node. The next program module completes the methodology. It searches for the minimum time to travel from the source to each target for all the subsets from this program module.

Finding Shortest Time Paths

The last program module's output data now becomes the input data to the program that executes the Bellman-Ford method to find the shortest time paths from the source node to all other nodes.

Referring to the flowchart in Figure 40, each of the subset networks at a desired cost is evaluated. For each subset, the value of the shortest time path from the source to each target is found. Then those values are added and divided by the total number of target nodes to find the mean time to travel between the source and the

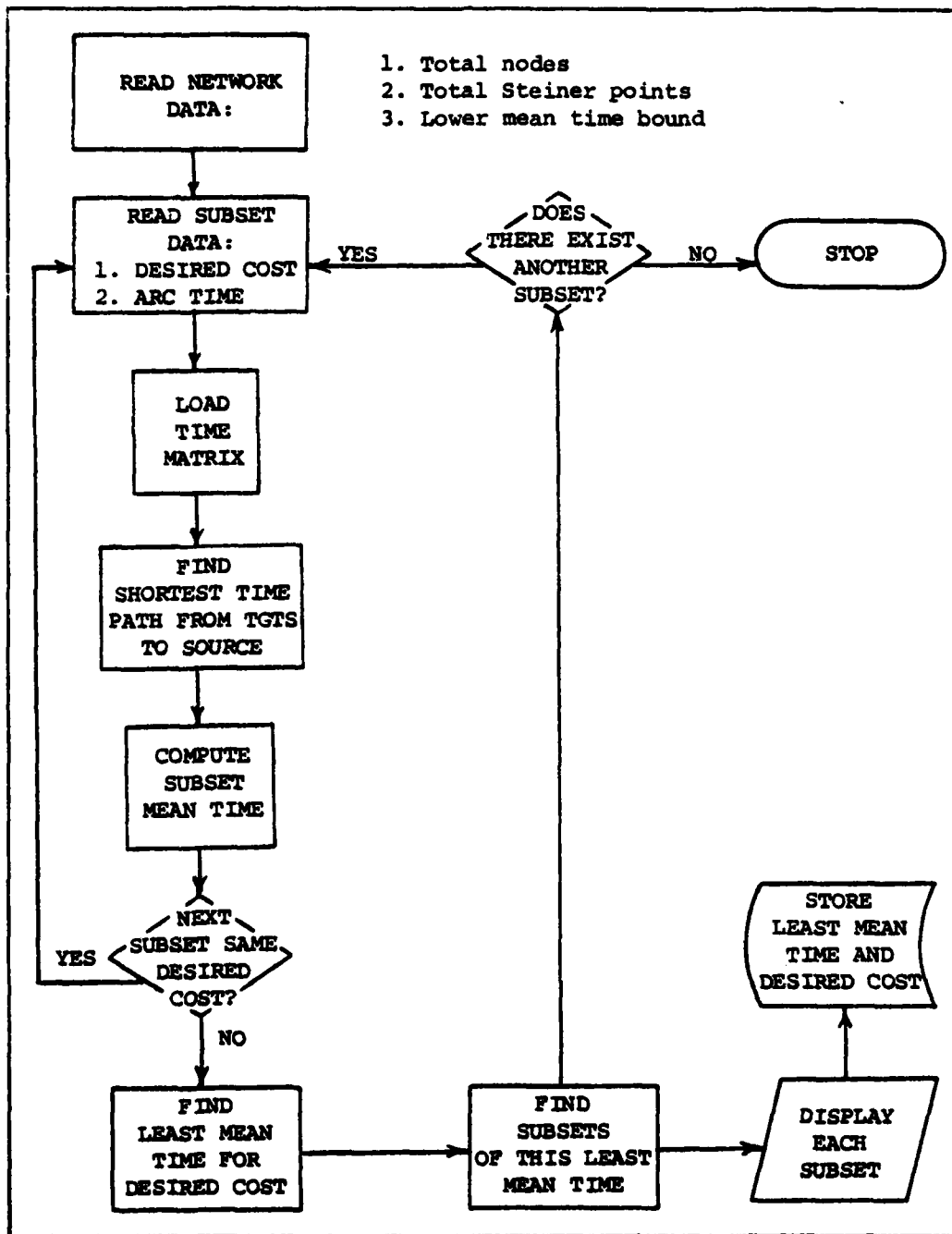


Fig. 40. Minimum Mean Time for Desired Cost Subset Flowchart

targets. The subsets that display the minimum of these mean times are listed (terminal or paper output). Additionally, the desired cost of the subsets being evaluated and the minimum mean time to travel associated with the subsets are externally stored on tape or disc.

