# ABSTRACT

## Development of a Composite User Equilibrium and System Optimization Assignment Model

**by**
**Kimberly M. Stump**

The Urban Transportation Modeling System (UTMS) is a set of procedures used by transportation planners to predict the volume of traffic that will flow through a network, and how the traffic is routed. This paper will focus on the final component of UTMS, traffic assignment, which assigns the traffic flows to actual routes in the network.

In this paper, two composite models which model both User Equilibrium and System Optimal assignments are presented. The composite models are solved using the GAMS (General Algebraic Modeling System) software, a powerful mathematical programming tool. The first model was based on Beckman's Formulation, (Beckman, MacGuire, Winsten, 1956). The second model was developed by the author and utilizes some unique features of the GAMS software in order to solve the problem of User Equilibrium. Finally, the results of the example problem are used to develop general conclusions regarding the applicability of the model, as well as areas of future improvement and research.

# Development of A Composite User Equilibrium and System Optimization Assignment Model

by
Kimberly M. Stump

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science

Interdisciplinary Program in Transportation

January, 1993

Blank Page

## APPROVAL PAGE

**Development of a Composite User Equilibrium and System Optimization Assignment Model**

**Kimberly M. Stump**

1/8/93

**Dr. Lazar N. Spasovic, Thesis Advisor**
**Assistant Professor, School of Industrial Management,**
**Center for Transportation Studies and Research, NJIT**

1/8/93

**Dr. Louis J. Pignataro, Committee Member**
**Distinguished Professor, Transportation Engineering and**
**Director, Center for Transportation Studies and Research, NJIT**

1/8/93

**Dr. Athanassios K. Bladikas, Committee Member**
**Associate Professor, Industrial and Management Engineering and Associate**
**Director, Center for Transportation Studies and Research, NJIT**

## BIOGRAPHICAL SKETCH

**Author:** Kimberly M. Stump

**Degree:** Master of Science in Transportation

**Date:** January, 1993

**Date of Birth:**

**Place of Birth:**

**Undergraduate and Graduate Education:**

- Master of Science in Transportation, New Jersey Institute of Technology, Newark, NJ, 1993

- Bachelor of Science in Civil Engineering, University of Delaware, Newark, DE, 1987

**Major:** Transportation (Planning)

**Professional Positions:**

- Project Engineer, Delaware Department of Transportation, Dover, DE, 1988-1991

- Research Assistant, New Jersey Institute of Technology, Newark, NJ, 1991-1992

- Traffic Engineer, Garmen Associates, Montville, NJ, 1992-1993

## ACKNOWLEDGEMENT

The author wishes to express her sincere gratitude to her supervisor, Professor Lazar N. Spasovic, for his direction and friendship throughout this research.

The author also wishes to thank Professors Louis J. Pignataro and Athanassios K. Bladikas for serving as members of the committee.

Finally, the author wishes to thank her family for their words of encouragement and moral support, and Kenneth Hausman for his guidance and friendship throughout this research project.

## TABLE OF CONTENTS

## LIST OF TABLES

**Table** **Page**

## LIST OF FIGURES

**Figure** **Page**

# CHAPTER 1

# INTRODUCTION

The Urban Transportation Modeling System (UTMS) is a set of techniques which is used by transportation planners to model urban travel supply and demand. UTMS is used to predict the number of trips made within an urban area during different times of the day, and where these trips originate and their destination. In addition, UTMS predicts the mode by which these trips are made and predicts the routes taken through the transportation network. The UTMS process is shown in Figure 1.1.

