INCORPORATING DIRECT DEMAND MODELING AND IMPROVED MODAL CHOICE INTO FREIGHT COMMODITY MOVEMENT MODELS

by

PAUL ROBILLARD
B.Sc., University of Montreal (1960)
M.Sc., University of Montreal (1961)

Submitted in partial fulfillment of the requirements for the degree of

Master of Science

at the

Massachusetts Institute of Technology
January 1974

Signature redacted
Signature redacted
Signature redacted
ABSTRACT

INCORPORATING DIRECT DEMAND MODELLING AND IMPROVED MODAL CHOICE INTO FREIGHT COMMODITY MOVEMENT MODELS

by

PAUL ROBILLARD

Submitted to the Department of Civil Engineering on January, 1974, in partial fulfillment of the requirements for the degree of Master of Science

Models have been developed for use in studying the response of the transportation system on a network of transportation facilities. This work looks at an existing model system and adds new modelling capabilities to this system.

The model system is modified such that level of service variables are calculated for each mode. Both socio-economic data and level of service variables are used in conjunction with a family of direct demand models (Special Product Model-SPM), McLynn model, etc..., to perform the modal distribution of network flows. The model system also has been modified to include a facility for performing the assignment of flows using an incremental approach. These model improvements are tested and integrated into the original system.

This thesis shows in detail how the modification was accomplished and develops the underlying theory.

Thesis Supervisor: P. O. Roberts
Title: Professor of Civil Engineering
ACKNOWLEDGEMENT

The author would like to express his appreciation to those individuals and organizations who contributed their support to this effort. Special thanks go to Professor Paul O. Roberts, Head of the Transportation Systems Division, for his guidance, his enthusiasm about this work and his many helpful suggestions during the research and the preparation of this thesis. Gratitude is also felt for Professor Wayne Pecknold who has assisted me throughout my stay at M.I.T. The author also acknowledges the financial support of the firm Lalonde, Valois, Lamarre, Valois and Associates Incorporated, who has made this year at M.I.T. possible. Special gratitude is felt for the president of this firm, Mr. Bernard Lamarre. Finally, the author would like to express his debt and his appreciation to his wife, Marthe, for the moral support and the encouragement that she gave me during the less pleasant moments of this enterprise.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>TITLE PAGE</td>
<td>1</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>2</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>3</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>4</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>7</td>
</tr>
<tr>
<td>LIST OF MAPS</td>
<td>8</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>9</td>
</tr>
<tr>
<td>CHAPTER 1 INTRODUCTION</td>
<td>10</td>
</tr>
<tr>
<td>Section 1.1 General Overview of the Thesis and of the Harvard Brookings Model</td>
<td>10</td>
</tr>
<tr>
<td>Section 1.2 Summary of the Content of Each Chapter in this Work</td>
<td>15</td>
</tr>
<tr>
<td>CHAPTER 2 DESCRIPTION OF THE MACROECONOMIC TRANSPORT SIMULATOR AND IDENTIFICATION OF SOME PROBLEMS</td>
<td>18</td>
</tr>
<tr>
<td>Section 2.1 General Description of the Macroeconomic Sector</td>
<td>18</td>
</tr>
<tr>
<td>Section 2.2 General Description of the Transport Sector</td>
<td>21</td>
</tr>
<tr>
<td>Section 2.2.1 Disaggregation</td>
<td>23</td>
</tr>
<tr>
<td>Section 2.2.2 Editing</td>
<td>25</td>
</tr>
<tr>
<td>Section 2.2.3 Routing, Mode Choice, Distribution and Network Assignment</td>
<td>26</td>
</tr>
<tr>
<td>Section 2.2.4 Cost Performance Calculations</td>
<td>30</td>
</tr>
<tr>
<td>Section 2.2.5 Transport Prices, Investments, and Systems Summary</td>
<td>31</td>
</tr>
<tr>
<td>Section 2.3 R-factors</td>
<td>32</td>
</tr>
<tr>
<td>Section 2.3.1 Some Problems on Routing, Mode Choice, Distribution and Assignment</td>
<td>33</td>
</tr>
</tbody>
</table>
Section 2.3.2 Outline of Improvements to the Four Stages in the Assignment Process .................. 37

CHAPTER 3 DIRECT DEMAND MODELS AND THE HARVARD BROOKINGS MODEL ............ 39
  Section 3.1 Direct Demand Models .................................................. 39
  Section 3.2 The Special Product Model (SPM) .................................. 43
  Section 3.3 The McLynn, Sarc-Kraft and Baumol-Quandt Models .............. 46
  Section 3.4 Data Needed to Calibrate and Use a Direct Demand Model ......... 48

CHAPTER 4 CRITERION AND METHODOLOGY FOR THE SELECTION OF LEVEL OF SERVICE VARIABLES ................................................................. 54
  Section 4.1 Existing Tree Program .................................................... 54
  Section 4.2 The New Tree Program ................................................... 56
  Section 4.3 Routing, Mode Choice, and Assignment Using NWTREE and Direct Demand Models ................................................................. 61

CHAPTER 5 CAPACITY RESTRAINTS AND EQUILIBRIUM ASSIGNMENT ..................... 63
  Section 5.1 An Iterative Search to Reach Equilibrium ............................ 63
  Section 5.2 Incremental Assignment ................................................... 65
    Section 5.2.1 Increments Equal to the Supply of One Sub-commodity at a time ...... 67
    Section 5.2.2 Increments Equal to a Given Percentage of the Supply of All Subcommodities at a Time .............................................. 69
  Section 5.3 R-factors, Average R-factors and Their Use .......................... 71

CHAPTER 6 NUMERICAL EVALUATION OF THE MODIFIED SYSTEM ......................... 74
  Section 6.1 Data Used for Evaluation ................................................ 74
    Section 6.1.1 Network Used for Evaluation .................................. 75
    Section 6.1.2 Supply and Demand Data Used .................................. 78
<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Characteristics of the Links</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>(Reference 7)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Recent Exports and Forecast of Exports</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>(Reference 7)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Recent Imports and Forecast of Imports</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>(Reference 7)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Exports for 1980 by Origins and Destinations</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>(Reference 7)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Imports for 1980 by Origins and Destinations</td>
<td>86</td>
</tr>
<tr>
<td></td>
<td>(Reference 7)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Basic Grouping of Commodities for Computer Input</td>
<td>87</td>
</tr>
<tr>
<td>7</td>
<td>Minimum R-path by Mode</td>
<td>90</td>
</tr>
<tr>
<td>8</td>
<td>Minimum R-path by Mode</td>
<td>91</td>
</tr>
<tr>
<td>9</td>
<td>Minimum R-path by Mode</td>
<td>92</td>
</tr>
<tr>
<td>10</td>
<td>Matrices of Level of Service Variables</td>
<td>94</td>
</tr>
<tr>
<td>11</td>
<td>Matrices of Level of Service Variables</td>
<td>95</td>
</tr>
<tr>
<td>12</td>
<td>Matrices of Flows (McLynn Model)</td>
<td>96</td>
</tr>
<tr>
<td>13</td>
<td>Matrix of Flows for Nieb (L.P.)</td>
<td>97</td>
</tr>
<tr>
<td>14</td>
<td>Matrices of Flows (SPM1-model)</td>
<td>98</td>
</tr>
<tr>
<td>15</td>
<td>Matrices of Flows (Sarc-Kraft model)</td>
<td>99</td>
</tr>
<tr>
<td>16</td>
<td>Matrices of Flows (SPM1-model)</td>
<td>100</td>
</tr>
<tr>
<td>17</td>
<td>Link Flows for One Increment</td>
<td>103</td>
</tr>
<tr>
<td>18</td>
<td>Link Flows for Two Increments</td>
<td>106</td>
</tr>
<tr>
<td>Maps</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1</td>
<td>Road Classification.................................</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>(Map for 1970)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Transport Network Used...............................</td>
<td>77</td>
</tr>
<tr>
<td>3</td>
<td>Link Flows for One Increment.........................</td>
<td>104</td>
</tr>
<tr>
<td>4</td>
<td>Link Flows for Two Increments......................</td>
<td>107</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figures</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-a</td>
<td>Functional Components of the Macroeconomic Model</td>
<td>14</td>
</tr>
<tr>
<td>1-b</td>
<td>Sketch of a Possible Component of a Network</td>
<td>34</td>
</tr>
<tr>
<td>2</td>
<td>Macro-Flowchart of Assignment Process, Part 2</td>
<td>115</td>
</tr>
<tr>
<td>3</td>
<td>Macro-Flowchart of Assignment Process, Part 3</td>
<td>116</td>
</tr>
<tr>
<td>4</td>
<td>Macro-Flowchart of Assignment Process, Part 4</td>
<td>117</td>
</tr>
<tr>
<td>5</td>
<td>Macro-Flowchart of Subroutine DRCDMD</td>
<td>119</td>
</tr>
<tr>
<td>6</td>
<td>Macro-Flowchart of Subroutine TRNDIS</td>
<td>122</td>
</tr>
<tr>
<td>7</td>
<td>Macro-Flowchart of Subroutine DISTRC</td>
<td>123</td>
</tr>
<tr>
<td>8</td>
<td>Macro-Flowchart of Subroutine DISTRD</td>
<td>126</td>
</tr>
<tr>
<td>9</td>
<td>Macro-Flowchart of Subroutine DISTRE</td>
<td>128</td>
</tr>
<tr>
<td>10</td>
<td>Macro-Flowchart of Subroutine SPM1</td>
<td>131</td>
</tr>
<tr>
<td>11</td>
<td>Macro-Flowchart of Main Program</td>
<td>135</td>
</tr>
<tr>
<td>12</td>
<td>Macro-Flowchart of Subroutine NWTREE</td>
<td>137</td>
</tr>
<tr>
<td>13</td>
<td>Macro-Flowchart of Subroutine NADDON</td>
<td>138</td>
</tr>
<tr>
<td>14</td>
<td>Macro-Flowchart of Subroutine RCOMP</td>
<td>140</td>
</tr>
<tr>
<td>15</td>
<td>Macro-Flowchart of Subroutine LNUPDT</td>
<td>141</td>
</tr>
</tbody>
</table>

It is becoming more and more recognized that the funds that are allocated to transportation should not be established without a good understanding of the interactions of transportation and the other economic activity. In fact it is well known that the location of industrial investment for instance, is affected by the amount of transportation investment and it's location. Similarly, the pattern of economic growth and development will interact with the demand for transportation facilities and their locations. Thus it is important to realize that any long run transportation planning should take these factors into consideration. So the interdependency between the transport system and the economy and also the interactions within the transport system should be accounted for in any planning process.

As a consequence, the models that should be used to carry on the analysis must be able to deal with these different components of the planning process. Similarly, time-staging of investments in the transportation sector requires a modelling tool which is capable of dealing with systems interactions and also with the time component of any investment plan. It follows that the system approach to these questions must be dealt with models of large and broad capabilities. These models should be able to evaluate alternative plans for investment while accounting for the other two components mentioned above, namely systems interactions and time-staging. In addition, in order to be able to deal with the transportation problem globally, the model should be able to simulate freight commodity
movements as well as passengers movements. It should evaluate in some way the interactions between these two important sectors of traffic flow.

It follows that these attributes are some of the desired properties of a system model for planning. During the last decade, a certain number of attempts were made to design and build models that would have all or some of these attributes. While this was going on, a great deal of work was carried out in the area of transportation demand modelling, model formulations, definition of the relevant variables and choice of possible and plausible functional relationships between these variables. It is now becoming more and more recognized that the most pressing needs are for improving existing models. This should be done by the additions of new modelling capabilities made available as a result of the research in these different sectors. In reference [2], the author has suggested a set of immediate actions to be taken and this work is done as an effort to accomplish this goal among others.

Before any of these suggested improvements could be undertaken, the question was which model or model system to use as a starting point. Looking back at all the desired properties of the models as well as the structure of the model itself with respect to the contemplated improvements, the author decided to choose the Harvard Brookings model as a starting point. This model was developed during the years 1965 to 1969 and is very well described in reference [1]. This model system was developed to solve the transportation system planning problems in developing countries. As originally designed, it is capable to simulate the interactions between transportation investment and the economy. It is also capable of dealing
with the time dimension which is involved in the process as well as to approach the transportation problem from a system point of view. It was conceived as a model system to deal mostly with commodity movement. The addition of the modelling capacity given by the direct demand models will hopefully make the model a better tool for passenger movement as well as adding to the modelling capacity of the freight commodity movements.

This model has been used in transportation planning by Professor Paul O. Roberts and his colleagues in a study of transportation planning in Colombia. During the last five years, the firm with which the author is associated has used the transport sector of the model in many transportation studies in developing countries. It has been found to be a usefull tool and the improvements that have been done should stimulate furthermore the interest in using it in that type of planning studies.

Used as a system approach to transportation planning, the model performs the following operations. First the model simulates the general economic conditions within which the transport system is operating. Also it must forecast the demand for transportation as well as evaluate how the transport network is going to be used. Finally the model determines the cost performances characteristics of each link in the network and find out the economic implications of changes in the system performance. The main functional components of the Macroeconomic transport simulator are shown in figure 1-a taken from reference [1]. In the following chapter, a more detailed analysis of the operations of these functions will be given for the economic and for the transport sector. These explanations will be
in sufficient detail to help to understand the method that was adopted in order to carry on the additions to the model system.
FUNCTIONAL COMPONENTS OF THE MACROECONOMIC MODEL

Final Demand

Industrial Production

Regional supply and demands, regional production cost and prices

Transport Model

Interregional flows, transport output, transport costs, transport charges

Incomes, prices, wage rates

Figure 1-a
1.2 Summary of the Content of each Chapter in This Work

The methods of systems analysis have been increasingly used for transportation planning in the last decade. Models and model systems have been developed to deal with the problems of urban, regional and even national transportation. In chapter one a general overview of the problems addressed in this work is undertaken. First, a brief description of the main questions analysed here is given and then some reasons why the Harvard Brookings model was used as a starting base are mentioned. Chapter two goes into a more detailed description of the main functions in the Harvard Brookings systems.

First the macro-economic sector of the model is considered and it's relation with the transport sector are analysed. Then, a description of the main components of the transport sector is carried out. This is done in somewhat more detail than in the economic sector in order to promote understanding of the reasons why certain changes were made to the system. It is necessary to have a good understanding of the main functions of the model and a good system global view in order to be able to follow the logic used in modifying and adding to the system. At the end of chapter two, some problems with routing, mode choice, distribution and assignment are identified in the actual model and a brief outline of how this work attempts to solve these problems is given.

The next chapter is concerned with the definitions of the variables making up direct demand models and their characteristics. It also tries to identify what the additions of these new capabilities in the Harvard model would mean in terms of modifications to the system. The special
product model (SPM) is discussed in chapter three as well as the McLynn and the Sarc-Kraft models.

Having identified what is needed in order to use a direct demand model, chapter four deals with the identification of changes to implement in the model if one wants to use the level of service variables by mode as they are needed. This results in the need for a new tree algorithm which is going to generalize the one originally available in the system. The details involved in doing this work and their relations with routing, mode choice and assignment are all analysed.

Though the improvements so far were more concerned with mode choice and distribution, the whole process including assignment was looked at and modifications were made in order to have the possibility of carrying the assignment in an incremental way. These questions are the object of chapter five where a few methods of assignment are compared. The details of the implementation in the Harvard Brookings model are all analysed and the different possibilities of defining an increment are also considered. The details of actual implementation (coding and debugging and testing) were all made and tested for one of the procedure considered.

The results of the tests mentioned above are essentially reported in chapter six. The numerical evaluation of the modified system was made on data from a transportation study done in 1969 by a transportation firm with which the author is associated. The network and the supply and demand data used for the evaluation of the new capabilities is discussed and the results of the different tests can be found in chapter six.

Chapter seven gives a more detailed discussion of each new subroutines and of the flowchart of their operations. The methodology and the documentation
is explained in sufficient detail so as to facilitate any changes in the programs themselves if this was ever desired. Finally in the conclusions, some discussions of the results and some recommendations for future work in the area are made. In two appendices, a complete listing of all the input data used to make the runs as well as a listing of all the new sub-routines can be found out. These are only a part of the programs which define the whole system. The new ones are modular and can be inserted in any version of the model without any major changes to the actual system.
"Description of the Macro-Economic Transport Simulator and Identification of Some Problems"

2.1. General description of the macro economic sector.

The Macro-Economic Transport Simulator (METS) is used principally for transport planning and in the analysis of developing countries. It is made up of two important sectors, the macro-economic sector and the transport sector. It is the purpose of this model to perform a detailed analysis of commodity flows as well as determine operating characteristics of the system. When this analysis is complete, the information obtained is passed and incorporated into the economic simulation sector.

While a more detailed description of the transport sector is given later, a brief resume, based on [1]*, of the main functions performed in the macro economic sector follows.

The basic structure of the computations performed by the macro economic model can be classified into four principal categories: a) final demand for goods, b) industrial production, c) incomes and d) prices.

In order to get the first component of the final demand in each region, the model uses an empirical relationship between the personal consumption of commodity I in region M to a fraction of the total consumer expenditures in that region and also the price of that commodity in that region. This is done for both domestic and imported goods.

The next components of the final demand analysed by the model are the fixed investment, the demand for investment goods and also the inventory investment. After having determined the total investment made in each region and on each commodity by the industry and the government, the model uses a relation between the demand for investment goods I in region M to the investment expenditures mentioned above. Similarly the imports of investment goods, I, are also estimated. Then, the model determines the
amount of inventory investment in good I in region M using a linear relation between the outputs of the corresponding industry in that region and the fraction that will be put into inventory.

In a final step, the total demand for each good in each region is determined by summing up the three components described above and adding to these the government purchases and the exports of that good.

Having the final demand for goods the model proceeds to determine the industrial production. This is done by using an input-output table. In fact if A is a matrix of coefficients such that A(I,J) is the amount of good I needed to produce one unit of good J, then if \( X = [X(1), X(2), \ldots, X(I), \ldots] \) is the vector of total demand for all goods and \( Y = [Y(1), Y(2), \ldots, Y(J), \ldots] \) is a vector of outputs for all goods, then we have the following relation:

\[
X = AX + Y \tag{1}
\]

or transformed

\[
X = (I-A)^{-1} Y
\]

The level of the industrial production just determined is the desired one. However, because of balance of payments problems in many economies the economy simulated will operate under import quotas. If the desired level of imports is less than the quotas, no change is then necessary and the simulation proceeds forward. If the level of imports exceeds quotas, the estimates of final demand and of production will be adjusted in the model by shifting part of this gap to domestic industries. If the excess demand for imports cannot be eliminated by that shift, then the model will have to make direct reductions in the final demand for imports. These adjustments will make the
simulation results consistent with exogenously specified import quotas.

After having completed the calculations of industrial production, the economic model proceeds to make a regional allocation of production. Then the regional and national output levels are checked against the regional and national maximums in each industry and calculations of the total regional demand for each commodity by industry are performed. The last steps before going into the transport sector determines the unit production costs exclusive of transportation costs, the wage income and the average material cost. Finally, the total value of exports of industry I as well as the total value of imports of each good in current prices are determined. These are used to calculate the balance of trade.

The functions described above are done under the control of the main program METS, by the subroutines ECONOM, INOUT, CONTIN and BEFORE. All the calculations discussed above up to the unit production costs are done in ECONOM except the determination of the level of industrial production which is a result of INOUT. The subprogram CONTIN takes over from ECONOM and calculates unit production costs and the quantities following them as described above. The subroutine BEFORE is the last one in the economic part of the model which is used before the transport sector. This last part of the model is then executed under the control of METS.

Once the transport sector has completed its calculations, the subroutine SUMFLO, which is the interface between the transport and the economic sector, is executed. Then, the economic model takes over from SUMFLO and the tasks described below are done in subroutine YREND.

First, for each industry and each region, the model determines the revenues, the profits and the retained and expected earnings as well as the
invested earnings. Then, average and marginal costs of supplying each region with each commodity are calculated along with dividends and the total regional income. After the calculations of indirect taxes collected by each industry, the model determines final demand and the gross national product for the year as the difference between final demand and production imports. In addition, some aggregate summary statistics for the economic activities which will provide a description of the pace of economic development are collected.

This completes the simulation for this year and under the control of METS the whole cycle for next year is repeated with the additions of some possible exogenous changes in the economic and the transport sector.

2.2. "General description of the transport sector."

Under the control of METS, the transport sector is entered after the execution of CONTIN and of BEFORE, two subroutines in the economic sector. All the operations in the transport sector are performed under the control of the subroutine TRNSPT. There are two interfaces between the transport sector and the economic sector; one before, one after. The work, in this case, is done in the subprograms AGGRE, GETIT and DISAG. The other is at the end of TRNSPT and is executed just before reentering the economic sector with the execution of YREND. The tasks in this interface are carried out in SUMFLO.

The structure of the transport model can be better described in terms of its principal computational steps, namely: 1) commodity disaggregation, 2) editing of network data, 3) routing, mode choice, distribution and network assignment, 4) costs performance calculations, 5) transport prices and system
performance measures and, 6) Summary.

The overall computational procedure can be summarized as follows. It first begins with the dissaggregation sector which prepares seasonal supply and demand vectors for each subcommodity. This is followed by the edit sector which is concerned with network edit and an initial link performance file computation. In a third step, all the minimum path trees, calculations of distribution and assignment of link flows, as well as an update of the network link file are then performed. The next stage in the procedure is centered on the evaluation of the link performance characteristics by using the proper cost models. It is also possible (if requested) to evaluate pricing policies and to perform the computation of network system performance measures, the determination of vehicle requirements and availability. Finally, it is necessary to calculate transport summaries and to reaggregate the transport flows and costs in such a way that they are available on a regional and industry basis in order to use them in the economic sector. Each of those steps will be the object of a more detailed analysis in the following sections.

All the calculations done in the steps above can be interrupted and resumed at two important points in the process. This is possible due to a checkpoint and restart procedure incorporated in the model. This is an additional feature under the control of TRNSPT and it allows the users to stop the calculations in the transport sector at these two important steps and an analysis of the results can then be performed at each stage before a decision to proceed is taken. This procedure does not require additional input data but only a proper storage and retrieval of the files already available. In fact, all the parameters necessary to restart the
calculations are stored by the model at the checkpoint stage and they are used again to resume the computation as if the run had never been interrupted.

2.2.1 Disaggregation

The supply and demand disaggregation is done in TRNSPT by the following routines: SUPDEM, PICKUP, GETIT, AGGRE, and DISAG. The data as given by the economic model on supplies, costs and demands must be aggregated in order to provide the transport sector with single matrices for the values of these elements. The subroutine AGGRE performs the tasks of consolidating the data on supplies and demands and costs for imported and domestic goods. Once this is completed, the control is passed to GETIT which reads and edits the disaggregation factor table.

There are three options available in the use of these tables. First, these tables can be entered completely from a card file. In this case GETIT will edit the data and prepare a file for these tables in order that they can be used in subsequent runs and years. The second option is used when the file just mentioned above is going to be utilized. Then, GETIT in this case will move the tables into memory. The third option uses a file which had been previously prepared and a card file which contains corrections and changes to be made to the first file. In that case GETIT will perform the editing and the changes and it will create a new updated file of these disaggregation factor tables.

Using the results of AGGRE and GETIT the next step is to complete the supply and demand disaggregation which is done in DISAG. It will
dissaggregate the supply values as provided above into subcommodity supplies at each node and on a seasonal basis. While it is doing this process, DISAG can accommodate exogenously specified demands and it will adjust the supplies and demands that it already has to incorporate the new demands before it starts to do the normal supply and demand computations. The final result will be a file named LSDTAP which contains for each season:

a) A list of supply nodes for each subcommodity.

b) A list of supply quantities for each node mentioned in a.

c) A list of the demand nodes for each subcommodity.

d) A list of the demanded quantities at each node in c.

This file will be used later in the distribution section. At this stage control is given back to TRNSPT which will proceed to edit the network data with the subroutine EDITOR.

Though the transport model has been designed to use the basic regional supply and demand data furnished by the economic sector, there may be times when the use of the full macro-economic model is not warranted or the use of it is made difficult because of lack of sufficient data. In these cases the transport sector can still be used if sufficient data exists on flows and production quantities. In fact, SUPDEM can be utilized in two additional ways, one is where exogenous regional supply and demand data is specified and dissaggregation factors tables for every commodity given or another is where exogenous dissaggregated supply and demand data is specified and again given for every subcommodity. In this last case, SUPDEM will act as an edit routine and it will create and update dissaggregated supply and demand files.
2.2.2 Editing

In the preceding section the supply and demand data has been put on nodes of the network, so it is necessary that this network be updated or created if it is a new one. This is accomplished by the subroutine EDIT under the control of EDITOR and TRNSPT. One of the main functions of EDIT is to add, delete, or change and modify link data in the network link file. Also entirely new network link file can be created.

The editing process in EDIT is carried out using a card file and a previously constructed link file LNTAPE. It reads each link in LNTAPE and checks to see if this link is in the card file. If so, it is either deleted completely or changed and stored on the link file LNEW, an updated version of LNTAPE. This process is repeated for all links in LNTAPE. Then, if there are links to be added to the LNEW file, this is done at the end as the last step in the editing process. At the completion of this task we have an updated link file on LNEW. The other main function of EDIT is to use this file LNEW and to perform a first estimation of the link utilization vector and of the link performance vector for each link in LNEW.

In fact, when performing this initialization of LNEW, EDIT will update the link utilization and the link performance vectors for each link in LNEW. It will accomplish this by making a first approximation to the link flow volumes in tons and in vehicles. These estimates are then used with the link characteristics to determine approximate link performance vectors using a special modal cost performance models. Thus, at the end of this second task EDIT has produced a link file, LNEW, which is going to be used later in the
distribution section. For further reference in the transport sector the names of the files are switched and LNTAPE becomes LNEW. This editing process is then completed and control is passed to TRNSPT.

2.2.3 Routing, Modal choice, distribution, network assignment.

At this stage in the transport sector, an updated file of supply and demand data LSĐTAP is now available and also an updated link file LNTAPE has just been released from EDIT. So the next tasks are concerned with routing, mode choice, distribution and assignment and to do so, TRNSPT passes control to the subroutine ASSIGN which will begin the calculations necessary to perform these four tasks for each season; that is the work will be repeated as many times as there are seasons involved in this year simulation. Therefore, all the steps described below will be for a given season.

Using the data from the link file for this particular season, the subprogram RCOMP will calculate the R-factors for each link and for each subcommodity and store these on the R-factor file named RFCTAP. These R-factors are calculated by multiplying the vector of subcommodity preference with the vector of link performance for the adequate vehicle class which contains five elements, namely the waiting time (WT), the travel time (TT), the variance of travel time (VTT), the probability of loss (PLS), the cost (C). The elements of the subcommodity preference vector (CPV) are weights that a subcommodity (K) gives to the components of the performance vector (LPV) for a given class of vehicle to which this subcommodity is assigned. Thus, for every link \( L \) we have:

\[
RFAC(L,K) = LPV(L) \times CPV(K)
\]
When this task in RCOMP is completed, there are three files outstanding: the two files mentioned previously and the new one RFCTAP.

It is at this stage that the calculations of the minimum R-paths trees and mode choice are undertaken. In fact in the subprogram NETFL0, which is the next one executed, an iterative sequence of operations is begun. Indeed for each season, for each subcommodity and for each supply mode (the HOME NODE) of this subcommodity, a minimum R-path tree to all other nodes in the network is built in the subroutine TREE. It returns, as a result, the minimum R-sums and the corresponding costs-sums from each mode in the tree to the HOME NODE. When this tree is completed, the cumulative R (CUMR) and the corresponding cumulative costs (CUMC) from the home node to all other demand nodes for that subcommodity are calculated. At this stage, another file is opened (TRETAP) and each tree is stored on TRETAP for further use in the assignment phase. The trees that are available determine the routing from the home node to all other nodes in the network. Also, because the routing is done on a minimum R-factors basis and without taking account of the modes used to go from the home node to any other node, the routing will, at the same time determine the mode choice used by the flows of this subcommodity. This process is repeated for all supply nodes and the matrix of cumulative costs (CUMC), in which the rows correspond to supply nodes and the columns to the demand nodes, are stored on another file (TRNSUM) and they will be used at the end of the transport sector in SUMFL0 where transport summary and the interface to the economic sector is done. The program NETFL0 is now ready to undertake the distribution of flows from the supply nodes to the
demand nodes for that subcommodity in that particular season. In order to perform these tasks, the transport sector as currently programmed has access to two models DISTRA, DISTRB, the gravity model and the linear programming model respectively. These models use the CUMR as resistance or generalized costs factors from origin to destination (supply and demand nodes) and also use the supply and demand data on the LSDTAP file to determine the flows (FLOW) between these nodes. In the linear programming algorithms, generalized transportation costs are minimized subject to the usual restrictions on supply and demand. The gravity model applies successive adjustments to the attraction factors until the restrictions on supply and demand are met to a specified degree of accuracy. Then, the flows in FLOW which are in tons per day are transformed into dollars per season and stored on the TRNSUM file to be used later in SUMFLO for transport summary purposes like the matrix CUMC.

Now that the distribution of flows is completed for that particular subcommodity in that season, the program NETFLO is ready to carry on the assignment of these flows to the network. This task is carried out in the subroutine ADDON. The first thing done is to retrieve from TRETAP the particular tree for each supply node. Then using the matrix of flows (FLOW), which is converted back to tons per day, the algorithm proceeds from the demand nodes to the particular supply node for that tree and assigns the corresponding flows to the network links belonging to that tree. This is repeated for every supply node. Now that the routing, distribution, mode choice and assignment is completed, NETFLO will go to the next subcommodity in the same season and repeat the calculations above,
starting with the minimum R-path tree calculations. Once this process has been terminated for all subcommodities, NETFLO has all it needs to make this season's update of the link file LNTAPE. In fact, since the assignment has accumulated the flows on the links in the network from all subcommodities, the subroutine LNUPT will carry on a link update of the link file LNTAPE.

First, one component of the link utilization is already available namely the link volumes in tons. A second component is calculated in LNUPT, that is the link volumes in number of vehicles. This is done by dividing the tonnages by the payload of the vehicle corresponding to each vehicle class. These two link utilization figures are stored in LNEW with all other data on link characteristics and link performances all of which are on LNTAPE. Also at the same time, the link performance data for the next season is transfer from LNTAPE to a temporary file LNSTOR which will be used later in the next season calculations by RCOMP in order to calculate the R-factors. The work for this season is done, thus the file LNEW is switched with LNTAPE and the subroutine ASSIGN proceeds to the next season calculations by repeating the whole process until the last season is done.

Once this is finished, control is passed to TRNSPT and at this stage we have a link file LNTAPE which contains the link characteristics, the link utilization data in vehicular volumes and tonnages. The last task left (to produce a complete update of the link file LNTAPE) is to calculate the five components of the link performance vector for each vehicle class.
2.2.4 Cost Performance Calculation

The calculations of the link performance vectors are carried out in the subroutine LNKOST which is executed under the control of COSTER and TRNSPT. This work is done using the link file LNTAPE as input; then the link performance vector is updated using the proper cost models and the results are stored in the file LNEW. In doing these tasks, the two directions of a link are used at the same time because certain performance measures depend on both directions (as for example the back haul calculations). Using the proper mode in the link characteristics of that link, the correct modal cost model is entered and the evaluation of the performance measures is made using the link characteristic and the link utilization vector which was updated before. The measures of waiting time, travel time, probability of loss and cost to the operators are calculated. These costs, which are not generally the costs to the users of transportation, can be varied using the pricing policy routine which allows the study of different pricing schemes.

For each one of ten possible modes and submodes and each vehicle class, a rate is available and the calculated cost could be replaced by the rate if the mode and submode of that link correspond to one in the table and if the corresponding entry in the table of rates is positive.

The next step is the calculation of systems performance, LNKOST proceeds to collect systems performance measures by cumulating the data from each link which is being updated and stored in the file LNEW. Cumulative sums are carried out for waiting times, travel times, transfer times, vehicle requirements and availabilities, costs and revenues, ton-miles and vehicle miles.
for each mode, for each season and for each vehicle class. Once all the links in LNTAPE have been evaluated, again the files LNTAPE and LNEW are switched and then some measure of vehicle requirements availabilities ratio is determined.

2.2.5 "Transport prices, investments, and systems summary"

The calculations of some measures of performance in LNKOST being done, the transport sector proceeds to determine this year's transport investment, labor costs of the transport sector and to complete the transport summary and the interface to the economic sector. This is achieved in the subroutines TINVST, LABOR, and SUMFLO. First, the vehicle depreciation by system and class is computed and it is used in the calculations of the vehicle investment for next year. These investments are then distributed on a regional basis. They will be used later on in the economic sector. Next, the labor wage bill in the transport sector is determined in LABOR and is distributed on a regional basis.

Finally, in order to provide input to the economic sector of the system, it is necessary to reaggregate the data from a node-subcommodity level to a region-industry level. These tasks are done in SUMFLO, using the summary file TRNSUM which was created earlier in the transport sector and contains the cumulative costs (CUMC), and the flows, (FLOW). To complete these tasks, it uses the supply and demand work file TABTAP which was created in the forward coupling stage while doing the dissaggregation process. Three matrices of inter-regional flows are prepared: one for each industry measured in dollars of the industry output and two others measured in value of transport required at current
transport prices and at base year prices. Finally, the transport marginal matrix is determined as well as the transport industry revenues using current prices. This last step terminates the calculations in the transport sector (TRNSPT) this year and the control is passed on to the economic sector and more precisely to the subroutine YREND.

2.3 R-factors

In the preceding sections an overall view of the operations of the macro economic and transportation simulator has been given. The remainder of this work is going to be concerned with the routing, mode choice, distribution and assignment sector of the system. A more detailed analysis of each of those functions is done and some problems that one would face when using the system are identified.

Once the calculations have been initiated in ASSIGN for a given season, the R-factors are then determined for all the subcommodities on all links in the network using the commodity preference vector (CPV) and this season link performance vector (LPV). It is important to realize that if one assumes that any two subcommodities have the same CPV components, then the R-factors for each link for these two subcommodities are going to be the same as long as they use the same vehicle class when they are moved over the network. The reason for this is due to the fact that the LPV is not changed when these R-factors are calculated sequentially on subcommodities. This is going to have a direct impact on the routing, mode choice, distribution and assignment as will be seen in the next section.
2.3.1 Some problems on Routing, Mode choice, Distribution and Assignment

The file RFCTAP at this stage contains the R-factors for all subcommodities for this season. Then the subroutine NETFLO initiates a sequence of calculations done in an iterative way on subcommodities. The R-factors are used in the tree building program (TREE) to determine routes from each supply nodes of a subcommodity to all other nodes in the network. Now, the routes, so selected, determine at the same time the different modes over which this subcommodity flow is distributed over the network. In fact, when the routes are selected, the transfer links allow the tree algorithm to change from one mode to another mode by using these links. The R-costs of such changes between modes must be such that the corresponding transfer links are on the minimum R-path determined by the algorithm.

The sketch in Figure 1-b will illustrate the situation. Suppose that the minimum R-path route from I to J is going sequentially through the following nodes: I, 1, 3, 4, 2, J. This means that the route going through I, 1, 2, J was not minimum but had a higher R-costs than the first one mentioned above. So the two parallel links (1,2) and (3,4), one on highway and one on rail, compete for the traffic going from 1 or 3 to 2 or 4. Now the minimum R-path uses (3,4) and not (1,2). So the mode choice made by the traffic flowing into node 1 is going to reach node 2 exclusively on rail. Now it could very well be that the R-sum on the path I, 1,2, J is just a little more than the minimum R-sum on I, 1, 3, 4, 2, J. This is a type of "all or nothing" mode split and this situation is not going to reproduce well the way the traffic chooses modes in the real world. So, the mode split resulting from the minimum R-path routes is subject to some weakness from that point of view.
Now as another consequence of this minimum R-path routing, it is going to assign all the flows generated by the supply $S_I$ at node I to the path I, 1, 3, 4, 2, J for subcommodity A. Thus, if one assumes that a subcommodity B has node I as a supply node and that some of this supply is going to be shipped to node J, then if the CPV vector of this subcommodity is the same as the one for A, then the same mode split will result since the same route will be selected. Thus the flow of B will be assigned to the same links as the flow of A. This is a result of the fact that the assignment is done sequentially, one subcommodity at a time, and that there is no update of any kind of the link performance vector which would account for the flows that have already been assigned to this path. Indeed, the resistance factor for subcommodities A and B have been calculated before any assignment has begun and thus the resulting R-factors are based on the same link performance vector. In the assignment process then any path that uses link (3,4) has chosen to do so without any consideration of the volume and the flows already assigned to links (1,2) or (3,4). Thus, the assignment is an all or nothing assignment and since no consideration is given to the existing volumes in the choice of the links to assign future volumes or subsequent flow, there is no capacity restraint involved in the assignment process. In other words, it is quite conceivable that the resulting volumes and flows on (3, 4), that would follow from the assignment of the flows to the network for all the subcommodities concerned, would exceed the capacity of the link by a large factor. Also, it is possible that, while the rail mode is over-used, the highway mode on (1,2) would be very underloaded. Therefore, the mode split would not be accurate and the volumes predicted on these links would not be realistic.
Another problem that results from this assignment is some oscillation of the flows assigned. In fact if (3,4) is overloaded and (1,2) underloaded, then, based on these link utilizations, any recalculations of the link performance characteristics which are normally done at the end of the assignment would produce low figures for the performance measures on (3,4) and high or good values on (1,2). If this link file was used again in a subsequent run of the model, the R-factors for all subcommodities for link (3,4) would be much higher than they were on the previous run while the R-factor on (1,2) may be sensibly the same as before. In this case, the minimum R-factor route from I to J may very well use (1,2) and not (3,4) as it did before. Thus, the flows assigned to (3,4) may be shifted almost all to (1,2) and it is possible to have the opposite situation where (1,2) would be overloaded while (3,4) would be under-used. In either case, the volumes and the flows predicted would not be realistic and the transport sector does not have any sure method to avoid this problem. The fact that the supply and demand for the different subcommodities is spread all over the network is going to prevent the situation described above in some cases but it is quite possible that the problem will arise locally in a network.