The program then repeats the process for the subsets associated with the next desired cost. The externally stored output of the desired costs evaluated and their associated minimum mean time allows the user to determine the non-dominated solution set (NDSS) of minimum cost and minimum time networks. The visual output data consists of the minimum mean times of the networks associated with the desired costs evaluated and the associated networks' arc cost, time, and node information. This information allows the user to graphically reconstruct the networks that may be selected for construction based on the NDSS of networks.

With the completion of this last program module, the analyst and the decision maker now have information that they can use to understand the network of nodes and possible ways to connect the nodes. The user must first determine the relevant cost, time, and geographic information to start the process. User interaction between the program modules should help in defining the network system. Output from each program module gives the user the necessary supporting documentation to analyze the system and

understand the operation of the programs. This information can now be used in the decision-making process, which will be the topic of the next chapter.

IV. Use of Results

The application of the methodology developed in this thesis to the decision-making process requires a strong interface with the appropriate decision maker. The decision maker's inputs are used to bound and re-bound the solution space until a single preferred solution or at most a set of solutions remain to which the decision maker is indifferent.

The first bounds are established by combining the results of Lee's and Prim's algorithms in the methodology. These parts of the methodology identify the global maximum and minimum costs for feasible road networks. These will be referred to as *maxcost* and *mincost*, respectively. The decision maker is made aware of the bounds and asked if they are within the limits of acceptable range of cost or "attainment levels." Assuming the option to not construct the network has been previously eliminated from the decision maker's options, the mincost value will become the decision maker's lowest attainment option. However, the upper attainment level may be selected well below the maxcost or best operational option value. The strict preference of operations is to have the network with the least mean time to travel between the source and end points,

which is also the more costly network option to build. Regardless, these new bounds will be used to bracket the incremental desired costs needed in the methodology to find all possible subsets of networks that equal those incremental desired costs. These increments, proportional to the difference between the cost bounds, are controlled only by the bounds selected. A decision must be made later if the increments should be reduced, but only after the methodology has given the information that is necessary for creation of the efficient frontier of non-dominated solutions. This non-dominated solution set is the decision maker's primary decision tool.

Selection from among this final set of solutions can be accomplished by three basic methods.

Referencing Figure 41, the first method for selection between the four points would be to choose the solution with the most "intuitive appeal" to the decision maker. This simple direct approach implies the decision maker is well aware of self-preference between the attributes of cost and time. Lacking this, one must analytically determine the decision maker's preferred choice.

The following illustrates another method of finding the decision maker's choice. If all solutions have acceptable cost, one equitable selection procedure may be to compare the improvement in time to the increase in cost from the least cost option. This asks the question,