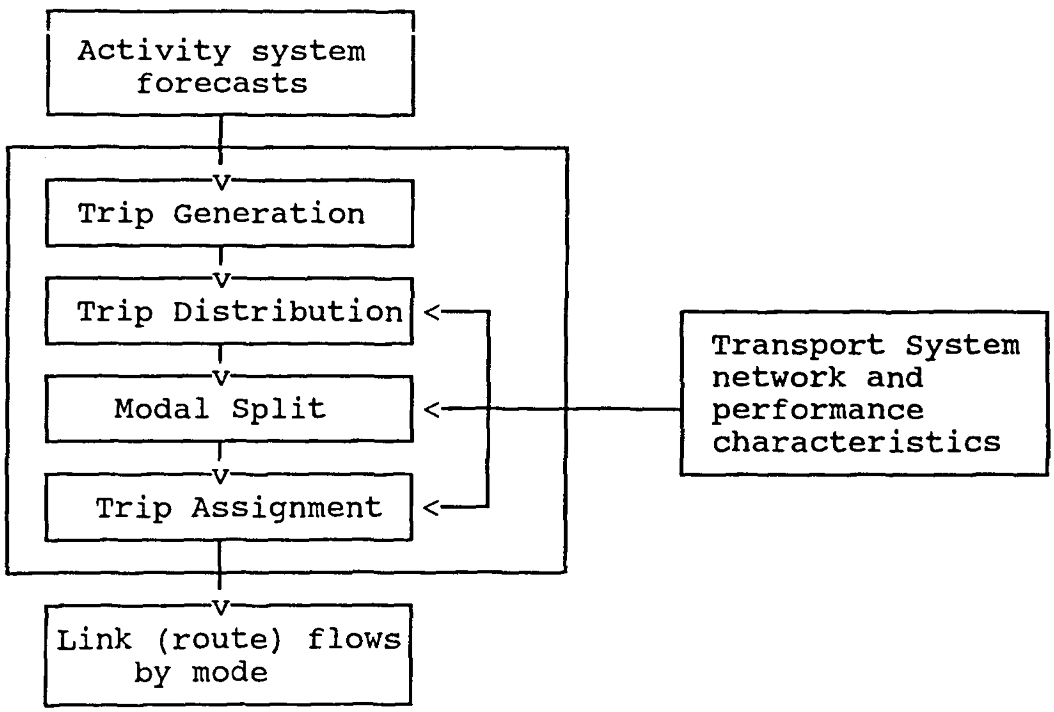



**Figure 1.1 The Urban Transportation Modeling System** (Source: *Urban Transportation Planning A Decision-Oriented Approach*, Michael D. Meyer and Eric J. Miller, 1984)

In order to simplify this complex process, the UTMS has been broken down into four discrete stages. The first stage in UTMS is the trip generation component. This step involves the analysis of land use and population statistics in order to estimate traffic flows. The second stage is trip distribution, which predicts the destination of the flows from each of the origins, usually according to a gravity model (Meyer, M and E. Miller, 1984). The gravity model uses impedance (i.e., distance or travel time) to distribute traffic among origins and destinations. Modal split, the third step in UTMS, projects the split of the flows among the available modes of transport, such as highway and rail. The last component of the UTMS is traffic assignment, which assigns the traffic flows to actual routes in the network. It is this final component, the development of a traffic assignment model, which will be the subject of this paper.

The trip assignment function of UTMS is a process in which traffic flows between each origin-destination (O-D) pair are allocated to actual routes in a given network. The assignment procedures are based on two principles, which were developed by Wardrop (1952):

> (1) Wardrop's First Principle: each individual selects a route between his origin-destination pair which will minimize his own travel cost. In this type of assignment, called "User Equilibrium", a traveler can not improve upon his own individual travel cost by changing to a different route.
>
> (2) Wardrop's Second Principle: each individual selects a route so that the system-wide total transportation cost is minimized. In this type of assignment, called "System Optimization", a traveler can not improve upon the system-wide average cost of travel by changing to a different route.

In an uncongested network, application of both principles would yield the same route assignments. This result occurs because in an uncongested network, the links operate under free flow conditions, and the presence of additional travelers will not increase the travel cost of traversing a route. All of the travelers choose the minimum cost path, minimizing both individual user and total system travel cost. In a congested network, however, the two principles do not generate the same route assignments. Traveling on a congested route could be the minimum path for an individual traveler, but would result in additional travel cost for everyone who has chosen this route. Even if this cost is very small, it will be multiplied by the number of travelers on the route and result in raising the total travel cost for that O-D pair significantly. Moreover, if a single link is utilized by several routes, then congestion on this link can have far reaching effects on the total system cost.

The most realistic assignment of flows through an unregulated network is based on a User Equilibrium model. If travelers have a choice, they will choose to minimize their own travel costs, at the expense of others. Unfortunately, the User Equilibrium assignment process has proven to be more difficult to model than the System Optimization process. Consequently, many transportation planning software packages use trip assignment models which are based on techniques which only approximate the User Equilibrium model, or which are based on a simplified form of the System Optimization model.