It is also important to notice that the minimum R-paths selected have attached to them the R-sums which give the generalized costs from each supply node to any other node in the network and thus in particular to add demand nodes for any given subcommodity. The result is a matrix of resistance factors CUMR used in the distribution algorithms of the transport sector. The two distribution models already available (linear programming and the gravity model) use only one explicit cost factor in order to perform the distribution.
The direct effect of each component of the LPV is somewhat confounded with the effects of the other components due to the nature and the definition of the R-factors. It would be desirable to have access to more models and also to distribution models which can use the direct effect of each component of the link performance vector: waiting time, travel time, variance of travel time, probability of loss and costs. It is likely that better simulation results will be available on the distribution of subcommodity flows if individual components of the level of service variables are used instead of their linear combination.

2.3.2 "Outline of improvements to the four stages of assignment"

In the preceding discussions, some problems related to the assignment process of the transport sector have been identified. In order to improve on some of the problems or to add or expand upon the existing flexibility and capacity of the simulation of flows using the model, it is now the object of the following chapters to show into details how this can be accomplished. Direct demand models, as defined later on, will offer an additional way to determine, at the same time the distribution and the modal split of the subcommodity flows. This can be done only if the level of service variables for each mode are made available. Thus, in order to obtain their values, new procedures to determine routes and trees over a specific mode will be analysed and introduced into the model. Also, in order to deal with the questions related to capacity restraint, oscillation of flows assigned to the network, new assignment procedures will be discussed and incorporated into the transport
sector. The questions related to the implementation will be addressed and an analysis of how the implementation have been done (or should be done in some cases) will be discussed in the following chapters.
Chapter III. "Direct Demand Models and the Harvard Brookings Model"

3.1 "Direct demand models"

In a study of some fundamental properties of travel demand models, the author in [2] suggests a direction which should be followed in the implementation of new model systems or in the improvement of existing models. Among the guidelines proposed is the suggestion that a model system should allow explicit treatment of any number of market segments and should also allow each market segment to exhibit different behavior patterns. These differences should be expressed in the structured form of the model function as well as in the values of the parameters. The user should have the possibility of selecting the functional form he desires and the computational procedures should allow simultaneous use of several different forms corresponding to different market segments. This availability of alternative forms in a single model system would allow the analyst to do sensitivity analysis of alternative models as well as alternative sets of model parameters.

It is also suggested that any model system should be such that a set of possible level of service variables can be used. In [2], the author determines immediate actions that should be taken. First, modifications to some of the existing systems should be undertaken in order to handle a range of level of service variables. Capabilities of analyzing a single multimodal network should also be implemented. Whenever it is possible, existing models should be modified to allow the use of the Special Product Models (SPM) to be defined later. It is also proposed that a single model system be developed to compute equilibrium using a direct approach and the explicit form of the General Share Model (GSM).
The iteration procedures should also be studied in order to achieve convergence to equilibrium in both the indirect and the direct approach.

Taking into consideration the problems that are discussed in chapter one regarding the usage of the Harvard Brookings model, it will become clear that the program of developments and improvements suggested in [2] and summarized above, is going to help to solve some of these problems. Some of the results discussed and some of the new methodology implemented in the following chapters are a step forward in the directions suggested above. The modified Harvard model will then be a better tool for national transportation planning.

The following quantities will be needed to define certain models. First, let us make the following definitions:

\[ V_{klmp} = \text{The flow in tons going from node } k \text{ to node } l \text{ by mode } m \text{ and path } p. \]

\[ V_{klm} = \text{The flow in tons going from node } k \text{ to node } l \text{ by mode } m. \]

\[ V_{kl} = \text{The flow in tons going from node } k \text{ to node } l. \]

\[ V_k = \text{The flow in tons going from node } K. \]

\[ V_t = \text{The total flow in tons.} \]

\[ A = \text{Vector of variables describing the socioeconomic data or activity system. In the case described here the elements of } A \text{ are the supply and demand data, population, income, etc...} \]

\[ a = \text{a vector of parameters applying to } A. \]

\[ X_{klmp} = \text{a vector of } N \text{ level of service variables } (L^0 = 1, \ldots, N) \]

which are used to describe the transportation system characteristics as seen by the flows from node K to node l by mode m on path p.
In this work the components of \( X_{k1mp} \) will be the R-factors defined above, the waiting time, the travel time, the variance of travel time, the probability of loss of a unit load of goods, and the cost.

\[
X = \left\{ X_{k1mp} \right\} = \text{the set of all level of service variables for all nodes, all modes and all paths.}
\]

\( w = \) A vector of parameters applying to \( X \).

\( R_{k1mp} = \) the combined effect of all the level of service characteristics of all modes as they influence the flows from node \( k \) to node \( l \) by mode \( m \) on path \( p \), (a so-called "resistance to flow").

This variable \( R_{k1mp} \) is a function of \( X \) and \( w \), that is

\[
R_{k1mp} = f(X, w) \quad (2)
\]

Also let:

\( R_{k1mq,p} = \) The combined effect of all level of service variables of mode \( q \) as they influence the flows going from node \( K \) to node \( l \) on mode \( m \) using the path \( p \).

\( z = \) The combined effect of all the activity system variables.

The variable \( Z \) is a function of both \( A \) and \( a \) that is:

\[
z = g(A,a) \quad (3)
\]
And finally:

\[ Y = \text{The combined effect of all the activity system and of the level of service characteristic.} \]

The variable \( Y \) is a function of the activity system, \( Z \) and the resistance to flow, \( R \), that is:

\[
Y = h(Z, R) = h(A, a, X, w)
\]  \hspace{1cm} (4)

Having now defined the fundamental quantities used to describe a transportation system, it is possible to define a general demand model system using the following functional relation:

\[
V_{klmp} = f(V_{klmp}, Y)
\]  \hspace{1cm} (5)

This is a functional relationship between the flow \( V_{klmp} \) and the system itself, \( Y \). It must be noted that in general this relation is implicit because \( V_{klmp} \) appears on both sides of the equation. The functional form of the equations are different when the indices \( k, l, m, p \) change. In this work, only special forms of the function \( f \) will be considered. This is a consequence of the fact that a detailed implementation is also carried out. But, there is an important class of models all of which are a particular case of the general relation above. They are defined as the explicit system of demand models and the variables \( V_{klmp} \) appear only on the left side of the demand model relationship. That is:

\[
V_{klmp} = f(Y)
\]  \hspace{1cm} (6)
and in more detail we have:

\[
V_{k\text{mp}} = f\left(A_k, A_1, a_k, a_1, X_{k\text{mp}}, \{X_{k\text{mp}m}\}_{p}, W\right) \quad (7)
\]

The Mc Lynn, the Sarc-Kraft and the Baumol Quandt models are all special cases of this explicit general demand model. Each will be discussed later.

In references [2] and [3], a very general model, the general share model (GSM) is defined. In fact the GSM defines a family of models among which most of the models mentioned so far are particular cases. Indeed, in [2] it is shown that any explicit demand system can be expressed as a GSM and more generally, the general direct demand model (GDDM) is a special case of the GSM. Also, any of the well known explicit direct demand models are particular cases of the GDDM.

Another class of model which is a particular case of the GSM is the Special Product Model 1 (SPM-1) and when special values are given to some of the parameters of the SPM-1, the special product model 2 and 3 (SPM-2, SPM-3) are obtained. It can be shown that the McLynn model is a particular case of the SPM-2 as well as of the GDDM and also that the Sarc-Kraft model is a particular case of SPM-8 and of the GDDM. Thus, it is clear that the SPM classes of models are very general and can be used to represent the travel behavior patterns of different market segments.

3.2 "The Special Product Model (SPM)"

The transport sector of the Harvard model has been modified in order that different subcommodities can use several models for distribution and modal split. The details of the implementation will be discussed later.
But it is important to see the new flexibility that will be added to the
model by the addition of the SPM models.

The functional form of the SPM-1 model is as follows:

\[ V_{klm} = \left( \frac{R_{klm}}{R_{kl}} \right)^{\frac{Z_1 \cdot R_{kl}^{a_1}}{\sum Z_i R_{kl}^{a_2} \cdot a_2}} \left( \frac{\sum Z_i R_{kl}^{a_3} \cdot a_3}{\sum Z_i R_{kl}^{a_3} \cdot a_3} \right) \]  

(7)

where \( V_{klm} \) is the flows from node \( k \) to node \( l \) by mode \( m \) as defined above.

The quantities \( Z_1, Z_k \) and \( R_{klm} \) have been defined before. It is important to
realize that the \( \cdot \) in the indices to \( R_{klm} \) means that the summation has been
carried on the values of that indice and thus:

\[ R_{klm} = \sum_{m=1}^{M} R_{klm} \]  

(8)

The general form in equation (7) leads to other particular models when the
value of some of the parameters are specialized. In fact, if the following
restrictions are imposed on the \( a_{ij} \), the SPM-1 model reduces to SPM-3, and SPM-2

Assume that:  
\[ a_{10} = a_{11} = a_{12} = a_1 \]  
\[ a_{20} = a_{21} = a_2 \]  
\[ a_{30} = a_3 \]  

(9)

Then, the resulting functional form is called the SPM-2 and it can be
written as follows:  
\[ V_{klm} = R_{klm} \left[ Z_k Z_1 \right] \left[ \frac{a_1}{R_{kl}}, \frac{a_2}{Z_1 R_{kl}}, \frac{a_3}{Z_1 R_{kl}} \right] \]  

(10)

The SPM-3 is obtained from equation (10) if imposing the condition
\[ a_1 = a_2 = a_3 = 1. \] Then the resulting functional form is:

\[ V_{klm} = R_{klm} Z_k Z_1 \]  

(11)
Thus, it is certain that the implementation of the general functional form of the SPM-1 will be an important addition and an improvement to the flexibility of the Harvard model. The behavior of different market segments can be better modelled using this large family of models. In METS, every sub-commodity and every class of passengers are assigned to a particular distribution model. Thus, it will be possible to select parameters that will reflect the behavior of each subcommodity and passenger movements.

At this stage it must be realized that particular forms of the functions defining $Z_k$, $Z_l$ and $R_{klm}$ must be chosen to carry out the implementation completely. There are many possibilities to choose from. If a functional form is wanted and is not in the available system, the addition to the Harvard model of SPM-1 was formed in such a way that it will be very easy to add this capability to the actual existing transport sector. The following forms for $R_{klm}$ have been added to the original transport model:

$$R_{klm} = w_{mo} \prod_{i=1}^{p} x_{mi}^{w_{mi}}$$  \hspace{1cm} (12)

where $\{x_{klmi}\}_{i=1}^{p}$ is the set of level of service variables. The index $i$ runs over the different variables used and the $k$, $l$, $m$ indices have the meaning already used above. In addition to the form in equation (12), the forms in equation (13) and (14) have been implemented in the Harvard model.

$$R_{klm} = \frac{\sum_{w} w_{mi} \prod_{i=1}^{p} x_{klmi}^{w_{mi}}}{\sum_{q} w_{qo} \prod_{i=1}^{p} x_{klqi}^{q_{qi}}}$$  \hspace{1cm} (13)
3.3 "The Mc Lynn, Sarc-Kraft and Baumol-Quandt models"

These explicit direct demand models are well known and have interesting properties. They all have been used in important studies and are part of the new version of the transport sector. In order to develop their functional forms and some of their properties let us assume that the socio economic variables used are the population $P$ and the income $I$.

The Mc Lynn model is given by:

$$ V_{klm} = a_{m1} a_{m2} a_{m3} a_{m4} P_{k}^{w_{m1}} P_{l}^{w_{m2}} P_{l}^{w_{m3}} P_{l}^{w_{m4}} X_{klmi}^{w_{q1}} $$

This model is a particular case of the SPM-2 as mentioned before. In fact, if one let $a_3 = 1$, $a_2 = 1$ in the SPM-2 equation (10), the following result is obtained:

$$ V_{klm} = Z_k Z_l \frac{R_{klm}}{R_{k1}} (R_{k1})^a $$

When one defines

$$ Z_k Z_l = a_{m1} a_{m2} a_{m3} a_{m4} $$

and

$$ R_{klm} = w_{m1} w_{m2} w_{m3} w_{m4} X_{klmi} $$

then
the preceding expression is obtained. The ratio $R_{klm}/R_{kl}$ is the modal share of mode $m$ and is defined as $\Omega_{k1m}$. With this definition, the elasticities of the flows $V_{klm}$ with respect to any level of service variable can be shown to be given by:

$$E(V_{klm}, X_{klmi}) = \omega_{mi}[(1-lm)(\Omega_{k1m})^{-1}]$$  \tag{17}$$

The cross elasticities are given by:

$$E(V_{klm}, X_{klqi}) = \omega_{qi}[(d-1)(\Omega_{k1q})^{-1}]$$  \tag{18}$$

The functional form of the Sarc-Kraft model is given by the following relation:

$$V_{klm} = a_{mk1}p_{10}a_{m21}p_{11}a_{m31}a_{m41} \prod_{i=1}^{p} x_{k1qi}$$  \tag{19}$$

Equation (19) is a particular case of SPM-3 where $Z_k Z_1$ are defined as in the Mc Lynn case but where the $R_{klm}$ is given by:

$$R_{klm} = \prod_{q=1}^{q} x_{k1qi}^{-1}$$

In this case, the elasticities of the flows with respect to any level of service variable are

$$E(V_{klm}, X_{klmi}) = \omega_{mmi}$$

The cross elasticities are:

$$E(V_{klm}, X_{klqi}) = \omega_{mqi}$$

The Baumol-Quandt model is an abstract mode model in which the parameters are independent of the modes. The functional relationship
which defines the model is given by:
\[ V_{klm} = a_{m0} p_k a_{m1} p_{l1} a_{m2} i_k a_{m3} i_{l1} a_{m4} \sum_{i=1}^{p} x_{klim} w_{li} \frac{x^{(b)}_{kli}}{w_{0i}} \]

where \( x^{(b)}_{kli} \) is the value of \( x_{klim} \) for the best mode \( m \). This model is a particular case of the GDDM as the other two models mentioned above. The elasticities and cross elasticities of the flows with respect to any level of service variable are:
\[ E(V_{klm}, x_{klim}) = w_{li} \]
\[ E(V_{klm}, x_{klq1}) = (w_{01} - w_{11}) \text{ or } 0, \text{ depending on whether the mode } q \text{ is the best mode or not.} \]

There are many other possible models that could be considered for implementation, but these last three are the most well known and they have been added to the transport sector independently from the SPM-1 which is more general and which can generate a whole family of models by varying the parameters and the functional form for \( Z_k, Z_l, \) and \( R_{klm} \).

3.4 "Data needed to calibrate and use a direct demand model"

Looking at the functional form of an explicit direct demand model (equation (7)), it is clear that, if one wants to use the model, the values of the parameters must be known. Also, the values of the socio-economic variables appearing in the equations must be available. Finally, it is necessary to have the values of the level of service variables for each combination of mode pair, for each mode and each path. The functional relations will then determine the flows for all combinations of \( k, l, m, p \).
once the above data is available. So the implementation in the Harvard Model system of any one of the models mentioned before implies that the transport sector has these quantities when they will be needed. In the original version of the model, it was possible to use only one level of service variable for distribution purposes, namely the cumulative R-generalized cost (CUMR). This matrix contains the generalized cost for each combination of supply and demand nodes for a given subcommodity. These costs are obtained from the minimum R-sums corresponding to the minimum R-path from one supply node to another node in the network. These values in CUMR are not mode specific in the sense that they are obtained from the minimum R-path algorithm which search for an absolute minimum over all available modes in the network. Thus, if two or three modes are on the minimum R-path when going from I to J, then the corresponding values in CUMR involves use of the level of service variables from these modes.

The use of any direct demand model requires that the values of the level of service variables from I to J be available for each mode that is being considered in the network. The transport sector must be modified in order that the system can calculate these level of service variables for each mode. Also, the original version of the transport sector uses as R-factors linear functions of the basic variables already available in the Harvard model namely; waiting time, travel time, variance of travel time, probability of loss and cost. Thus, it is important that the modifications to the transport sector be such that the new model will use the individual level of service variables as mentioned above. Indeed
to comply with the suggested directions made in [2], the model has been modified in such a way that for each mode the direct demand model can use any predetermined selection or combination of the level of service variables among the six possible ones already mentioned. For each mode, the transport sector must calculate six matrices of level of service variables; the rows corresponding to the supply node and the columns to the demand nodes. A capability must exist for selecting any combination of these variables and use them with a vector of parameters \( w \) to calculate the \( R_{klm} \). Obviously, the functional form used to specify the way that the above matrix will enter the calculations of the \( R_{klm} \) will have to be specified.

The next step is to calculate the effect of the socio economic variables. This is accomplished by using the available socio economic data on each supply and demand node namely the subcommodity supply and demand, the population, income, etc... This is combined with the parameters as specified by the vector \( a \) and the functional form of the function \( Z \). The model has been modified to use this new socio economic data in addition to the already existing subcommodity supply and demand. The modifications as they have been implemented allow the use of the population, the income and other additional variables specified at a node. Any three vectors of data which is attached at nodes, can be used in conjunction with the subcommodity supply and demand to determine the effect of the socio economic data.

In the original version of the transport sector, the subroutines TREE and NETFLO perform the calculations of the matrix CUMR; TREE determines the minimum R-path and NETFLO uses the corresponding R-sums
to obtain the CUMR matrix. This matrix is used by DISTRA and DISTRB for distribution purpose using the gravity and the linear programming model. To accomplish the modifications mentioned above the TREE program and NETFLO have been extensively modified and also new distribution models have been added to the actual ones DISTRA and DISTRB.

If one assumes access in the model to all the data mentioned above on the level of service variables for each mode and the data on the socio economic variables, then there is still an important question left unanswered; that is, how can the models be used to reproduce actual conditions. In other words, how is the calibration of the model going to be performed in order that the simulation is a satisfactory reproduction of the existing conditions. The model, as modified, provides sets of parameters for groups of subcommodities.

The first thing to do is to select, based on past experience or on other research results, or empirical investigations, a set of parameters that looks to be the most convenient for each group of subcommodities. Secondly, a set of origin and destination pairs should be selected for each subcommodity and the flows between the pairs of nodes should be determined based on the result of an origin and destination survey or through the use of data collected from other sources giving the subcommodity flows of interest. Using the selected parameters the model should be run with the supply and demand data for each subcommodity and the resulting flows on each mode should be analysed. The total flows between all pairs of supply and demand nodes should also be addressed. In carrying out this analysis, one should use the results of the origin and destination survey
on the flows for the selected pairs of nodes and compare them with the actual model predictions. Then, based on the results of these comparison, successive revisions of the values of the parameters of the distribution model should be done in order to improve the correspondence and reduce the discrepancies between the predicted and the observed flows.

Another way of dealing with the problem is to use the data on the flows for the selected origin and destination pairs with the corresponding predicted values of the level of service variables and the socio-economic data as input to a multiple regression program. The regression analysis is used to determine the set of parameters that will give the best fit between the level of service variables and the socio-economic data on one hand and the observed flows on the other hand. This approach is useful for some type of distributions model like the Sarc-Kraft, but in other cases, one has to use a non-linear regression program because of the impossibility to linearize the distribution model in its parameters. This is the case with the Mc Lynn model for instance. The calibration also involves some search for the model structure that would best explain the observed flows. Different functions form for the $R_{klm}$ and the $Z_k, Z_1$ are compared at the same time as the regression analysis or the iterative process described in the first approach is performed. The resulting calibration that one obtains with these approaches can always be improved upon. As it is stated in [1], while modelling is the art of building into the structure of the replicating mechanism only that detail which is needed, the calibration process is the selection of parameters that will make the model replicate the history of the original process as
faithfully as possible. Calibration is therefore subject to the same arbitrariness and artfulness as original model design.
Chapter IV "Criterion and Methodology for the Selection of Level of Service Variables"

4.1 Existing Tree Program

As mentioned in the previous chapter the mode choice, routing and assignment are based on the minimum R-path routes determined by the subroutine TREE under the control of NETFLO. It has been observed that, in order to use direct demand models, the level of service variables must be known for every mode in the network. To see what is involved in making the modifications needed to achieve these results, it is important that the functions performed by the tree-building program be well understood.

The algorithm which is implemented in TREE could be formally described using the following definitions:

\[ S_k = \text{a set of } k \text{ nodes at stage } k \text{ in the computing process for which the shortest path (minimum R-sum path) to any one of these } k \text{ nodes from a given base or home node is known.} \]

Also, let \( i \) be any one of these nodes in \( S_k \). Let us define \( d_i = \text{the least distance (minimum R distance) of node } i \text{ to the home node.} \) Also, let \( J_i \) be a node not in \( S_k \) and such that \( J_i \) is the closest (minimum R-distance) node to node \( i \) in \( S_k \). Define \( D_i \) as the distance (measured in units of R-sums) of node \( J_i \) to node \( i \). Then, the algorithm implemented in TREE chooses as the \((K+1)\text{th}\) node to enter in \( S_k \) and so to define \( S_{(K+1)} \) that node which makes \((d_i+D_i)\) a minimum. In carrying out the calculations above, the TREE program does not consider which mode is being used to go from one node to the next. As a result, the trees and the minimum
R-paths may make use of a combination of modes when going from the home node to any other node in the network.

To see why certain modifications were made to the algorithm when using direct demand models, a more detailed description of the steps involved and of how the implementation was done in the program TREE is necessary. Let us define a few additional variables: NHOME will stand for the home node considered; INODES and JNODES are the two node numbers defining a link and the link is traversed in the direction from INODES to JNODES. Thus INODES is referred to as the beginning node and JNODES as the end of the link. Also, let RSUM (NODE) be defined as the minimum R-factors sum from the node NODE to NHOME and let us define RCUM (LINK) as the cumulative R-factor obtained by the following relation:

\[ RCUM(LINK) = RSUM(INODES) + R(LINK) \]  \hspace{1cm} (20)

where INODES is the beginning node of the link LINK and R(LINK) is the R-factor of that link.

The following tasks, identified in the algorithm and essential to it will now be described:

1) The implemented algorithm starts at NHOME and determines all the links connected to NHOME or having NHOME as INODES. As these links enter the link table, the cumulative R-factors RCUM(LINK) are determined for each link by using the relation defined in equation (20). In order to complete the table at this initial stage, the algorithm will search through all of the links in the network.

2) Then a search of the RCUM table is performed over all the links in the table to determine the minimum RCUM. Corresponding to this minimum RCUM is a link number which is noted by the algorithm as well.
as the JNODES of that link. The tree algorithm will calculate the R-sum RSUM(JNODES) which corresponds to the term JNODES. The array NN(JNODES) is then used to store the link number having JNODES as end node and this link is deleted from the RCUM table.

3) The algorithm now uses this JNODES=NM (found in task 2 above) as the new base node. Another search through the whole set of links in the network is performed and the RCUM of all the links connected to or having as INODES, JNODES = NM are calculated. As this is accomplished these new links are added to the actual RCUM table. At the end of this stage, we have in the RCUM table all the previous RCUM plus the new ones added in the process just described less the one that was deleted in task 2 above.

4) With this new table of RCUM, the tasks 2, 3 and 4 are repeated.

It is clear that this procedure will ultimately go through all the nodes in the network and determine the link from any given node (JNODES of that link) that is leading you to a node (INODES of that link) which is closest in terms of R-Sum to the home node NHOME. While carrying out these tasks, the algorithm also calculates the cumulative cost sum CSUM(NODE) corresponding to the RSUM(NODE).

4.2 "The New Tree Program (NWTREE)"

The new tree building subroutine is a generalization of the actual TREE subroutine. It is possible to specify the minimum R-path based on absolute minimum R-sum. It is also possible to get as output
of the program, in addition to trees, the corresponding, accumulated sums for the level of service variables, namely: waiting time, travel time, variance of travel time, probability of loss and cost. Moreover the trees are mode specific. This last concept requires a more detailed explanation. It is to help one understand the modifications performed to the TREE building program to obtain the NWTREE.

First, as implemented, the algorithm uses the R-factors as a resistance to the utilization of a link the same as was done in TREE. However, it also collects the corresponding values of the other five components of the level of service vectors. If cost or travel time is desired as the resistance variable, it is easy to use since all the individual components are available and carried at all times.

Using either one of these possible resistance variables, a minimum R-path from the home node NHOME to any destination (NDEST) in the network is said to be over a specific mode, m, if either one of the two situations described below in A1 and A2 are true:

A1) If NHOME is the INODES of a link with mode m or if the link is a transfer link, then the algorithm described above (in tasks 1, 2, 3 or 4) should be used to search for a minimum with the restriction that the link considered has mode m or is a transfer link. As long as this condition is true we proceed as such.

A2) If at some point the conditions described are not true, that is there are no more links connecting to the nodes in \( S_R \) which are of mode m or are a transfer link, then the algorithm described under tasks 1, 2, 3, 4 should be used without any restriction, that is, it should make use of all connecting links regardless of mode. When operating under this last situation, the
algorithm will reach more and more nodes, thus at some point it will find a
node which is the INODES of a transfer link (if any are left) and in that
case the next step done by the modified algorithm is to use this last INODES
as the only connecting node in $S_R$ and it proceeds as described in paragraph
Al since the conditions of Al are satisfied.

8) If NHOME is not the INODES of a link with mode m or of a transfer link,
the algorithm described by tasks 1,2,3, 4 should be used without any restrictions
and the procedure followed should be exactly the one described above in A2.

The description of the algorithm modified could be summarized as follows:

1) As long as there are links with mode m or transfer links connecting
to the nodes in $S_R$, the algorithm considers only these as candidates for the
minimum R-path.

2) If at some stage $R'$, there is no more links to satisfy the condition
in 1, all links connecting to $S_R$, are now considered. As soon as a transfer
link is reached, it is the only one considered. If the JNODES of that trans-
fer link is on mode m, then the situation reverts to that in 1 described
above. If it is not on a link with mode m, we are back to the situation de-
scribed at the beginning in 2 and the modified algorithm proceeds as such.

A more formal definition of a tree over a specific mode can be made
using the sets $S_R$ and the variables defined before. At some stage $R$, either
there is one node $i$ in $S_R$ and at least one node $j_i$ not in $S_R$ and at least one
connecting link between $i$ and $j_i$ such that the mode of the link is either
m or a transfer link or this last condition is not true. If it is true we
proceed as in the original algorithm with the restriction that only mode
m or transfer links are considered. If it is not true, the algorithm is used
without any restriction so long as there is no connecting link which is a
transfer. At this stage $R'$, only the node $i$ in $S_R$, which is on this transfer link is considered and the algorithm is used with that restriction to reach stage $S(R' + 1)$. At stage $S(R' + 1)$, we have one or the other of the situations described above. The resulting routes going from $I$ to $J$ over a specific mode can be made completely of links of each mode if it is physically possible to go from $I$ to $J$ with routes using links of each one of the modes. If it is not physically possible that all the routes be of this kind, then certain routes will have links belonging to other modes. This fact is realistic in the sense that if a shipper is preparing to move goods from $I$ to $J$ and if he cannot use mode $m$ only, he will then ship over a combination of modes $m, m', m''...$ in order to physically transport his commodities from $I$ to $J$. If his primary mode is $m$, he will use modes $m', m''...$ as a mean of establishing connections with the primary mode $m$.

In the implementation of the modified algorithm in NWTREE, it is clear that all the tasks previously done in TREE also have to be done by NWTREE. Thus NWTREE must give as outputs minimum $R$-path trees which are the absolute minimum over all modes. The basic structure of the TREE program are kept in NWTREE but the following additions and modifications to steps 1,2,3,4 described above were made. Step 1 was left unchanged except for the search which is done over all links in the network to find all the links having the node $NM$ as INODES. This approach is expensive in computer time especially in large networks where all the links are scanned to find only two or three connecting links. At an earlier stage in subroutine ASSIGN and prior to entering NETFLO, the calculations of all the nodes that are the JNODES of links having the node $NM$ as INODES are carried out in NETURN. There is only one restriction; if there are more than three of these JNODES, then the array will not be complete and a negative sign will be placed in the third component
of the array for that INODES. The subroutine NWTREE will make use of
this array whenever there are three or less such JNODES. This will save
a lot of loops which are done over all links, each time a new node enters
the table or the set $S_R$. Whenever there are more than three JNODES, the
algorithm will proceed as before and go through a search over all links in
the network. The use of this array should reduce computer time if one
takes into consideration the number of times the loop is iterated.

Step 2 was modified in the following way: a search of the RCUM table is
now made to see if the mode of at least one link in the table is equal to
$m$, the specific mode in which we are interested in or if the mode of at least
one of these links is a transfer link. An indicator called MATCH is set
accordingly; to one if at least one of the conditions above are met, if not,
MATCH is set to two. If the algorithm should not be mode specific in the
search, then MATCH is set to three and NWTREE will do the same tasks as
TREE. Next, another search of the RCUM table is performed over all the links
in the table in order to determine the "constrained" minimum satisfying
the conditions defined by mode $m$ and the indicator MATCH. Having found this
constrained minimum RCUM, the link number of the link corresponding to this
minimum is noted and then step two continues as before with the additional
restriction that if we have gotten to a node at some point over mode $m$,
then we make sure that we stay on mode $m$.

Steps three and four are carried out without major changes with the
exception that in addition to returning the RSUM as TREE does, NWTREE
will also return the waiting time sum (WTSUM), the travel time sum (TTSUM),
the variance of travel time sum (VTTSUM), the probability of loss sum
(PLSSUM) and the cost sum (CSUM). These are the cumulative sums corresponding
to RSUM which was used to determine mode specific trees.
With the NWTREE program, the transport sector can calculate the level of service variables for each mode. There is currently a maximum of six variables that can be used namely WT, TT, VTT, PLS, cost and the CUMR (the abbreviations have been defined above). In the model as modified, it is possible to select any combination of these six components to use in the computing process. Thus, if one uses the McLynn model, the Sarc-Kraft model or any of the SPM models, the flows from node r to node 1 by mode m, \( V_{rlm} \), are determined by the socio-economic variables but also to a great extent the values of \( V_{rlm} \) are influenced by these levels of service variables. Thus, the mode split is based on the relative differences and values of the level of service variables of each mode. So it is quite likely in general that the modal split will be better and more realistic than the one obtained directly from the routing and the minimum R-paths of the original system.

There is an improvement in the routes used from any supply node to any demand node. In fact, there are as many routes as there are modes with a maximum of four modes as the system currently exists. Moreover, the mode choice is better due to the fact that individual level of service variables influence it directly and not through the R-factors, where they are only one term of a linear sum. The assignment, resulting from the flows and from the minimum R-paths routes, is somewhat improved because the matrix of flows for each mode is assigned to the network over the mode specific routes determined by NWTREE. The flows from the supply nodes to the demand nodes are therefore better distributed over the modes than the assignment that would have resulted from the original routing and assignment. The link capacity will not be exceeded as often or in the same proportion as before because the flows are
distributed over all modes. Still, there is no capacity restraint and the oscillation problem mentioned above will appear from time to time unless other methods are used to deal directly with these questions. They will, in fact, be addressed in the following chapter. Nevertheless, the trees over specific modes and the assignment of mode specific flows over the routes determined by these trees have improved the mode split and the assignment to the network of the flows.

A possibility offered by the new tree building program will allow the user to specify now that a subcommodity should move and be assigned to a given predetermined mode. In the modified version of the model, this flexibility will be useful because in many real world situations some subcommodities are shipped almost all by one mode. This feature will help the user of the system to replicate that situation in the simulation process.
5.1 "An Iterative search to reach equilibrium".

In references (4) and (5), appear descriptions of the most widely used assignment methods. The introduction of an incremental traffic assignment technique is also developed and tested on data. In section 2.3.2, some of the problems that could result in the original version of the transport sector regarding capacity restraints were discussed and identified. At this stage, one iterative method is suggested in order to deal with the questions raised and modifications that should be made to the Harvard model to achieve these goals, are explained.

The transport sector actually performs the assignment in the following sequence: for each season, the subcommodity flows generated by one of the distribution models discussed above, are assigned to the network using the minimum R-path from each supply node to each demand node. Once the assignment has been performed for all subcommodities for this season the transport sector proceeds to update the link utilization vector for this season. It then repeats these operations for the next season, and so on. Each cost performance model for each mode is called in turn to update the link performance vector. At this stage the model can be interrupted in the calculations before going back into the economic sector by making use of the checkpoint restart procedure that is explained in chapter two. This feature makes it possible to iterate this sequence of calculations using as input the updated link file obtained after the cost performance calculations.

In that case, the model returns to the beginning of the transport sector in EDITOR and proceed to make a run using this link file as input without editing except that the link file will be copied from LNTAPE to LNEW and that LNTAPE is switched back to LNEW. The whole sequence of calculations
in the transport sector is then repeated using the old file of supply and demand data LSDTAP. In doing so, the model will produce in this second run, a new file of R-factors on RFCTAP because the R-factors and their components will be based on the updated file LNTAPE. The model uses a new file of trees on TRETAP because of the new file of R-factors. Both the flows and the assignment will be affected by the new R-factors and their components.

The links which had flow assigned exceeding capacity in the first run, will normally have smaller flows in the second run because the R-factors have increased for the second run. These over capacitated flows will be distributed and diverted to under-utilized links in the network if there are any. This iterative process can be repeated but there is no guarantee that successive iterations will converge to a stable assignment. The results should be analysed after each iteration and the decision to perform more iterations will be based on a specific criterion. For example, the observations that the total number of vehicles hours in the system is fairly constant from one iteration to the other is one possible criterion.

There are many possible global measures that can be used to decide on the iteration to stop the process. One possibility is based on the total number of vehicle miles. This would stop the iterations when this last quantity is fairly constant from one run to the other. Another possible rule is to decide to stop when there is a given percentage or more of the R-factors for this iteration that are to within a given distance from the R-factors of the preceding iteration.

Some of the performance measures just mentioned are already available in the system like the total number of vehicle-miles. Measures based on the differences of the R-factors from one iteration to the other will require the following modifications. First, since the R-factors are calculated for all
subcommodities for a given season and are stored in the file RFCTAP and since
the same file is used to store the R-factors for the second and for all
succeeding seasons, it is necessary that the file be saved from one season to
the next and also from one run to another. Thus, at the end of the calculations
for each season the content of the file RFCTAP should be transferred to an
auxiliary file, RFCRUN. This file will be closed only at the end of the last
season considered in the model. The fortran numbers corresponding to RFCRUN
from one run to the other should be different in such a way that there is
access at the end of a run to the last two files of R-factors corresponding
to the last two runs. Once an iteration has been completed, a special program
can be run to perform the evaluation of the measure based on R-factors.
This routine should use as input the last two files of R-factors identified
by RFCRUN.

The iterative approach to assignment described above is in a way quite
similar to the U.S. Bureau of Public Roads assignment procedure. It is only
between iterations that the level of service variables are updated and in no
way is anything done on that question while carrying on the assignment. This
approach to assignment will be referred to as method A.