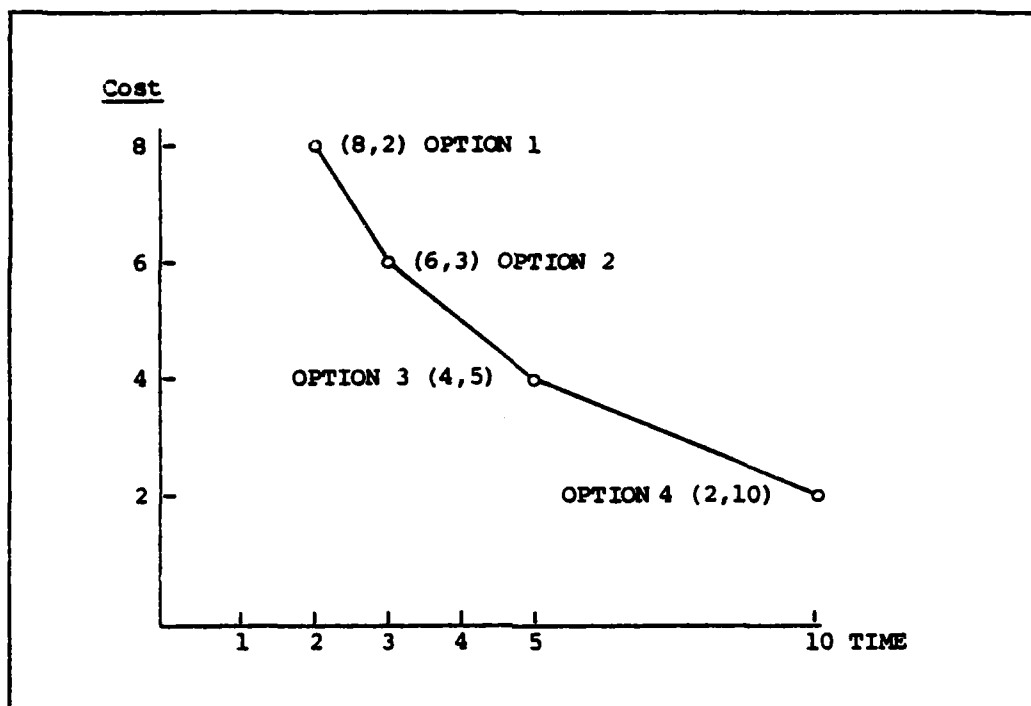


Fig. 41. Non-Dominated Solution Set (NDSS)

"What is the marginal rate of return for each option?"

In the above example, option 3 is selected with a marginal return of 2.5, as shown in Table 6.

TABLE 6
PREFERENCE BY MARGINAL RETURN

Option	Change in Time/Change in Cost Ratio
1	1.33
2	1.75
3	2.50

Third, a more general approach to building a preference structure is to establish the value the decision maker places on time and cost, construct value functions for each, and then score each option against a weighted joint value function. The solicitation process for this procedure is well documented (Ref 11:96-11). For the sample value functions in Figure 42, the decision maker would be indifferent between options 3 and 4. The attributes are assumed viewed by the decision maker as independent, allowing for an additive joint value function.

This indifference structure could be established independently without forming value functions by querying the decision maker for indifference tradeoffs for each option. Assuming the decision maker's actual value function is a monotonically increasing function, where less test and less time are preferred to more of each attribute then the option whose difference curve lies closest to the origin has the highest value. Figure 43 shows a set of indifference curves superimposed on the non-dominated solution set.

The indifference tradeoffs may be located outside the feasible solution space. They merely represent the cost-time points the decision maker would willingly trade for a given option. Again, options 3 and 4 fall on the same indifference curve, showing the hypothetical decision maker is consistent.