Although the User Equilibrium model more closely resembles a typical uncontrolled highway network, it is important to realize that a System Optimization model could be applied if there were a central authority which governed route choice, and whose goal was to minimize the total system travel cost. Implementing this plan has never been a tangible goal until the advent of recent technological advances in traffic management. The Intelligent Vehicle Highway System (IVHS) concept encourages the use of advanced information and congestion technology in order to better utilize the existing transportation network. Route guidance should play a major role in the proposed IVHS systems, but the specifics concerning route assignment have not been widely discussed. It is clear that the goal of the central authority which disseminates route choice information has a great effect on which path is recommended. Currently, for profit companies such as "Shadow Traffic", sell congestion and travel time information, primarily to radio stations. Unlike these private companies, which seek to route users to minimize their individual delays, the authority may wish to route users in order to minimize total system delay. A prevailing question should be: "Does the authority assign trips according to User Equilibrium or System Optimization?". Both scenarios should be examined in depth before making this important policy decision. Therefore, a modeling framework which combines both User Equilibrium and System Optimization assignment models would be useful to the planner considering an IVHS system.

In Chapter 2, some of the commonly used traffic assignment software models, including MinUTP and TRANPLAN, are examined. In Chapter 3, the flow conditions which arise under User Equilibrium and System Optimization assignments for networks with fixed demand are examined. A simple example problem to illustrate the difference between User Equilibrium and System Optimization is presented. In Chapter 4, two composite models which model both User Equilibrium and System Optimal assignments, are presented. The composite models are solved using the GAMS (General Algebraic Modeling System) software, a powerful mathematical programming tool. The first model was based on Beckman's Formulation, (Beckman, MacGuire, Winsten, 1956). The second model was developed by the author and utilizes some unique features of the GAMS software in order to solve the problem of User Equilibrium. Finally, in Chapter 5, the results of the example problem are used to develop general conclusions regarding the applicability of the model, as well as areas of future improvement and research.

# CHAPTER 2

# EXISTING METHODS TO SOLVE THE ASSIGNMENT PROBLEM

## 2.1 Introduction

There are many different personal computer (PC) transportation software packages available to perform the functions of transportation planning-- trip generation, distribution, modal split and trip assignment. Some of these models rely on techniques which try to estimate the User Equilibrium assignment, while others rely on the simpler System Optimization to assign traffic.

Most software packages contain one or several of the commonly used methods of solving the trip assignment problem. These are the all-or-nothing, all shortest-paths, or assignment by a stochastic method. Most UTMS software packages also contain features which allow the user to incrementally load the network with the travel demand. While these different methods attempt to approximate User Equilibrium conditions, the algorithms which are used by the software packages do not replicate the User Equilibrium solution. In addition, no UTMS software package could be found which modeled both User Equilibrium and System Optimization.

## 2.2 Existing Transportation Planning Packages

MinUTP, developed by Comsis Corporation, is a popular transportation planning software package (Comsis, July 1992). The software employs three methods for path routing in the trip assignment function:

(1) All-or-Nothing - all trips are assigned to a single minimum path. Even if there are multiple paths with the same minimum travel time, only one of the paths is used.

(2) All-Shortest-Paths - all trips are assigned to the minimum path, but if there are multiple minimum paths, the trips are equally divided among these links.

(3) Stochastic - if there are multiple efficient paths, the trips are divided among these paths according to an exponential function based on their relative efficiency (i.e., travel time).

Each of these assignment models are very simple, since they consider only the path free flow travel time in order to allocate traffic. Delay caused by congestion on the network links is not calculated. Further, in a congested network, these types of assignments do not model route choice according to either User Equilibrium or System Equilibrium.