5.2 "Incremental Assignment"

The incremental traffic assignment technique was developed by M.L. Manheim
and B.V. Martin and reported on it in (4) and (5). The main features of that
approach can be summarized as follows: given a transportation network, a zone
pair is selected at random from all possible combinations of zone pairs; then
a minimum path algorithm, using time as the resistance variable, determines the
minimum path between the zone pairs so selected. Then, using a curve called
the generation rate characteristic (a demand curve), the potential volume
to be assigned between the zone pairs is calculated and a small increment of
this last determined volume is assigned to the path already selected. This additional volume is added to the existing volumes and the volume delay curves are used to update the travel time changed by the new link flows. The steps of the algorithm are repeated until all the zone pairs have been considered and until the interzonal potential volume determined by the generation rate curve has been assigned. The generation rate curve indicates the percentage of the input interzonal volume that will be realized as a function of the unit travel time between zone pairs. The curve is an indicator of the network congestion. If this percentage is 100% at any travel time, then we have assumed that the demand is not affected by the existing volumes and our assignment is one with fixed demands as opposed to the case where this percentage would decrease as the travel time increases. These situations are more fully discussed in (6).

In the following sections, a form of equilibrium assignment is investigated and some details of the implementation in the Harvard model are analysed under two slightly different approaches. In the Harvard model the equivalent of a zone pair as defined in (5) can be considered to be a pair of supply and demand nodes for a given subcommodity. If one wanted to approach the incremental assignment in the transport sector in this way, it would be almost impossible to think of updating the choosen path between a given pair because of the large number of such pairs to expect in actual usage of the system. It is quite common to have fifty to sixty subcommodities with an average of eight to ten nodes as supply nodes and about the same number of demand nodes for a given subcommodity in a given season. If we have four seasons, the number of pairs to consider for one increment can be expected to be of the order of 2500. It is suggested therefore that two possible variants of the incremental assignment as developed in (5) be investigated.
and compared for their advantages and disadvantages.

Due to resource limitations, only one method has been implemented and tested in the Harvard model. The other one will only be discussed but no numerical results have been obtained from it. Both approaches assume that the potential volume to be assigned is 100% for any values of the level of service variables. Instead of sampling on a pair of nodes, the sampling will be on the subcommodity number that will be assigned at a give time in the process as explained later. The way the increments are defined and the time in the process at which the updating of the link performance vector will be done are going to define the major differences between the two approaches.

5.2.1 "Increments equal to the Supply of one subcommodity at a time."

Empirical results have shown that a technique that will update the link performance vector (LPV) while the assignment process is going on should be preferred to method A, because it has a tendency to produce a better distribution of the volumes and flows and so to avoid to a certain extent over-capacity and oscillations. Moreover, any method which will update the LPV as the assignment goes on can be coupled with method A and the iteration procedure. It is always possible to make iterations at the completion of one run as explained above in method A.

In the following approach (called method B), it is suggested that the modifications take advantage of the procedure used to carry on the assignment in the model. Indeed, it is suggested that one form of the increments can be defined as the flows generated by the supply and demand data on a given subcommodity. Thus for a given season and a given subcommodity, the R-factors corresponding to it will be calculated. In the routing, mode choice, distribution and assignment of that subcommodity, the R-factors just mentioned should be used to generate the flows. As these are assigned to the network, they will contribute to changes in the level of service variables. After the assignment
of the flows from the supply nodes to all the demand nodes for that subcommodity, the link performance vector of each link in the network will be updated. This should be done using the proper cost performance models, executed under the control of the subroutine LNKOST. In this approach the main objection that is going to arise to such a procedure is that the routes used by a subcommodity are dependent on the sequence into which the subcommodities are processed regardless of the fact that this sequence is determined randomly or not. In fact, as soon as a subcommodity flow has been assigned to certain links, the next subcommodity will experience new values of the level of service variables and the routes chosen will then be influenced by the preceding subcommodity assignments.

In order to carry on the implementation of this method, it is necessary that changes be made in the calculations of the R-factors in RCOMP. Indeed it will not be necessary to calculate the R-factors for all subcommodities but only for the one that is being considered at any given moment. In the section of the algorithm concerned with the calculations of the trees, they will be found for all the supply nodes of a given subcommodity. Also, as mentioned above, it is not feasible to perform the updating of the link file LNTAPE after the assignment of the flows generated by one supply node only. Instead this will be done after the assignment of the flows for all the supply nodes. The best way to achieve this file update is to use the subroutine LNKOST and have it called by the subroutine ADDON after it has completed the flow assignment to the different links in the link file. The subroutine LNKOST will be used to update the link performance part of the link file.

It is possible to simplify the work involved in making this updating of LNTAPE. In fact, instead of using LNKOST with all the extended sophisticated technological models, one can develop some crude and simplified models to
be used for updating purposes at that stage and which would require less
calculations than LNKOST. After that all the subcommodities have been assigned,
the costing part of the Harvard model is called. Only then would the sophis-
ticated models be used to update LNTAPE. This would complete the work done
on this subcommodity into this season. The whole process should then be
repeated for the next subcommodity.

5.2.2 "Increments equal to a given percentage of the supply of all subcommodities
at a time."

The different treatment that each subcommodity receives under the use of
method B was a major argument to postponing the implementation until sufficient
experience was gained with the usage of the iteratives methods described
before but above all until sufficient empirical results are obtained with the
following approach, called method C. In case C, the cycles on each season and
on each subcommodity in the assignment process will not be changed at all.
But, each time that the system will consider the supply and demand of a
subcommodity, only a given percentage of the total supply data is going to be
assigned. After a given percentage, $\Delta$, of all the subcommodities have been
assigned for all seasons and after the link utilization vectors have been up-
dated in ADDON to account for the flows generated by that $\Delta$ per cent, the
link performance vector of each link in the file LNTAPE are updated using the
subroutine LNKOST with all the so-called sophisticated technological models.
It must be noted that the sequence into which the subcommodities are processed
is going to be random and thus the sequence for the assignment of the first
$\Delta$ % will be different than for the second $\Delta$ % and so on. Thus, the system
will assign consecutively $\Delta$ % of all subcommodities supply and the new
flows generated at each time will be added to the flows of the preceeding
cycle, where a \( \Delta \% \) of the supply was assigned. The increments used or the value of \( \Delta \) is under the control of the user and can take any value compatible with the time required to perform the assignment. It is clear that the smaller \( \Delta \) is, the better the assignment should be. However, more computer time is needed to carry on the assignment in this case.

The tasks involved in the implementation of the method C will require that the matrix of flows for each vehicle class and each link be stored after each assignment of amount \( \Delta \), because the cost sector of the transport model is overlaid over the assignment process subroutines. In addition, the subroutine ADDON must be modified to account for the fact that second and consecutive flow assignment will be added to the preceding flow in order that the total cumulative flow be obtained for every link. The basic logic of the system will stay as it is. Also the fact that, in each assignment of amount \( \Delta \), the sequence into which the subcommodities are assigned is random, this is going to remove some form of bias that the link assignment process might have with regard to subcommodity sequence. In method B, this bias can be removed by assigning only an amount equal to \( \Delta \% \) of the supply of each subcommodity instead of the 100 \% as suggested. Then one can select at random the sequence used in method C; though this is perfectly correct, it is not feasible to attempt this because of economical reasons. In fact, if method C and B use the same increment \( \Delta \), then method B will require a lot more time to complete the updating part of the process, since the updating is done after each subcommodity assignment with the sophisticated technological models. This may not be the case though, if the number of subcommodities considered is small say 5 or 6, but more experimental work would be needed to decide.
5.3 "R-factors, average R-factors and their use."

It is a known fact that, in the transport sector, the part of the Harvard model dealing with the calculations of the trees for each supply node of each subcommodity in a given season accounts for an important percentage of the total computer time for a run. This is one of the reasons why investigations of the average R-factors is done. Some of the other reasons can be summarized as follows. First, the introduction of the direct demand models has increased the number of trees that must be calculated since routes are obtained over each mode. Second, the introduction of method A and method C in order to improve on the capacity and oscillation problems will add to the amount of computer time needed for a run. In method C for instance, the number of trees actually calculated with this method will be a multiple equal to the number of cycles needed to assign 100% at a rate of $\Delta\%$ per cycle. Similarly the iteration procedure will require as many more times the original number of tree calculation as there are iterations made.

To calculate the average R-factors, a subcommodity preference vector (CPV) is suggested where the values are all equal to 1.00 for each subcommodity. Using this special CPV, the resulting R-factors for each subcommodity should be calculated. It is important to observe that, since each subcommodity is assigned to a vehicle class and that the five components of the link performance vectors are different for each vehicle class, the number of average R-factors for a given link will be equal to the number of vehicle class used by all the subcommodities. In order to see the advantages that will be available with these average R-factors, let us see by how much computer time will be reduced for one run. The Harvard model determines a tree for each supply node of each subcommodity. If many subcommodities have the same
supply nodes in common, as is the case in many situations, then the actual
algorithm will calculate a tree each time a node is the supply node of a
given subcommodity. All the nodes in the network which are the supply nodes of at
least one subcommodity are found. Then for these nodes, the modified algorithm
should determine a tree for each one of them for as many times as there are
subcommodities having these nodes as supply node and making use of different
vehicle class. In order to do that, the modified approach will use average
R-factors. If a node is a supply node of two subcommodities, then the routes
and the modes used in shipping any quantity of supply to a common demand node
are going to be the same if the two subcommodities use the same vehicle class.
It is important to realize that the quantities going over each route and the
mode split is not going to be in the same proportion for the two subcommodities
because, at the distribution level, the R-sums, (RSUM) the waiting time sums
(WTSM), the travel time sums (TTSUM), the variance of travel time sums (VTTSUM),
the probability of loss sums (PLSSUM), the cost sums (CSUM) will be obtained
for each subcommodity, by applying the proper CPV to the values of the corre-
ponding quantities obtained with the special CPV. Thus the savings in terms
of computer time come from the fact that only one tree will be built for a given
node in most cases.

After sufficient experience and empirical knowledge have been gained with
the direct demand models and method C, the implementation of the changes
necessary to use the average R-factors will then be done. They will be
significant changes in the overall logic of the actual system. In order to
incorporate them into the transport sector with the least amount of modifications,
the following tasks must be done. The first step if it is done now will
compute the R-factors file RFCTAP but with the special CPV. This should be
done for all subcommodities. The next step is the use of a special subroutine
that will find out all the nodes that are supply nodes at least once. Then a tree for each one of these nodes should be calculated and stored with the RSUM, WTSUM, TTSSUM, PLSSUM, CSUM. These calculations should be performed for a given node as many times as this node is used as a supply node of subcommodities using different vehicle classes. These tasks and the execution of this new subroutine should be done before entering NETFLO. Now, as NETFLO begins to carry on the work sequentially on subcommodities, the actual tasks concerned with the tree building and the calculations of the matrix CUMR, CUMWT, CUMTT, CUMVTT, CUMPLS, CUMC will be optional depending on whether the R-factors are going to be used or the average R-factors. In this last case, a search of the file on TRETAP, which contains RSUM, WTSUM, TTSSUM, VTTSUM, PLSSUM and CSUM, will be made and the calculations of the proper matrix of level of service variables will be performed for the subcommodity being looked at this particular moment. These matrices CUMR, CUMWT, CUMTT, CUMVTT, CUMPLS, CUMC will be obtained from their corresponding sums above by applying to them the proper values of the subcommodity preference vector as specified for each subcommodity. Once this is complete, the calculations will proceed without additional changes up to the assignment stage for that subcommodity. When the assignment is begun, the only difference between the R-factors and average R-factors approaches will be that the trees corresponding to any row of the flow matrix will have to be searched, in the case of average R-factors, on the file TRETAP instead of having only these for this subcommodity with the R-factors case. These changes will affect the routing, mode choice, distribution and assignment in the direct demand part of the system as well as in the actual systems. This can be seen from the system macroflow chart (figures 1, 2, 3, 4).
Chapter VI "Numerical Evaluation of the modified system."

6.1 "Data used for evaluation"

To evaluate the modifications to the Harvard Brookings model discussed and analysed in the preceding chapters, a considerable amount of data is needed. The data is of two main kinds: 1) the supply and demand data for a given set of subcommodities and 2) transportation network data. In order to keep the costs of data collection down as much as possible and to stay within the limits of a rather small computer time budget, the transportation network had to be a small one. Instead of defining a hypothetical network without real world content for use as the base for the tests, the network previously used in a study done for the African State of Niger was employed. This strategy also provided comparison runs of the original system on the same data. These results could be used to guide the testing phase of the modifications and additions to the Harvard model.

Following a 1970 agreement between the Canadian International Development Agency (CIDA) and the consulting firm Lamarre Valois International, a study was made of the different transportation systems used by the African State of Niger in the freight commodity movements generated by its foreign trade. A main objective of the study was to evaluate the present and future needs of the state in the transportation of imported and exported commodities with particular attention being given to possible access to the ocean of Niger. A general strategy for the development of transportation systems that would deal with the foreign trade transportation problems of Niger was defined.

The results of the study were a set of recommendations on the choice of the most advantageous general transportation system to serve the State of Niger.
The details of the study and its results were reported in (7). The study dealt with the evaluation of the actual and proposed systems with respect to the nature and the extent of the foreign trade of Niger. Principal interest was devoted to certain categories of commodities and to the transportation systems available and utilized by these commodities. Another task was to use available data and studies done by other agencies to forecast the expected volumes of foreign trade with the State of Niger. This was done for the main categories of good over their origins and destinations, as well as their volumes and values.

In parallel with the preceding task, the study had to look at possible transportation systems alternatives for carrying the flows generated by these commodities with particular attention given to ocean access.

6.1.1 "Network used for evaluation"

To test the additions and changes to the model, the network described in the study above was used. This network has highway links, rail links, water links and transfer links between these different modes. The total number of links is 108 and among these there are 66 highways, 6 rails, 6 water and 30 transfer links. The total number of nodes in the network is 44 and nodes 1 to 32 are on highway links, nodes 33 to 38 are on rail links while nodes 39 to 42 are on the water links. The remaining two nodes are on transfer links. The maps 1 and 2, that follow, give a good picture of the network. These maps are reproduced from the report mentioned above in reference (7). The map 1 is concerned with the highway links and gives a classification of the different links in the highway network. The map 2 gives a complete picture of the overall network with the names of the different nodes in it, their location on a map of the State of Niger and of the neighboring states of Nigeria, Dahomey, Togo, Ghana, Algeria, and the Ivory Coast.
Map 1 - Road Classification Map
Map 2 - Transport Network Used
In table 1, one can find the main characteristics of each link in the network. In fact, the name of the place represented by the node at the ends of each link is identified with the node number. The length in miles of each link is also given with an index showing the class number into which this link is classified and the possible class to which this link is expected to be upgraded. For the year 1973, 1980 and 1990 an estimate is given of the local traffic in millions of tons miles for each one of these links and each one of these years.

The cost constants A and B shown in Table 1 are related to a formula defined in (7) in sections 3.2 and 5.2 to determine the annual cost of transportation to Niger resulting from external trade on an import or export route. The factor A is related to the annual fixed charges of the investments on that link while the factor B is there to take account of the variable costs of transportation. This relation serves the purpose of distributing the infrastructure costs in an acceptable proportion between the local traffic and the external traffic based on the actual external annual tonnages using this link.

It should be noted that some of the links are, in fact, multiple links. For example, the link 1-9 is in fact two links, the link 1-10 from Niamey to Koupela and the link 10-9 from Koupela to Ouagadougou. This can be verified using map 2.

All the network data, used in the tests, is listed and given in appendix 2. It is part of the input along with the supply and demand data to be discussed next.

6.1.2 "Supply and demand data used"

The Niger economy is essentially based on agricultural production and this sector accounts for almost 60% of the gross national product. To arrive at
<table>
<thead>
<tr>
<th>Vecteur No.</th>
<th>Vecteur nom</th>
<th>Longueur en milles</th>
<th>En 1973</th>
<th>Amél. consid.</th>
<th>Constantes de coût</th>
<th>Trafic local en millions de tonnes-milles</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8</td>
<td>Niamey - Dosso</td>
<td>86.6</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>1-9</td>
<td>Niamey - Ouagadougou</td>
<td>324.0</td>
<td>2</td>
<td>1</td>
<td>2,000</td>
<td>4,400</td>
</tr>
<tr>
<td>8-11</td>
<td>Dosso - Gaya</td>
<td>101.0</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>11-14</td>
<td>Gaya - Parakou</td>
<td>180.0</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>14-26</td>
<td>Parakou - Bohicon</td>
<td>185.0</td>
<td>3</td>
<td></td>
<td>4,400</td>
<td>.0490</td>
</tr>
<tr>
<td>26-16</td>
<td>Bohicon - Cotonou</td>
<td>70.0</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>4-8</td>
<td>Dosso - Birni N'konni</td>
<td>175.0</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>4-2</td>
<td>Birni N'konni - Maradi</td>
<td>150.7</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>2-12</td>
<td>Maradi - Kano</td>
<td>167.0</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>2-23</td>
<td>Maradi - Takieta</td>
<td>114.8</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>3-23</td>
<td>Zinder - Takieta</td>
<td>33.2</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.0238</td>
</tr>
<tr>
<td>12-23</td>
<td>Kano - Takieta</td>
<td>138.3</td>
<td>1</td>
<td></td>
<td>4,400</td>
<td>.028</td>
</tr>
</tbody>
</table>

Table 1 - Characteristics of the Links
### CARACTERISTIQUES DES VECTEURS DU SYSTEME DE TRANSPORT (suite)

<table>
<thead>
<tr>
<th>Vecteur no.</th>
<th>Vecteur nom</th>
<th>Longueur en milles</th>
<th>Constantes de coût</th>
<th>Trafic local en millions de tonnes-mille</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHEMIN DE FER.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33-38</td>
<td>Ouagadougou Abidjan</td>
<td>715.0</td>
<td>11,100</td>
<td>.0147</td>
</tr>
<tr>
<td></td>
<td>Abidjan</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34-37</td>
<td>Kano Lagos</td>
<td>690.0</td>
<td>0</td>
<td>.0235*</td>
</tr>
<tr>
<td>35-36</td>
<td>Parakou</td>
<td>272.0</td>
<td>5,760</td>
<td>.0133</td>
</tr>
<tr>
<td>NAVIGATION</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39-40</td>
<td>Niamey Gaya</td>
<td>183.0</td>
<td>1,120</td>
<td>7 mois</td>
</tr>
<tr>
<td>40-41</td>
<td>Gaya Yelwa</td>
<td>129.0</td>
<td>890</td>
<td>11 mois</td>
</tr>
<tr>
<td>41-42</td>
<td>Yelwa Warri</td>
<td>728.0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

* Ce coût a été supposé indépendant de l'apport de trafic du Niger et constant dans le temps.
CARACTÉRISTIQUES DES VECTEURS DU SYSTÈME DE TRANSPORT (suite)

<table>
<thead>
<tr>
<th>Vecteur No.</th>
<th>Vecteur nom</th>
<th>Longueur en milles</th>
<th>Classe</th>
<th>Améli. consid.</th>
<th>Constantes de coût sans améli.</th>
<th>Constantes de coût avec améli.</th>
<th>Trafic local en millions de tonnes-milles</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-22</td>
<td>Zinder</td>
<td>67.0</td>
<td>3</td>
<td>0</td>
<td>.0490</td>
<td>.00238</td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>Sabonkafi</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.80</td>
</tr>
<tr>
<td>22-19</td>
<td>Sabonkafi</td>
<td>44.6</td>
<td>2</td>
<td>2,000</td>
<td>.0412</td>
<td>.00238</td>
<td>2.52</td>
</tr>
<tr>
<td></td>
<td>Tchin Garagen</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.52</td>
</tr>
<tr>
<td>19-18</td>
<td>Tchin Garagen</td>
<td>127.2</td>
<td>4</td>
<td>0</td>
<td>.0991</td>
<td>.00238</td>
<td>4.25</td>
</tr>
<tr>
<td></td>
<td>Abelama</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4.25</td>
</tr>
<tr>
<td>18-7</td>
<td>Abelama</td>
<td>39.7</td>
<td>2</td>
<td>2,000</td>
<td>.0417</td>
<td>.00238</td>
<td>1.33</td>
</tr>
<tr>
<td></td>
<td>Agadez</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.33</td>
</tr>
<tr>
<td>7-5</td>
<td>Agadez</td>
<td>154.</td>
<td>3</td>
<td>1</td>
<td>4,400</td>
<td>.0490</td>
<td>0.86</td>
</tr>
<tr>
<td></td>
<td>Arlit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.86</td>
</tr>
<tr>
<td>7-17</td>
<td>Agadez</td>
<td>37.2</td>
<td>3</td>
<td>1</td>
<td>4,400</td>
<td>.0490</td>
<td>.84</td>
</tr>
<tr>
<td></td>
<td>Assaouas</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.84</td>
</tr>
<tr>
<td>17-20</td>
<td>Assaouas</td>
<td>205.5</td>
<td>3</td>
<td>1</td>
<td>4,400</td>
<td>.0490</td>
<td>1.15</td>
</tr>
<tr>
<td></td>
<td>Ibeceten</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.15</td>
</tr>
<tr>
<td>20-21</td>
<td>Ibeceten</td>
<td>31.6</td>
<td>3</td>
<td>1</td>
<td>4,400</td>
<td>.0490</td>
<td>1.63</td>
</tr>
<tr>
<td></td>
<td>Tahoua</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.63</td>
</tr>
<tr>
<td>21-4</td>
<td>Tahoua</td>
<td>74.5</td>
<td>3</td>
<td>1</td>
<td>4,400</td>
<td>.0490</td>
<td>6.77</td>
</tr>
<tr>
<td></td>
<td>Birni N'Konni</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6.77</td>
</tr>
<tr>
<td>11-13</td>
<td>Gaya</td>
<td>115.0</td>
<td>2</td>
<td>2,000</td>
<td>.0412</td>
<td>.00238</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>Yelwa</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0</td>
</tr>
<tr>
<td>10-27</td>
<td>Koupela</td>
<td>222.0</td>
<td>2</td>
<td>1</td>
<td>4,400</td>
<td>.0412</td>
<td>30.0</td>
</tr>
<tr>
<td></td>
<td>Kande</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>30.0</td>
</tr>
<tr>
<td>27-15</td>
<td>Kande</td>
<td>308.0</td>
<td>1</td>
<td>4,400</td>
<td>.0238</td>
<td>.00238</td>
<td>41.6</td>
</tr>
<tr>
<td></td>
<td>Lomé</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>41.6</td>
</tr>
</tbody>
</table>
the supply and demand data necessary to run the model, a study of the different sectors of production was performed. The tonnages for different commodities that were exported in the years 1962 to 1968 and the forecasts for 1980 is given in the table 2. This table is a reproduction of the one given in (7). Similarly, the imports of the State of Niger were analysed for the same years above for the main goods imported into the country. The results are reproduced in table 3 for the main tonnages of the most important goods imported by Niger. This analysis is a first step toward a table of the regions of origin and destination of the main goods that are exported. These results are summarized in table 4. Table 5 is a similar table of tonnages of the imported goods by country of origin and region of destination. This is done in order to arrive, in a final stage, to supply and demand data which is assigned for each commodity to the nodes of the transportation network. The imported and exported goods are then grouped into 11 different categories on the basis of their physical characteristics, their cost and their origin. The goods that belong to the different categories and their tonnages are reported in table 6. The computer names of these 11 commodities are URAN, NIEB, ARAC, UILE, EXPO, FUEA, FUEL, IBVS, IHVS, IBVA, IHVA. The final step is to assign the tonnages for each of these 11 commodities to nodes in the network. The final result of this task, based on the preceding tables mentioned, is given in appendix 2 in the listing of the supply and demand computer input data. Each field in this data set is identified in this appendix so that it is easy to go from table 6 to this listing.

6.2 "Results of the tests"

This section shows the results of the tests performed on the new approach.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Bovins</td>
<td>2,044</td>
<td>20,304</td>
<td>16,203</td>
<td>22,189</td>
<td>20,200</td>
<td>19,507</td>
<td>21,302</td>
<td>59,400</td>
</tr>
<tr>
<td>Ovins et Caprins</td>
<td>2,520</td>
<td>2,011</td>
<td>2,168</td>
<td>2,140</td>
<td>1,342</td>
<td>835</td>
<td>10,120</td>
<td></td>
</tr>
<tr>
<td>Poisson</td>
<td>2,044</td>
<td>1,320</td>
<td>532</td>
<td>78</td>
<td>0</td>
<td>0</td>
<td>80</td>
<td>660</td>
</tr>
<tr>
<td>Oignons frais</td>
<td>6,257</td>
<td>1,680</td>
<td>1,246</td>
<td>4,432</td>
<td>3,925</td>
<td>3,331</td>
<td>2,561</td>
<td>3,850</td>
</tr>
<tr>
<td>Oignons deh.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>840</td>
</tr>
<tr>
<td>Haricots (Nièbe)</td>
<td>15,564</td>
<td>8,283</td>
<td>2,820</td>
<td>15,858</td>
<td>4,512</td>
<td>6,284</td>
<td>14,108</td>
<td>110,000</td>
</tr>
<tr>
<td>Mil et Sorgho</td>
<td>3,805</td>
<td>3,200</td>
<td>824</td>
<td>1,459</td>
<td>0</td>
<td>0</td>
<td>65</td>
<td>11,000</td>
</tr>
<tr>
<td>Farine de Mil</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>49,500</td>
</tr>
<tr>
<td>Riz</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>11,000</td>
</tr>
<tr>
<td>Sucre</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>11,000</td>
</tr>
<tr>
<td>Arachides décort.qu.</td>
<td>76,259</td>
<td>100,502</td>
<td>82,684</td>
<td>95,042</td>
<td>179,924</td>
<td>193,600</td>
<td>178,922</td>
<td>49,500</td>
</tr>
<tr>
<td>Huile d’arachides</td>
<td>2,378</td>
<td>3,455</td>
<td>6,040</td>
<td>5,292</td>
<td>9,668</td>
<td>8,229</td>
<td>4,171</td>
<td>62,500</td>
</tr>
<tr>
<td>Tourteau d’arachides</td>
<td>6,081</td>
<td>4,404</td>
<td>19,303</td>
<td>0</td>
<td>17,938</td>
<td>8,232</td>
<td>6,057</td>
<td>11,000</td>
</tr>
<tr>
<td>Cuir et Peaux</td>
<td>541</td>
<td>463</td>
<td>437</td>
<td>620</td>
<td>484</td>
<td>398</td>
<td>554</td>
<td>8,660</td>
</tr>
<tr>
<td>Colza égrené</td>
<td>1,170</td>
<td>2,142</td>
<td>2,096</td>
<td>2,243</td>
<td>2,847</td>
<td>3,196</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Viande fraiche</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>657</td>
<td>532</td>
</tr>
<tr>
<td>Uranium et autres min</td>
<td>7</td>
<td>67</td>
<td>68</td>
<td>69</td>
<td>89</td>
<td>99</td>
<td>110</td>
<td>3,400</td>
</tr>
<tr>
<td>Légumes et fruits</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1,100</td>
</tr>
<tr>
<td>Divers</td>
<td>7,710</td>
<td>3,463</td>
<td>2,541</td>
<td>47,103</td>
<td>11,094</td>
<td>6,963</td>
<td>3,862</td>
<td>0</td>
</tr>
<tr>
<td>Exportations totales</td>
<td>122,890</td>
<td>150,831</td>
<td>136,851</td>
<td>196,406</td>
<td>252,874</td>
<td>251,326</td>
<td>236,125</td>
<td>414,540</td>
</tr>
</tbody>
</table>

Table 2 - Recent Exports and Forecast of Exports
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Lait, Produits Laitiers, Oeufs</td>
<td>442</td>
<td>314</td>
<td>424</td>
<td>479</td>
<td>869</td>
<td>923</td>
<td>788</td>
<td>1,418</td>
</tr>
<tr>
<td>Fruits Tropicaux</td>
<td>3,342</td>
<td>1,702</td>
<td>536</td>
<td>936</td>
<td>101</td>
<td>185</td>
<td>184</td>
<td>234</td>
</tr>
<tr>
<td>Farine de Froment</td>
<td>3,355</td>
<td>1,607</td>
<td>1,581</td>
<td>1,598</td>
<td>3,956</td>
<td>3,875</td>
<td>3,082</td>
<td>3,905</td>
</tr>
<tr>
<td>Noix de Cola</td>
<td>2,886</td>
<td>944</td>
<td>1,255</td>
<td>3,306</td>
<td>3,848</td>
<td>4,028</td>
<td>2,886</td>
<td>4,400</td>
</tr>
<tr>
<td>Sucre</td>
<td>7,451</td>
<td>7,101</td>
<td>6,830</td>
<td>7,102</td>
<td>9,257</td>
<td>7,889</td>
<td>3,740</td>
<td>0</td>
</tr>
<tr>
<td>Préparation de Céréales</td>
<td>416</td>
<td>398</td>
<td>282</td>
<td>425</td>
<td>495</td>
<td>551</td>
<td>424</td>
<td>798</td>
</tr>
<tr>
<td>Préparation de Légumes et Fruits</td>
<td>652</td>
<td>486</td>
<td>554</td>
<td>450</td>
<td>679</td>
<td>860</td>
<td>895</td>
<td>3,410</td>
</tr>
<tr>
<td>Bière</td>
<td>5,121</td>
<td>2,347</td>
<td>2,126</td>
<td>2,500</td>
<td>2,103</td>
<td>2,429</td>
<td>1,869</td>
<td>220</td>
</tr>
<tr>
<td>Vin, Alcool et Vinaigre</td>
<td>2,729</td>
<td>1,887</td>
<td>1,621</td>
<td>1,556</td>
<td>1,826</td>
<td>1,799</td>
<td>1,403</td>
<td>2,184</td>
</tr>
<tr>
<td>Ciment Hydraulique</td>
<td>18,820</td>
<td>10,762</td>
<td>14,816</td>
<td>14,298</td>
<td>6,838</td>
<td>176</td>
<td>8</td>
<td>9,900</td>
</tr>
<tr>
<td>Sel</td>
<td>8,405</td>
<td>6,102</td>
<td>7,376</td>
<td>5,546</td>
<td>9,110</td>
<td>14,988</td>
<td>9,216</td>
<td>16,500</td>
</tr>
<tr>
<td>Hydrocarbures</td>
<td>25,760</td>
<td>38,083</td>
<td>37,951</td>
<td>34,050</td>
<td>47,210</td>
<td>45,558</td>
<td>49,206</td>
<td>201,300</td>
</tr>
<tr>
<td>Spécialités Pharmaceutiques</td>
<td>154</td>
<td>70</td>
<td>133</td>
<td>55</td>
<td>121</td>
<td>261</td>
<td>175</td>
<td>1,540</td>
</tr>
<tr>
<td>Savon</td>
<td>760</td>
<td>635</td>
<td>527</td>
<td>1,378</td>
<td>1,133</td>
<td>879</td>
<td>189</td>
<td>2,750</td>
</tr>
<tr>
<td>Pneumatiques</td>
<td>328</td>
<td>281</td>
<td>1,053</td>
<td>377</td>
<td>485</td>
<td>717</td>
<td>715</td>
<td>7,645</td>
</tr>
<tr>
<td>Bois et Ouvrages en Bois</td>
<td>891</td>
<td>623</td>
<td>849</td>
<td>881</td>
<td>1,753</td>
<td>1,626</td>
<td>1,904</td>
<td>5,995</td>
</tr>
<tr>
<td>Papier et Carton</td>
<td>631</td>
<td>399</td>
<td>685</td>
<td>831</td>
<td>981</td>
<td>802</td>
<td>1,056</td>
<td>3,696</td>
</tr>
<tr>
<td>Fil, Tissus et Vêtements</td>
<td>3,127</td>
<td>2,395</td>
<td>6,466</td>
<td>6,315</td>
<td>5,393</td>
<td>6,288</td>
<td>4,513</td>
<td>2,200</td>
</tr>
<tr>
<td>Sacs et Sachets</td>
<td>1,177</td>
<td>1,907</td>
<td>2,410</td>
<td>2,511</td>
<td>3,034</td>
<td>5,088</td>
<td>2,316</td>
<td>3,025</td>
</tr>
<tr>
<td>Produits Métalliques</td>
<td>4,094</td>
<td>4,348</td>
<td>6,632</td>
<td>5,478</td>
<td>9,391</td>
<td>6,261</td>
<td>5,810</td>
<td>32,890</td>
</tr>
<tr>
<td>Machines et Appareils Non Electriques</td>
<td>638</td>
<td>439</td>
<td>699</td>
<td>866</td>
<td>1,547</td>
<td>1,003</td>
<td>1,986</td>
<td>3,685</td>
</tr>
<tr>
<td>Appareils et Moteurs Electriques</td>
<td>591</td>
<td>712</td>
<td>920</td>
<td>254</td>
<td>920</td>
<td>704</td>
<td>733</td>
<td>2,200</td>
</tr>
<tr>
<td>Automobiles, Autres Véhicules et Pièces</td>
<td>2,208</td>
<td>1,374</td>
<td>2,318</td>
<td>2,868</td>
<td>2,547</td>
<td>3,874</td>
<td>4,431</td>
<td>12,980</td>
</tr>
<tr>
<td>Produits Chimiques</td>
<td>336</td>
<td>266</td>
<td>605</td>
<td>458</td>
<td>582</td>
<td>49,775</td>
<td>1,540</td>
<td>5,995</td>
</tr>
<tr>
<td>Divers</td>
<td>16,518</td>
<td>4,220</td>
<td>7,758</td>
<td>12,367</td>
<td>14,906</td>
<td>19,470</td>
<td>18,013</td>
<td>33,000</td>
</tr>
</tbody>
</table>

**Importations totales** 111,267 89,629 106,627 107,283 129,500 130,516 116,011 406,114

*Table 3 - Recent Imports and Forecast of Imports*
<table>
<thead>
<tr>
<th>Produits</th>
<th>Saisons</th>
<th>Exportations Utilisant Un Mode de Transport Moderne</th>
<th>Exportations</th>
<th>ORIGINES</th>
<th>DESTINATIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>%</td>
<td>%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bovins</td>
<td>50 50</td>
<td>59,400</td>
<td>1,480</td>
<td>330</td>
<td>1,480</td>
</tr>
<tr>
<td>Ovins et caprins</td>
<td>– –</td>
<td>10,120</td>
<td>0</td>
<td>385</td>
<td>390 130 140</td>
</tr>
<tr>
<td>Poissons</td>
<td>60 40</td>
<td>660</td>
<td>660</td>
<td>160</td>
<td>500</td>
</tr>
<tr>
<td>Oignons frais</td>
<td>90 10</td>
<td>3,850</td>
<td>3,850</td>
<td>1,650</td>
<td>2,200</td>
</tr>
<tr>
<td>Oignons deshydratés</td>
<td>60 40</td>
<td>840</td>
<td>840</td>
<td>2,750</td>
<td>2,750</td>
</tr>
<tr>
<td>Haricots</td>
<td>1 39</td>
<td>110,000</td>
<td>110,000</td>
<td>49,500</td>
<td>22,000</td>
</tr>
<tr>
<td>Mil et sorgho</td>
<td>20 80</td>
<td>11,000</td>
<td>11,000</td>
<td>11,000</td>
<td>11,000</td>
</tr>
<tr>
<td>Farine de mil</td>
<td>50 50</td>
<td>49,500</td>
<td>49,500</td>
<td>5,500</td>
<td>5,500</td>
</tr>
<tr>
<td>Riz</td>
<td>60 40</td>
<td>11,000</td>
<td>11,000</td>
<td>11,000</td>
<td>11,000</td>
</tr>
<tr>
<td>Sucre</td>
<td>60 40</td>
<td>11,000</td>
<td>11,000</td>
<td>11,000</td>
<td>11,000</td>
</tr>
<tr>
<td>Arachides décortiquées</td>
<td>90 10</td>
<td>49,500</td>
<td>49,000</td>
<td>8,800</td>
<td>13,200</td>
</tr>
<tr>
<td>Huile brute d'arachides</td>
<td>60 40</td>
<td>62,500</td>
<td>62,000</td>
<td>20,020</td>
<td>62,700</td>
</tr>
<tr>
<td>Tourteaux d'arachides</td>
<td>60 40</td>
<td>11,000</td>
<td>11,000</td>
<td>3,300</td>
<td>11,000</td>
</tr>
<tr>
<td>Cuir</td>
<td>60 40</td>
<td>1,620</td>
<td>1,620</td>
<td>80</td>
<td>1,140 1,270 150 30 160</td>
</tr>
<tr>
<td>Peaux</td>
<td>60 40</td>
<td>7,040</td>
<td>5,640</td>
<td>920</td>
<td>2,200 1,100 4,400 3,300</td>
</tr>
<tr>
<td>Viande</td>
<td>60 40</td>
<td>11,000</td>
<td>11,000</td>
<td>11,000</td>
<td>2,200 1,100 4,400 3,300 2,200 1,100 4,400 3,300</td>
</tr>
<tr>
<td>Uranium et autres minéraux</td>
<td>60 40</td>
<td>3,410</td>
<td>3,400</td>
<td>1,030</td>
<td>3,400</td>
</tr>
<tr>
<td>Légumes et fruits</td>
<td>99 1</td>
<td>1,100</td>
<td>1,100</td>
<td>1,100</td>
<td>550</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>414,540</td>
<td>345,090</td>
<td>139,340</td>
<td>39,270 56,675 104,710 181,650 42,720 4,560 35,200 28,760 46,890</td>
</tr>
</tbody>
</table>