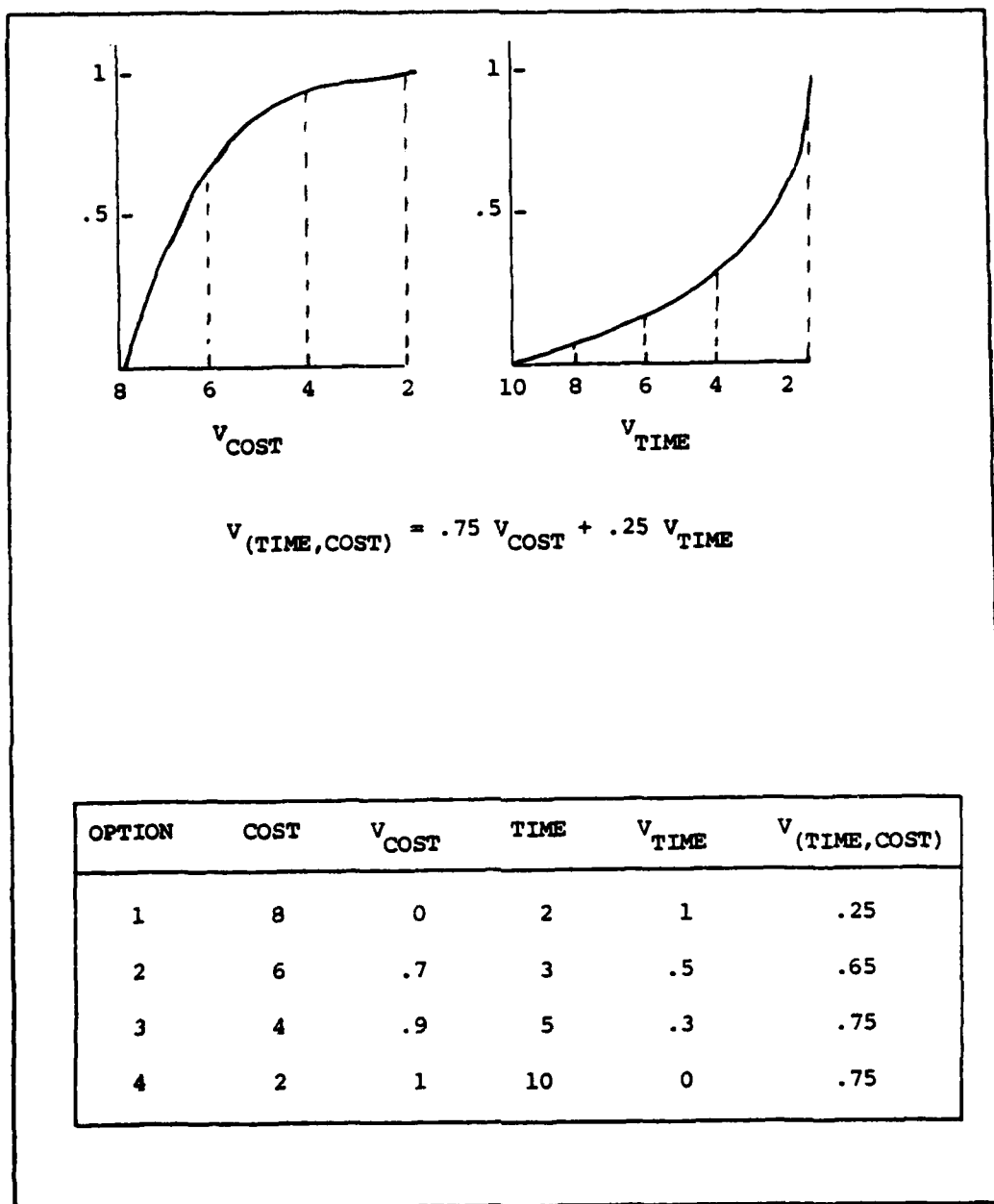


Fig. 42. Value Functions of Time and Cost

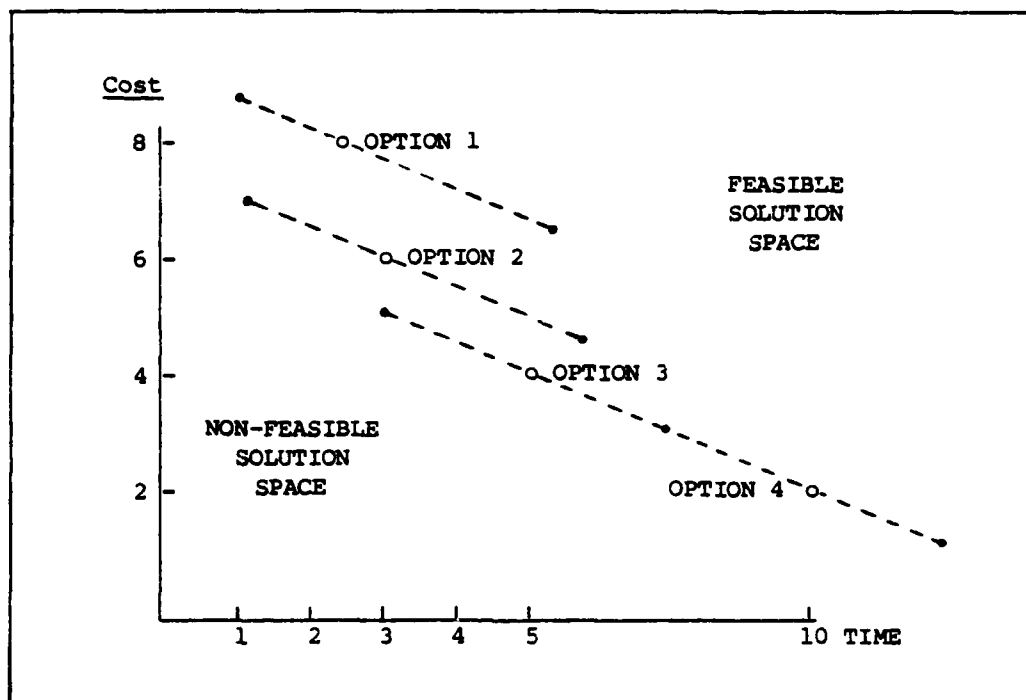


Fig. 43. Indifference Curves

Now that the preferred solution has been selected by any of the mentioned methods, the size of the cost increments can be questioned. If they are larger than the nominal cost of construction through a single cell of the grid structure, then the decision maker must decide if the cost difference between the choice and adjacent options is significant enough to warrant additional investigation. Factors to be questioned for significance would be cost estimate errors and the relative value of adjacent options. If increments are deemed too large, then the analyst would choose the costs of the adjacent options as new upper and

lower bounds and rerun the methodology. This changes the inputs to the methodology where all subset networks associated with the desired cost increments are found. These interactions with the decision maker and the methodology would continue until an acceptable option is chosen by the decision maker.

Since the example NDSS is incremented by two cost units, this increment may be too large. The methodology is initiated again with the cost bounds given by options 2 and 4. This will generate single unit incremental costs and possibly a new NDSS. The hypothetical results are shown in Figure 44, with the decision maker's indifference curves plotted.