In order to assign the traffic more realistically, MinUTP allows the user to incrementally load the network with the traffic. Assignment is based on the All-or-Nothing minimum path assignment, however, only a portion of the total traffic is assigned on each pass. Adjusted travel times based on the Bureau of Public Roads (BPR) congestion curve are then calculated, and are used to determine the minimum path for the next increment of traffic (U.S. Bureau of Public Roads, 1964). The BPR Curve adjusts travel time on a highway link in proportion to the volume to capacity ratio (V/C) for that link. The Bureau of Public Roads (BPR) Congestion Curve equation is presented below:

$$Time_{adjusted} = Time_{freeflow} * \{1 + c*(volume/capacity)^4\}$$

where:

$Time_{adjusted}$ = travel time on congested link

$Time_{freeflow}$ = travel time on uncongested link

$c = 0.15$

*volume* = link volume

*capacity* = link capacity

MinUTP allows for up to ten passes of this kind. Final assignment values can be based on the last iteration, an average of all iterations, a summation of incremental accumulations, or an "Equilibrium volume adjustment". The title "Equilibrium" is misleading because it does not refer to User Equilibrium but, according to the MinUTP User's Manual (1992), to a "process whereby link volumes are obtained by weighing the volumes from each iteration". Further, the Manual states that this process is considered by some to be the most appropriate technique to use in trip assignment. Indeed, this method does consider congestion and its effect on link impedances, and weight certain iterations in an attempt to approximate the User Equilibrium solution. Still, this method is constrained by the ten iteration limit, and for a large network, may vary from the User Equilibrium solution.

TRANPLAN, distributed by the Urban Analysis Group, is another popular transportation modeling software (The Urban Analysis Group, 1990). Like MinUTP, it employs three different types of assignment (loading) models. These assignment models are listed below:

(1) All-or-Nothing - all trips are loaded on the minimum paths (impedances may be based on time, distance, cost, or other user specified parameters).

(2) Stochastic Highway Load - trips are assigned to all reasonable paths, each path receiving a fraction of trips which are proportional to a user specified diversion parameter.

(3) Equilibrium Highway Load - iterative series of all-or-nothing assignments, with an adjustment of travel times in accordance with the BPR curve. The trips are assigned so as to minimize the impedance of each trip.

Like MinUTP, TRANPLAN has several user options to be used in conjunction with the basic assignment models so that they model actual route choice more realistically. The user may choose multiple pass runs, adjusting the time impedances link by link. Incremental loading of the all-or-nothing assignment model is also allowed, with limited iterations available.

TRANPLAN'S Equilibrium Highway load assignment is somewhat different from MinUTP's Equilibrium assignment model. The basis of TRANPLAN's Equilibrium load model is stated in the description of the Equilibrium Highway Load located in TRANPLAN's User Manual (The Urban Analsis Group, 1990), which states:

> "Equilibrium, in the context of transportation assignments, occurs when no trip can be made by an alternate path without increasing the total travel time of all trips in the network." [1]

This implies that the model is based on System Optimization of the network. This is not the case, however. The assignment algorithm assigns the volumes so that the link volumes are as close as possible to the User Optimized equilibrium loadings. The assignment is an iterative process, and the final assignment is a weighted average of the iterations. Again, the actual model attempts to estimate the User Equilibrium routing but does not explicitly calculate the solution.

---

[1] The Urban Analysis Group, *TRANPLAN: User Manual, Version 7.0*, 1990, pg. 4-1.

# CHAPTER 3

# AN EXAMPLE OF USER EQUILIBRIUM VERSUS SYSTEM OPTIMIZATION

## 3.1 Introduction

When transportation demand is fixed, the following flow conditions arise if network assignment is made under System Optimization (for each O-D pair):

$$MC_{p1} = MC_{pm} \leq MC_{pm+1} \leq MC_{pn}$$

where:

$MC_p$ = marginal cost on path p

$h_{pj} > 0 \qquad j = 1, \ldots m$

$h_{pj} = 0 \qquad j = m+1, \ldots n$

Therefore, it is possible to solve for System Optimization by calculating the marginal costs for all paths for each O-D pair, and assigning the traffic to the paths with the minimum marginal costs. The logic behind this definition of a System Optimized flow pattern is straightforward. The objective of an assignment in accordance with Wardrop's Second Principle is to minimize the total system cost. The marginal cost on path P is defined as the "cost" of adding one unit of flow onto path P. Consequently, the flow will be assigned to the path with the lowest marginal cost, so as to keep the total system cost at a minimum.