Table 4 - Exports for 1980 by Origins and Destinations
<table>
<thead>
<tr>
<th>Produits</th>
<th>Importations Totales</th>
<th>DESTINATIONS</th>
<th>ORIGINES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Ouest</td>
<td>Centre-ouest</td>
</tr>
<tr>
<td>Lait, Produits Laitiers, Œufs</td>
<td>1,418</td>
<td>792</td>
<td>143</td>
</tr>
<tr>
<td>Fruits Tropicaux</td>
<td>234</td>
<td>85</td>
<td>47</td>
</tr>
<tr>
<td>Farine de Froment</td>
<td>3,905</td>
<td>2,147</td>
<td>431</td>
</tr>
<tr>
<td>Noix de Cola</td>
<td>4,400</td>
<td>1,584</td>
<td>880</td>
</tr>
<tr>
<td>Sucre</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Préparations de Céréales</td>
<td>798</td>
<td>287</td>
<td>160</td>
</tr>
<tr>
<td>Préparations de Fruits et Légumes</td>
<td>3,410</td>
<td>1,228</td>
<td>682</td>
</tr>
<tr>
<td>Bière</td>
<td>220</td>
<td>121</td>
<td>22</td>
</tr>
<tr>
<td>Vins, Alcools, Vinaigre</td>
<td>2,184</td>
<td>1,201</td>
<td>237</td>
</tr>
<tr>
<td>Cigarettes</td>
<td>464</td>
<td>255</td>
<td>52</td>
</tr>
<tr>
<td>Ciment</td>
<td>9,900</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sel</td>
<td>16,500</td>
<td>5,940</td>
<td>3,300</td>
</tr>
<tr>
<td>Hydrocarbures</td>
<td>201,300</td>
<td>99,165</td>
<td>11,000</td>
</tr>
<tr>
<td>Spécialités Pharmaceutiques</td>
<td>1,540</td>
<td>847</td>
<td>169</td>
</tr>
<tr>
<td>Savons et Détectgents</td>
<td>2,750</td>
<td>2,640</td>
<td>0</td>
</tr>
<tr>
<td>Pneumatiques</td>
<td>7,645</td>
<td>4,400</td>
<td>220</td>
</tr>
<tr>
<td>Bois et Ouvrages en Bois</td>
<td>5,955</td>
<td>2,156</td>
<td>1,199</td>
</tr>
<tr>
<td>Papier et Carton</td>
<td>3,696</td>
<td>2,090</td>
<td>407</td>
</tr>
<tr>
<td>Fils, Tissus et Vêtements</td>
<td>2,200</td>
<td>1,210</td>
<td>242</td>
</tr>
<tr>
<td>Sacs et Sachets</td>
<td>3,025</td>
<td>660</td>
<td>385</td>
</tr>
<tr>
<td>Produits Métalliques</td>
<td>32,890</td>
<td>14,349</td>
<td>2,041</td>
</tr>
<tr>
<td>Machines et Appareils Non Electriques</td>
<td>3,685</td>
<td>2,024</td>
<td>380</td>
</tr>
<tr>
<td>Appareils et Moteurs Electriques</td>
<td>2,700</td>
<td>1,210</td>
<td>110</td>
</tr>
<tr>
<td>Automobiles, Véhicules et Pièces</td>
<td>12,980</td>
<td>7,156</td>
<td>660</td>
</tr>
<tr>
<td>Produits Chimiques</td>
<td>49,775</td>
<td>26,397</td>
<td>103,833</td>
</tr>
</tbody>
</table>

Importations Totales | 169,801 | 103,833 | 49,900 | 56,184 | 305,263 | 64,804 | 19,415 | 14,025 | 2,607 |

*Ghana
<table>
<thead>
<tr>
<th>MARCHANDISES REELLES</th>
<th>Rapport</th>
<th>PREVISIONS</th>
<th>HYPOTHESE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Input saipiner 1973</td>
<td>1980</td>
<td>1990</td>
</tr>
<tr>
<td>Uranium</td>
<td>URAN 63/37 1,760</td>
<td>3,400</td>
<td>3,400</td>
</tr>
<tr>
<td>Niébé</td>
<td>NIEB 0/100 33,198</td>
<td>110,000</td>
<td>140,000</td>
</tr>
<tr>
<td>Arachides décortiquées</td>
<td>ARAC 90/10 104,500</td>
<td>49,500</td>
<td>0</td>
</tr>
<tr>
<td>Huile d'arachide</td>
<td>UILE 60/40 36,300</td>
<td>62,500</td>
<td>85,000</td>
</tr>
<tr>
<td>Tourteau d'arachides</td>
<td>EXPO 63/37 24,200</td>
<td>105,460</td>
<td>157,200</td>
</tr>
<tr>
<td>Mil et sorgho brut</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Farine de mil</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oignons frais</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oignons déshydratés</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Riz</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sucre</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cuir</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peaux</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hydrocarbures pour Arlit</td>
<td>FUEA 60/40 22,000</td>
<td>40,700</td>
<td>45,000</td>
</tr>
<tr>
<td>Hydrocarbures (autres destinations)</td>
<td>FUEL 60/40 68,517</td>
<td>160,600</td>
<td>412,000</td>
</tr>
<tr>
<td>Fruits tropicaux</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Farine de froment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Noix de cola</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sucre</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Préparations de céréale</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Préparations de fruits et légumes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bière</td>
<td>(2)IBVS 60/40 43,450</td>
<td>79,996</td>
<td>156,000</td>
</tr>
<tr>
<td>Vins, alcools, vinaigre</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cigarettes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ciment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sel</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Savon</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bois, ouvrages en bois</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Papiers et carton</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sacs et sachets</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Produits métalliques</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lait, produits laitiers</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spécialités pharmaceutiques</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pneumatiques</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fils, tissus, vêtements</td>
<td>(3)IHVS 60/40 37,620</td>
<td>60,944</td>
<td>27,000</td>
</tr>
<tr>
<td>Machines et app. non-électriques</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appareils et moteurs électriques</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Véhicules, automobiiles et pièces</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6 - Basic Groupings of Commodities for Computer Input
The method of building trees is evaluated for its impact on the calculations of the level of service variables matrix by mode. Using the distribution models added to the model, the flows generated by these models are evaluated. Finally, the method of calculating equilibrium flows will be tested and evaluated.

An important piece of data that is necessary in the use of the direct demand models is a set of parameters for each model. For the McLYnn and the Sarc-Kraft models, the literature and the results of some studies on passenger movements (such as those in reference (8)) can be used to guide the choice of the parameters. In the case of the Special Product Model (SPM-1), even less information is available. Though it is important to determine the parameters in such a way that we have a calibrated model as discussed in section 3.4, this was not our primary concern here but rather to make sure that the distribution of flows obtained from the various models, is a correct one given a set of parameters. Before this can be done it is necessary that the modified system be tested to insure that the algorithms and models are giving good and consistent results. The runs necessary to get a good calibrated model are in addition to the runs that were made to insure that the different models perform well numerically. The computer time that would have been needed was just not available to do all of these tasks at this stage. The parameters that were used to test the McLYnn, the Sarc-Kraft and the SPM-1 models are all listed in the appendix 1 in the BLOCK DATA subroutine. The arrays for the parameters are ALPHA, BETA, GAMMA and DELTA. They are well identified in the listing and in the next chapter.

6.2.1 "Tests on the tree and NWTREE"

The first tests on the subroutines used to build trees over specific modes were performed using a main program that reads in the necessary data
and passes control to NWTREE which then calculates the trees and the corresponding sums for each one of the six possible levels of service variables. When the NWTREE subroutine was incorporated into the transport sector, it was again tested. The system is organized such that, under control, the following output can be obtained: for any given subcommodity, for each mode and for each supply node a table of the trees can be displayed with the corresponding cumulative sums of the level of service variables. The tables 7,8,9 entitled "Minimum R-path on a given mode with a given home node" are examples of the type of output that is available. To find the minimum R-path in the table from any node N to the home node, the table is entered in the column NODE at the entry N. On the line corresponding to N, the link number of the link which belongs to the minimum R-path and leads to node N is available in the second column. The six consecutive columns of the table are the corresponding RSUM, WTSUM, TTSUM, VTTSUM, PLSSUM, CSUM. The last column INODES gives the INODES of the link identified in column two. Thus with the value of INODES as found, column one is entered at INODES. This procedure should lead ultimately to the HOME NODE and the corresponding link would be 0. The tables 7,8,9 are for 3 different modes with 3 different home nodes for the particular network used.

6.2.2 "Matrix of Level of Service Variables"

As explained in chapter two, the level of service variables corresponding to a given subcommodity and a given mode are obtained directly from the tables of minimum R-path discussed in the previous section. In fact, if one looks at any node in table 9, then the pair of nodes formed by this node and the Home node determines the CUMWT corresponding to the value of WTSUM. This is
## Table 7 - Minimum R-paths by Mode

<table>
<thead>
<tr>
<th>NODE</th>
<th>LINK</th>
<th>RSUM</th>
<th>WTSUM</th>
<th>TTSUM</th>
<th>VTTSUM</th>
<th>PLSSUM</th>
<th>CSUM</th>
<th>INODES</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>40.5</td>
<td>1.0</td>
<td>13.4</td>
<td>0.6</td>
<td>3.2</td>
<td>22.2</td>
<td>24</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>13.0</td>
<td>0.4</td>
<td>5.2</td>
<td>0.2</td>
<td>1.2</td>
<td>5.9</td>
<td>24</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>23.3</td>
<td>0.6</td>
<td>8.2</td>
<td>0.4</td>
<td>2.2</td>
<td>11.9</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>24</td>
<td>59.6</td>
<td>1.6</td>
<td>20.7</td>
<td>1.0</td>
<td>4.5</td>
<td>31.8</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>92</td>
<td>134.5</td>
<td>3.9</td>
<td>87.1</td>
<td>39.9</td>
<td>2.0</td>
<td>37.5</td>
<td>43</td>
</tr>
<tr>
<td>7</td>
<td>26</td>
<td>49.5</td>
<td>1.3</td>
<td>17.6</td>
<td>0.8</td>
<td>4.0</td>
<td>25.7</td>
<td>17</td>
</tr>
<tr>
<td>8</td>
<td>15</td>
<td>34.6</td>
<td>0.9</td>
<td>11.7</td>
<td>0.6</td>
<td>2.7</td>
<td>18.7</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>237.1</td>
<td>7.1</td>
<td>163.7</td>
<td>7.3</td>
<td>5.6</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>55.8</td>
<td>1.4</td>
<td>18.2</td>
<td>0.9</td>
<td>3.7</td>
<td>31.7</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>29</td>
<td>41.4</td>
<td>1.0</td>
<td>13.7</td>
<td>0.7</td>
<td>3.2</td>
<td>22.7</td>
<td>8</td>
</tr>
<tr>
<td>12</td>
<td>40</td>
<td>11.5</td>
<td>0.3</td>
<td>3.9</td>
<td>0.2</td>
<td>1.0</td>
<td>6.8</td>
<td>23</td>
</tr>
<tr>
<td>13</td>
<td>35</td>
<td>53.0</td>
<td>1.2</td>
<td>16.6</td>
<td>0.8</td>
<td>3.7</td>
<td>30.6</td>
<td>11</td>
</tr>
<tr>
<td>14</td>
<td>84</td>
<td>213.6</td>
<td>5.7</td>
<td>146.0</td>
<td>6.4</td>
<td>2.5</td>
<td>53.0</td>
<td>35</td>
</tr>
<tr>
<td>15</td>
<td>94</td>
<td>134.5</td>
<td>3.9</td>
<td>87.1</td>
<td>3.9</td>
<td>2.0</td>
<td>37.5</td>
<td>43</td>
</tr>
<tr>
<td>16</td>
<td>96</td>
<td>134.5</td>
<td>3.9</td>
<td>87.1</td>
<td>3.9</td>
<td>2.0</td>
<td>37.5</td>
<td>43</td>
</tr>
<tr>
<td>17</td>
<td>48</td>
<td>46.7</td>
<td>1.3</td>
<td>15.9</td>
<td>0.8</td>
<td>3.5</td>
<td>24.2</td>
<td>20</td>
</tr>
<tr>
<td>18</td>
<td>50</td>
<td>45.9</td>
<td>0.7</td>
<td>9.6</td>
<td>0.5</td>
<td>1.2</td>
<td>33.9</td>
<td>19</td>
</tr>
<tr>
<td>19</td>
<td>52</td>
<td>17.3</td>
<td>0.2</td>
<td>5.3</td>
<td>0.2</td>
<td>0.7</td>
<td>12.8</td>
<td>22</td>
</tr>
<tr>
<td>20</td>
<td>54</td>
<td>33.4</td>
<td>1.0</td>
<td>12.8</td>
<td>0.6</td>
<td>3.0</td>
<td>16.1</td>
<td>21</td>
</tr>
<tr>
<td>21</td>
<td>17</td>
<td>31.3</td>
<td>0.9</td>
<td>12.2</td>
<td>0.6</td>
<td>2.7</td>
<td>14.8</td>
<td>4</td>
</tr>
<tr>
<td>22</td>
<td>11</td>
<td>8.4</td>
<td>0.2</td>
<td>2.2</td>
<td>0.1</td>
<td>0.5</td>
<td>5.5</td>
<td>3</td>
</tr>
<tr>
<td>23</td>
<td>13</td>
<td>2.6</td>
<td>0.1</td>
<td>0.7</td>
<td>0.0</td>
<td>0.5</td>
<td>1.3</td>
<td>7</td>
</tr>
<tr>
<td>24</td>
<td>55</td>
<td>10.9</td>
<td>0.4</td>
<td>4.7</td>
<td>0.2</td>
<td>0.7</td>
<td>4.9</td>
<td>23</td>
</tr>
<tr>
<td>25</td>
<td>9</td>
<td>15.7</td>
<td>0.4</td>
<td>5.9</td>
<td>0.3</td>
<td>1.7</td>
<td>7.3</td>
<td>2</td>
</tr>
<tr>
<td>26</td>
<td>45</td>
<td>142.2</td>
<td>4.2</td>
<td>91.7</td>
<td>4.1</td>
<td>2.5</td>
<td>40.3</td>
<td>16</td>
</tr>
<tr>
<td>27</td>
<td>33</td>
<td>70.1</td>
<td>1.7</td>
<td>22.6</td>
<td>1.1</td>
<td>4.2</td>
<td>40.5</td>
<td>10</td>
</tr>
<tr>
<td>28</td>
<td>57</td>
<td>72.9</td>
<td>1.8</td>
<td>23.4</td>
<td>1.1</td>
<td>4.5</td>
<td>42.1</td>
<td>27</td>
</tr>
<tr>
<td>29</td>
<td>59</td>
<td>75.9</td>
<td>1.8</td>
<td>24.3</td>
<td>1.2</td>
<td>4.7</td>
<td>43.9</td>
<td>28</td>
</tr>
<tr>
<td>30</td>
<td>61</td>
<td>82.7</td>
<td>2.1</td>
<td>28.3</td>
<td>1.3</td>
<td>5.0</td>
<td>45.9</td>
<td>29</td>
</tr>
<tr>
<td>31</td>
<td>63</td>
<td>91.2</td>
<td>2.4</td>
<td>32.3</td>
<td>1.5</td>
<td>5.2</td>
<td>49.7</td>
<td>30</td>
</tr>
<tr>
<td>32</td>
<td>65</td>
<td>98.1</td>
<td>2.7</td>
<td>36.3</td>
<td>1.7</td>
<td>5.5</td>
<td>51.8</td>
<td>31</td>
</tr>
<tr>
<td>33</td>
<td>68</td>
<td>181.8</td>
<td>6.1</td>
<td>115.7</td>
<td>5.2</td>
<td>2.2</td>
<td>52.6</td>
<td>38</td>
</tr>
<tr>
<td>34</td>
<td>81</td>
<td>67.0</td>
<td>1.3</td>
<td>51.5</td>
<td>2.2</td>
<td>1.2</td>
<td>10.8</td>
<td>12</td>
</tr>
<tr>
<td>35</td>
<td>72</td>
<td>158.4</td>
<td>4.7</td>
<td>98.0</td>
<td>4.4</td>
<td>2.2</td>
<td>49.0</td>
<td>36</td>
</tr>
<tr>
<td>36</td>
<td>98</td>
<td>134.5</td>
<td>3.9</td>
<td>87.1</td>
<td>3.9</td>
<td>2.0</td>
<td>37.5</td>
<td>43</td>
</tr>
<tr>
<td>37</td>
<td>69</td>
<td>114.5</td>
<td>3.5</td>
<td>79.1</td>
<td>3.5</td>
<td>1.5</td>
<td>27.0</td>
<td>34</td>
</tr>
<tr>
<td>38</td>
<td>104</td>
<td>134.5</td>
<td>3.9</td>
<td>87.1</td>
<td>3.9</td>
<td>2.0</td>
<td>37.5</td>
<td>43</td>
</tr>
<tr>
<td>39</td>
<td>85</td>
<td>102.8</td>
<td>4.6</td>
<td>61.4</td>
<td>1.6</td>
<td>3.5</td>
<td>31.7</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>87</td>
<td>103.7</td>
<td>4.6</td>
<td>61.7</td>
<td>1.7</td>
<td>3.5</td>
<td>32.2</td>
<td>11</td>
</tr>
<tr>
<td>41</td>
<td>89</td>
<td>115.4</td>
<td>4.8</td>
<td>64.6</td>
<td>1.8</td>
<td>4.0</td>
<td>41.8</td>
<td>13</td>
</tr>
<tr>
<td>42</td>
<td>106</td>
<td>134.5</td>
<td>3.9</td>
<td>87.1</td>
<td>3.9</td>
<td>2.0</td>
<td>37.5</td>
<td>43</td>
</tr>
<tr>
<td>43</td>
<td>101</td>
<td>120.2</td>
<td>3.6</td>
<td>83.1</td>
<td>3.7</td>
<td>1.7</td>
<td>28.0</td>
<td>37</td>
</tr>
<tr>
<td>44</td>
<td>99</td>
<td>140.2</td>
<td>4.2</td>
<td>91.1</td>
<td>4.1</td>
<td>2.2</td>
<td>38.5</td>
<td>36</td>
</tr>
<tr>
<td>NODE</td>
<td>LINK</td>
<td>RSUM</td>
<td>WTSUM</td>
<td>TTSUM</td>
<td>VTTSUM</td>
<td>PLSSUM</td>
<td>CSUM</td>
<td>INODES</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>------</td>
<td>-------</td>
<td>-------</td>
<td>--------</td>
<td>--------</td>
<td>------</td>
<td>--------</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>30.1</td>
<td>0.8</td>
<td>10.5</td>
<td>0.5</td>
<td>2.0</td>
<td>16.3</td>
<td>25</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>0.1</td>
<td>1.0</td>
<td>13.5</td>
<td>0.6</td>
<td>3.2</td>
<td>22.2</td>
<td>23</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>19.7</td>
<td>0.6</td>
<td>7.5</td>
<td>0.4</td>
<td>1.0</td>
<td>10.3</td>
<td>8</td>
</tr>
<tr>
<td>24</td>
<td>24</td>
<td>56.0</td>
<td>1.5</td>
<td>20.0</td>
<td>0.9</td>
<td>3.2</td>
<td>30.3</td>
<td>7</td>
</tr>
<tr>
<td>92</td>
<td>92</td>
<td>281.3</td>
<td>15.8</td>
<td>210.7</td>
<td>8.7</td>
<td>1.5</td>
<td>44.6</td>
<td>41</td>
</tr>
<tr>
<td>26</td>
<td>26</td>
<td>45.9</td>
<td>1.3</td>
<td>16.9</td>
<td>0.8</td>
<td>2.7</td>
<td>24.2</td>
<td>17</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>8.4</td>
<td>0.3</td>
<td>4.0</td>
<td>0.2</td>
<td>0.5</td>
<td>5.4</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>21.2</td>
<td>0.5</td>
<td>6.5</td>
<td>0.3</td>
<td>1.0</td>
<td>12.9</td>
<td>10</td>
</tr>
<tr>
<td>52</td>
<td>52</td>
<td>57.9</td>
<td>1.3</td>
<td>16.8</td>
<td>0.8</td>
<td>4.0</td>
<td>35.6</td>
<td>22</td>
</tr>
<tr>
<td>54</td>
<td>54</td>
<td>29.9</td>
<td>0.9</td>
<td>12.1</td>
<td>0.6</td>
<td>1.7</td>
<td>14.5</td>
<td>21</td>
</tr>
<tr>
<td>17</td>
<td>17</td>
<td>27.7</td>
<td>0.9</td>
<td>11.5</td>
<td>0.5</td>
<td>1.5</td>
<td>13.3</td>
<td>8</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>49.0</td>
<td>1.2</td>
<td>15.7</td>
<td>0.7</td>
<td>3.7</td>
<td>27.6</td>
<td>3</td>
</tr>
<tr>
<td>56</td>
<td>56</td>
<td>38.0</td>
<td>1.0</td>
<td>12.8</td>
<td>0.6</td>
<td>2.7</td>
<td>20.8</td>
<td>24</td>
</tr>
<tr>
<td>27</td>
<td>27</td>
<td>32.1</td>
<td>0.8</td>
<td>11.0</td>
<td>0.5</td>
<td>2.5</td>
<td>17.3</td>
<td>2</td>
</tr>
<tr>
<td>19</td>
<td>19</td>
<td>27.4</td>
<td>0.7</td>
<td>9.8</td>
<td>0.5</td>
<td>1.5</td>
<td>14.9</td>
<td>1</td>
</tr>
<tr>
<td>41</td>
<td>41</td>
<td>233.8</td>
<td>12.7</td>
<td>168.8</td>
<td>5.5</td>
<td>1.7</td>
<td>45.1</td>
<td>14</td>
</tr>
<tr>
<td>33</td>
<td>33</td>
<td>29.6</td>
<td>0.7</td>
<td>9.2</td>
<td>0.4</td>
<td>1.0</td>
<td>18.3</td>
<td>10</td>
</tr>
<tr>
<td>57</td>
<td>57</td>
<td>32.4</td>
<td>0.7</td>
<td>10.0</td>
<td>0.5</td>
<td>1.2</td>
<td>20.0</td>
<td>7</td>
</tr>
<tr>
<td>39</td>
<td>39</td>
<td>35.5</td>
<td>0.6</td>
<td>10.9</td>
<td>0.5</td>
<td>1.5</td>
<td>21.8</td>
<td>28</td>
</tr>
<tr>
<td>61</td>
<td>61</td>
<td>42.3</td>
<td>1.1</td>
<td>14.9</td>
<td>0.7</td>
<td>1.7</td>
<td>23.8</td>
<td>29</td>
</tr>
<tr>
<td>63</td>
<td>63</td>
<td>50.8</td>
<td>1.4</td>
<td>18.9</td>
<td>0.9</td>
<td>2.0</td>
<td>27.5</td>
<td>30</td>
</tr>
<tr>
<td>65</td>
<td>65</td>
<td>57.6</td>
<td>1.7</td>
<td>22.9</td>
<td>1.1</td>
<td>2.2</td>
<td>29.6</td>
<td>31</td>
</tr>
<tr>
<td>79</td>
<td>79</td>
<td>76.4</td>
<td>1.5</td>
<td>54.5</td>
<td>2.3</td>
<td>1.2</td>
<td>16.9</td>
<td>9</td>
</tr>
<tr>
<td>81</td>
<td>81</td>
<td>96.1</td>
<td>2.0</td>
<td>61.8</td>
<td>2.7</td>
<td>2.7</td>
<td>26.8</td>
<td>12</td>
</tr>
<tr>
<td>83</td>
<td>83</td>
<td>272.1</td>
<td>13.2</td>
<td>210.7</td>
<td>7.2</td>
<td>1.5</td>
<td>39.6</td>
<td>14</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
<td>281.3</td>
<td>15.8</td>
<td>210.7</td>
<td>8.7</td>
<td>1.5</td>
<td>44.6</td>
<td>44</td>
</tr>
<tr>
<td>102</td>
<td>102</td>
<td>281.3</td>
<td>15.8</td>
<td>210.7</td>
<td>8.7</td>
<td>1.5</td>
<td>44.6</td>
<td>43</td>
</tr>
<tr>
<td>104</td>
<td>104</td>
<td>281.3</td>
<td>15.8</td>
<td>210.7</td>
<td>8.7</td>
<td>1.5</td>
<td>44.6</td>
<td>43</td>
</tr>
<tr>
<td>85</td>
<td>85</td>
<td>62.3</td>
<td>3.6</td>
<td>48.0</td>
<td>1.0</td>
<td>0.2</td>
<td>9.5</td>
<td>1</td>
</tr>
<tr>
<td>73</td>
<td>73</td>
<td>142.8</td>
<td>8.3</td>
<td>111.0</td>
<td>4.0</td>
<td>0.5</td>
<td>19.0</td>
<td>39</td>
</tr>
<tr>
<td>75</td>
<td>75</td>
<td>177.6</td>
<td>10.5</td>
<td>139.7</td>
<td>5.4</td>
<td>0.7</td>
<td>21.3</td>
<td>40</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>261.3</td>
<td>15.2</td>
<td>202.7</td>
<td>8.3</td>
<td>1.0</td>
<td>34.1</td>
<td>41</td>
</tr>
<tr>
<td>105</td>
<td>105</td>
<td>267.1</td>
<td>15.5</td>
<td>206.7</td>
<td>8.5</td>
<td>1.2</td>
<td>35.1</td>
<td>42</td>
</tr>
<tr>
<td>107</td>
<td>107</td>
<td>267.1</td>
<td>15.5</td>
<td>206.7</td>
<td>8.5</td>
<td>1.2</td>
<td>35.1</td>
<td>42</td>
</tr>
</tbody>
</table>

Table 8 - Minimum R-paths by Mode
<table>
<thead>
<tr>
<th>NODE</th>
<th>LINK</th>
<th>RSUM</th>
<th>WTSUM</th>
<th>TTSUM</th>
<th>VTSUM</th>
<th>PLSSUM</th>
<th>CSUM</th>
<th>INODES</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>27.5</td>
<td>0.6</td>
<td>8.2</td>
<td>0.4</td>
<td>2.0</td>
<td>16.3</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td>10.5</td>
<td>0.2</td>
<td>3.0</td>
<td>0.0</td>
<td>1.2</td>
<td>5.9</td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>10.4</td>
<td>0.2</td>
<td>3.0</td>
<td>0.1</td>
<td>1.0</td>
<td>6.0</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>24</td>
<td>46.6</td>
<td>1.2</td>
<td>15.5</td>
<td>0.7</td>
<td>3.2</td>
<td>25.9</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>92</td>
<td>81.9</td>
<td>2.1</td>
<td>27.7</td>
<td>1.3</td>
<td>4.0</td>
<td>46.8</td>
<td>43</td>
</tr>
<tr>
<td>7</td>
<td>26</td>
<td>36.5</td>
<td>0.9</td>
<td>12.6</td>
<td>0.6</td>
<td>2.7</td>
<td>19.8</td>
<td>17</td>
</tr>
<tr>
<td>8</td>
<td>15</td>
<td>21.6</td>
<td>1.5</td>
<td>6.5</td>
<td>0.3</td>
<td>1.5</td>
<td>12.9</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>32</td>
<td>48.7</td>
<td>1.1</td>
<td>14.7</td>
<td>0.7</td>
<td>3.0</td>
<td>29.2</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>42.9</td>
<td>1.0</td>
<td>13.0</td>
<td>0.6</td>
<td>2.5</td>
<td>25.8</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>29</td>
<td>28.4</td>
<td>0.6</td>
<td>8.5</td>
<td>0.4</td>
<td>2.0</td>
<td>16.9</td>
<td>8</td>
</tr>
<tr>
<td>12</td>
<td>5</td>
<td>10.7</td>
<td>0.2</td>
<td>3.3</td>
<td>0.2</td>
<td>0.5</td>
<td>6.5</td>
<td>2</td>
</tr>
<tr>
<td>13</td>
<td>35</td>
<td>40.1</td>
<td>0.9</td>
<td>11.4</td>
<td>0.5</td>
<td>2.5</td>
<td>24.8</td>
<td>11</td>
</tr>
<tr>
<td>14</td>
<td>37</td>
<td>40.1</td>
<td>0.9</td>
<td>12.1</td>
<td>0.6</td>
<td>2.5</td>
<td>24.0</td>
<td>11</td>
</tr>
<tr>
<td>15</td>
<td>94</td>
<td>81.9</td>
<td>2.1</td>
<td>27.7</td>
<td>1.3</td>
<td>4.0</td>
<td>46.8</td>
<td>43</td>
</tr>
<tr>
<td>16</td>
<td>61.9</td>
<td>1.5</td>
<td>19.7</td>
<td>0.9</td>
<td>3.5</td>
<td>36.3</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>48</td>
<td>33.7</td>
<td>0.9</td>
<td>11.7</td>
<td>0.6</td>
<td>2.2</td>
<td>18.4</td>
<td>20</td>
</tr>
<tr>
<td>18</td>
<td>27</td>
<td>44.2</td>
<td>1.2</td>
<td>16.4</td>
<td>0.8</td>
<td>3.2</td>
<td>22.6</td>
<td>7</td>
</tr>
<tr>
<td>19</td>
<td>52</td>
<td>27.8</td>
<td>0.5</td>
<td>6.3</td>
<td>0.3</td>
<td>2.0</td>
<td>18.7</td>
<td>22</td>
</tr>
<tr>
<td>20</td>
<td>54</td>
<td>20.5</td>
<td>0.6</td>
<td>7.6</td>
<td>0.4</td>
<td>1.7</td>
<td>10.2</td>
<td>21</td>
</tr>
<tr>
<td>21</td>
<td>17</td>
<td>18.3</td>
<td>0.5</td>
<td>7.0</td>
<td>0.3</td>
<td>1.5</td>
<td>8.9</td>
<td>4</td>
</tr>
<tr>
<td>22</td>
<td>11</td>
<td>18.9</td>
<td>0.4</td>
<td>5.2</td>
<td>0.2</td>
<td>1.7</td>
<td>11.3</td>
<td>3</td>
</tr>
<tr>
<td>23</td>
<td>56</td>
<td>7.9</td>
<td>0.2</td>
<td>2.3</td>
<td>0.1</td>
<td>0.7</td>
<td>4.6</td>
<td>24</td>
</tr>
<tr>
<td>24</td>
<td>7</td>
<td>2.0</td>
<td>0.0</td>
<td>0.5</td>
<td>0.0</td>
<td>0.5</td>
<td>1.4</td>
<td>2</td>
</tr>
<tr>
<td>25</td>
<td>9</td>
<td>2.7</td>
<td>0.1</td>
<td>0.7</td>
<td>0.0</td>
<td>0.5</td>
<td>1.4</td>
<td>2</td>
</tr>
<tr>
<td>26</td>
<td>41</td>
<td>57.0</td>
<td>1.4</td>
<td>18.3</td>
<td>0.9</td>
<td>3.0</td>
<td>33.5</td>
<td>14</td>
</tr>
<tr>
<td>27</td>
<td>33</td>
<td>57.1</td>
<td>1.3</td>
<td>17.4</td>
<td>0.8</td>
<td>3.0</td>
<td>34.6</td>
<td>10</td>
</tr>
<tr>
<td>28</td>
<td>57</td>
<td>59.9</td>
<td>1.4</td>
<td>18.2</td>
<td>0.9</td>
<td>3.2</td>
<td>36.2</td>
<td>27</td>
</tr>
<tr>
<td>29</td>
<td>59</td>
<td>63.0</td>
<td>1.4</td>
<td>19.1</td>
<td>0.9</td>
<td>3.5</td>
<td>38.0</td>
<td>28</td>
</tr>
<tr>
<td>30</td>
<td>61</td>
<td>69.7</td>
<td>1.7</td>
<td>23.1</td>
<td>1.1</td>
<td>3.7</td>
<td>40.1</td>
<td>29</td>
</tr>
<tr>
<td>31</td>
<td>63</td>
<td>78.3</td>
<td>2.0</td>
<td>27.1</td>
<td>1.3</td>
<td>4.0</td>
<td>43.8</td>
<td>30</td>
</tr>
<tr>
<td>32</td>
<td>43</td>
<td>73.8</td>
<td>2.1</td>
<td>29.2</td>
<td>1.3</td>
<td>4.5</td>
<td>47.7</td>
<td>15</td>
</tr>
<tr>
<td>33</td>
<td>79</td>
<td>103.9</td>
<td>2.1</td>
<td>62.7</td>
<td>2.7</td>
<td>3.2</td>
<td>33.2</td>
<td>9</td>
</tr>
<tr>
<td>34</td>
<td>81</td>
<td>66.0</td>
<td>1.2</td>
<td>51.3</td>
<td>2.2</td>
<td>0.7</td>
<td>10.5</td>
<td>12</td>
</tr>
<tr>
<td>35</td>
<td>83</td>
<td>95.4</td>
<td>1.9</td>
<td>60.1</td>
<td>2.6</td>
<td>2.7</td>
<td>28.0</td>
<td>14</td>
</tr>
<tr>
<td>36</td>
<td>98</td>
<td>81.9</td>
<td>2.1</td>
<td>27.7</td>
<td>1.3</td>
<td>4.0</td>
<td>46.8</td>
<td>43</td>
</tr>
<tr>
<td>37</td>
<td>102</td>
<td>81.9</td>
<td>2.1</td>
<td>27.7</td>
<td>1.3</td>
<td>4.0</td>
<td>46.8</td>
<td>43</td>
</tr>
<tr>
<td>38</td>
<td>104</td>
<td>81.9</td>
<td>2.1</td>
<td>27.7</td>
<td>1.3</td>
<td>4.0</td>
<td>46.8</td>
<td>43</td>
</tr>
<tr>
<td>39</td>
<td>85</td>
<td>89.8</td>
<td>4.2</td>
<td>56.2</td>
<td>1.4</td>
<td>2.2</td>
<td>25.8</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>87</td>
<td>90.8</td>
<td>4.2</td>
<td>56.5</td>
<td>1.4</td>
<td>2.2</td>
<td>25.8</td>
<td>1</td>
</tr>
<tr>
<td>41</td>
<td>89</td>
<td>102.4</td>
<td>4.5</td>
<td>59.9</td>
<td>1.5</td>
<td>2.7</td>
<td>34.3</td>
<td>13</td>
</tr>
<tr>
<td>42</td>
<td>106</td>
<td>81.9</td>
<td>2.1</td>
<td>27.7</td>
<td>1.3</td>
<td>4.0</td>
<td>46.8</td>
<td>43</td>
</tr>
<tr>
<td>43</td>
<td>95</td>
<td>67.6</td>
<td>1.8</td>
<td>23.7</td>
<td>1.1</td>
<td>3.7</td>
<td>37.3</td>
<td>16</td>
</tr>
<tr>
<td>44</td>
<td>99</td>
<td>87.6</td>
<td>2.4</td>
<td>31.7</td>
<td>1.5</td>
<td>4.2</td>
<td>47.8</td>
<td>36</td>
</tr>
</tbody>
</table>

Table 9 - Minimum R-paths by Mode
similar for all level of service variables. The tables 10 and 11 are examples of matrices of selected level of service variables for each mode. For instance, table 10 gives the waiting time (WT) costs per ton for the origin nodes 1,2,3,4 and the destination nodes 16,37,38,43. This table of WT costs is for mode 50. Table 10 gives, for the same mode, the travel time costs and the variance of travel time costs for the same set of origin and destinations. Table 11 is for modes 40 and 30 and contains the matrix of WT costs and the matrix of costs (TRN CHARGE). These matrices of Level Of Service variables are used subsequently by the different distribution models to determine the mode split and the resulting flows over each mode. This is the object of the following section.

6.2.3 "Matrix of Flows for each mode"

Having calculated the matrix of level of service variables for each mode and for each component of the Level of Service vector, this data is used for each subcommodity to calculate the mode split and the flows from the supply nodes to the demand nodes. The basic distribution models used are discussed in chapter three and the details of the computer implementation will follow in the next chapter.