This sensitivity analysis on incremental size has added an option to the NDSS that is now preferred over both options 3 and 4. The increment cannot be reduced further; therefore, the final solution has been selected.

To summarize, once the final network has been selected by the decision maker for construction, the decision maker has chosen the network that has been determined for the least cost and the least mean time to travel from the source to each end point. The methodology has determined the networks for their best cost and time attributes. The decision maker has selected a network based on preference of the importance of these two attributes considered together.

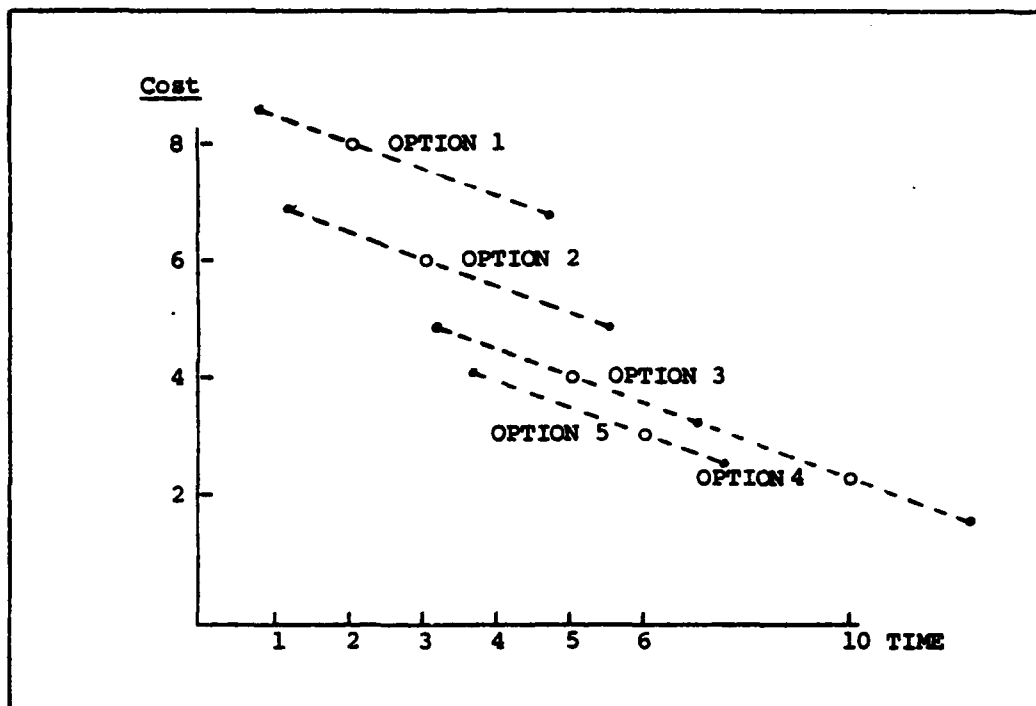


Fig. 44. Reiterated Non-Dominated Solution Set with Indifference Curves

V. Conclusion and Recommendations

The original problem of where to build a suitable road network for the transportation of the MX missile from the Designated Assembly Area to its cluster area in the Nevada-Utah basing plan can now be attempted by the use of the methodology presented in this thesis.