Wardrop's Second Principle can also be expressed in the following mathematical programming problem form:

$$\text{Minimize} \quad Z = \sum_{a \in A} c_a(f_a) * f_a$$

subject to:

$$Tij - \sum_{p \subseteq Pij} h_p = 0 \quad \text{for } \forall \text{ i,j}$$

$$fa - \sum_{p \subseteq Pij} \delta_{ap} * h_p = 0 \quad \text{for } \forall\ i,j$$

$$h(p_j) \geq 0 \quad j = m+1, \dots n$$

where:

$\delta_{ap}$ = 1 if arc $a$ is included in path $p$

$\delta_{ap}$ = 0 otherwise

$T_{ij}$ = travel demand from $i$ to $j$

$h_p$ = flow on path $p$

$f_a$ = flow on arc $a$

$c(f_a)$ = cost function

The objective function calculates total system cost, so the minimization of this function produces the desired flow pattern for System Optimization. The first constraint ensures that flow on path P includes the flows for all of the arcs contained in that path. The second constraint results in the conservation of flow along all of the utilized paths, from each origin to destination. This formulation produces the same flow pattern as was previously described for System Optimization, and is relatively simple to solve.

Solving for User Equilibrium produces this particular flow pattern:

$$C_{p1} = C_{pm} \leq C_{pm+1} \leq C_{pn}$$

where:

$C_p$ = cost of travel on path $p$

$h_{pj} > 0 \quad j = 1, \dots m$

$h_{pj} = 0 \quad j = m+1, \dots n$

In this case, the flow pattern relies only on the minimum path cost for each O-D pair. If there is a lower cost path available to a user, he/she will seek to switch to that path. The path switching continues until all of the users have minimized their own individual travel time. The mathematical programming formulation of this problem is not straightforward. Beckman (1956) introduced a form, often called "Beckman's Equivalent Optimization Problem", which is presented below:

$$\text{Minimize } Z = \sum_{a \in A} \int_0^{fa} c(fa)\,dfa$$

subject to:

$$T_{ij} - \sum_{p \subseteq P_{ij}} h_p = 0 \quad \text{for } \forall\ i,j$$

$$fa - \sum_{p \subseteq P_{ij}} \delta_{ap} * h_p = 0 \quad \text{for } \forall\ i,j$$

$$h_p \geq 0$$

where:

$\delta_{ap}$ = 1 if arc $a$ is included in path $p$

$\delta_{ap}$ = 0 otherwise

$T_{ij}$ = travel demand from $i$ to $j$

$h_p$ = flow on path $p$

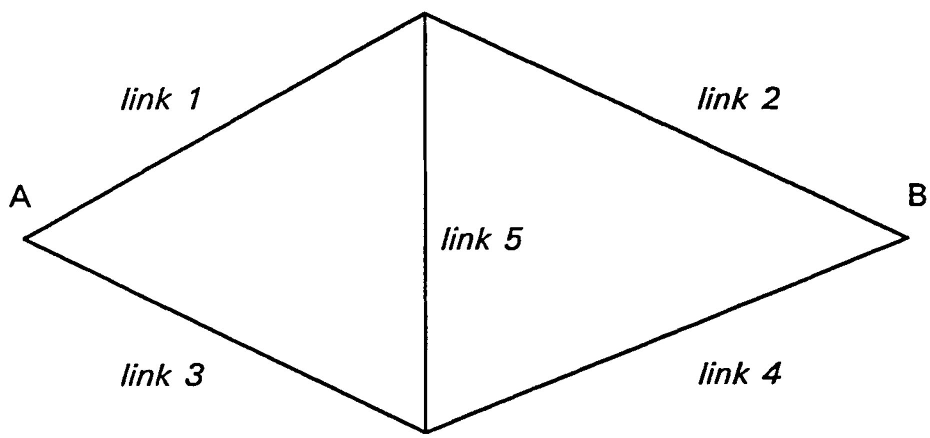
$f_a$ = flow on arc $a$

$c(f_a)$ = cost function

In order to solve this problem it is essential that the cost function be differentiable. Then the derivative may be taken and set equal to zero to determine where the minimum occurs. Although this particular equation has no economic or engineering meaning, it useful in determining the User Equilibrium flow patterns.

## 3.2 The Sample Problem

In order to better illustrate the difference between the User Equilibrium and System Optimal solutions, a small example problem is presented. The parameters of the problem are presented in Figure 3.1. The impedance is defined using the Bureau of Public Roads (BPR) travel time equation, and the fixed demand is 1500 trips from origin A to destination B.