In table 12 below the flows over each one of the modes 30,40,50 from the supply node 1,2,3,4 to the demand nodes 43,37,16,38 are given. This table is concerned with the flows of the subcommodity NIEB. It must be noted that the direct demand model to which NIEB was assigned is the McLynn Model. As expected, the flows are non-zero for almost all entries in the table. This contrasts with the linear programming algorithm applied to the same subcommodity. In fact table 13 gives the flows over all the modes from the supply to the demand nodes when the distribution is obtained from the linear programming
| ORIGINS |          |          |          |          |          |          |          |          |
|---------|----------|----------|----------|----------|----------|----------|----------|
|         | CIAM     | ZIND     | SUMS     | BIERN    | 300      | 300      | THIS LAST COST TABLE IS FOR MODE 50 |
|         | 1        | 2        | 3        | 4        | SUMS     | SUMS     | TT COSTS PER TON FOR NIEB             |
|---------|----------|----------|----------|----------|----------|----------|
| NIAM    | ZIND     | SUMS     | BIERN    | 300      |          |          |                                      |
| 43      | 39 MER   | 1.59     | 1.78     | 2.17     | 1.55     | 7.09     |                                      |
| 37      | 37 LAGO  | 1.89     | 2.08     | 2.47     | 1.85     | 8.29     |                                      |
| 16      | 16 COTO  | 1.29     | 1.48     | 1.87     | 1.25     | 5.89     |                                      |
| 38 ABID | 38 ABID  | 1.89     | 2.08     | 2.47     | 1.85     | 8.29     |                                      |
| 300 SUMS | 6.66     | 7.41     | 8.97     | 6.51     | 29.55    |          |                                      |
|         |          |          |          |          |          |          |                                      |
| THIS LAST COST TABLE IS FOR MODE 50 |

VAR OF TT COSTS PER TON FOR NIEB

Table 10 - Matrix of Level of Service Variables
TRN CHARGE PER TON FOR NIEB

<table>
<thead>
<tr>
<th>NIAM</th>
<th>ZIND</th>
<th>SUMS</th>
<th>ORIGINS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MARA</td>
<td>3</td>
<td>BIRN 300</td>
</tr>
</tbody>
</table>

|      |      |      |         |
| 43   | MER  | 35.08| 51.37   | 57.25      | 45.39189  |
| 37   | LAGO | 44.58| 60.87   | 66.75      | 54.89227  |

|      |      |      |         |
| D    | 16   | COTO | 44.58   | 60.87     | 66.75      | 54.89227  |
| E    | 38   | ABID | 44.58   | 60.87     | 66.75      | 54.89227  |
| 300  | SUMS | 83233| 99257   | 49210.0687| 0.37       |

THIS LAST COST TABLE IS FOR MODE 40

DECISION COSTS PER TON FOR NIEB

<table>
<thead>
<tr>
<th>NIAM</th>
<th>ZIND</th>
<th>SUMS</th>
<th>ORIGINS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MARA</td>
<td>3</td>
<td>BIRN 300</td>
</tr>
</tbody>
</table>

|      |      |      |         |
| 43   | MER  | 129.54 | 19120 | 21129.5498.48 |
| 37   | LAGO | 143.781 | 134511 | 47123.80495.50 |
| D    | 16   | COTO | 143.781 | 1345134 | 45143.78555.44 |
| E    | 38   | ABID | 123.801 | 1345134 | 45143.78535.46 |
| 300  | SUMS | 894999 | 50503 | 59540.901111 |

THIS LAST COST TABLE IS FOR MODE 30

WT COSTS PER TON FOR NIEB

<table>
<thead>
<tr>
<th>NIAM</th>
<th>ZIND</th>
<th>SUMS</th>
<th>ORIGINS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MARA</td>
<td>3</td>
<td>BIRN 300</td>
</tr>
</tbody>
</table>

|      |      |      |         |
| 43   | MER  | 3.93 | 3.62 | 3.63 | 3.84 | 15.02 |
| 37   | LAGO | 4.23 | 3.32 | 3.33 | 3.54 | 14.42 |
| D    | 16   | COTO | 4.23 | 3.92 | 3.93 | 4.14 | 16.22 |
| E    | 38   | ABID | 3.63 | 3.92 | 3.93 | 4.14 | 15.62 |
| 300  | SUMS | 16.03 | 14.77 | 14.83 | 15.67 | 61.30 |
## Flows in Tons per Day for NIEB

### Table 12 - Matrices of Flows (McLynn Model)

<table>
<thead>
<tr>
<th></th>
<th>NIAM 1</th>
<th>ZIND 3</th>
<th>SUMS 4</th>
<th>BIRN 5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NIA</strong>M</td>
<td>43</td>
<td>37</td>
<td>16</td>
<td>38</td>
</tr>
<tr>
<td><strong>ZIN</strong>D</td>
<td>MER 43</td>
<td>LAGO 37</td>
<td>COTO 16</td>
<td>ABID 38</td>
</tr>
<tr>
<td><strong>SUMS</strong></td>
<td>1.77</td>
<td>0.58</td>
<td>0.12</td>
<td>0.30</td>
</tr>
<tr>
<td><strong>BIRN</strong></td>
<td>MARA 2</td>
<td>ZIND 4</td>
<td>SUMS 300</td>
<td>300</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>NIAM 1</th>
<th>ZIND 3</th>
<th>SUMS 4</th>
<th>BIRN 5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NIA</strong>M</td>
<td>43</td>
<td>37</td>
<td>16</td>
<td>38</td>
</tr>
<tr>
<td><strong>ZIN</strong>D</td>
<td>MER 43</td>
<td>LAGO 37</td>
<td>COTO 16</td>
<td>ABID 38</td>
</tr>
<tr>
<td><strong>SUMS</strong></td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td><strong>BIRN</strong></td>
<td>MARA 2</td>
<td>ZIND 4</td>
<td>SUMS 300</td>
<td>300</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>NIAM 1</th>
<th>ZIND 3</th>
<th>SUMS 4</th>
<th>BIRN 5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NIA</strong>M</td>
<td>43</td>
<td>37</td>
<td>16</td>
<td>38</td>
</tr>
<tr>
<td><strong>ZIN</strong>D</td>
<td>MER 43</td>
<td>LAGO 37</td>
<td>COTO 16</td>
<td>ABID 38</td>
</tr>
<tr>
<td><strong>SUMS</strong></td>
<td>0.03</td>
<td>0.01</td>
<td>0.00</td>
<td>0.02</td>
</tr>
<tr>
<td><strong>BIRN</strong></td>
<td>MARA 2</td>
<td>ZIND 4</td>
<td>SUMS 300</td>
<td>300</td>
</tr>
</tbody>
</table>

---

**FLOWS IN TONS PER DAY FOR NIEB**

---

**ORIGINS**

---

These flows in tons per day are for Mode 50.

---

**ORIGINS**

---

These flows in tons per day are for Mode 40.

---

**ORIGINS**

---

These flows in tons per day are for Mode 30.

---

**********TRACER TDIS 5**********
Table 13 - Matrix of Flows (Linear Programming Model)

<table>
<thead>
<tr>
<th>ORIGINS</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>SUMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MER</td>
<td>1.89</td>
<td>0.61</td>
<td>1.51</td>
<td>0.24</td>
<td>4.25</td>
</tr>
<tr>
<td>LAGO</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.42</td>
<td>1.42</td>
</tr>
<tr>
<td>COLO</td>
<td>0.24</td>
<td>0.0</td>
<td>0.0</td>
<td>0.24</td>
<td></td>
</tr>
<tr>
<td>ABID</td>
<td>0.71</td>
<td>0.0</td>
<td>0.0</td>
<td>0.71</td>
<td></td>
</tr>
<tr>
<td>SUMS</td>
<td>2.83</td>
<td>0.61</td>
<td>1.51</td>
<td>1.65</td>
<td>6.60</td>
</tr>
</tbody>
</table>
### FLOWS IN TONS PER DAY FOR NIEB

<table>
<thead>
<tr>
<th>ORIGINS</th>
<th>NIAM 1</th>
<th>ZIND 2</th>
<th>SUMS 4</th>
<th>BIRN 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIAM</td>
<td>ZIND</td>
<td>SUMS</td>
<td>BIRN</td>
<td>(SPM1 Model)</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>43</td>
<td>MER</td>
<td>3.25</td>
<td>0.08</td>
<td>0.02</td>
</tr>
<tr>
<td>37</td>
<td>LAGO</td>
<td>0.35</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>16</td>
<td>COTO</td>
<td>0.52</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>38</td>
<td>ABID</td>
<td>0.19</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>300</td>
<td>SUMS</td>
<td>4.31</td>
<td>0.10</td>
<td>0.02</td>
</tr>
</tbody>
</table>

**These flows in tons per day are for MODE 50**

### FLOWS IN TONS PER DAY FOR NIEB

<table>
<thead>
<tr>
<th>ORIGINS</th>
<th>NIAM 1</th>
<th>ZIND 2</th>
<th>SUMS 4</th>
<th>BIRN 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIAM</td>
<td>ZIND</td>
<td>SUMS</td>
<td>BIRN</td>
<td>(SPM1 Model)</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>43</td>
<td>MER</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>37</td>
<td>LAGO</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>16</td>
<td>COTO</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>38</td>
<td>ABID</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>300</td>
<td>SUMS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**These flows in tons per day are for MODE 40**

### FLOWS IN TONS PER DAY FOR NIEB

<table>
<thead>
<tr>
<th>ORIGINS</th>
<th>NIAM 1</th>
<th>ZIND 2</th>
<th>SUMS 4</th>
<th>BIRN 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIAM</td>
<td>ZIND</td>
<td>SUMS</td>
<td>BIRN</td>
<td>(SPM1 Model)</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>43</td>
<td>MER</td>
<td>0.05</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>37</td>
<td>LAGO</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>16</td>
<td>COTO</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>38</td>
<td>ABID</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>300</td>
<td>SUMS</td>
<td>0.07</td>
<td>0.01</td>
<td>0.01</td>
</tr>
</tbody>
</table>

**These flows in tons per day are for MODE 30**

Table 14 - Matrices of Flows (SPM1 Model)
Table 15 - Matrices of Flows (Sarc-Kraft Model)

<table>
<thead>
<tr>
<th>ORIGINS</th>
<th>NIAM 2</th>
<th>ZIND 300</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 MARA</td>
<td>3 SUMS</td>
</tr>
<tr>
<td>43 MER</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>300 SUMS</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NIAM</td>
<td>ZIND</td>
<td>43</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>43</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

These flows in tons per day are for mode 40.

<table>
<thead>
<tr>
<th>NIAM</th>
<th>ZIND</th>
<th>43</th>
<th>300</th>
<th>1</th>
<th>SUMS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>43</td>
<td>MER</td>
<td>0.12</td>
<td>4.18</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SUMS</td>
<td>0.12</td>
<td>4.18</td>
</tr>
</tbody>
</table>

These flows in tons per day are for mode 30.

Table 16 - Matrices of Flows (SPMI - Model)
model, while the sum of the flows matrix for mode 30,40,50 in table 12 generates a distribution of flows of a quite different nature. This is anticipated since with the linear programming algorithm, there is a maximum of \( m + n-1 \) non-zero flows where \( m \) is the number of rows and \( n \) the number of columns in the flow matrix.

In table 14, the resulting flows for the same subcommodity NIEB with the same supply and demand data is given. The distribution model used was the Special Product Model (SPM-1) with the \( R_{lm} \) calculated using formula (12) from chapter three. If we compare the mode split of this model with the one obtained from the McLYnn model in table 12, it is clear that the total flow on mode 50 is more or less the same. For the other two modes, the flows are small and they are zero on mode 40 in both cases. The flows on mode 50 from each origin to each destination, however, are quite different. This is certainly due to the parameters used to arrive at these flows in each model.

Table 15 and 16 show the flows on each mode for another subcommodity ARAC which was supplied at 3 nodes (1,2,3) and demanded at one node (43). In table 15, the results are obtained using the Sarc-Kraft model while the flows in table 16 are the results of the Special Product Model 1 (SPM-1) with the \( R_{lm} \) computed from formula (14) in chapter three. It is seen that all the flows are on mode 50 in both cases with the flows from each origin to each destination in the same order of magnitude as predicted from each model.

6.2.4 "Test runs on Incremental Assignment"

In order to analyse the results of the assignment of flows with the incremental method, the following evaluation was made. One subcommodity was supplied at 4 nodes (1,2,3,4) and demanded at 4 nodes (43, 37, 16, 38). A network link file was used in which the flows were zero on all
links prior to the first incremental assignment. The link performance vectors of each link had initial values that were good measures of the relative performance of each link with respect to one another.

A first assignment was made using this link file and it was carried out in one increment for the subcommodity mentioned above. This amounts to making the assignment the same as in the original system in which no updating of the link performance vectors was carried out before all the flows were assigned. The results of this test have given the following flows in tons per day as shown in table 17. The same flows, as they were assigned to the network, are shown on map 3 which displays the flows from the supply nodes to the demand nodes. The total flow going into the demand nodes is 902 tons per day.

The same supply and demand data was assigned to the network in another run of the system. This time, however, the assignment was carried out in two increments with the link performance vectors and the flows being the same as the previous case prior to the first incremental assignment. After one increment (or one half of the supply) was assigned, the link performance vectors were updated by calculating new values of the performance measures, taking into consideration the flows already assigned. The total link flows that resulted from these two equal increments are shown in table 18. The map 4 shows the network with the resulting flows of the two successive incremental assignments. The link performances were updated between the two assignments. In both cases the total flow into the demand nodes is the same, that is 902 tons per day. Also in both cases, the demand model used and the parameters were the same. The model was the SPM-1 model with formula (12) in chapter three for the $R_{1m}$ values.
<table>
<thead>
<tr>
<th>Link</th>
<th>INODE</th>
<th>JNODE</th>
<th>FLOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td></td>
<td>714</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>25</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>23</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td></td>
<td>174</td>
</tr>
<tr>
<td>4</td>
<td>25</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>33</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>12</td>
<td>34</td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Link</th>
<th>INODE</th>
<th>JNODE</th>
<th>FLOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>4</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>8</td>
<td>11</td>
<td></td>
<td>888</td>
</tr>
<tr>
<td>10</td>
<td>9</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>14</td>
<td></td>
<td>888</td>
</tr>
<tr>
<td>14</td>
<td>26</td>
<td></td>
<td>888</td>
</tr>
<tr>
<td>26</td>
<td>16</td>
<td></td>
<td>888</td>
</tr>
<tr>
<td>23</td>
<td>24</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>33</td>
<td>38</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>34</td>
<td>37</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>43</td>
<td></td>
<td>888</td>
</tr>
<tr>
<td>37</td>
<td>43</td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>
MAP 3 LINK FLOWS FOR ONE INCREMENT
The two maps 3 and 4 show very interesting results concerning the effect of incremental assignment. First table 3 shows that with one increment most of the flow (888 tons per day) follows a route going through the nodes (1, 8, 11, 14, 26, 16, see map 3). This is essentially a corridor going through Dahomey reach the ocean. There are two other small corridors, one going through the nodes 3, 23, 24, 2, 12, 34, 37 to Lagos. The flow is only 4 tons per day. The other corridor is going through nodes 1, 10, 9, 38 to Abidjan. But again the flow is only 10 tons per day.

The map 4, on the other hand, has the same two small corridors as above with flows of 2 tons per day and 7 tons per day respectively. But the main flow is split between two main routes. One of them is the same as in the previous case but the flow is now 474 tons per day, or about one half of what it was previously. The other corridor goes through the nodes 10, 27, 28, 29, 30, 31, 32, 15, 43 and has a flow of 419 tons per day (see map 4). What has happened is that after the assignment of the first increment, the flows were mostly in the first corridor ending at node 16. But after the link performance vectors were updated, the cheapest route was no longer the one defined by this corridor but rather was the one ending at node (15 or 43) and going through Togo. This shows that incremental assignment should have the effect of loading the network in a more uniform way than the one increment assignment method previously used.
<table>
<thead>
<tr>
<th>Link</th>
<th>INODE</th>
<th>JNODE</th>
<th>FLOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td></td>
<td>373</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td></td>
<td>426</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>25</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>23</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td></td>
<td>101</td>
</tr>
<tr>
<td>4</td>
<td>25</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>25</td>
<td>4</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>33</td>
<td>38</td>
<td></td>
<td>7</td>
</tr>
<tr>
<td>34</td>
<td>37</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>33</td>
<td></td>
<td>7</td>
</tr>
<tr>
<td>12</td>
<td>34</td>
<td></td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Link</th>
<th>INODE</th>
<th>JNODE</th>
<th>FLOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>11</td>
<td></td>
<td>474</td>
</tr>
<tr>
<td>10</td>
<td>9</td>
<td></td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>27</td>
<td></td>
<td>419</td>
</tr>
<tr>
<td>11</td>
<td>14</td>
<td></td>
<td>474</td>
</tr>
<tr>
<td>14</td>
<td>26</td>
<td></td>
<td>474</td>
</tr>
<tr>
<td>32</td>
<td>15</td>
<td></td>
<td>419</td>
</tr>
<tr>
<td>26</td>
<td>16</td>
<td></td>
<td>474</td>
</tr>
<tr>
<td>23</td>
<td>24</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>27</td>
<td>28</td>
<td></td>
<td>419</td>
</tr>
<tr>
<td>28</td>
<td>29</td>
<td></td>
<td>419</td>
</tr>
<tr>
<td>29</td>
<td>30</td>
<td></td>
<td>419</td>
</tr>
<tr>
<td>30</td>
<td>31</td>
<td></td>
<td>419</td>
</tr>
<tr>
<td>31</td>
<td>32</td>
<td></td>
<td>419</td>
</tr>
<tr>
<td>15</td>
<td>43</td>
<td></td>
<td>419</td>
</tr>
</tbody>
</table>
MAP 4 LINK FLOWS FOR TWO INCREMENTS
Chapter VII  "Description of the Subroutines and their Operations."

7.1 "Alphabetical listing of Fortan names and variables and their definitions."

ADDON  name of a subroutine in the transport sector
AGGRE  name of a subroutine in the transport sector
ALPHA  Array for the parameters of the McLynn model.
ASSIGN name of a subroutine in the transport sector.
BETA   Array for the parameters of the Sarc-Kraft model.
BSHEET name of a subroutine in the transport sector.
CFAC   Array for the five components of the R-factor.
CPV    Array for the commodity preference vector.
CSUM   Array for the cumulative cost sums.
CUMC   Matrix of cumulative costs.
CUMPLS Matrix of probability of loss costs.
CUMR   Matrix of cumulative R-factors.
CUMTT  Matrix of cumulative travel time costs.
CUMVTT Matrix of cumulative variance of travel time costs.
CUMWT  Matrix of cumulative waiting time costs.
DAYS   Array for the number of days in each season.
DELTA  Array of parameters for the SPM-1 models.
DISAG  name of a subroutine in the transport sector.
DRCDMD name of a subroutine in the transport sector.
ECONOM name of a subroutine in the economic sector.
EDIT   name of a subroutine in the transport sector.
EDITOR name of a subroutine in the transport sector.
FLOW   Matrix of flows for a given mode.
GAMMA  Array of parameters for the Baumol-Quandt model.
ICLASS  vehicle class to which a subcommodity is assigned.
IDISTB  distribution model assigned to a given subcommodity.
INODES  Array of the beginning node of a link.
JNODES  Array of the end node of a link.
KOM    index of subcommodities.
KOMNAM  Array of the names of the subcommodities.
LABOR  name of a subroutine in the transport sector.
LNEW  Fortran name of the updated network link file.
LNKOST  name of a subroutine in the transport sector.
LNSTOR  Fortran name of a temporary network link file.
LNTAPE  Fortran name of the network link file.
LOSTAP  Fortran name of the file of level of service variables.
LSDTAP  Fortran name of the file of supply and demand data.
MODE  Array of the mode of each link.
NADDON  name of a subroutine in the transport sector.
NCLASS  number of actual vehicle class in the model.
NETFLO  name of a subroutine in the transport sector.
NIN  Fortran number of the system input file.
NITEMS  number of components in the link performance vector.
NLINKS  number of links in the network.
NNODES  number of nodes in the network.
NOCOMS  number of subcommodities in the simulation.
NODNAM  Array of names for the nodes in the network.
NOPARM  number of parameters in the link characteristic vectors.
NOUT  Fortran number for the system output file.
NOVOLS  number of components in the link utilization vector.
NSEZNS  number of season in the simulation.
NWTREE  name of a subroutine in the transport sector.
PLSSUM  Cumulative sums for the probability of loss.
RCOMP   name of a subroutine in the transport sector.
RDVOLT  name of a subroutine in the transport sector.
RFAC    Array for the R-factors.
RFCTAP  Fortran file number for the R-factors file.
RSUM    Cumulative sums of the R-factors.
SUBDEM  Array of the demand data for a given subcommodity.
SUBSUP  Array of the supply data for a given subcommodity.
SUMFLO  name of a subroutine in the transport sector.
TFLOW   Array for the flows for all modes.
TREE    name of subroutine in the transport sector.
TRETAP  Fortran number for the file of trees.
TRNDIS  name of a subroutine in the transport sector.
TRNSPT  name of a subroutine in the transport sector.
TRNSUM  Fortran number for the transport summary file.
TTSUM   Cumulative sum of the travel times.
VALKOM  values of the subcommodities.
VTTSUM  Array of the cumulative sum of the variance of travel time.
WTSUM   Array of the cumulative sum of the waiting time.
YREND   name of a subroutine in the economic sector.
7.2 "Macro-flowchart of the Assignment Process"

The routing, mode choice, distribution and assignment of the original version of the macroeconomic and transport simulator has been described in chapter two. At this stage, the purpose is to show in flowchart form the modifications and the additions of the direct demand models to the transport sector. Looking at figure 1, 2, 3, 4, the overall operations performed in the transport sector under the control of the subroutine ASSIGN are summarized in such a way that their interdependence is shown.

First the subroutine ASSIGN calls in the subroutine NETURN which determines the possible turns that can be done in the network from any node. Then a loop over all the possible seasons is initiated and the subroutine RDWOLT reads in the tables of link performance vectors for this particular season. The file RFCTAP is created for all subcommodities in this season by RCOMP. Then a loop on each subcommodity is initiated and the RFCTAP and the LOSTAP files are read into memory; the subroutine BSHEET (IS) will produce a balance sheet for this subcommodity. At this stage in NETFLO, a decision is made as to whether the distribution of this subcommodity flow is going to be done with the linear programming and the gravity model or if the distribution is going to use a direct demand model. To achieve this, a test is made on IDISTB (KOM) which specifies the model used for this particular subcommodity. If the value is less than or equal to two then the logic followed will use the modeling capability offered by the L.P. and the gravity model. Otherwise a direct demand model is used among the McLynn, the Sarc-Kraft, the Baumol-Quandt and the SPM-1 models. Form this stage to the final assignment, the two
possible paths defined by IDISTB run in parallel and they perform similar
tasks. Let us therefore describe the case where we have a direct demand
model.

The subroutine DRCDMD is called by NETFLO and it will perform essentially
two tasks: the first one is to build trees for each mode and for each supply
node of the subcommodity being looked at and to store on the file TRETAP all
the results defining the trees as well as all the following sums: RSUM, WTSUM,
TTSUM, VTTSUM, PLSSUM, CSUM. The other task is to calculate the level of
service variables and to store the cumulative values of these variables CUMR,
CUMWT, CUMTT, CUMVT, CUMPLS, CUMC on a file LOSTAP. Then the subroutine TRNDIS
is called to determine the distribution of the flows using one of the direct
demand model specified. Then TRNDIS will store the summary statistics on the
file TRNSUM. Finally the subroutine NADODON will perform the flow assignment on
each link in the network. Control is then returned to NETFLO and that
completes the calculations for this subcommodity. The same tasks are
repeated for all subcommodities and then the subroutine LNUPDT is called in
order to update the link file LNTAPE for this season. The process is repeated
for all seasons in order to complete the assignment process. If the model
used is not a direct demand model, a similar sequence of operation is done.

7.3 "Description of the Operations of each subroutines"

In this section, those subroutines which are additions to the transport
sector or is a modification of an existing subroutine, will be described in
detail with regard to operations and functions. Also, a macro-flowchart
of the main operations in each subroutine will be given. The new routines are
the following: DRCDMD, TRNDIS, DISTRC, DISTRD, DISTRE, DISTRF, BLOCKDATA, MAIN.
Those which were modified in order to perform new tasks and incorporate direct demand modeling were: RCOMP, NWTREE, NADDON, LNUPDT.

7.3.1 "Subroutine DRCDMD".

This program is called by NETFLO at the beginning of the assignment process. It will calculate the trees for each mode and also determine the level of service variables matrix CUMR, CUMWT, CUMTT, CUMVTT, CUMPLS, CUMC. While the trees are determined the subroutine calculates the cumulative sums RSUM, WTSUM, TTSUM, VTTSUM, PLSSUM, CSUM. The subroutine has an argument the array NN used to store the trees information. In carrying out the calculations, DRCDMD uses two sequential files TRETAP and LOSTAP.
FIGURE 1 MACRO-FLOWCHART OF ASSIGNMENT PROCESS, PART 1

Set up network turns
Call return in assign

Do for each season
I=1 NSEZNS
Read in LPV for that season
Call RDVOLT

Compute R-factors for all commodities for this season

Do for each subcommodity
KOM=1,NOCOMS
Read R-factors, read supply and demand for subcommodity KOM
Write out a balance sheet for commodity KOM, call BSHEET (IS)

Do for each supply node
I=1,II
Build a minimum R-tree, determine resulting costs.
Dump out the tree
Sum up the RsuM and CsuM one row at a time to form CUMR, CUMC tables
Save the CUMC, CUMR tables for the transport summary computations
FIGURE 2 MACRO_FLOWCHART OF
ASSIGNMENT PROCESS, PART 2

1. PERFORM SUBCOMMODITY DISTRIBUTION USING ONE OF SEVERAL DISTRIBUTION MODELS.
2. SAVE FLOWS FOR TRANSPORT SUMMARY—FLOW.
3. CALL, PACK, ADDON, REPACK.
4. DO FOR EACH SUPPLY NODE:
   - ADDON
   - RDTREE
   - REREAD MIN. PATH TREES.
5. DO FOR EACH DEMAND NODE:
   -_node
   - CONVERT FROM DOLLARS TO TONS AND ASSIGN FLOW TO NETWORK.
6. CALL UNPACK, LNUPDT, REPACK.
7. IN LNUPDT:
   - UPDATE LINK FILE WITH THE SEASONAL ASSIGNMENT AND PREPARE LNSTOR FOR R-COMPUTATION NEXT SEASON.
FIGURE 3 MACRO-FLOWCHART OF ASSIGNMENT PROCESS, PART 3

NRCOMP

NRCOMP COMPUTES THE R-FACTORS FOR ALL COMMODITIES FOR THIS SEASON AND STORE R-FACTORS + THEIR 5 COMPONENTS.

READ R-FACTORS + THE 5 COMPONENTS.
READ SUPPLY AND DEMAND FOR SUB-COMMODITY KOM.

PATH FOR ROUTING, MODE CHOICE, ASSIGNMENT, USING DIRECT DEMAND.

DO FOR EACH MODE CALL DRCMD

DO FOR EACH SUPPLY NODE

BUILD THE TREES FOR EACH MODE AND EACH SUPPLY NODE. STORE THE TREES ON TRETAP AS WELL AS THE 6 CUMULATIVE SUMS: RSUM, WTSUM, TTSUM, VTTSUM, PLSSUM, CSUM.

TREES SUMS
DO FOR EACH MODE
DO FOR EACH SUPPLY NODE
READ IN TREES + CUMULATIVE SUMS.
DO FOR EACH DEMAND NODE
CALCULATE THE SIX LEVEL OF SERVICE (L.O.S.) VARIABLES MATRIX: CUMR, CUMTT, CUMVTT, CUMPLS, CUMC.
STORE THE SIX L.O.S. VARIABLES MATRIX FOR THAT MODE ON LOSTAP
DO FOR EACH MODE
READ L.O.S. VARIABLES, FROM LOSTAP
DISTRC, DISTRD, DISTRE
CALL PROPER DIRECT DEMAND MODEL;
MCLYNN, SARC-KRAFT, BAUMOL-QUANDT,
AND CALCULATE FLOWS FOR EACH MODE,
AND STORE IN TFLOW (I,J, MODE)
STORE AVERAGE CUMR, CUMC, AND TOTAL Fl OWS OVER ALL MODES IN THE TRANSPORT SUMMARY FILE.
DO FOR EACH SUPPLY NODE
DO FOR EACH DEMAND NODE
REREAD TREES WITH CUMULATIVE SUMS
STORE AVERAGE CUMR, CUMC, AND TOTAL Fl OWS OVER ALL MODES IN THE TRANSPORT SUMMARY FILE.
DO FOR EACH DEMAND NODE
CONVERT FLOWS TO TONS AND ASSIGN FLOWS TO THE NETWORK ONE MODE AT A TIME.

Figure 4 Macro-Flowchart of Assignment Process, Part 4
The following subroutines are called by DRCDMD:

Subroutine NWTREE
Subroutine OUTPUT

The output generated by the program for each supply node is: tables of the minimum R-path trees for each mode with all the cumulative sums RSUM, WTSUM, TTSUM, VTTSUM, PLSSUM, CSUM. The following tables on decisions costs and level of service variables are also available as output: CUMR, CUMWT, CUMTT, CUMVTT, CUMPLS, CUMC. In figure 5, a macro-flowchart of the operations of DRCDMD is shown.

7.3.2 "Subroutine TRNDIS"

This program is called by NETFLO after the execution of DRCDMD in order to continue the tasks of the assignment process. Using the L.O.S. matrix for each mode, it will calculate the flows from the supply nodes to the demand nodes using one of several direct demand models. It will also store on a file some transport summary statistics like the CUMC, and the matrix of flows FLOW. Then the subroutine will transfer control to subroutine NADDON to complete the assignment work.

TRNDIS has as arguments IS, the season number that is being considered, and the array NN which is used to store the trees information.

The following subroutines are called by TRNDIS:

Subroutine DISTRC
Subroutine DISTRD
Subroutine DISTRE
Subroutine DISTRF
Subroutine UNPACK
Subroutine NADDON
Subroutine REPACK

Also in order to perform the tasks of distribution, the following files are used:
DO FOR EACH MODE
DO FOR EACH SUPPLY NODE

DETERMINE IF SUBCOMMODITY KOM IS GOING TO USE A SPECIFIC MODE, USE ALL POSSIBLE MODES, OR USE ONE TREE OVER ALL MODES.

DO FOR EACH SUPPLY NODE

DETERMINE TREES AND CUMULATIVE SUMS: RSUM, WTSM, TTSM, VTTSM, PLSSUM, CSUM.

DO FOR EACH NODE ON TREE

PRINT A TABLE CONTAINING THE TREES AND THE CUMULATIVE SUMS

STORE THE TREES AND CUMULATIVE SUMS ON TREATAP

DO FOR EACH MODE
DO FOR EACH SUPPLY NODE

READ TREES AND CUMULATIVE SUMS

DO FOR EACH DEMAND NODE

CALCULATE LEVEL OF SERVICE VARIABLES CUMR, CUMWT, CUMVT, CUMVTT, CUMPLS, CUMC.

STORE THE LEVEL OF SERVICE VARIABLES FOR EACH MODE ON LOSTAP

PRINT TABLES OF OUTPUT FOR THE LEVEL OF SERVICE VARIABLES, CUMR, CUMWT, CUMVT, CUMVTT, CUMPLS, CUMC.
the level of service file LOSTAP and the transport summary file TRNSUM. Output tables can also be produced by TRNDIS on the flows (FLOW) that have been calculated by the distribution models. A macro-flowchart of the operations in TRNDIS is available and shown in figure 6.

7.3.3 "Subroutine DISTRC"

This subroutine is called by TRNDIS to perform the distribution of the flows. The model used in DISTRC is the McLynn model. It uses the level of service variables CUMR, CUMWT, CUMTT, CUMVT, CUMPLS, CUMC or any selected combinations of them to determine the flows over each mode. In doing this, it uses the supply and demand data. The formula coded in the subroutine can be written as:

$$
T_{FLOW}(I,J,m) = \frac{\sum_{m=1}^{K} b_{om} A_{m} B_{m} C_{m} D_{m} E_{m} F_{m}}{a_{om} SUBSUP(I)^{a_{1m}} \times SUBDEM(J)^{a_{2m}}}
$$

where

$$
A_{m} = CUMR(I,J)_{m}^{b_{1m}}
$$

$$
B_{m} = CUMWT(I,J)_{m}^{b_{2m}}
$$

$$
C_{m} = CUMTT(I,J)_{m}^{b_{3m}}
$$

$$
D_{m} = CUMVT(I,J)_{m}^{b_{4m}}
$$

$$
E_{m} = CUMPLS(I,J)_{m}^{b_{5m}}
$$

$$
F_{m} = CUMC(I,J)_{m}^{b_{6m}}
$$
As mentioned above any selection of the L.O.S. can appear, DISTRC has as argument the variable KMOD which is the mode for the distribution that is being done at a given moment. A macro-flowchart of the calculations carried out in DISTRC is shown in figure 7.
DO FOR EACH MODE

READ L.O.S. MATRIX FROM LOSTAP:
CUMR, CUMWT, CUMTT, CUMVTT, CUMPLS,
CUMC

CALL THE PROPER DISTRIBUTION MODEL
TO CALCULATE FLOWS

STORE THE CUMULATIVE COSTS CUMC ON
THE TRANSPORT SUMMARY FILE

DETERMINE THE TOTAL FLOWS AND THE
TOTAL SUPPLY AND NORMALISE FLOWS.

PRINT TABLE OF FLOWS

STORE THE FLOWS ON THE TRANSPORT
SUMMARY FILE.

STORE THE FLOWS TEMPORARILY ON
LOSTAP IN ORDER TO USE THEM IN
NADDON.

DO THE LINK FLOWS ASSIGNMENT USING
UNPACK, NADDON, AND REPACK.

FIGURE 6 MACRO-FLOWCHART FOR THE SUBROUTINE TRNDIS
DO FOR EACH SUPPLY NODE

DO FOR EACH DEMAND NODE

SELECT THE CLASS OF PARAMETERS TO WHICH THIS SUBCOMMODITY IS ASSIGNED

CALCULATE SUPPLY AND DEMAND EFFECTS ON THE FLOWS.

CALCULATE THE LEVEL OF SERVICE VARIABLES EFFECTS ON THE FLOWS USING A SELECTION OF VARIABLES.

RETURN
7.3.4 "Subroutine DISTRD"

This subroutine is called by TRNDIS to determine the flows resulting from the use of the Sarc-Kraft distribution model. Any selection of the level of service variables CUMR, CUMWT, CUMTT, CUMVTT, CUMPLS, CUMC can be used and also the effect of the supply and demand data on the flows is evaluated. The flows are calculated as follows:

\[
T_{FLOW}(I,J,m) = SE_{I,J,m} \cdot P_{I,J,m}
\]

where

\[
SE_{I,J,m} = a_{om} \cdot SUBSUP(I)^{a_{1m}} \cdot SUBDEM(J)^{a_{2m}}
\]  \hspace{1cm} (22)

and where

\[
P_{I,J,m} = \sum_{k=1}^{K} A_k B_k C_k D_k E_k F_k
\]

where

\[
A_k = CUMR(I,J)_k^{b_{ikm}}
\]

\[
B_k = CUMWT(I,J)_k^{b_{2km}}
\]

\[
C_k = CUMTT(I,J)_k^{b_{3km}}
\]

\[
D_k = CUMVTT(I,J)_k^{b_{4km}}
\]

\[
E_k = CUMPLS(I,J)_k^{b_{5km}}
\]

\[
F_k = CUMC(I,J)_k^{b_{6km}}
\]
The subroutine has an argument KMOD which is the mode used in the calculations of the distribution at this moment. In order to find out the flows, DISTRD uses the level of service file LOSTAP. In figure 8, a macro-flowchart of DISTRD is displayed.

7.3.5 "Subroutine DISTRE"

This subroutine is called by TRNDIS when the value of IDISTB(LOM)=5. It is used to determine the flows between the supply and the demand nodes for a Baumol-Quandt model. This is an abstract mode model and it can use any selection of the six components of the L.O.S. vectors to determine the flows. In doing so, the effect of the supply and demand data is also accounted for. If one defines:

\[ A_{1m} = (CUMR(I,J)_b)^{b_1} \frac{(CUMR(I,J)_m)^{b_2}}{(CUMR(I,J)_b)} \]

\[ A_{2m} = (CUMWT(I,J)_b)^{b_3} \frac{(CUMWT(I,J)_m)^{b_4}}{(CUMWT(I,J)_b)} \]

\[ A_{3m} = (CUMTT(I,J)_b)^{b_5} \frac{(CUMTT(I,J)_m)^{b_6}}{(CUMTT(I,J)_b)} \]

\[ A_{4m} = (CUMVTT(I,J)_b)^{b_7} \frac{(CUMVTT(I,J)_m)^{b_8}}{(CUMVTT(I,J)_b)} \]

\[ A_{5m} = (CUMPLS(I,J)_b)^{b_9} \frac{(CUMPLS(I,J)_m)^{b_{10}}}{(CUMPLS(I,J)_b)} \]

\[ A_{6m} = (CUMC(I,J)_b)^{b_{11}} \frac{(CUMC(I,J)_m)^{b_{12}}}{(CUMC(I,J)_b)} \]
SELECT THE CLASS OF PARAMETERS TO WHICH THIS SUBCOMMODITY IS ASSIGNED

FOR EACH MODE USED
DO FOR EACH SUPPLY NODE
DO FOR EACH DEMAND NODE

CALCULATE THE SUPPLY AND DEMAND EFFECT ON THE FLOWS

USE THE L.O.S. VARIABLES FOR THE CALCULATION OF $P_{i,j,m}$

READ IN THE L.O.S. VARIABLES FOR THE OTHER MODES IN ORDER TO DETERMINE THEIR EFFECT ON FLOWS.