The goal of this thesis has been to find a methodology to provide the decision maker with a set of best possible networks. These networks were found for their best routing in terms of least cost and minimum mean time to travel from the source to each end point. From this set of solutions, the decision maker can choose the road network that best satisfies preferences for cost and time.

The methodology is started by finding the paths of least construction cost and time to travel between the source and each end point. This determination of best paths became a major objective because network solution procedures assume the knowledge of these paths.

From this information, the upper and lower bounds on the cost of networks to connect the source to all end points were determined. The possible cost networks between these two bounds were then sorted for the best time or minimum mean time to travel networks. These

remaining network choices, each found for their best characteristics of cost and time, formed the efficient frontier of possible networks that the decision maker could choose to build. This decision maker's choice is based on the decision maker's preference or tradeoff beliefs for the relative importance of construction cost and mean time to travel.

The methodology has aggregated many operations research algorithms and methods together to provide the decision maker with a choice of best alternatives. This aggregation demonstrates the power and usefulness of the algorithms and methods that have been developed over the past twenty years. Additionally, the use of Multiple Objective Optimization Theory, to find the set of best possible networks for the decision maker's choice, demonstrates the usefulness of this theory that is reviving because of increasing computer capability. The methodology's sequencing of the algorithms and methods is new. The modular use and computerization of the methodology led to its easy verification and implementation. The methodology, although designed as a result of the MX Designated Transportation Network routing problem, has general application to other similar problems.

Recommendations for Use

Although the MX missile's Nevada-Utah basing plan has been suspended, the methodology developed for this thesis has other applications. Of course, if the Nevada-Utah MX missile basing plan is ever revived, the methodology can be used by the planners at the Ballistic Missile Office and Headquarters Strategic Air Command for the routing of the DTN. However, the immediate U.S. requirement to base the MX missile in existing missile launch facilities may find a use for the methodology.

Using the methodology for MX missile deployment in existing missile launchers would serve to identify which existing roads in the missile complex should be upgraded to support the weight of the MX missile and support the increased traffic volume for missile emplacement. The existing missile transportation roads that support the U.S. Intercontinental Ballistic Missile fleet are rectilinear in nature, as they follow the general mile section roads of rural areas. Therefore, the direct application of the rectilinear network the methodology develops is recommended for making road improvement decisions under the current MX missile deployment plan. Other uses may include the massive and costly routing of coal and lumber haul roads or cross country power lines and towers.

The methodology's application to coal strip mining and lumber haul roads to be built in new resource areas

is fairly straightforward. The applications to power line routing would probably not include time. However, the minimization of cost to build the towers and lines and the minimization of power loss may be the objectives.

Certain other applications of the individual parts of the methodology occurred to the researchers. Lee's algorithm to find best path connections may be used in a variety of routing problems. For example, given a battle area of known geography and air defenses, the routing of aircraft missions to avoid the air defenses may be made by weighting the cells of the grid of the area with probability of kill to the aircraft or the amount of exposure time the aircraft has to the defenses. The algorithm should find the path of least hazard to the aircraft.

Certainly, these applications of the methodology and parts of the methodology's algorithms are not all-inclusive. These suggestions should serve to foster more applications of the network and decision theory algorithms and methods on other problems. As a result of the research and use of the algorithms, many questions surfaced that would be good subjects for further research.

Recommendations for Further Research

The recommendations for further research fall into the general category of making the methodology better and

more efficient for the user. Some of the recommendations deal with lifting some of the assumptions.

Lee's algorithm found the least cost and minimum time rectilinear paths. User interaction was required if a straight line path was appropriate. Modifications may be designed to cause the algorithm to find euclidean paths. Jeffery Hoel does mention, in an article on variations of Lee's algorithm (Ref 7:23), that additional path directions may be included using hexagon shaped cells in the grid structure as an approximation to a euclidean solution. The researchers believe that an overlapping octagonal cell structure may provide a closer approximation, but may be very difficult to implement.