Fixed Demand = 1500 trips from A to B.

| Link Characteristics | | | | |
|---|---|---|---|---|
| Link | Capacity [vph] | Speed [mph] | Distance [miles] | Direction |
| 1 | 1500 | 50 | 5.0 | one-way |
| 2 | 1500 | 50 | 5.0 | one-way |
| 3 | 1200 | 35 | 3.0 | one-way |
| 4 | 1200 | 35 | 3.0 | one-way |
| 5 | 1200 | 35 | 1.0 | two-way |

**Figure 3.1 Sample Problem**

Under User Equilibrium, the trips are assigned by applying the logic in Wardrop's First Principle. Each user will choose a path so that his travel time is minimized, and that he can not improve upon that travel time by taking a different path. In this simple network, there are only four possible paths from A to B, that are defined as follows: Path 1 consists of links 1 and 2; Path 2, links 3 and 4; Path 3, links 1, 5 and 4; and Path 4, links 3, 5 and 2.

The problem is to assign flows such that the cost for each individual user is minimized. In order to find the minimum value for this function, it is necessary to take the derivative and set this equal to zero. The key to determining the solution to the sample problem is the differentiable cost function, the BPR Curve. When the BPR curve is substituted for c(x), the minimum occurs at:

$$fa * Time_{freeflow} *\{1+ 0.03*(volume/capacity)^4\} = 0$$

This is the basis the formulation of Beckman's Model. The solution to this equation produces the User Equilibrium flow pattern. The solution to the sample problem is presented in Table 3.1, listed under "User". Note that the trips are distributed so that the resultant path travel times are identical.

The System Optimization solution uses the logic of Wardrop's Second Principle to arrive at a solution, in order to minimize the system's total travel time. In this case, the marginal costs of each cost function are determined in order to determine the optimal flow. The marginal path cost is simply the partial derivative of the BPR curve, with respect to flow (or the traffic volume in our equation)

The System Optimal solution for this sample problem is presented in Table 3.1, listed under "System". Note that a majority of the travelers are assigned to path 2, which has the lower capacity and the lower impedance. Many of the trips are assigned to path

1, which has a higher travel time then path 2. This allows path 2 to maintain its minimal travel time for a selected group of users, thereby lowering the total system travel cost.

**Table 3.1 Summary of Solutions for the Sample Problem**

| | Path 1 | | Path 2 | | Path 3 | | Path 4 | | Total |
|---|---|---|---|---|---|---|---|---|---|
| Model | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | System Cost [min] |
| User | 268 | 12.0 | 1232 | 12.0 | 0 | 12.9 | 0 | 12.9 | 18,000 |
| System | 642 | 12.1 | 139 | 10.7 | 0 | 12.9 | 0 | 12.9 | 16,923 |

As can be seen from this short example, the User Equilibrium assignment can vary quite significantly from the System Optimal assignment. It is important to note that although path 1 was clearly the minimum path for most of the travelers, the total demand was greater than the capacity. Path 2 was able to handle any or all of the travel demand without becoming congested.

### 3.3 Solutions to the Sample Problem Using MinUTP and TRANPLAN

How would MinUTP and TRANPLAN perform on this simple example? The problem was solved using both software packages. The MinUTP results were calculated by two different methods: using the All-or-Nothing (AON) assignment, and the AON assignment with the Equilibrium Volume adjustment (3 iterations). Two different assignment models were also run for TRANPLAN: the AON assignment was determined, along with the Equilibrium Highway Load assignment (3 iterations). Although this problem is very small, the results provide an indicator of how the different software packages perform. These results are presented in Table 3.2.