RETURN
where \( b \) as a subscript means the corresponding L.O.S. variable for the best mode, then the flows in the subroutine are determined by:

\[
TFLOW(I,J,m) = a_0 \text{SUBSUP}(I) \text{SUBDEM}(J) A_{1m} A_{2m} A_{3m} A_{4m} A_{5m} A_{6m}
\]

The parameters are identified by the array GAMMA (1) to GAMMA (15) and they are in the following order: \( a_0, a_1, b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9, b_{10}, b_{11}, b_{12} \). The variable KMOD, the mode for which the calculations are done, is an argument of the subroutine. In figure 9, we have a macro-flowchart of the subroutine.

7.3.6 "Subroutine SPM-1"

The SPM-1 model is implemented and the general form of the equation used to calculate the flows is given in equation (7) where the sets of \( Z_l \) and \( R_{rlm} \) have to be explicitly defined. In the subroutine, the functional form of the \( R_{rlm} \) used is given by equations (12), (13) and (14) where the \( X_{rlmi} \) are the level of service variables \( i \) for mode \( m \) between the nodes \( r \) and \( l \). As an example, equation (14) becomes:

\[
R_{klm} = \frac{1}{M} \sum_{q=1}^{M} A_B C_D E_F
\]

where

\[
A_q = \text{CUMR}(K,l)_q^{w_{1mq}}
\]

\[
B_q = \text{CUMWT}(K,l)_q^{w_{2mq}}
\]

\[
C_q = \text{CUMTT}(K,l)_q^{w_{3mq}}
\]
CALCULATE THE COEFFICIENTS $A_{1m}$, $A_{2m}$, $A_{3m}$, $A_{4m}$, $A_{5m}$, $A_{6m}$

CALCULATE FLOWS FOR EACH MODE

RETURN
\[ D_q = \text{CUMVT}(K, 1)_q \]

\[ E_q = \text{CUMPLS}(K, 1)_q \]

\[ F_q = \text{CUMC}(K, 1)_q \]

Now the functional form of \( Z \) used in the subroutine is given by the following relations:

\[ Z_K = \text{SUBSUP}(K)^{c1}_m \text{POP}(K)^{c2}_m \text{INCM}(K)^{c3}_m \text{SOC}(K)^{c4}_m \]

\[ Z_1 = \text{SUBDEM}(1)^{c5}_m \text{POP}(1)^{c6}_m \text{INCM}(1)^{c7}_m \text{SOC}(1)^{c8}_m \]

In the equations POP, INCM, SOC are any three socio-economic variables given at supply and demand nodes. SUBSUP and SUBDEM are the subcommodity supply and demand data.

In this case also, any selection of the L.O.S. variables can be chosen. Now the parameters defined by the matrix \( A_{ij} \) in the equation (7) are identified in the subroutine to the matrix \( \Delta \) which is a \( 3 \times 3 \) matrix. Thus we have:

\[
\Delta = \begin{pmatrix}
A_{11} & A_{12} & A_{13} \\
A_{21} & A_{22} & A_{23} \\
A_{31} & A_{32} & A_{33}
\end{pmatrix}
\]

It should be noted that some of the elements \( A_{ij} \) are not used in equation (7) but it is more convenient to define the full matrix. The parameters in the
expressions for the $Z_r$ and the $R_{r1m}$ will be obtained from the array 
ALPHA in the case of the expressions (12) and (13) and from the array 
BETA in the case of the equation (14).

SPM1 is called by the subroutines RKLM12 and RKLM3 which are used 
to calculate the expressions (12), (13), (14) for the $R_{r1m}$ (see chapter 
three). The subroutine RKLM12 calculates the first two expressions (12) and 
(13) and the results of the $R_{r1m}$ calculations are temporarily stored in the TFLOW 
in order to pass them to SPM1. Similarly RKLM3 is used to calculate the 
expression (14) and to store the results in TFLOW. Then SPM1 is called 
and the calculations of the flows are done from this stage 
in this subroutine. In figure 10, a macro-flowchart of the operations in SPM1 is 
shown.

7.3.7 "BLOCK DATA"

This block data subprogram contains all the parameters necessary to 
use the direct demand models and the special product models implemented 
in the modified transport sector. The direct demand models parameters 
declared are those of the McLynn, the Sarc-Kraft, the Baumol-Quandt models. 
The family of models defined by the SPM-1 are also available in this block 
data.

The first array is INDEX (.). It is used to select any subset of 
the six possible levels of service variables defined in the direct demand 
models. If INDEX (1)=2, INDEX(2)=4 and all other values of INDEX(.) are 
zero, the L.O.S. defined by CUMWT and CUMPLS would then be used.

The variable LOSTAP is the Fortran number of the file used to store 
the L.O.S. matrix of variables.
DECIDE IF EXPRESSION (13) IS CONSIDERED OR NOT AND IF (13) IS USED MAKE CHANGES TO TFLOW.

CALCULATE $Z_1$

CALCULATE 1st AND 2nd RATIOS IN FORMULA (7).

CALCULATE $Z_r$ IN FORMULA (7).

CALCULATE 3rd RATIO IN FORMULA (7)

CALCULATE THE 4th TERM IN FORMULA (7)

RETURN
The array MODL(.) defines the modes (other than the transfer mode) that should be considered when the model calculates trees and routes for specific modes. If highway, rail and water are the available modes, then the data in MODL(.) should be MODL(1)=20, MODL(2)=30, MODL(3)=40 and all the other values of MODL(.) are zero.

The array MODKOM(.) has all its values equal to zero unless a given subcommodity KOM is assigned to a specific mode when the flows from supply to demand nodes are assigned to the network. The argument of MODKOM(.) is a subcommodity number KOM and the value of MODKOM(KOM) is the mode used.

The array ALPHA(.,.,.) contains the parameters of the McLynn model. The first index refers to the 10 parameters of a McLynn model for a given mode and a given mode class. The second index refers to the modes used in the network and defined in MODL(.). The third index applies to the mode class defined in MODCLS(.). Thus for a given mode and a given mode class, the 10 values of the first index are $a_{0m}$, $a_{1m}$, $a_{2m}$, $b_{0m}$, $b_{1m}$, $b_{2m}$, $b_{3m}$, $b_{4m}$, $b_{5m}$, $b_{6m}$ in that order as in equation (21) in section 6.3.3 concerned with the subroutine DISTRC. The mode class are defined in the array MODCLS(.). It is used to assign each subcommodity to a given class of parameters for a given model. Thus if the McLynn model is used, IDISTB(KOM) is equal to 3 and if the subcommodity KOM uses the set of parameters defined by class 4, then MODCLS(KOM) should be made equal to 4. As it is implemented now in the transport sector, there is a possibility of a maximum of 5 mode classes to choose from.
The array BETA(.,.,.) contains the parameters of a Sarc-Kraft model.
The second and third index have the same role as in ALPHA and define the
mode and the mode class. The mode can be any number less than four and the
mode class can be any class. The first index when it varies from one to
27 defines the parameters of a Sarc-Kraft model for a given mode and a given mode
class. They are as follows:

\[ \begin{align*}
a_{om}, a_{1m}, a_{2m} & \\
b_{11m}, b_{12m}, b_{13m}, b_{14m} & : \text{CUMR} \\
b_{21m}, b_{22m}, b_{23m}, b_{24m} & : \text{CUMWT} \\
b_{31m}, b_{32m}, b_{33m}, b_{34m} & : \text{CUMTT} \\
b_{41m}, b_{42m}, b_{43m}, b_{44m} & : \text{CUMVTT} \\
b_{51m}, b_{52m}, b_{53m}, b_{54m} & : \text{CUMPLS} \\
b_{61m}, b_{62m}, b_{63m}, b_{64m} & : \text{CUMC} 
\end{align*} \]

When \( m \) is equal to 1, we have the first 27 values of the parameters in the model
and \( m \) takes the values 1,2,3 or 4 for each possible mode. The first sub-
script on the \( b_{ijm} \) refers to the variable used in the level of service
variables. Thus, as in equation (22) in section 7.3.4, there is one set of
parameters for each level of service variable considered.

The array GAMMA(.) defines the parameters of the Baumol-Quandt model and
are well identified in equation (23) and the subsequent material in section
7.3.5.
The next array DELTA is a matrix of coefficients $A_{ij}$ used in the SPM-1 model defined by equation (7). The identification of the coefficients is made in this equation in section 7.3.6.

7.3.8 "Main Program"

This program was written in the first phase of tests on the subroutine NWTREE. It is the controlling program and as such in the first task it reads all the necessary data on the network and the 5 components of the R-factors in order to calculate the trees and the corresponding cumulative sums. The main program calls the subroutine NWTREE which does all the calculations on the data.

Then the main program produces as output a table of the input data and also a table of the minimum R-path for each mode from a given home node. This table contains also the cumulative sums and is identical to the tables 7, 8, 9. In figure 11, a macro-flowchart of the operations of the main program is displayed.

7.3.9 "Subroutine NWTREE"

This subroutine is called by DRCDMD during the assignment process. It is used to calculate trees from a given HOME NODE over a specific mode. At the same time, the calculations of the corresponding cumulative sums RSUM, WTSUM, TTTSUM, VTTSUM, PLSSUM, CSUM are carried out. The tree information given by NWTREE is stored in the array NN. For a given node value of the argument of NN(.), the number stored in this location is the link number of the link having this node as JNODES and being on the minimum R-path going to that node from the home node.
FIGURE 11  MACRO-FLOWCHART OF THE MAIN PROGRAM

READ INPUT DATA ON NETWORK AND COMPONENTS OF THE R-FACTORS.

REPRODUCE THE INPUT DATA AS OUTPUT.

DO FOR EACH HOME NODE

DO FOR EACH MODE

CALCULATE TREES AND THE CORRESPONDING CUMULATIVE SUMS IN SUBROUTINE WNTREE

PRODUCE A TABLE OF RESULTS LIKE TABLE 9, CHAPTER 6.

STOP
The subroutine NWTREE has as arguments the home node NHOME, the R-factors RFAC and the array of components of the RFAC, namely CFAC. Other arguments are the array RCUM and NCUM used as working locations in NWTREE. In addition the arguments contain the output of NWTREE, namely the tree in the array NN and the cumulative sums in the arrays RSUM, UTSUM, TTSUM, VTTSUM, PLSSUM, CSUM. The final argument MODL is the mode over which the tree should be determined.

The following figure 12 gives a macro-flowchart of the operations of NWTREE.

7.3.10 "Subroutine NADDON"

This subroutine is called at the assignment stage in the process by the program TRNDIS. It is used to assign the flows over each mode in the transportation network. It uses the flows calculated at a previous stage and the trees which were stored before reaching this step.

The subroutine NADDON has as arguments the season number, corresponding to the assignment being done, the number of modes over which the flows have to be assigned (MMMOD), the matrix of flows for a given mode (FLOW), the array NN into which the trees over specific mode are retrieved and ADDKOM a vector which cumulates the total flows over each link as they are assigned. While carrying out these calculations, NADDON uses the file LOSTAP on which the flows are temporary stored before entering NADDON. Also it makes use of the file TRETAP which allows the system to retrieve the trees over each mode and the paths used by the flows of each subcommodity. In figure 13, a macro-flowchart of the operations of NADDON is shown.
INITIALISE WORKING VECTORS NCUM, RCUM AND THE SIX ARRAYS OF SUMS.

DETERMINE THE NEXT ENTRIES INTO THE RCUM TABLE.

SEARCH THE CUM TABLE FOR PROPER VALUE OF MODL AND SET THE VALUE OF MATCH ACCORDINGLY.

SEARCH THE CUM TABLE FOR THE SMALLEST RCUM WHICH IS COMPATIBLE WITH MATCH.

REMOVE MINIMUM RCUM FROM THE TABLE.

STORE THE LINK NUMBER IN THE ARRAY NN AND DETERMINE THE CORRESPONDING FIVE CUMULATIVE SUMS.

HAVE WE GONE THROUGH ALL THE NODES IN THE NETWORK?

RETURN YES
DO FOR EACH MODE

READ THE FLOWS FOR THAT MODE WHICH ARE TEMPORARILY STORED ON LOSTAP.

DO FOR EACH SUPPLY NODES

READ THE TREE FOR THAT MODE AND FOR THAT SUPPLY NODE FROM TRETAP

DO FOR EACH DEMAND NODE

USING THE TREE, WORK YOUR WAY BACK FROM THE DEMAND NODE TO THE SUPPLY NODE AND ASSIGN THE FLOWS TO THE LINKS ON THE MINIMUM R-PATH.

RETURN
7.3.11 "Subroutine RCOMP"

This subroutine is called by ASSIGN before that the system enters the program NETFLO. It is calculating the R-factors and the 5 components of the R-factors based on the commodity preference vector CPV and the link performance vector for a given season which is stored in the array VOLTON. The subroutine, as modified, will keep the components of the R-Factors as well as the R-factors for later use in the transport sector. While it is doing these tasks, RCOMP creates a file of R-factors and of the components on the file RFCTAP. It is also providing the user with the option of printing tables of R-factors for future references. A macro-flowchart of the operations is given in figure 14.

7.3.12 "Subroutine LNUPDT"

This subroutine has been modified in the updating process in order to account for the incremental assignment where the updating for each increment means that the quantities actually assigned must be added to the preceding quantities already assigned. This subroutine has only one argument IS. It is the season number for which the updating is done. LNUPDT is called by NETFLO at the end of the assignment process. In order to carry out these tasks, LNUPDT uses the old network link file LNTAPE and will update the file and store the results in the link file LNEW. At the same time, it creates a file of link performance vectors LNSTOR to be used in the next season assignment in RCOMP. The macro-flowchart in figure 15 shows the overall operations of LNUPDT.
FIGURE 14 MACRO-FLOWCHART OF THE SUBROUTINE RCOMP
DO FOR EACH SUBCOMMODITY
DO FOR EACH COMPONENT OF R

CALCULATE THIS COMPONENT OF R AND CUMULATE FOR THE R-FACTORS

PRODUCE A TABLE OF OUTPUT OF THE R-FACTORS AND OF THE COMPONENTS.

STORE THE R-FACTORS FOR THIS SUBCOMMODITY AND THE FIVE COMPONENTS.

RETURN
DO FOR EACH LINK

READ THE LINK FILE LNTAPE

CALCULATE THE NUMBER OF VEHICLES IN EACH CLASS AND UPDATE THE NEW LINK FILE LNEW

CONSTRUCT THE FILE LNSTOR TO BE USED BY RCOMP NEXT SEASON

RETURN
Chapter VIII Conclusions

In this work, a review of the Harvard Brookings Model has been undertaken and those difficulties and problems encountered when using the model identified. Some of these had been recognized for quite some time by users of the model system as the result of doing applied work. In this report some modifications and additions were made to the model. In doing so, an effort has been made to use to our advantage, as much as possible, the actual features of the system with its logical structure. The modelling capacity of a whole family of direct demand models (the SPM-I models) as well as the Sarc-Kraft model, the Baumol-Quandt model and the McIynn model have been added to the model. The modularity of the original system has made this possible. In order to achieve the additions of new modelling capacity, additional data and variables were needed. This is why the tree algorithm was modified to obtain the added level of information by mode that is necessary in a multimodal network using this form of direct demand model.

The assignment process is normally composed of the following steps: 1) construction of trees, 2) routing, 3) mode choice, 4) distribution and assignment of the flows to the network. If one task involved the additions of modelling capacity to the system, another one was concerned with methods to carry out the assignment process and the implementation of a version of incremental assignment. This assignment technique can be used as well as the original one which has been preserved.

All these new features have been coded and tested with a network and with data that was part of a study where the model system was used. This data
has not been used to make extensive comparisons of the new modelling tools with those originally in the system but rather it was used to test the new methodology and insure that it was performing the numerical calculations as it was intended. The performance of the models should be compared to other models. This should be done in greater detail than it was possible to do in this work. In doing these comparisons, some experimental knowledge will be gained and it could then be used to decide upon the most appropriate model for a given situation.

The data was also used to test the incremental assignment technique. Again it is clear that more experimental tests and results will be needed to compare the implemented incremental assignment procedure with other methods. It is necessary to gather this knowledge in order to decide upon the desirability of implementing some computer costs reduction improvements (such as the average R-factors method). It would also be interesting to compare the incremental assignment approach with the results of assignments made by the iterative method available in the present model systems.

This work has concentrated on the first phase, an important step in the implementation, and a first evaluation of the improvements. The author has reason to believe that some of these expensive tests and comparisons will be done in the near future in the course of a transportation study in which he will be involved.

This work is a case where some of the guidelines defined in (2) were applied in improving upon an existing system. It is hoped that this work has contributed to enlarging and broadening the scope and the generality of a model system. It is clear that it was a real advantage to be able to start from a well organized model system and improve upon it. It is finally hoped
that this is only a first phase in this process and that some of the suggested improvements raised and discussed can be added to the modified system in the near future.
REFERENCES


APPENDIX 1
COMPUTER LISTINGS OF ALL SUBROUTINES
This program is called in NETFLO and is the controlling program for the distribution of subcommodity flows over each mode. It is going to perform transportation statistics summary as well as going to make the assignment of these flows to the network.

Paul Robillard, July, August 1973

Common /QTAPES/ NTAPE, ACUT, RECTAP, TRETAP, IMACRC, ITRANS, NINM,
    1 NOTHER, NOTM, Q00001, ITRNEW, Q00002, LNTAPE, LNEW, LSTDAP, Q00003,
    2 TRNSUM, Q00004, LNSTOR, C00005, TABTAP, Q00006, TABNEW, Q00007,
    3 HEADIM(6), HEADIT(6), HEADLN(6), HEADLS(6), HEADTA(6), HEADRF(6),
    4 HEADTR(6), HEADTS(6), HEADZZ(6), TAPNAM(20)
Common /TOIMEM/ IMX, IPX, ISX, IXT, JSX, K1L, K1X, KMX, LNX, MMX, MNX, 1NTF0120
    1 NCX, NDX, NNX, NPX, NSX, NT, NTX, NVX, NX, NXX, NZX 1NTF0130
Common /CBKKCM/ KOMNAME(100), VALKOM(100), ICLASS(100), CPV(100,5),
    2 IDISTB(100), EXPONENT(100), HCOEF(100), TCOEF(100)
Common /MINDEX/ INDEX(6)
Common /MODUSE/ NBMOD, MBMOD, MODT, LOSTAP, MODL(10), MODKOM(100)
    1 MODCLS(100)
Common /CKOM / KOM
Common /NODESI/ NODDEM, SPACE, NODSUP
Common /CIIJ1 / I, J
Common /DAYSEZ/ DAYS
Common /CIAVOL/ IA VOL
Common /SUBKOM/ SUBDEM, SUBRUC, SUBSUP, DIFF, DUMMY
Common /CCFLOW / FLOW
Common /NAMMOD/ BLANK, ACODNAM
Dimension NN(300)
Dimension IAVOLJ(900,5,4)
Dimension FLOW(20,40)
Dimension TFLOW(20,40,5)
Dimension NODDEM(40), NODSUP(20)
Dimension DAYS(4)
Dimension CUMWT(20,40), CUMTT(20,40)
Dimension CUMVTT(20,40), CUMPLS(20,40)
Dimension CUMC(20,40), CUMR(20,40)
DIMENSION SUBDEM(40),SUBRUC(20)
DIMENSION SUBSUP(20),DIFF(40),DUMMY(180),ADDKOM(1500)
DIMENSION P(20,40)
EQUIVALENCE (SUBDEM(1),ADDKCM(1))
EQUIVALENCE (IAVOLJ(1,1,3),CUMR(1))
EQUIVALENCE (IAVOLJ(601,1,3),CUMWT(1))
EQUIVALENCE (IAVOLJ(701,2,3),CUMTT(1))
EQUIVALENCE (IAVOLJ(601,3,3),CUMVTT(1))
EQUIVALENCE (IAVOLJ(501,4,3),CUMPLS(1))
EQUIVALENCE (IAVOLJ(401,5,3),CUMC(1))
EQUIVALENCE (IAVOLJ(301,1,4),TFLOW(1,1,1))
EQUIVALENCE (IAVOLJ(201,2,4),TFLOW(1,1,2))
EQUIVALENCE (IAVOLJ(101,3,4),TFLOW(1,1,3))
EQUIVALENCE (IAVOLJ(1,4,4),TFLOW(1,1,4))
EQUIVALENCE (IAVOLJ(801,4,4),TFLOW(1,1,5))
EQUIVALENCE (ADDKCM(301),P(1))
INTEGER TRNSUM
INTEGER TRETAP
INTEGER RFCTAP
DATA SUM /4HSUMS/
REWIND LCSTAP
LOGICAL WRITC
ISN=IS
DO 10 I=1,20
DO 11 J=1,40
FLOW(I,J)=0
P(I,J)=0
11 CONTINUE
10 CONTINUE
PERTON=VALKOM(KOM)
C
READ LOS VARIABLES FOR EACH MODE AND PERFORM DISTRIBUTION
C
DO 1 KMCD=1,MBMCD
READ(LOSTAP) KM,MBM,KOMTAP,I6,J6, ((CUMR(I,J),I=1,II),J=1,JJ),
 1   ((CUMWT(I,J),I=1,II),J=1,JJ),
C TEST TO SEE IF KOMTAP=KCM

IF(KOMTAP.EQ.KOM) GO TO 2
WRITE(NOUT,3) KOMTAP,KOM
3 FORMAT(1HO, 'INTRNDIS ,KOMTAP=', I5, 2X, 'WHILE KOM=', I5)
   CALL EXIT

CALL THE PROPER DIRECT DEMAND DISTRIBUTION MODEL FOR THIS
SUBCOMMUNITY AND PERFORM THE DISTRIBUTION.

USE MC-LYAN MODEL

2 IF(IDISTR(KOM).EQ.3) CALL DISTRC(KMOD)

USE SARC-KRAFT MODEL

IF(IDISTR(KOM).EQ.4) CALL DISTRD(KMOD)

USE BAUMCL-QUANDT-ABSTRACT MODE MODEL

IF(IDISTR(KOM).EQ.5) CALL DISTRE(KMOD)

IF(IDISTR(KOM).EQ.6) CALL RKLM12(KMOD)
IF(IDISTR(KOM).EQ.7) CALL RKLM12(KMOD)
IF(IDISTR(KOM).EQ.8) CALL RKLM3(KMOD)
IF(IDISTR(KOM).GE.9) WRITE(NOUT,4) KOM
4 FORMAT(1HO, 'FOR SUBCOMMUNITY', I5, 2X, 'A WRONG DISTRIBUTION MODEL
1 IS SPECIFIED')
   IF(IDISTR(KOM).GE.9) CALL EXIT

PAGE 149
Determine the average cumr over all modes and convert these averages cumulative r-factors to a dollard of subcommodity basis. These will be written on the transport summary file (TRNSUM) and will be used by subroutine SUMFLO for summary statistics.

DO 13 I=1,II
DO 14 J=1,JJ

It is assumed that we do not deal with more than 4 modes at one time.

FLOW(I,J) = FLOW(I,J) + CUMC(I,J)/(MBMOD*PERTON)
14 CONTINUE
13 CONTINUE
1 CONTINUE
REWIND LOSTAP

Store averages cumulative c-factors.
WRITE(TRNSUM) KOM,II,JJ, ((FLOW(I,J), I=1,II),J=1,JJ)

Normalyse total flows in TFLOW such that the total flows predicted are equal to total supply.

TFLOW=0.
DO 40 KMOD=1,MBMOD
DO 41 I=1,II
DO 42 J=1,JJ
TFLOW=TFLOW+TFLOW(I,J,KMCC)
42 CONTINUE
41 CONTINUE
40 CONTINUE
TOTAL SUPPLY

TSUP=0.0
DO 43 I=1,II
   TSUP=TSUP+SUBSUP(I)
43 CONTINUE

CALCULATE NORMALIZED TOTAL FLOWS....

DO 44 KMOD=1,MBMOD
   DO 45 I=1,II
      DO 46 J=1,JJ
         TFLOW(I,J,KMOD)=TFLOW(I,J,KMOD)*(TSUP/TFLW)
46 CONTINUE
45 CONTINUE
44 CONTINUE

TABLES OF FLOWS FOR EACH MODE CAN BE WRITTEN OUT.

CALL OUTPUT(3,1,WRITC)
IF(.NOT.WRITC) GO TO 15
DO 16 K=1,MBMOD
   DO 17 I=1,II
      DO 18 J=1,JJ
         FLOW(I,J) = TFLOW(I,J,K)
18 CONTINUE
17 CONTINUE
   CALL TABLE(FLOW,BLANK,24HFLOWS IN TONS PER DAY ,SUM)
   WRITE(NOUT,19) MODL(K)
19 FORMAT(1HO,'THESE FLOWS IN TONS PER DAY ARE FOR MODE',I5)
16 CONTINUE
15 CONTINUE

CONVERT THE FLOWS INTO TONS/SEASON AND STORE THEM ON THE TRANSPORT SUMMARY FILE. IN THIS CASE, THE SUM OF THE FLOWS OVER ALL MODES

PAGE 151
SUM ALL THE FLOWS OVER ALL MODES AND CONVERT THEM AND STORE THEM

CONTINUE

STORE FLOWS / SEASON ON TRNSUM FILE.

WRITE(TRNSUM) KCM,II,JJ,((FLOW(I,J),I=1,II),J=1,JJ),

CALL OUTPUT(3,7,WRITO)

IF(.NOT.WRITO) GO TO 25

CALL TABLE(FLOW,4H-2P,24HFLOW/SEASON(100TONS),SUM)

CONTINUE

CONVERT ALL THE FLOWS IN TFLOW IN DOLLARDS/SEASON BEFORE GOING
TO PROCEED TO THE ASSIGNMENT.

CONTINUE

CONTINUE

CONTINUE
CALL TRACER(4HTDIS,6)

THE DISTRIBUTION OF FLOWS HAVE BEEN DETERMINED. NOW WE WILL USE
THE MINIMUM PATH TREE FOR EACH MODE DETERMINE D BEFORE TO
ASSIGN THE FLOWS TO THE NETWORK.

BECAUSE OF CORE SPACE USE, TFLOW IS GOING TO BE ERASED BY UNPACK,
SO STORE TFLOW ON LOSTAP IN ORDER TO RETRIEVE IT IN ADDON....

REWIND LOSTAP
DO 30 KMOD=1,MBMOD
K=KMOD
WRITE(LOSTAP) K,((TFLOW(I,J,K),I=1,II),J=1,JJ)
30 CONTINUE
REWIND LOSTAP
CALL UNPACK
CALL TRACER(4HTDIS,7)
CALL NADDON(ISN,MBMOD,FLOW,NN,ADDKCM)
CALL TRACER(4HTDIS,8)
CALL REPACK
REWIND TRETAP
RETURN
END
SUBROUTINE DISTRC(KMOD)

THIS IS THE MCLYNN MODEL USED WITH A POSSIBILITY OF 6 L.O.S. VAR...

COMMON /QTAPES/ NTAPE ,NOUT ,RFCTAP,TRETAP,IMACRC,ITRANS,NINM ,
1 NOTHER,NOUTM ,Q00001,ITRNEW,Q00002,LNTAPE,LNEW ,LSDTAP,Q00003,
2 TRNSUM,Q00004,LNSTOR,Q00005,TARTAP,Q00006,TABNEW,Q00007,
3 HEADIM(6),HEADIT(6),HEADLN(6),HEADLS(6),HEADTA(6),HEADRF(6),
4 HEADTR(6),HEADTS(6),HEADZZ(6),TAPNAM(20)
COMMON /SUBKCM/ SUBDEM, SUBRUC, SUBSUP, DIFF, DUMMY
COMMON / DIRDMD / ALPHA(16,4,5), BETA(33,4,5), GAMMA(21), DELTA(3,3)
COMMON/MODUSE/ NMOD,MFBMOD,MODT,LOSTAP,MODL(10),MODKOM(100)
!
1 MODCLS(100)
COMMON/UCKOM / KOM
COMMON /CMCMMON / CK
 COMMON /CMCOMMON / CI
 COMMON /CPCOMPON Idl
 COMMON /NSIMMON / N

DIMENSION IAVOLJ(900,5,4)
DIMENSION ADDKOM(150C)
DIMENSION SUBDEM(40),SUBRUC(20)
DIMENSION SUBSUP(20),DIFF(40),DUMMY(180)
DIMENSION TFLOW(20,40,5),P(20,40)
DIMENSION C(20,40,6)
EQUIVALENCE (SUBDEM(1),ADKOM(1))
EQUIVALENCE (IAVOLJ(1,1,3), C(1,1,1))
EQUIVALENCE (IAVOLJ(801,1,3),C(1,1,2))
EQUIVALENCE (IAVOLJ(701,2,3),C(1,1,3))
EQUIVALENCE (IAVOLJ(601,3,3),C(1,1,4))
EQUIVALENCE (IAVOLJ(501,4,3),C(1,1,5))
EQUIVALENCE (IAVOLJ(401,5,3),C(1,1,6))
EQUIVALENCE (IAVOLJ(301,1,4),TFLOW(1,1,1))
EQUIVALENCE (IAVOLJ(201,2,4),TFLOW(1,1,2))
EQUIVALENCE (IAVOLJ(101,3,4),TFLOW(1,1,3))
EQUIVALENCE (IAVOLJ(1,4,4),TFLOW(1,1,4))
EQUIVALENCE (IAVOLJ(801,4,4),TFLOW(1,1,5))
EQUIVALENCE (ADKOM(301),P(1))

THE THIRD DIMENSION IN ,BETA, ALPHA, GAMMA, REFERS TO COMMODITY
NUMBER OR TO ONE OF THE CLASS 1 TO 5 THAT COMMODITY KOM IS
ASSIGNED...

FOR THE MOMENT LET US USE IKOM=MODCLS(KOM)

IKOM=MODCLS(KOM)
IF(IKOM.GT.5) IKOM=1

C BY DEFAULT IKOM=1 IF IKCM IS GT.5

WRITE(NOUT,50) INDEX,II,JJ,KMOD,MAMOD,MODL,MODKOM,MODCLS

50 FORMAT(1HC,25(15))
CC 51 K=1,6
DO 52 I=1,II
WRITE(NOUT,53) ( C(I,J,K),J=1,JJ)
53 FORMAT(1HC,15F8.2)
52 CONTINUE
51 CONTINUE
IRETN=0
DO 1 I=1,II
DO 2 J=1,JJ
CST=1.0
DO 3 INN = 1,6
IN=INDEX(INN)
IF((IN.GT.6).OR.(IN.LE.0)) GO TO 3

C TEST IF C(I,J,IN) LESS THAN OR EQUAL TO ZERO...

IF(C(I,J,IN).GT.0) GO TO 4
WRITE(NOUT,25)
25 FORMAT(1HC,'IN DISTRC THERE IS A ZERO COST ELEMENT')
IRETN=IRETN+1
C(I,J,IN)=0.0001
4 K=4+IN
A=ALPHA(K,KMOD,IKOM)
CST=CST*(C(I,J,IN)**A)
3 CONTINUE
CST=CST*ALPHA(4,KMOD,IKOM)
P(I,J)=P(I,J)+CST
SOC=ALPHA(1,KMOD,IKOM)
A=ALPHA(2,KMOD,IKOM)
B=ALPHA(3,KMOD,IKOM)
A11 = ALPHA(11,KMOD,IKCM)
A12 = ALPHA(12,KMOD,IKCM)
A13 = ALPHA(13,KMOD,IKCM)
A14 = ALPHA(14,KMOD,IKCM)
A15 = ALPHA(15,KMOD,IKCM)
A16 = ALPHA(16,KMOD,IKCM)
ISUP=NCDSUP(I)
IDEM=NODDEM(J)
SCC = SCC*(SUBSUP(I)**A1)*(SUBDEM(J)**B)*(POP(ISUP)**A11)*(
1 TINCM(ISUP)**A12) *(SOCIO(ISUP)**A13) *(POP(IDEM)**A14) *
2 (TINCM(IDEM)**A15) *(SOCIO(IDEM)**A16)
TFLOW(I,J,KMCD)=SOC*CST
2 CONTINUE
1 CONTINUE
IF(IRETN.EQ.0) GO TO 5
WRITE(6,6) IRETN,KMCD
6 FORMAT(1HO,'NUMBER OF ZERO ELEMENTS IN MODE',2(I5))
C
C TEST TO SEE IF WE ARE AT THE LAST MODE
C
5 IF(KMOD.EQ.MBMOD) GO TO 7
RETURN
7 DC 10 K=1,MBMOD
DO 8 I=1,II
DO 9 J=1,JJ
TFLOW(I,J,K)=TFLOW(I,J,K)/P(I,J)
8 CONTINUE
8 CONTINUE
10 CONTINUE
C
C THE FLOWS PRODUCED BY THE MCLYNN MODEL ARE NOW RETURNED...
RETURN
END
SUBROUTINE DISTR0(KMCD)
C
C THIS IS THE SARC KRAFT MODEL USED WITH A POSSIBILITY OF 6 L.O.S.
COMMON /CTAPES/ NTAP, NOUT, RFCTAP, TRETAP, IMACRC, ITRANS, NINM
1  NOTHER, NOUTM, Q00001, ITNNEW, Q00002, LNTAPE, LNEW, LSDTAP, Q00003,
2  TRNSUM, Q00004, LNSDOR, Q00005, TARTAP, Q00006, TAHNEW, Q00007,
3  HEADIN(6), HEADIT(6), HEADLN(6), HEADLS(6), HEADTA(6), HEADRF(6),
4  HEADTR(6), HEADTS(6), HEADZZ(6), TAPNEW(20)
COMMON/MODUSE/ NBM009MBMODGMODTiLOSTAPiMCOL(10)tMODKCM(100)
I  MCCCLS(100)
COMMON /SUBKCM/ SUBDEM, SUBRUC, SUBSUP, DIFF, DUMMY
COMMON /DIRMD/ ALPHA(16, 4, 5), BETA(33, 4, 5), GAMMA(21), DELTA(3, 3)
COMMON /CKOM/ KOM
COMMON /CIJJ/ II, JJ
COMMON /MINDEX/ INDEX(6)
COMMON /CIAVCL/ IAVCL
COMMON /NODES1/ NODDEM(40), SPACE, NODSUP(20)
COMMON /SCCATA/ POP(300), TINCM(300), SOCIO(300)
DIMENSION IAVOLJ(900, 5, 4)
DIMENSION ADDKCM(1500)
DIMENSION SUBDEM(40), SUBRUC(20)
DIMENSION SUBSUP(20), DIFF(40), DUMMY(180)
DIMENSION TFLOW(20, 4, 5), P(20, 40)
DIMENSION C(20, 40, 6)
EQUIVALENCE (SUBDEM(1), ADDKCM(1))
EQUIVALENCE (IAVOLJ(1, 1, 3), C(1, 1, 1))
EQUIVALENCE (IAVOLJ(601, 1, 3), C(1, 1, 2))
EQUIVALENCE (IAVOLJ(701, 2, 3), C(1, 1, 3))
EQUIVALENCE (IAVOLJ(601, 3, 3), C(1, 1, 4))
EQUIVALENCE (IAVOLJ(501, 4, 3), C(1, 1, 5))
EQUIVALENCE (IAVOLJ(401, 5, 3), C(1, 1, 6))
EQUIVALENCE (IAVOLJ(301, 1, 4), TFLOW(1, 1, 1))
EQUIVALENCE (IAVOLJ(201, 2, 4), TFLOW(1, 1, 2))
EQUIVALENCE (IAVOLJ(101, 3, 4), TFLOW(1, 1, 3))
EQUIVALENCE (IAVOLJ(1, 4, 4), TFLOW(1, 1, 4))
EQUIVALENCE (IAVOLJ(801, 4, 4), TFLOW(1, 1, 5))
EQUIVALENCE (ADDCM(301), P(1))
THE THIRD DIMENSION IN $\beta$, $\alpha$, $\gamma$, REFERS TO COMMODITY NUMBER OR TO ONE OF THE CLASS 1 TO 5 THAT COMmodity KOM IS A ASSIGNED...