The methodology uses Lee's algorithm to find only one least cost or minimum path back from the target to the source. However, many of these paths may exist between the points. Different trace back decision rules or enumeration may allow the finding of all possible least cost and minimum time paths that could be evaluated for the solution set of networks on the efficient frontier. Application of the methodology to all these paths may slightly alter the shape of the efficient frontier. This is because more paths with different weights of construction cost and time to travel will be available for the multiple objective optimization in the methodology.

The methodology uses heuristic decision rules to find the proper subsets of steiner points to construct the minimum steiner tree associated with the least cost paths. The proven optimal solution is a complete enumeration of subsets of $N-2$ or less steiner points, where N is the number of nodes including steiner points (Ref 12:291). Empirically, the rules seem to find the proper solution. One may want to determine whether or not these heuristic decision rules can, in all cases, find the minimum steiner tree.

If good cost information is available, life cycle cost (LCC) rather than acquisition cost and travel time could be used as cell weights in the grid used by Lee's algorithm. The primary difference in this approach is the life cycle cost of a road network is dependent on the network design. In other words, the arcs of each network option could have different LCCs based on a parameter of the number of destinations served by that arc. As more destinations are served by an arc, the operations and maintenance cost of the arc would increase. This requires the use of a dynamic cell weight function that factors in all variables of a LCC estimating equation that is a function of the number of multiple destinations served by that cell. The lower cost bounding of this approach may not be a minimum spanning tree, but a sum of the cost to reach each destination. This could prove to be an exceptionally

efficient solution technique if all measures of operational effectiveness could be quantified into a dollar value and the entire network minimized for cost.

In conclusion, this research effort has married operations research and decision theory techniques into a solution methodology adaptable to a wide range of routing problems with conflicting objectives. As with any research, many questions are left unanswered, but they leave fertile ground for inquisitive practitioners of these sciences.

A conscious effort was made to avoid stating this methodology can find the "optimal" solution. The result from the methodology is merely a reflection of the assumptions and the accuracy of the input parameters to the methodology's analytical, deterministic approach to the solution of a combinatorial optimization problem.

Bibliography

1. Brewer, M. Design Automation of Digital Systems. Vol. 1. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1972.
2. DeWispelare, A. R. "Algorithm Efficiency in Generating Non-Dominated Solution Sets," Proceedings of the IEEE 12th Annual Symposium on System Theory, 218-222 (1980).
3. _____ and A. P. Sage. "Our Combined Multiple Objective Optimization Theory and Multiple Attribute Utility Theory for Evaluation and Choicemaking," Journal of Large Scale Systems, 2: 1-19 (1981).
4. Element Study M-15. "DDA Support Facility Optimization," Document No. D295-10356-1, Model MX. Boeing Company, Seattle, Washington, 22 August 1980.
5. Elliott, Major Mike. Manager MX Siting Group, Air Force Regional Corps of Engineers. Telephone interview. Norton AFB, California, 12 May 1981.
6. "Facility Development Specification for MX Designated Transportation Network," Specification No. 5-M-X-41922A. Martin Marietta Corporation, March 1981.
7. Hoel, J. H. "Some Variations of Lee's Algorithm," IEEE Transactions on Computers, C-25, No. 1: 19-24 (January 1976).
8. Horowitz, E. and S. Sahni. Fundamentals of Computer Algorithms. Potomac, Maryland: Computer Science, Inc., 1976.
9. Jones, Major Grinnel. MX Requirements Officer, Headquarters Strategic Air Command (SAC)/XPQM. Personal interview. Offutt AFB, Nebraska, 26 March 1981.
10. _____ and Major Robert Petrusky. MX Requirements Officers, Headquarters Strategic Air Command (SAC)/XPQM. Telephone interview. Offutt AFB, Nebraska, 21 April 1981.

11. Keeney, R. L. and H. Raiffa. Decisions with Multiple Objectives: Preferences and Value Tradeoffs. New York: John Wiley and Sons, 1976.
12. Lawler, E. L. Combinatorial Optimization: Networks and Matroids. New York: Holt, Rinehart, and Winston, 1976.
13. Lee, C. Y. "An Algorithm for Path Connections and Its Applications," IRE Transactions on Electronic Computers. EC-10: 346-353 (September 1961).
14. Lyle, Dwayne. Systems Analyst, Martin Marietta Corporation. Telephone interview. Denver, Colorado, 22 May 1981.