FOR THE MOMENT LET US USE IKOM=MODCLS(KOM)

IKOM=MODCLS(KCM)
IF(IKOM.GT.5) IKOM=1

BY DEFAULT IKOM=1 IF IKCM IS GT.5 .......... 

WRITE(NCUT100) INDEX, II, JJ, KMOD, MBMOD, MODL, MODKCM, MODCLS
50 FORMAT(1HC929(I5))
CC 51 K=1,6
DO 52 I=1,II
WRITE(NCUT100) (C(I,J,K), J=1, JJ)
53 FORMAT(1HC929(15F3.2))
52 CONTINUE
51 CONTINUE
IRETN=0

INITIALIZE MATRIX $P(I, J)$ TO 1

CO 20 I=1,II
CC 30 J=1, JJ
P(I,J)=1.
30 CONTINUE
20 CONTINUE

FOR THE FIRST TIME GOING THROUGH LOOP 3 PUT L=1

L=1
8 DO 1 I=1,II
CC 2 J=1, JJ
DO 3 INN=1,6
IN=INDEX(INN)
IF((IN.GT.6).OR.(IN.LE.0)) GO TO 3

TEST IF C(I,J,IN) LESS THAN OR EQUAL TO ZERO...

IF(C(I,J,IN).GT.0) GO TO 4
WRITE(NCUT,5)
5 FORMAT(1HC, 'IN DISTRD THERE IS A ZERO COST ELEMENT!')
  IRETN=IRETN+1
  C(I,J,IN)=0.0001
4 K=3+(IN-1)*4+L
  B=BETA(K,KMOD,IKOM)
P(I,J)=P(I,J)*(C(I,J,IN)**B)
3 CONTINUE
IF(L.LT.MBMOD) GO TO 11
SCC = BETA(1,KMOD,IKCM)
A = BETA(2,KMOD,IKCM)
B = BETA(3,KMOD,IKCM)
A28 = BETA(28,KMOD,IKCM)
A29 = BETA(29,KMOD,IKCM)
A30 = BETA(30,KMOD,IKCM)
A31 = BETA(31,KMOD,IKCM)
A32 = BETA(32,KMOD,IKCM)
A33 = BETA(33,KMOD,IKCM)
ISUP=NODSUP(I)
IDEM=NODDEM(J)
SOC = SOC *(SUBSUP(I)**A) *(SURED(J)**B) *(POP(ISUP)**A28) *
  (TINC(M(ISUP)**A29)*(SOC(SUP(ISUP)**A30) *(POP(IDEM)**A31) *
     (TINC(M(I2)IDEM)**A32)*(SOC(I2)IDEM)**A33)
11 CONTINUE
CALL TRACER (4HCTRD,1)
2 CONTINUE
1 CONTINUE
IF(MBMOC.EQ.1) GO TO 6
TEST TO SEE IF L=MBMOD

IF(L.EQ.MBMOD) GO TO 6
CALL TRACER (4HOTRD,2)
L=L+1
READ(LOSTAP) KM,MBM,KOMTAP,I6,J6,((C(I,J,K),I=1,II),J=1,JJ),
1 K=1,6)

TEST TO SEE IF KOMTAP=KCM

IF(KOMTAP.EQ.KOM) GO TO 7
WRITE(NCUT,9) KOMTAP,KOM
9 FORMAT(1HO,'IN DISTRO KCMTAP=',I5,2X,'WHILE KCM=',I5)
CALL EXIT
7 IF(L.LE.MBMOD) GO TO 8
6 REWIND LOSTAP

THIS IS THE END FOR THIS MODE.

IF(IRETN.EQ.0) GO TO 15
WRITE(NCUT,16) IRETN,KMCD
16 FORMAT(1HO,'NUMBER OF ZERO ELEMENTS IN MODE',2(I5))

THE FLOWS PRODUCED BY THE SARC-KRAFT MODEL ARE NOW RETURN.....

15 CONTINUE
RETURN
END

SUBROUTINE RKLM12(KMOD)

THIS SUBROUTINE CALCULATES THE R(K,L,M) FORMULAS.........
THEN IT WILL CALL THE SUBROUTINE SPM-1 TO CALCULATE THE FLOWS

TFLOW(K,L,M) WITH THIS R(K,L,M)..............................

COMMON /QTAPE, NTAPE, NOUT, RFCTAP, TRETAP, IMACRO, ITTRANS, NINM,
1 NOTHER, NOUTM, QO0001, ITNEW, QO0002, LNTAPE, LNEW, LSOTAP, QO0003,
2 TRNSUM, QO0004, LNSTOR, QO0005, TBAITAP, QO0006, TABNEW, QO0007,
3 HEADIM(6), HEADIT(6), HEADLN(6), HEADLS(6), HEADTA(6), HEADDR(6),
4 HEADTR(6), HEADTS(6), HEADZZ(6), TAPNAM(20)

COMMON /SUBKOM, SUBDEM, SUBRUC, SUBSUP, DIFF, DUMMY

COMMON /DIRDMO, /ALPHA(16,4,5), BETA(33,4,5), GAMMA(21), DELTA(3,3)

COMMON/MODUSE, /NBMOD, MBMOD, MODT, LOSTAP, MDGL(10), MODKOM(100),
1 MOECLS(100)

COMMON /CKOM, /KOM

COMMON /CII, JJ

COMMON /MINDEX, INDEX(6)

COMMON /IAVOLJ, IAVOLJ

COMMON /SCDATA, /POP(300), TINCM(300), SOCIO(300)

COMMON /NODESI, NODDEP(40), SPACE, NCDEP(20)

DIMENSION IAVOLJ(900,5,4)

DIMENSION ADDKOM(1500)

DIMENSION SUBDEM(40), SUBRUC(20)

DIMENSION SUBSUP(20), DIFF(40), DUMMY(180)

DIMENSION TFLOW(20,40,5), P(20,40)

DIMENSION C(20,40,6)

EQUIVALENCE (SUBDEM(1), ADDKOM(1))
EQUIVALENCE (IAVOLJ(1,1,3), C(1,1,1))
EQUIVALENCE (IAVOLJ(601,1,3), C(1,1,2))
EQUIVALENCE (IAVOLJ(701,2,3), C(1,1,3))
EQUIVALENCE (IAVOLJ(601,3,3), C(1,1,4))
EQUIVALENCE (IAVOLJ(501,4,3), C(1,1,5))
EQUIVALENCE (IAVOLJ(401,5,3), C(1,1,6))
EQUIVALENCE (IAVOLJ(301,1,4), TFLOW(1,1,1))
EQUIVALENCE (IAVOLJ(201,2,4), TFLOW(1,1,2))
EQUIVALENCE (IAVOLJ(101,3,4), TFLOW(1,1,3))
EQUIVALENCE (IAVOLJ(1,4,4), TFLOW(1,1,4))
EQUIVALENCE (IAVOLJ(601,4,4), TFLOW(1,1,5))
EQUIVALENCE (ADDKOM(301),P(1))

THE THIRD DIMENSION IN , , , , , REFERS TO COMMODITY
NUMBER OR TO ONE OF THE CLASS 1 TO 5 THAT COMMODITY KOM IS
ASSIGNED...

FOR THE MOMENT LET US USE IKOM=MODCLS(KOM)

IKOM=MODCLS(KOM)
IF(IKOM.GT.5) IKOM=1

BY DEFAULT IKOM=1 IF IKC M IS GT.5 ........

WRITE(NCUT,50) INDEX,II,JJ,KMOC,MBMOD,MODL,MODKCM,MODCLS
50 FORMAT(1H0,25(I5))
DC 51 K=1,6
DO 52 I=1,II
WRITE(NCUT,53) ( C(I,J,K),J=1,JJ)
53 FORMAT(1HC*15F8.2)
52 CONTINUE
51 CONTINUE
IRETN=0
DO 2 I=1,II
DO 3 INN = 1,6
IN=INDEX(INN)
IF((IN.GT.6).OR.(IN.LE.C)) GO TO 3

TEST IF C(I,J,IN) LESS THAN OR EQUAL TO ZERO...
IF(C(I,J,IN).GT.0) GO TO 4
WRITE(NCUT,25)
25 FORMAT(1HO,'IN RKLM12 THERE IS A ZERO COST ELEMENT')
IRETN=IRETN+1
C(I,J,IN)=0.0001
4 K=4+IN
   A=ALPHA(K,KMCD,IKOM)
   CST=CST*(C(I,J,IN)**A)
3 CONTINUE
   CST=CST*ALPHA(4,KMOD,IKOM)

STORE IN TFLOW THE ARRAY OF R(K,L,M) TO BE USED BY SPM-1......

TFLOW(I,J,KMCD)=CST
2 CONTINUE
1 CONTINUE
   IF(IRETN.EQ.C) GO TO 5
   WRITE(NCUT,6) IRETN,KMCD
6 FORMAT(1HC, 'NUMBER OF ZERO ELEMENTS IN MODE', 2(I5))

TEST TO SEE IF WE ARE AT THE LAST MODE

5 IF(KMOD.EQ.MBMOD) GO TO 7
   RETURN
7 CALL SPM1
   RETURN
END

SUBROUTINE RKLM3 (KMCD)

THIS SUBROUTINE CALCULATES THE R(K,L,M) FORMULAS.......... 

THEN IT WILL CALL THE SUBROUTINE SPM-1 TO CALCULATE THE FLOWS
TFLOW(K,L,M) WITH THIS R(K,L,M).............

COMMON /CTAPES/ NTAPE,NCUT,RFCTAP,TRETAP,IMACRC,ITRANS,NINM,
   NOTHER,NOUTM,Q00001,IRNEW,Q00002,LNTAPE,LNEW,LSDTAP,Q00003,
   TRNSUM,Q00004,LSNCRQ,Q00005,TABTAP,Q00006,TABNEW,Q00007,
   HEADIM(6),HEADIT(6),HEADLN(6),HEADLS(6),HEADTA(6),HEADRF(6),
   HEADTR(6),HEADTS(6),HEADZZ(6),TAPNAM(2C)
 COMMON/MCDUSE/ NBMOD,MBMOD,MODT,LOSTAP,MODL(10),MODKOM(100) 

PAGE 163
1 MODCLS(100)
COMMON /SUBKCM/, SUBDEM, SUBRUC, SUBSUP, DIFF, DUMMY
COMMON /DIRMDM/, ALPHA(16,4,5), BETA(33,4,5), GAMMA(21), DELTA(3,3)
COMMON /CKOM/, KOM
COMMON /CIIJ/, II, JJ
COMMON /MINDEX/, INDEX(6)
COMMON /CIAVOL/, IAVOLJ
DIMENSION IAVOLJ(900,5,4)
DIMENSION ADDKOM(150C)
DIMENSION SUBDEM(40), SUBRUC(20)
DIMENSION SUBSUP(20), DIFF(40), DUMMY(180)
DIMENSION TFLOW(20,4C,5), P(20,40)
DIMENSION C(20,40,6)
EQUIVALENCE (SUBDEM(1), ADDKOM(1))
EQUIVALENCE (IAVOLJ(1,1,3), C(1,1,1))
EQUIVALENCE (IAVOLJ(601,1,3), C(1,1,2))
EQUIVALENCE (IAVOLJ(701,2,3), C(1,1,3))
EQUIVALENCE (IAVOLJ(601,3,3), C(1,1,4))
EQUIVALENCE (IAVOLJ(501,4,3), C(1,1,5))
EQUIVALENCE (IAVOLJ(401,5,3), C(1,1,6))
EQUIVALENCE (IAVOLJ(301,1,4), TFLOW(1,1,1))
EQUIVALENCE (IAVOLJ(201,2,4), TFLOW(1,1,2))
EQUIVALENCE (IAVOLJ(101,3,4), TFLOW(1,1,3))
EQUIVALENCE (IAVOLJ(1,4,4), TFLOW(1,1,4))
EQUIVALENCE (IAVOLJ(601,4,4), TFLOW(1,1,5))
EQUIVALENCE (ADDKOM(301), P(1))

THE THIRD DIMENSION IN , BETA, ALPHA, GAMMA, REFERS TO COMMODITY
NUMBER OR TO ONE OF THE CLASS 1 TO 5 THAT COMMODITY KOM IS A
ASSIGNED...

FOR THE MOMENT LET US USE IKOM=MODCLS(KOM)

IKOM=MODCLS(KOM)
TF(IKOM.GT.5) IKOM=1
BY DEFAULT IKCM=1 IF IKCM IS GT.5  ..........  

WRITE(NCUT,50) INDEX,II,JJ,KMOC,MBMOD,MODL,MODKOM,MODCLS  
50 FORMAT(1HO,25(I5))  
CC 51 K=1,6  
DO 52 I=1,II  
WRITE(NCUT,53) ( C(I,J,K),J=1,II)  
53 FORMAT(1HO,15F8.2)  
52 CONTINUE  
51 CONTINUE  
IRETN=0  

INITIALIZE MATRIX P(I,J) TO 1  

DO 20 I=1,II  
CC 30 J=1,JJ  
P(I,J)=1.  
30 CONTINUE  
20 CONTINUE  

FOR THE FIRST TIME GOING THROUGH LOOP 3 PUT L=1  
L=1  
E DO 1 I=1,II  
CC 2 J=1,JJ  
DC 3 INA=1,6  
IN=INDEX(INN)  
IF((IN.GT.6).OR.(IN.LE.4)) GO TO 3  

TEST IF C(I,J,IN) LESS THAN OR EQUAL TO ZERO...  

IF(C(I,J,IN).GT.0) GC TC 4  
WRITE(NCUT,5)  
5 FORMAT(1HO,'IN RKLM3 THERE IS A ZERO COST ELEMENT')  
IRETN=IRETN+1
C(I,J,IN) = 0.0001
4 K = 3 + (IN - 1) * 4 + L
B = BETA(K, KMOD, IKOM)
P(I, J) = P(I, J) * (C(I, J, IN) ** 8)
3 CONTINUE
IF(L.LT.MBMOD) GO TO 11
TFLOW(I, J, KMOD) = P(I, J)
11 CONTINUE
CALL TRACER (4HDTRD, 1)
2 CONTINUE
1 CONTINUE
IF(MBMOD.EQ.1) GO TO 6

TEST TO SEE IF L = MBMOD

IF(L.EQ.MBMOD) GO TO 6
CALL TRACER (4HDTRD, 2)
L = L + 1
READ(LOSTAP) KM, MBM, KOMTAP, I6, J6, ((C(I, J, K), I=1, II), J=1, JJ),
1 K=1, II

TEST TO SEE IF KOMTAP = KCM

IF(KOMTAP.EQ.KOM) GO TO 7
WRITE(NOUT, 9) KOMTAP, KCM
9 FORMAT(1HO, 'IN DISTRD KOMTAP=', I5, 2X, 'WHILE KOM=', I5)
CALL EXIT
7 IF(L.LE.MBMOD) GO TO 8
6 REWIND LOSTAP

THIS IS THE END FOR THIS MODE.

IF(IRETN.EQ.0) GO TO 15
WRITE(NCUT, 16) IRETN, KMOD
16 FORMAT(1HO, 'NUMBER OF ZERO ELEMENTS IN MODE', 2(I5))
IF(KPCO.EQ.MEMOD) GC TC 73
RETURN
73 CALL SPM1
RETURN
END
SUBROUTINE SPM1

THIS SUBROUTINE CALCULATES THE FLOWS WITH THE GENERAL SPM-1 MODEL

COMMON /QTAPES/ NTAPE, NOUT, RFCTAP, TRETAP, IMACRC, ITTRANS, NINM,
1 NOOTHER, NOUTM, QQO001, ITRNEW, QQ0002, LNTAPE, LNEW, LSOTAP, QQ0003,
2 TRNSUM, QQ0004, LNSTCR, QQ0005, TABTAP, QQ0006, TARBNEW, QQ0007,
3 HEADIM(6), HEADIT(6), HEADLN(6), HEADLS(6), HEADTA(6), HEADRF(6),
4 HEADTR(6), HEADTS(6), HEADZL(6), TAPNAM(20)
COMMON /SUBKCM/ SUBDEM, SUBRUC, SUBSUP, DIFF, DUMMY
COMMON / DRCMD / ALPHA(16,4,5), BETA(33,4,5), GAMMA(21), DELTA(3,3)
COMMON/MODUSE/ NBMOD, MBMOD, MODT, LOSTAP, MODL(10), MODKOM(100)
1 MODCLS(100)
COMMON / CKOM / KOM
COMMON / CIIJJ / II, JJ
COMMON / MINDEX / INDEX(6)
COMMON / CIAVCL / IAVCLJ
COMMON / CBKCOM / KOMNAM(100), VALKOM(100), ICLASS(100), CPV(100,5),
2 IDISTB(100), EXPONT(100), TCOEF(100), TCOEF(100)
COMMON / SCDATA / POP(300), TINC(300), SOCIO(300)
COMMON / NODES1 / NODDEM(40), SPACE, NODSUP(20)
DIMENSION IAVCLJ(900,5,4)
DIMENSION ADDKOM(1500)
DIMENSION SUBDEM(40), SLBRUC(20)
DIMENSION SUBSUP(20), DIFF(40), DUMMY(180)
DIMENSION TFLCW(20,40,5), P(20,40)
DIMENSION C(20,40,6)
EQUIVALENCE (SUBDEM(1), ADDKOM(1))
EQUIVALENCE (IAVCLJ(1,1,3), C(1,1,1))
EQUIVALENCE (IAVOLJ(801,1,3),C(1,1,2))
EQUIVALENCE (IAVOLJ(701,2,3),C(1,1,3))
EQUIVALENCE (IAVOLJ(601,3,3),C(1,1,4))
EQUIVALENCE (IAVOLJ(501,4,3),C(1,1,5))
EQUIVALENCE (IAVOLJ(401,5,3),C(1,1,6))
EQUIVALENCE (IAVOLJ(301,1,4),TFLOW(1,1,1))
EQUIVALENCE (IAVOLJ(201,2,4),TFLOW(1,1,2))
EQUIVALENCE (IAVOLJ(101,3,4),TFLOW(1,1,3))
EQUIVALENCE (IAVOLJ( 1,4,4),TFLOW(1,1,4))
EQUIVALENCE (IAVOLJ(801,4,4),TFLOW(1,1,5))
EQUIVALENCE (ADDKOM(301),P(1))
IKOM=MODCLS(KCM)
IF(IKOM.GT.5) IKOM=1

IF WE USE THE FORMULA (12) THEN GO TO 1, OTHERWISE WE USE FORMULA (13) AND WE DO THE FOLLOWING CALCULATIONS......

IF,IDISTB(KOM),NE.7) GO TO 1
DO 2 I=1,II
DO 3 J=1,JJ
DO 4 K=1,MBMOD
CST=0.0
DO 5 L=1,MBMOD
CST=CST+TFLOW(I,J,L)
5 CONTINUE
TFLOW(I,J,K)=TFLOW(I,J,K)/CST
4 CONTINUE
3 CONTINUE
2 CONTINUE

CALCULATES THE FLOWS...
1 CTK1=0.0
CTK2=0.0
77 FORMAT(1H,1CF10.1)
78 FORMAT(1H,25F5.1)
CC 8 K=1, J1
CTL=0.0
CTL1=0.0
CTL2=0.0
DC 9 L=1, JJ
P(K,L)=G.C.C
CC 11 M=1, MBMOD
P(K,L)=P(K,L) + TFLOW(K,L,M)
11 CONTINUE
DC 10 KMCD=1, MBMOD
IDEM=NODEM(L)
IF((IDISTB(KOM), GE, 6), AND, (IDISTB(KOM), GE, 7)) GO TO 30
B=ALPHA(3, KMCD, IKOM)
A14 = ALPHA(14, KMCD, IKOM)
A15 = ALPHA(15, KMCD, IKOM)
A16 = ALPHA(16, KMCD, IKOM)
IDEM=NODEM(L)
ZL=(SUBDEM(L)**B)*(POP(IDC, M)**A14)*(TINCM(IDC, M)**A15)*(SOCIO(IDC, M)
1 **A16 )
GO TO 32
30 IF(IDISTB(KOM), GE, 8) CC TO 31
B=BETA(3, KMOD, IKCM)
A31=BETA(31, KMOD, IKCM)
A32=BETA(32, KMOD, IKCM)
A33=BETA(33, KMOD, IKCM)
IDEM=NODEM(L)
ZL=(SUBDEM(L)**B)*(PCP(IDC, M)**A31)*(TINCM(IDC, M)**A32)*(SOCIO(IDC, M)
1 **A33)
GO TO 32
31 ZL=SUBDEM(L)*POP(IDC, M)*TINCM(IDC, M)*SOCIO(IDC, M)
32 ZL=ZL*(P(K,L)**DELTA(1,1))
ZL1=ZL*(P(K,L)**DELTA(1,2))
ZL2=ZL*(P(K,L)**DELTA(1,3))
TFLOW(K,L,KMOD)=(TFLOW(K,L,KMOD)/P(K,L)) * ZL
10 CONTINUE
CTL=CTL+ZL
9 CONTINUE
DO 12 L=1,JJ
DO 13 KMCD=1,MBMCD
ISUP=NODSUP(K)
IF((IDISTB(KMB),NE.6) .AND. (IDISTB(KCOM),NE.7)) GO TO 40
A= ALPHA(2,KMOD,IKOM)
A11=ALPHA(11,KMOD,IKCM)
A12=ALPHA(12,KMOD,IKCM)
A13=ALPHA(13,KMOD,IKCM)
ISUP=NODSUP(K)
ZK=(SUBSUP(K)**A)*(POP(ISUP)**A11)*(TINCM(ISUP)**A12)*(SOCIO(ISUP))
GO TO 42
40 IF(IDISTB(KOM),NE.8) GC TO 41
A=BETA(2,KMOD,IKOM)
A28=BETA(28,KMOD,IKOM)
A29=BETA(29,KMOD,IKOM)
A30=BETA(30,KMOD,IKOM)
ISUP=NODSUP(K)
ZK=(SUBSUP(K)**A)*(POP(ISUP)**A28)*(TINCM(ISUP)**A29)*(SOCIO(ISUP))
GO TO 42
41 ZK=SUBSUP(K)*POP(ISUP)*TINCM(ISUP)*(SOCIO(ISUP))
42 CONTINUE
TFLOW(K,L,KMCD)=(TFLOW(K,L,KMCD)/CTL)*ZK*((CTL1)**DELTA(2,1))
13 CONTINUE
12 CONTINUE
CTK1=CTK1+(ZK*((CTL1)**DELTA(2,1)))
CTK2=CTK2+(ZK*((CTL2)**DELTA(2,2)))
8 CONTINUE
CTK2=CTK2*DELTA(3,1)
CTE=CTK2/CTK1
DO 15 K=1,II
DO 16 L=1,JJ
DC 17 M=1,MBMCD
TFLOW(K,L,M)=TFLOW(K,L,M)*CTE
17 CONTINUE
16 CONTINUE
15 CONTINUE
RETURN
END

SUBROUTINE NWTREE (NHCMN,RFAC,CFAC,RCUM,NCUM,NN,RSUM,WTSUM,
1 TTSUM,WTSUM,PLSUMP,CSUM,MODL)

MODIFIED BY PAUL ROBILLARD JULY 1973 IN ORDER TO BUILD TREES

COMMON /MAINC2/ NNODES,NLINKS
COMMON /CTAPES/ NTAPE,NCUT,RFCTAP,TRETAP,IMACRC,ITRANS,NINM,
1 NOTHER,NCUTM,QO0001,ITRNEW,QO0002,LNTAXE,LSOTAP,QO0003,
2 TRNSUM,QO0004,LTNSOR,QO0005,TAUTAP,QO0006,TABNEW,QO0007,
3 HEADIM(6),HEADIT(6),HEADLN(6),HEADLS(6),HEADTA(6),HEADRF(6),
4 HEADTR(6),HEADTS(6),HEADZZ(6),TAPNAM(20)
COMMON /DIMEN/ IMX,IPX,ISX,ITX,JDX,JSX,KLI,KLX,LMX,MMX,MNX,
1 NCX,NDX,NNX,NPX,NSX,NT2,NTX,NVX,NXX,NZX
COMMON /CINODE/ INODES
COMMON /CJNODE/ JNODES
COMMON /CMODE/ MODE
COMMON /CJD/ JD (900,3)
DIMENSION MODE (900)
DIMENSION RFAC (900)
DIMENSION CFAC (900,5)
DIMENSION RCUM (900)
DIMENSION NCUM (900)
DIMENSION NN (300)
DIMENSION RSUM (300)
DIMENSION WTSUM (300)
DIMENSION TTSUM (300)
DIMENSION VTTSUM (300)
DIMENSION PLSSUM (300)
DIMENSION CSUM(300)
DIMENSION INODES(900)
DIMENSION JNODES(900)

C
C INITIALIZE TABLES
DATA FMAX/33420214271. /

C
DO 1 I=1,NXX
NCUM(I)=0
1 RCUM(I) = FMAX
DO 2 I=1,NXX
2 RSUM(I) = FMAX
NTREE=1
NN(NHOME)=0

C C INITIALYSE THE SUMS FOR R,WT,TT,VTT,PLS,C TC ZERO

C
RSUM(NHOME)=0
WTSUM(NHOME)=0
TTSUM(NHOME)=0
VTTSUM(NHOME)=0
PLSSUM(NHOME)=0
CSUM(NHOME)=0
NCM=1
NCMI=1
NM=NHOME
ISET=1

C C BEGIN TREE COMPUTATIONS
C MAKE ENTRY INTO CUM TABLE FOR NODE NM
C FIRST, LOOK FOR NM

C
6 DO 7 IL=1,NLINS
I=IL
IF(ISET.EQ.1) GO TO 93
IMM=IMM+1
IF(IMM.EQ.4) GO TO 8
JNODD=JD(L,IMM)/1000
I=JD(L,IMM) - 1000*JNODD
IF(I.EQ.0) GO TO 8
93 IF(INODES(I)-NM) 7,3,7
FOUND NM
3 K=JNODD(I)

HAVE WE ALREADY GOTTEN HERE

IF(RSUM(K).NE. FMAX ) GO TO 7

IF 18, WE HAVEN'T GOTTEN HERE BEFORE SO MAKE ENTRY INTO CUM

18 RCUM(NCMI)=RSUM(NM)+RFAC(I)
IF(RFAC(I).LE.0) WRITE (NOUT,99) NM
99 FORMAT(1H 'RFAC=ZERO IN TREE COMPUTATION,NODE=',I3)
NCUM(NCMI)=I

INCREASE SIZE OF CUM TABLE FOR NEXT ENTRY

NCM=NCM+1
NCMI=NCM
7 CONTINUE
8 RMIN=FMAX

SEARCH CUM TABLE FOR LINKS WITH THE PROPER VALUE OF MODL. SET THE
VALUE OF MATCH ACCORDINGLY

MATCH=2
DO 100 K=1,NCM
I = NCUM(K)
IF(I.EQ.0) GO TO 100
IF(( MODE(I).NE.MODL).OR.(MODE(I).LE.18)) MATCH=1
100 CONTINUE
IF SMALLEST RCUM OVER ALL MODE IS WANTED, SET MATCH=3

IF(MODL.GT.100) MATCH=3

SEARCH CUM TABLE FOR SMALLEST RCUM

DO 9 K=1,NCM
   I=NCUM(K)
   IF(I.EQ.0) GO TO 9
   GO TO (101,102,102),MATCH
101 IF((MODE(I).NE.MDL).AND.(MODE(I).GT.18)) GO TO 9
102 IF(RMIN-RCUM(K)) 9,9,10

SAVE RMIN AND ITS INDEX IN THE TABLE

RMIN=RCUM(K)

L=LINE NUMBER IN THE LINK TABLE
L=NCUM(K)
NCMI=K
9 CONTINUE
   IF(RMIN.EQ.FMAX) GO TO 60

NOW REMOVE MIN RCUM FROM TABLE
RCUM(NCMI)=FMAX
REMOVE LINE NUMBER CORRESPONDING TO RCUM(NCMI)

NCUM(NCMI)=0
K=NODE NUMBER OF NEW NODE
K=JNODLES(L)
SEE IF RMIN IS LESS THAN PRESENT VALUE
IF(RSUM(K)-RMIN) 8,8,13
IT IS SO PUT IT IN TREE TABLE AND ADD 1 TONNTREE
MAKwSURE THAT IF WE HAVE GOTTEN AT K BEFORE OVER MODL THAT WE
STAY ON MODL.

13 IF(RSUM(K) LT FMAX) GO TO 8
RSUM(K)=RMIN

FIRST, COMPUTE THE WTSM, TTSUM, VTTSUM, PLSSUM, CSUM TO GET HERE

KL=INODES(L)
WTSM(K)=CFAC(L,1) + WTSM(KL)
TTSUM(K)=CFAC(L,2) + TTSUM(KL)
VTTSUM(K)=CFAC(L,3) + VTTSUM(KL)
PLSSUM(K)=CFAC(L,4) + PLSSUM(KL)
CSUM(K)=CFAC(L,5) + CSUM(KL)
NN(K)=L
NTREE=NTREE+1

IF WE HAVE A MIN PATH TO ALL NODES RETURN TO CALLLING PROGRAM

IF(NTREE-NNODES) 17, 50, 17
17 NM=K
ISET=1
IF(JD(L,3) LT O) ISET=1
IMM=0
GO TO 6
50 CONTINUE
RETURN
60 WRITE(NOUT,51) NHOME
51 FORMAT(1H 'BUILDING TREE FOR', I3, ' AND AT LEAST ONE NODE IS MISSI
1NG. RCUM EMPTY BEFORE NTREE EQ NNODES')
WRITE(NOUT,52) NTREE, NNODES
52 FORMAT(1X,215)
WRITE(NOUT,53) (RCUM(I), NCUM(I), I=1, NCM), (RSUM(I), NN(I), I=1, NNODES 1)
53 FORMAT(1X,12(F6,2,I4,F6,2,I4))
CALL EXIT
END
SUBROUTINE NAD0ON(IS, MMOD, FLOW, NN, ADDKOM)

C C
COMMON /TDIMEN/ IMX, IPX, ISX, ITX, JDX, JSX, KLI, KLX, KMX, LNX, MMX, MNX,
1 NCX, NDX, NNX, NPX, NT2, NTX, NXX, NNX, NzX
COMMON /MAINC2/ NNODES, NLINKS
COMMON /QTAPES/ NTAPE, NOUT, RFCTAP, TRETAP, IMACRO, ITRANS, NINM,
1 NOTHER, NOTUT, NQ00001, ITRENW, Q00002, LNTAPE, LNEW, LSĐTAP, Q00003,
2 TRNSUM, Q00004, LNSTOR, Q00005, TABTAP, Q00006, TABNEW, Q00007,
3 HEADIM(6), HEADIT(6), HEADLN(6), HEADLS(6), HEADTA(6), HEADRF(6),
4 HEADTR(6), HEADTS(6), HEADZZ(6), TAPNAM(20)
COMMON /CBKKOM/ KOMNAM(100), VALKOM(100), ICLASS(100), CPV(100,5),
1 IDISTB(100), EXPONT(100), HCOEF(100), TCOEF(100)
2 COMMON /CDISTN/ DISTNC %450<
COMMON /CINODE/ INODES
COMMON /CJO/ JD
COMMON /CIAVOL/ TAVOLJ
COMMON /CKOM/ KCM
COMMON /NODDEM/ NODDEM, SPACE, NCDSUP
COMMON /CIJJ/ JJ
COMMON /DAYSEZ/ DAYS
COMMON /MODUSE/ NBMOD, MBMOD, MCOT, LOSTAP, MODL(10), MODKOM(100),
1 MODCLS(100)
C
DIMENSION IAVOLJ %900, 5, 4<
DIMENSION NODDEM % 40<
DIMENSION NODSUP % 20<
DIMENSION DAYS % 4<
DIMENSION FLOW % 20, 40<
DIMENSION ADDKOM %1500)
DIMENSION ADDKOM %1500)
DIMENSION INODES %900<
DIMENSION NN %300<
DIMENSION AVOL %900, 5<
1ADD0060
1ADD0070
1ADD0080
1ADD0100
1ADD0120
1ADD0140
1ADD0150
1ADD0160
1ADD0170
1ADD0180
1ADD0190
1ADD0200
1ADD0210
1ADD0220
1ADD0230
1ADD0240
1ADD0250
1ADD0260
1ADD0270
1ADD0280
1ADD0300
PAGE 176
DIMENSION JD  %900, 3<
DIMENSION VOLJ  %900, 5, 3<

C
DIMENSION RSUM(300)
EQUIVALENCE %IAVOLJ,1,1<, VCLJ%1<<
EQUIVALENCE %IAVOLJ,1,4<, AVCL%1<<

C
INTEGER AVOL,JVOLJ
INTEGER TRETAP

C
THE FLOWS FROM EACH SUPPLY POINT TO EACH DEMAND POINT MUST BE
PLACED ON THE NETWORK

C
FIRST SET UP CERTAIN CONSTANTS THAT WILL BE NEEDED

C
TONDAY # VALKOM%KOM<*DAYS%IS<
ICLAS # ICLASS%KOM<

19 DO 20 L#1, NLINKS
ADDKOM%L< # 0.0
20 CONTINUE

C
MAKE A LOOP OVER EACH MODE TO ASSIGN FLOWS.
DO 40 KMOD=1,MBMOD

C
READ THE FLOWS FOR THAT MODE....

C
READ(LOSTOP) KK,((FLOW(I,J),I=1,II),J=1,JJ)

C
CHECK IF KK=K=KMOD

C
K=KMOD
IF(K.EQ.KK) GO TO 41
WRITE(NOUT,42) K,KK

PAGE 177
I. 'IS ON FLOW FILE')

CALL EXIT

CONTINUE

DO 22 I#1, II

IT IS NECESSARY TO PICK UP THE MINIMUM PATH TREE FOR THIS SUPPLY POINT. THIS IS THE SAME TREE THAT WAS USED TO DETERMINE THE MINIMUM COST ROUTE FOR THE DISTRIBUTION MODEL

FOR THIS PARTICULAR MODE LET US READ THE TRETAP

READ(TRETAP) NHOME,KM,MBM,MOD,(NN(L),L=1,NNX),

1 (RSUM(L),RSUM(L),RSUM(L),RSUM(L),RSUM(L),RSUM(L),L=1,NNX)

CHECK TO SEE THAT THE MODES ARE THE SAME

K=KMOD

IF((MBM.EQ.MBMOD).AND.(K.EQ.KM)) GO TO 43

WRITE(NOUT,44) K,KM,MBM,MBMOD

44 FORMAT(1HO,'ERRCR IN ADDON FOR THE MODES CONSIDERED',15)

CALL EXIT

CONTINUE

CHECK THE ORIGIN NODE TO MAKE SURE THAT THE CORRECT TREE WAS READ

NCRIG # NOCSUP%I<

IF %NHOMED.EQ.NCRIG< GO TO 25

WRITE %NOUT, 1023< NHOME,I

1023 FORMAT(19H2NHOMED ON TRETAP IS,15,12H FOR NOSUP%,12,1H<

CALL EXIT

DO 26 J#1, JJ

JDEST # NODDE%J<

CONVERT FLOW BACK IN TO TONS PER DAY FROM CARS PER SEASON
FLOW%I,J< # FLOW%I,J< / TONDAY
TMILES# 0.0
IF %NORIG.EQ. JDEST< GO TO 27

C SET DESTINATION NODE IN BACK NODE
NDS # JDEST
C SET M#LINK BY WHICH FLOW GETS TO BACK NODE
1 M#NN%NDS<
C ASSIGN FLOWS TO THAT LINK
IVOL# FLOW%I,J<100.0 & 0.5
AVOL%M, ICLAS< # IVOL & AVOL%M, ICLAS<
IVOL# FLOW%I,J< & 0.5

C THE DISTNC TABLE GIVES THE LENGTH OF EACH LINK PAIR IN MILES %KM<
C MILE # %M&I/2
TMILES # TMILES & DISTNC%MILE<
C SET UP THE ADDKOM VECTOR WHICH WILL GIVE THE NUMBER OF TONS OF
C THIS COMMODITY ADDED TO THE NETWORK IN THIS SEASON
C ADDKOM%M< # ADDKOM%M< & FLOW%I,J<
C SET K#LINK BY WHICH FLOW GETS TO AHEAD NODE
NDX1 # INODES%M<
K#NN%NDX1<
C ARE WE FINISHED
IF %K.LE.0< GO TO 6
5 D02M#1,3
C IS NODE IN TURN TABLE EQUAL TO THE BACK NODE
C IF %IABS%J%K,M< .NE. NDS< GO TO 2
JJJ=JD(K,M)/1000
IF(IABS(JJJ).NE.NDS) GO TO 2
C ASSIGN FLOW TO TURN VOLUME
3 VOLJ%K, ICLAS, M< # IVOL & VOLJ%K, ICLAS, M<
GO TO 4
C SET PRESENT NODE INTO BACK NODE
4 NDX1 GO TO 1
C CHECK TO SEE IF BACK NODE IS HCME NODE
6 IF %NDX1 .NE. Norig< GO TO 8
C
C CONVERT THE FLOW INTO DOLLARS PER SEASON
27 IF %TMILES .EQ. 0.0< TMILES # 1.0
   FLOW%I,J< # FLOW%I,J< # TMILES
C
26 CONTINUE
22 CONTINUE
40 CONTINUE
7 RETURN
C
C THERE IS A PROBLEM WITH THE TREE
C
8 WRITE %NOUT, 1008< NDX1,Norig,NHcME,%INODES%L<,L#1,900<,
   %NN%I,I#1,300<,%JD%I,J<,I#1,900<,J#1,3<
1008 FORMAT %19HLNDXI, Norig, NHcME,3I10/8HCINODES#/30%30I4/</4H0NN#/12
GO TO 24
C
C END
SUBROUTINE LNUPDT %IS<
C
C TRANS%S< -1LNUP %LNUPDT<
GECO REVISION 30 - CONVERT FLOWS IN AVOL BACK TO TONS / DAY
GECO REVISION 29 - DO NOT TRUNCATE PAYLOAD FOR TURN CALCULATION
ENELCO REVISION 4 - INCREASE MAX.NC.NODES TO 150

C MOST RECENT REVISION -- 25 NOV 1966 HSL
C
COMMON /CDATED/ DATED % 3<
COMMON /QTAPES/ NTAPE,NOUT,RFCTAP,TRETAP,IMACRC,ITRANS,NINM,
1 NOTHER,NOUTM,Q00001,ITRNEW,Q00002,LNTAPE,LNEW,LSDTAP,Q00003,
2 TRNSUM,Q00004,LISTOR,Q00005,TARTAP,Q00006,TABNEW,Q00007,
3 HEADIM(6),HEADIT(6),HEADLN(6),HEADLS(6),HEACTA(6),HEADRF(6),
4 HEADTR(6),HEADTS(6),HEADZZ(6),TAPNAM(20)
COMMON /MAINC3/ NCLASS,NCPARM,NOVOLS,NITEMS
COMMON /DAYSEZ/ DAYS
COMMON/CBKMOD/ MOCDAM%10 <,MODER%10 <,TVEHCL%15 <,
1 VEHCLS%10,5<,TPAYLO%15 <,PAYLOC%10,5<,TWMAXE%15 <,
2 WMAXE%10,5<,THRPDA%15 <,HRPDAY%10 <,TFIXED%15 <,
3 FIXDEC%10 <,RATE%10,5<,TDEPRA%15 <,
4 DEPRAT%10,5<,TDESRA%15 <,DESRA%10,5<,TVALUE%15 <,
5 VALVEH%10,5<,TPRCNT%15 <,PRCNT%10,10<,TFIXCA%15 <,
6 FIXCAP%10,10<,TFIXIN%15 <,FIXINV%10,10<,TDEPFX%15 <,
7 DEPRFX%10 <,TWAGRA%15 <,WAGRA%10,5<
COMMON /NAMNOO/ BLANK, NCDAM
COMMON /CIAVOL/ IA VOLJ
COMMON /CJD/ JD
COMMON /CINCR/ NINCRE,NRINCR
DIMENSION KATEC(3)
DIMENSION ITAM(5)
DIMENSION AVOL%900,5<
DIMENSION DAYS%4<
DIMENSION IA VOLJ%900,5,4<
DIMENSION JD%900,3<
DIMENSION NCDAM%300<
DIMENSION T L V %15<
DIMENSION TLPV%5,5,4<
DIMENSION TLUV%5,5,4<
DIMENSION VOLJ%900,5,3<
DIMENSION VOLTON%5,5,10<
EQUIVALENCE %IAVOLJ%1,1,4<,VOLJ%1<
EQUIVALENCE %IAVOLJ%1,1,4<,AVOL%1<
LOGICAL WRITO
INTEGER AVOL,VOLJ
C
C SET UP THE LINK NETWORK FILE TAPE
C
REWIND LNTAPE
REWIND LNEM
REWIND LNSTOR
C
ISN # IS
LNKEND # 0
LNKHL {# 0
LNKBEG # -9
C
CALL CHECK %4HLUPD,LNTAPE,HEADLN<
WRITE %LNEW< HEADLN, DATED
READ %LNTAPE< NNODES,NLINKS,NOPARM,NOVCLS,NCLASS,NITEMS,NSEZNS
WRITE %LNEW< NNODES,NLINKS,NOPARM,NOVCLS,NCLASS,NITEMS,NSEZNS
C
C THE VOLUMES AND TURNS OF EACH LINK FOR ALL FIVE CLASSES CAN BE
C WRITTEN IN TABLE FORMAT
C
CALL OUTPUT %3,10,WRITC<
IF %WRITO< WRITE %NOUT, 1011<
1011 FORMAT %LHI927X99HLINK FLCWt2CXt6HTURN
1923Xt6HTURN
4tllX#29HTURN
13 %-<#MORE THAN 3 TURNS/15H IN NODE C UT,4%29H NOD BULK GENL SP
12EC COMC PRIV\\\\
C
DO 100 L # 1, NLINKS
READ %LNTAPE< INODE, JNODE, MODE, %TLCV%1<, I#1,NOPARM<,
$ %TTLUV%1,J,IS<, I#1,NOVCLS<, J#1,NCLASS<,
$ %TLPV%1,J,IS<, I#1,NITEMS<, J#1,NCLASS<, IS#1,NSEZNS<
MODA#MODE
IF %MODA/10<.EQ.1< MODA#TLCV%1<
CALL WHATMODA#MODE,MODC<
C DO CCMPUTATIONS ONLY FOR SEASON ISN
DC 15 J # 1, NCLASS
NRN = NINCR - NINCR
TLUVVV = 0.0
IF(NRN.GT.0) TLUVVV = TLUVV(1, J, ISN)
TLUVV(1, J, ISN) = AVCL(L, J)/100 + TLUVVV
C TLUV%1, J, ISN < AVCL%L, J < / 100
ITAM(J) = TLUVV(1, J, ISN) + 0.5
TLUV%2, J, ISN < # TLUV%1, J, ISN < / PAYLOC%MOD, J <
ZPAY# PAYLOC%MOD, J <
DO 10 I # 3, NOVCLS
QVOL# VOLJ%L, J, I-2
IF(NRN.GT.0) TLUVVV = TLUVV(I, J, ISN)
TLUVV(I, J, ISN) = QVCL/ZPAY + TLUVVV
C TLUV%I, J, ISN < # QVCL/ZPAY
10 CONTINUE
15 CONTINUE
WRITE % LNEW < INODE, JNODE, MODE, % TLCV%I, I#1, NOPARM,<,
$ % TLUV%I, J, IS<, I#1, NOVCLS<, J#1, NCLASS<,
$ % LPV%I, J, IS<, I#1, NITEMS<, J#1, NCLASS<, IS#1, NSEZNS<
C DO 137 M=1,3
KATED(M) = JD(L, M)/1000
137 CONTINUE
IF % WRITO< WRITE % NCU<, 1012 < NODNAM%INODE<, INODE, NODNAM%JNODE<,
1 JNODE, % ITAM(J), J#1, NCLASS<,
2 (KATED(M), % VOLJ%L, J, M<, J#1, NCLASS<, M#1, 3<,
% JDL%L, M<, % VCLJ%L, J, M<, J#1, NCLASS<, M#1, 3<
1012 FORMAT %A6,13, A6, 4%I4, 5I5<
C
IF % ISN.EQ, NSEZNS< GO TO 100
C SET UP VOLTON WITH THE LPV SO THAT IT MAY BE USED BY RCOMP IN THE
C R-FACTOR COMPUTATION
C
C VOLTON WILL BE SAVED IN BLOCKS OF 10 LINKS EACH TO BE WRITTEN ON
C LNSTOR IN 252 WORD BINARY RECORDS. THE FIRST TWO WORDS ARE USED
C TO LOCATE THE TEN LINKS WITHIN THE FULL LINK FILE
C
SUBROUTINE ORCDM0 (NN)

C THIS PROGRAM WILL CALCULATE TREES FOR EACH MODE AND ALSO CAL-
C CULATE THE L.O.S. MATRIX OF VARIABLES FOR EACH MODE.

C PAUL ROBILLARD JULY 1973
DIMENSION DUMMY(180), ADDKCM(1500)
DIMENSION INODES(900)
DIMENSION JNODES(900)
EQUIVALENCE (SUBDAM(1), ADDKCM(1))
EQUIVALENCE (IAVOLJ(1,1,3), CFAC(1))
EQUIVALENCE (IAVOLJ(1,1,4), PFAC(1))
EQUIVALENCE (IAVOLJ(1,2,4), RCUM(1))
EQUIVALENCE (IAVOLJ(1,3,4), NCUM(1))
EQUIVALENCE (IAVOLJ(1,4,4), CUMR(1))
EQUIVALENCE (IAVOLJ(1,1,3), CUMWT(1))
EQUIVALENCE (IAVOLJ(1,2,3), CUMTT(1))
EQUIVALENCE (IAVOLJ(1,3,3), CUMVTT(1))
EQUIVALENCE (IAVOLJ(1,4,3), CUMPLS(1))
EQUIVALENCE (IAVOLJ(1,5,3), CUMC(1))
EQUIVALENCE (ADDCOM(301), RSUM(1))
EQUIVALENCE (ADDCOM(601), WTSUM(1))
EQUIVALENCE (ADDSOM(901), TTSUM(1))
EQUIVALENCE (ADDKOM(1201), VTTSUM(1))
EQUIVALENCE (FLOW(1,1), PLSSUM(1))
EQUIVALENCE (FLOW(1,01), CSUM(1))

INTEGER TRETAP
LOGICAL WRITO

C IS SUBCOMMODITY KOM ASSIGNED TO A SPECIFIC MODE IN MODKCM(KOM)

DATA SUM /4HSUMS/
MCDT=MODKCM(KOM)
NBMOD=0
DO 3 K=1,10
IF((MODL(K).EQ.MCDT).AND.(MODL(K).NE.0)) NBMOD=1
3 CONTINUE

C IS MODKOM(KOM).GT.100, IF YES THEN SUBCCOMMODITY KOM IS GOING TO
BE DISTRIBUTED WITH LOS VARIABLES OBTAINED OVER THE MINIMUM R-
PATH OVER ALL MODES

IF(MODT.GT.100) NBMOD=1
IF(MODT.GT.100) MODT=120

C IF MODKOM(KOM)=0, THEN THE SUBCOMMODITY WILL BE SHIPPED OVER ALL
C AVAILABLE MODES SPECIFIED IN MODL(10).
C
C THE NON-ZERO MODES IN MODL(.) ARE STORED IN MODL(1), MODL(2), ...
C
C DETERMINE MODES AVAILABLE IN MODL(.).
C
MBMOD=0
DO 1 K=1,10
IF(MODL(K).NE.0) MMBMOD=MBMOD+1
1 CONTINUE
IF(NBMODE.EQ.1) MMBMOD=1

C THE FOLLOWING LOOPS BUILD UP THE TREES FOR EACH MODE AND FOR
C EACH SUPPLY NODE AND STORE THEM ON TREETAP WITH THE SIX COMPONENTS
C OF THE L.O.S. SUMS,RSUM,WTSUM,TTSUM,VTTSUM,PLSSUM,CSUM.
C
DO 4 KMCD=1,MBMOD
MODLL=MODL(KMCD)
IF(NBMODE.EQ.1) MODLL=MODT
DO 5 I=1,II
CALL NWTREE(NODCSUP(I),RFAC,CFAC,RCUM,NCUM,NN,RSUM,WTSUM,TTSUM,
1 VTTSUM,PLSSUM,CSUM,MODLL)
4 CONTINUE

C PRINT THE TREES IF WANTED...
C
CALL OUTPUT(10,5,WRITO)
IF(.NOT.WRITO) GO TO 30
NHOMENODCSUP(I)
WRITE(NOUT,14) MODLL,NHOME
14 FORMAT(1H1,10X,'MINIMUM R-PATH ON MODE',I5,3X,'WITH HOME NODE=',
15 115)
WRITE(NOUT,12)
12 FORMAT(1H0,4X,'NODE',4X,'LINK',4X,'RSUM',4X,'WTSUM',3X,'TTSUM',)
DO 15 L=1,300
  IF(NN(L).LT.0) GO TO 15
  KK=NN(L)
  IF(KK.EQ.0) KK=300
  I9=INODES(KK)
  WRITE(NOUT,16) L,NN(L),RSUM(L),WTSSUM(L),TTSSUM(L),VTTSUM(L),
  1 PLSSUM(L),CSUM(L),I9
16  FORMAT(1H16,2(I8),96(F8.1),I8)
15  CONTINUE
30  CONTINUE

WRITE THE TREE FOR THIS NODE CN TRETAP AS WELL AS ALL THE
L.O.S. VARIABLES SUMS.

WRITE(TRETAP) NODSUP(I),KMOD,MBMOD,MODL,(NN(L),L=1,NNX),
  1 (RSUM(L),WTSSUM(L),TTSSUM(L),VTTSUM(L),PLSSUM(L),
  2 CSUM(L),L=1,NAX)
5  CONTINUE
4  CONTINUE
REWRITE TRETAP

FOR EACH MODE AND EACH SUPPLY NODE REREAD TRETAP AND CALCULATE THE
MATRIX OF L.O.S. VARIABLES ,CUMR,CUMWT,CUMTT,CUMVTT,CUMPLS,CUMC,

REWIND LOSTAP
DC 6 KMOD=1,MBMOD
DC 7 I=1,II
READ(TRETAP) NODS,KM,MBM,MG,(NN(L),L=1,NNX), (RSUM(L),WTSSUM(L),
  1 TTSSUM(L),VTTSUM(L),PLSSUM(L),CSUM(L),L=1,NAX)
DO 8 J=1,JJ
  IY=NODDEM(J)
  CUMR(I,J)=RSUM(IY)
  CUMWT(I,J)=WTSSUM(IY)
8  CONTINUE
CUMTT(I,J) = TTSUM(IY)
CUMVTT(I,J) = VTSUM(IY)
CUMPLS(I,J) = PLSSUM(IY)
CUMC(I,J) = CSUM(IY) + BASCRG(KOM)

8 CONTINUE
7 CONTINUE

FOR THIS MODE STORE THE L.G.S. MATRIX OF VARIABLES IN ORDER TO

USE THEM FOR THE DISTRIBUTION LATER.

WRITE(LOSTAP) KMOD, MBKOM, KCM, II, JJ, ((CUMR(I,J), I=1,II), J=1,JJ),
1 ((CUMWT(I,J), I=1,II), J=1,JJ),
2 ((CUMTT(I,J), I=1,II), J=1,JJ),
3 ((CUMVTT(I,J), I=1,II), J=1,JJ),
4 ((CUMPLS(I,J), I=1,II), J=1,JJ),
5 ((CUMC(I,J), I=1,II), J=1,JJ)

THESE TABLES CAN BE PRINTED IF THEY ARE WANTED

CALL OUTPUT(2,8,WRITO)
IF(WRITO) CALL TABLE(CUMR,BLANK,24HDECISION COSTS PER TON ,SUM)
WRITE(NOUT,50) MODL(KMCD)
CALL OUTPUT(10,1,WRITO)
IF(WRITO) CALL TABLE(CUMWT,BLANK,24HWHT COSTS PER TON ,SUM)
WRITE(NOUT,50) MODL(KMCD)
CALL OUTPUT(10,2,WRITO)
IF(WRITO) CALL TABLE(CUMTT,BLANK,24HTT COSTS PER TON ,SUM)
WRITE(NOUT,50) MODL(KMCD)
CALL OUTPUT(10,3,WRITO)
IF(WRITO) CALL TABLE(CUMVTT,BLANK,24HVAR OF TT COSTS PER TON ,SUM)
WRITE(NOUT,50) MODL(KMCD)
CALL OUTPUT(10,4,WRITO)
IF(WRITO) CALL TABLE(CUMPLS,BLANK,24HPB OF LCSS COST PER TON ,SUM)
WRITE(NOUT,50) MODL(KMCD)
CALL OUTPUT(2,5,WRITO)
IF(WRITO) CALL TABLE(CUMC,BLANK,24HTRN CHARGE PER TON ,SUM)
WRITE(NOUT,50) MODL(KMCD)
PARAMETERS FOR THE DIRECT DEMAND MODELS. THE MODELS ARE THE
MCLYNN MODEL, THE SARC-KRAFT MODEL, THE BEAUMOL QUANDT MODEL.

IN ADDITION A WHOLE CLASS OF MODELS WHICH ARE PARTICULAR CASES
OF THE GENERAL SHARE MODEL ARE ALSO AVAILABLE... IN PARTICULAR
THE PARAMETERS FOR THE FAMILY OF MODELS DEFINED BY THE SPECIAL
PRODUCT MODEL -1- ARE DEFINED HERE................

IT IS ASSUMED THAT THE SAME SET OF PARAMETERS ARE VALID FOR ALL
SUBCOMMODITY. IT IS POSSIBLE TO MODIFY THIS SO THAT WE CAN
SPECIFY PARAMETERS FOR EACH SUBCOMMODITY.

IN FACT WE HAVE 5 COMMODITIES POSSIBLE OR 5 CLASS TO ASSIGN
COMMODITY TO DISTRIBUTION MODEL AS IMPLEMENTED NOW.

COMMON/MODUSE/ NMODE, NMODE, MCDT, LOSTAP, MCDL(10), MODKM(100),
MODCLS(100)
COMMON / MINDEX / INDEX(6)
COMMON / DIRMDM / ALPHA(16,4,5), BETA(33,4,5), GAMMA(21), DELTA(3,3)
COMMON / SCDATA / POP(300), TINCM(300), SCCIC(300)
COMMON / CINCR / NINCRE, NINCR

INDEX(.) CONTAINS THE INDEX 1, 2, 3, 4, 5, 6. ANY SUBSET OF THESE CAN
BE IN INDEX(1), INDEX(2), .... THESE INDEX SPECIFY WHICH ONE OF
THE 6 POSSIBLE L.C.S. VARIABLES ARE USED IN THE DIRECT DEMAND
MODEL, THAT IS WHICH ONE OF CUMR, CUMWT, CUMTT, CUMVT, CUMPLS, CUMC,
IN THAT ORDER 1, 2, 3, 4, 5, 6, ARE USED...
DATA INDEX /1,2,3,4,5,6/

LCSTAP IS THE OUTPUT FILE NUMBER FOR THE L.C.S. VARIABLES...

DATA LCSTAP /10/

MCdl(*) CONTAINS THE DIFFERENT MODES OVER WHICH TREES AND L.Q.S.
VARIABLES SHOULD BE CALCULATED FOR COMMODITIES FREE TO TRAVEL
OVER ALL MODES......

DATA MODL /50,40,30,70/ DATA NINCRE /1/

MCdkom(*) IS ZERO FOR COMMODITIES FREE TO TRAVEL OVER ALL MODES
THEN MODL(*) CONTAINS THESE MODES... IF A COMMODITY KOM SHOULD
TRAVEL OVER A SPECIFIC MODE THEN MCdkom(KOM)=THE MODE OVER WHICH
THE COMMODITY SHOULD TRAVEL.......

DATA MCdkom /100*0/

DATA ALPHA /4*1.0, 6*1.0, 6*1.0, 4*1.0, 6*1.0, 6*1.0, 6*1.0, 256*0.0/
1 4*1.0, 6*1.0, 6*1.0, 4*1.0, 6*1.0, 6*1.0, 6*1.0, 256*0.0/

DATA BETA /3*1.0, -1*0, 3*1.0, -1*0, 3*1.0, -1*0, 3*1.0, 6*1.0,
1 -1*0, 3*1.0, -1*0, 3*1.0, -1*0, 3*1.0, 6*1.0,
3 3*1.0, 1.0, -1.0, 2*1.0, 1.0, -1.0, 2*1.0, 1.0, -1.0, 2*1.0,
4 1*0, -1.0, 2*1.0, 1.0, -1.0, 2*1.0, 1.0, -1.0, 2*1.0, 6*1.0,
5 3*1.0, 2*1.0, -1.0, 1.0, 2*1.0, -1.0, 1.0, 2*1.0, -1.0, 1.0,
6 2*1.0, -1.0, 1.0, 2*1.0, -1.0, 1.0, 2*1.0, -1.0, 1.0, 6*1.0,
7 3*1.0, 3*1.0, -1.0, 3*1.0, -1.0, 3*1.0, 1.0,
8 3*1.0, -1.0, 3*1.0, -1.0, 3*1.0, 1.0, 6*1.0, 518*0.0 /

DATA GAMMA /21*0.0 /

DATA DELTA /9*1.0 /

DATA POP /300*1.0 /
DATA TINCM / 300*1.0 /
DATA SOCIO / 300*1.0 /

MCDCLS(*) CONTAINS THE CLASS OF DIRECT DEMAND PARAMETERS TO USE FOR THIS COMMODITY... BECAUSE OF THE THIRD DIMENSION IN ALPHA AND BETA AS THEY ARE NOW THE MAXIMUM VALUE OF MCDCLS(KOME IS 5......

DATA MCDCLS /100*1 /

TABLE ALPHA ARE PARAMETERS OF THE MCFLYNN MODEL...

TABLE BETA ARE PARAMETERS OF THE SARC-KRAFT MODEL...

TABLE GAMMA ARE THE PARAMETERS OF THE BAUMOL-QUANDT ABSTRACT MCDE MODEL.....

END

SUBROUTINE RCOMP

MODIFICATIONS TO RCOMP INORDER TO CARRY ALL 5 COMPONENTS OF THE R-FACTORS WITH THE R-FACTORS THEMSELVES.

CFAC IS NOW DIMENSIONED CFAC(900,5) ONE FOR EACH OF THE 5 COMPONENTS...

MODIFICATIONS DONE BY PAUL ROBILLARD AUGUST 73
KCOM=SUBCOMMODITY NUMBER
ICLASS = SUBCOMMODITY
COMMON/MAINC1/NSEQNS,NINDUS,MREGNS,NOCCLS
COMMON /MAINC2/ NNODES, NLINKS
COMMON /MAINC3/ NCLASS, NCPARM, NOVCLS, NITEMS
COMMON /QTAPES/ NTAPE ,NOUT ,RFCTAP,TRETAP,IMACRC,ITRANS,NINM
COMMON /CBKKOM/, KCMNAM(100), VALKCM(100), ICLASS(100), CPV(100,5),
COMMON /CINODE/, INODES (900)
COMMON /CNODE/, JNODES (900)
COMMON /CMODE/, MCODE (900)
COMMON /CVOLTO/, VOLTON
COMMON /CNAMOC/, BLANK, NODNAM(300)
COMMON /OUTCOM/, ISWITC(10,10), ICTR(10,10)
DIMENSION RFAC (900)
DIMENSION CFAC (900,5)
DIMENSION VOLTON (5,5,900)
DIMENSION FACTOR (900)
DIMENSION RFTITL (9)
DIMENSION TYPE (5)
LOGICAL WRITO
INTEGER RFTITL, TYPE
INTEGER RFCTAP
EQUIVALENCE (CFAC(1,5), FACTCR(1))

DATA RFTITL ',36H 'S DECISION FACTOR FOR /
DATA TYPE ',20HWAITTRVLVARLPRCBCHR/G/

CALL TRACER(4HRCMP,1)
REWIND RFCTAP
71 DO 72 KOM = 1, NOCOMS
ICLAS = ICLASS(KCM)
VALCOM = VALKCM(KCH)
61 DO 62 L=1, NLINKS
RFAC(L) = 0.0
62 CONTINUE
DO 74 I=1, NITEMS
CPVEE = CPV(KOM,I)
IF (I.EQ.4) GO TO 77

COMPUTE R-FACTORS AND KEEP EACH COMPONENT OF R IN CFAC....

DO 76 L=1, NLINKS
FACTOR(L) = VOLTON(I,ICLAS,L)*CPVEE
RFAC(L) = RFAC(L) + FACTOR(L)
CFAC(L,I) = FACTOR(L)
CONTINUE

75 DO 77 L=1, NLINKS
FACTOR(L) = VOLTON(I,ICLAS,L)*CPVEE*VALCUM
RFAC(L) = RFAC(L) + FACTOR(L)
CONTINUE

80 CALL OUTPUT (1,I+5,WRITC)

IF ISWITC(1,5) = 0, NO TABLES OR PLOTS WILL BE PRINTED
IF ISWITC(1,5) = 1, BOTH TABLES AND PLOTS WILL BE PRINTED
IF ISWITC(1,5) = 2, ONLY TABLES WILL BE PRINTED
IF ISWITC(1,5) = 3, ONLY PLOTS WILL BE PRINTED

IN ALL CASES ISWITC(1,6) TO (1,10) WILL CONTROL WHICH COMMODITIES
AND ITEMS ARE TO BE PRINTED

KONTRL = ISWITC(1,5)
IF (KONTRL.LE.0) GO TO 74
IF (KONTRL.GE.3) GO TO 55
WRITE A TABLE OF THE R-FACTORS FOR EACH LINK

IF (.NOT. WRITO) GO TO 55

WRITE (NOUT, 1051) TYPE(I),KCMNKAM(KCM)

1051 FORMAT (1H1,37X,17HDECISION COSTS OF ,A7,23H AS SEEN BY SHIPPERS OF )

,AT//3(7X,5HINODE,6X,5HJNODE,5X,4HMODE,4X,8HADDITION))

53 DO 54 L=1,NLINKS,3
  I1 = INODES(L)
  J1 = JNODES(L)
  M = L + 1
  IF (M.GT.NLINKS) GO TO 54
  I2 = INODES(M)
  J2 = JNODES(M)
  N = L + 2
  IF (N.GT.NLINKS) GO TO 54
  I3 = INODES(N)
  J3 = JNODES(N)

54 WRITE (NOUT, 1054) I1,NODNAM(I1),J1,NODNAM(J1),MODE(L),FACTOR(L),
  I2,NODNAM(I2),J2,NODNAM(J2),MODE(M),FACTOR(M),
  I3,NODNAM(I3),J3,NODNAM(J3),MODE(N),FACTOR(N)

1054 FORMAT (3(I9,A6,I5,A6,I5,F13.2))

55 IF (.NOT.(KONTRL.EQ.1 .OR. KONTRL.EQ.3)) GO TO 74

RFTITL(1) = KCMNKAM(KCM)
RFTITL(8) = TYPE(I)
IF (WRITO) CALL PLCTA (1.0,FACTOR,RFTITL)

74 CONTINUE

WRITE( RFCTAP ) KOM, (RFAC(L), L=1,NLINKS),((CFAC(L,I),L=1,NLINKS)
  ,I=1,NITEMS)

72 CONTINUE
REWIND RFCTAP
APPENDIX 2

LISTINGS OF DATA USED FOR EVALUATION
&START
NJOBS=1, OVERLY=F, RESTRT=F
&END
EJCT
BLRB

CGSIGMA TRANS PROGRAM

JOB NUMBER- TEST01 DATE 31-8-1970

ETUDE AU NIGER
CGSIGMA INC.
LAMARRE VALCIS INTERNATIONAL LIMITÉE
CALIBRATION OF TRANSPORT MODEL

DATA SETS 1-001 GENERAL
2-002 NCDE NAMES
3-003 PLPVS
4-002 HIWAYS CONSTANTS
5-009 COMMODITY DATA
6-001 MODE DATA
7-002 VEHICLE DATA
8-003 RATE TABLE
9-006 SUPPLY DEMAND EDIT DATA
10-002 LINK EDIT DATA

+END
ENDL
&NEWJOB
JOB=1, NJOBS=1, NYEAR=1, NYREND=1, TJOCDATA=T, TYDATA=T
&END
&TDATA
ENTRY=2,

NSEZNS= 2,
NOCOMS= 11,
NNODES= 44,
NLINKS=108
NCLASS# 5,
NCPARM# 14,
NOVOLS# 5,
NITEMS# 5,
DAYS= 212., 153., 0., 0.,
ICTR# 100*0,

CPV= 4*0.0, -3.80, 3*0.0, -3.80, 91*0.0,
0.0006, 0.0009, 0.0004, 0.0011, 0.0004, 0.0011, 0.017, 0.182,
0.0004, 0.0011, 0.017, 0.000, 4*0.0, -2.30, 3*0.0, -2.30, 91*0.0,
50.0, 80.0, 360.0, 100.0, 32.0, 100.0, 1500.0, 16000.0,
100*1.0,
KOMNAM= 4HNIEB, 4HARAC, 4HUILE, 4HEXPO, 4HFUEL,
END

END
0, 1, 1, C, 0, 12, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 1, 0, 12, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 1, C, 0, 12, 0, 0, 0, 0, 0, 0,
0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 1, 1, C, 0, 0, 0, 0, 0, 0, 0, 0, 0,

$END

1 1NIEB 1NIAM 60000 010199
1 1NIEB 2MAR A 13000 010199
1 1NIEB 3ZIND 32000 010199
1 1NIEB 4BIRN 35000 010199
2 1NIEB 43PORT 90000 010199
2 1NIEB 37LAGO 30000 010199
2 1NIEB 16COTO 50000 010199
2 1NIEB 38ABID 15000 010199
1 2ARAC 2MAR A 1. 019010
2 2ARAC 43PORT 1. 019010
1 3HUILL 1NIAM 23000 016040
1 3HUILL 2MAR A 35000 016040
1 3HUILL 3ZIND 27000 016040
2 3HUILL 43PORT 85000 016040
1 4EXPO 1NIAM 66700 016337
1 4EXPO 2MAR A 27150 016337
1 4EXPO 3ZIND 45850 016337
1 4EXPO 4BIRN 17500 016337
2 4EXPO 16COTO 21550 016337
2 4EXPO 37LAGO 67750 016337
2 4EXPO 38ABID 25000 016337
2 4EXPO 43PORT 42900 016337
1 5FUEL 43PORT 322000 016337
1 5FUEL 37LAGO 90000 016040
2 5FUEL 1NIAM 270000 016040
2 5FUEL 2MAR A 44000 016040

PAGE 199
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td>50</td>
<td>86.6</td>
<td>0397</td>
<td>1.7</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>50</td>
<td>86.6</td>
<td>0397</td>
<td>1.7</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>50239.8</td>
<td>0397</td>
<td></td>
<td>4.8</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>50239.8</td>
<td>0397</td>
<td></td>
<td>4.8</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>50164.5</td>
<td>0397</td>
<td></td>
<td>3.3</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>50164.5</td>
<td>0397</td>
<td></td>
<td>3.3</td>
</tr>
<tr>
<td>2</td>
<td>24</td>
<td>50</td>
<td>24.8</td>
<td>0397</td>
<td>5.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>50 24.8 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>2</td>
<td>50 35.4 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td>50 35.4 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>2</td>
<td>50 67 .0817</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>3</td>
<td>50 33.2 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>3</td>
<td>50 33.2 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>50 173 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>50 173 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>2</td>
<td>50 74.5 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>4</td>
<td>50 74.5 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>50 1623 687</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>50 1623 687</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>50 154 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>50 154 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>50 37.2 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>50 37.2 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>7</td>
<td>50 37.2 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>7</td>
<td>50 37.2 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>11</td>
<td>50 101 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>8</td>
<td>50 101 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>50 85 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>9</td>
<td>50 85 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>27</td>
<td>50 222 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>10</td>
<td>50 222 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>13</td>
<td>50 115 .0687</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>11</td>
<td>50 115 .0687</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>14</td>
<td>50 180 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>14</td>
<td>50 180 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>11</td>
<td>50 180 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>23</td>
<td>50 138 687</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>12</td>
<td>50 138 687</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>26</td>
<td>50 185 99999</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>14</td>
<td>50 185 99999</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>32</td>
<td>50 22.4 .0397</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```
<table>
<thead>
<tr>
<th>32</th>
<th>15</th>
<th>50</th>
<th>22.4</th>
<th>0.0397</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>26</td>
<td>50</td>
<td>70.8</td>
<td>0.0397</td>
</tr>
<tr>
<td>26</td>
<td>16</td>
<td>50</td>
<td>70.8</td>
<td>0.0397</td>
</tr>
<tr>
<td>17</td>
<td>20</td>
<td>50</td>
<td>205.6</td>
<td>0.0397</td>
</tr>
<tr>
<td>20</td>
<td>17</td>
<td>50</td>
<td>205.6</td>
<td>0.0397</td>
</tr>
<tr>
<td>18</td>
<td>19</td>
<td>50</td>
<td>127.2</td>
<td>1.652</td>
</tr>
<tr>
<td>19</td>
<td>18</td>
<td>50</td>
<td>127.2</td>
<td>1.652</td>
</tr>
<tr>
<td>19</td>
<td>22</td>
<td>50</td>
<td>45.6</td>
<td>1.652</td>
</tr>
<tr>
<td>22</td>
<td>19</td>
<td>50</td>
<td>45.6</td>
<td>1.652</td>
</tr>
<tr>
<td>20</td>
<td>21</td>
<td>50</td>
<td>31.6</td>
<td>0.0397</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>50</td>
<td>31.6</td>
<td>0.0397</td>
</tr>
<tr>
<td>23</td>
<td>24</td>
<td>50</td>
<td>90.0</td>
<td>0.0397</td>
</tr>
<tr>
<td>24</td>
<td>23</td>
<td>50</td>
<td>90.0</td>
<td>0.0397</td>
</tr>
<tr>
<td>27</td>
<td>28</td>
<td>50</td>
<td>41.6</td>
<td>0.0397</td>
</tr>
<tr>
<td>28</td>
<td>27</td>
<td>50</td>
<td>41.6</td>
<td>0.0397</td>
</tr>
<tr>
<td>28</td>
<td>29</td>
<td>50</td>
<td>45.4</td>
<td>0.0397</td>
</tr>
<tr>
<td>29</td>
<td>28</td>
<td>50</td>
<td>45.4</td>
<td>0.0397</td>
</tr>
<tr>
<td>29</td>
<td>30</td>
<td>50</td>
<td>50.9</td>
<td>0.0397</td>
</tr>
<tr>
<td>30</td>
<td>29</td>
<td>50</td>
<td>50.9</td>
<td>0.0397</td>
</tr>
<tr>
<td>30</td>
<td>31</td>
<td>50</td>
<td>95.0</td>
<td>0.0397</td>
</tr>
<tr>
<td>31</td>
<td>30</td>
<td>50</td>
<td>95.0</td>
<td>0.0397</td>
</tr>
<tr>
<td>31</td>
<td>32</td>
<td>50</td>
<td>52.7</td>
<td>0.0397</td>
</tr>
<tr>
<td>32</td>
<td>31</td>
<td>50</td>
<td>52.7</td>
<td>0.0397</td>
</tr>
<tr>
<td>33</td>
<td>38</td>
<td>30</td>
<td>715.0</td>
<td>0.021</td>
</tr>
<tr>
<td>38</td>
<td>33</td>
<td>30</td>
<td>715.0</td>
<td>0.021</td>
</tr>
<tr>
<td>34</td>
<td>37</td>
<td>30</td>
<td>690.0</td>
<td>0.0235</td>
</tr>
<tr>
<td>37</td>
<td>34</td>
<td>30</td>
<td>690.0</td>
<td>0.0235</td>
</tr>
<tr>
<td>35</td>
<td>36</td>
<td>30</td>
<td>272.0</td>
<td>0.042</td>
</tr>
<tr>
<td>36</td>
<td>35</td>
<td>30</td>
<td>272.0</td>
<td>0.042</td>
</tr>
<tr>
<td>39</td>
<td>40</td>
<td>40</td>
<td>183.0</td>
<td>999.9</td>
</tr>
<tr>
<td>40</td>
<td>39</td>
<td>40</td>
<td>183.0</td>
<td>999.9</td>
</tr>
<tr>
<td>40</td>
<td>41</td>
<td>40</td>
<td>129.0</td>
<td>0.0176</td>
</tr>
<tr>
<td>41</td>
<td>40</td>
<td>40</td>
<td>129.0</td>
<td>0.0176</td>
</tr>
<tr>
<td>41</td>
<td>42</td>
<td>40</td>
<td>728.0</td>
<td>0.0176</td>
</tr>
<tr>
<td>42</td>
<td>41</td>
<td>40</td>
<td>728.0</td>
<td>0.0176</td>
</tr>
</tbody>
</table>
| 9 | 33 | 10 | 30 | 4.8 | 4.8 | 4.8 | 4.8