Vita

Captain Richard J. Baker

Captain Richard J. Baker graduated from Colorado State University in 1972 with a Bachelor of Science degree in the physical sciences. Concurrently, he received an Air Force commission through the ROTC program. His first assignment was as a Missile Launch Officer at Grand Forks AFB, ND. During this tour, he earned a Masters of Business Administration degree from the University of North Dakota through the Air Force Institute of Technology's (AFIT) Minuteman Education Program. In 1976, Captain Baker became a Missile Operations Evaluator with the 3901st Strategic Missile Evaluation Squadron (HQ SAC), Vandenberg AFB, CA. After three years on the missile evaluation team, he became a Development Test Program Manager with the 1st Strategic Aerospace Division (SAC), Vandenberg AFB, CA. During this assignment, he planned and directed SAC's participation in Minuteman and MX missile test and evaluation. In August 1980, he entered the AFIT School of Engineering. Upon graduation in March 1982, Captain Baker was assigned to the Defense Nuclear Agency, Washington, D.C.

Permanent Address: 615 S. Nevada Avenue

Colorado Springs, Colorado 80903

Vita

Major James E. Carter

Major James E. Carter is a graduate of Memphis State University, Memphis, TN, where he received a Bachelor of Science degree in mathematics and an AFROTC commission in January 1967. He graduated from Navigator Bombardier Training at Mather AFB, CA, in January 1969, and was assigned to the 28 Bombardment Wing (Heavy), Ellsworth AFB, SD. Serving in the 28 BMW, he progressed to Senior Standardization Radar Navigator and accumulated 104 combat missions in support of ARCLIGHT, BULLETSHOT and LINEBACKER operations in Southeast Asia. In September 1974, Major Carter was assigned to the 1st Combat Evaluation Group (SAC) (1CEVG), Barksdale AFB, LA. His tour of duty began as an Operations Officer for Detachment 9, Combat Skyspot Site, Osan, Korea. In July 1976, he was assigned as an Olive Branch Route Development Officer, Headquarters 1CEVG, until entering the School of Engineering, Air Force Institute of Technology (AFIT) in August 1980. Major Carter is also a graduate of the University of Southern California, Master of Science in Systems Management extension program, January 1979. Upon graduation from the AFIT School of Engineering in March 1982, he was assigned to the Aircraft Test and Evaluation Division, Headquarters Strategic Air Command, Offutt AFB, NE.

Permanent Address: 2290 Monroe Avenue

Memphis, Tennessee 38104

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/GST/OS/32M-3	2. GOVT ACCESSION NO. AD A115695	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) APPLICATION OF NETWORK AND DECISION THEORY TO ROUTING PROBLEMS		5. TYPE OF REPORT & PERIOD COVERED MS Thesis
7. AUTHOR(s) Richard J. Baker James E. Carter		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT/EN) Wright-Patterson AFB, Ohio 45433		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE March 1982
		13. NUMBER OF PAGES 112
		15. SECURITY CLASS. (of this report) UNCLASSIFIED UNCLASSIFIED
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES Approved for public release; IAW AFR 190-17 4 JUN 1982 FREDMIC C. LYNCH, Major, USAF Director of Public Affairs AIR FORCE INSTITUTE OF TECHNOLOGY (ATC) WRIGHT-PATTERSON AFB, OH 45433 LYNN E. WOLAYER Dean for Research and Professional Development		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) DECISION THEORY NETWORK THEORY ROUTING PROBLEMS MULTIPLE OBJECTIVES OPTIMIZATION		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This thesis presents a methodology for finding the routing of paths in a network. The methodology establishes the least construction cost paths and minimum time to travel paths between known points. Given the minimum time to travel network, the least cost path is determined. A decision theory, objective optimization theory is used to provide a decision.		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

↓
maker with a solution set of networks that have been optimized for their least construction cost and minimum travel time. The methodology may be applied to any class of problems where conflicting objectives exist in determining a network routing.
↗

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