**Table 3.2 Summary of Solutions for the Sample Problem Using MinUTP and TRANPLAN**

| | Path 1 | | Path 2 | | Path 3 | | Path 4 | | Total |
|---|---|---|---|---|---|---|---|---|---|
| Model | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | System Cost [min] |
| MinUTP | | | | | | | | | |
| AON | 0 | 12.0 | 1500 | 14.1 | 0 | 12.9 | 0 | 12.9 | 21,079 |
| Equil | 266 | 12.0 | 1234 | 12.0 | 0 | 12.9 | 0 | 12.9 | 18,012 |
| TRANPLAN | | | | | | | | | |
| AON | 0 | 12.0 | 1500 | 14.1 | 0 | 12.9 | 0 | 12.9 | 21,079 |
| Equil | 270 | 12.0 | 1230 | 12.0 | 0 | 12.9 | 0 | 12.9 | 17,987 |

As can be seen from the results, running an AON assignment for this sample problem produces a very unrealistic assignment. The AON assignment does not consider any congested travel times when choosing the shortest path. On the other hand, MinUTP's AON assignment with the equilibrium adjustment did quite well - the assignment is very close to the calculated User Equilibrium results. For a simple network such as the sample problem, the algorithms employed by MinUTP seem to replicate User Equilibrium quite well, with only a small margin of error. For larger networks which involve much more traffic volumes, however, the error may be a significant factor.

TRANPLAN's AON assignment calculates the assignment in the same manner as MinUTP's model, so the solution was exactly the same. Again, this method was very unrealistic. TRANPLAN's Equilibrium Highway Load assignment model is very close to User Equilibrium results. In fact, the small discrepency of this assignment with respect to the calculated value for User Equilibrium appear to be due to rounding errors. Still, these results are only an estimation of User Equilibrium.

# CHAPTER 4

# DEVELOPMENT OF THE COMBINED USER EQUILIBRIUM AND SYSTEM OPTIMAL ASSIGNMENT MODELS

## 4.1 Introduction

In order to compare a User Equilibrium model with a System Optimization model, a new formulation which would allow the problem to be solved both ways was developed. A more accurate User Equilibrium solution than the one supplied by the popular transportation software packages was sought, and two different mathematical programming formulations were used to model the network assignment process. By changing the objective functions and adding some constraints to our mathematical programs, the same data set is used to generate both the User Equilibrium and System Optimal solutions.

Using mathematical programming is not a new approach to solving the assignment problem, for many assignment algorithms are based on mathematical programming (Florian, M., et al., 1979; LeBlanc, L.J., et al., 1975). However, formulating and solving models for large networks was difficult. Since most "real world" networks were very large and would demand the formulation of hundreds of constraints, the Mathematical Programming method has never been practical outside of academic applications.

The advent of the GAMS (General Algebraic Modeling System) has provided a convenient, easy to use personal computer software package able to formulate and solve large mathematical programs. GAMS combines the use of relational databases with mathematical programming theory. Instead of explicitly defining long constraints and multi-term objective functions, GAMS is able to generate these equations, simply by reading the form of the equation and writing them over a user specified set of ranges.

This simplifies the data entry tremendously, and makes revisions and multi-runs much easier and quicker to perform.

The basic components of the GAMS Model are presented in Figure 4.1.

```
INPUTS

SETS
   Declaration
   Assignment of Members

Data (PARAMETERS, TABLES, SCALARS
   Declaration
   Assignment of Members

VARIABLES
   Declaration
   Assignment of Members
   (Optional) Assignment of Bounds
   and/or Initial Values

EQUATIONS
   Declaration
   Definition

MODEL and SOLVE Statements

(Optional) DISPLAY Statements
```

```
OUTPUTS

Echo Print

Reference Maps

Equation Listings

Status Reports

Results
```

**Figure 4.1: Structure of a GAMS Model** (Source: *GAMS: A User's Guide*, Brooke, , D. Kendrick, A. Meeraus, 1988)

The input data base is coded in free format, and data entry is accomplished through the use of tables. Instead of relying on an iterative process to estimate the solutions, the mathematical programs allow a direct solution for User Equilibrium and System Optimization to be calculated. Since the same data set can be used for each objective, the results can be compared quite easily. The GAMS Model is a database which is coded in free format, but must consist of statements in the GAMS language.

In order to illustrate the basics of GAMS and of the different model formulations, the input data base is presented in this chapter. First, the creation of the GAMS model based on Beckman's Formulation is described. In addition, the second model, the Author's Model, is presented, and the logic behind its formulation described. Rather than dwell on the syntax of GAMS, this paper shall concentrate on the basic components of the model, their definition, and their assignment. For more details regarding the GAMS language and software, the reader is directed to *GAMS: A User's Guide* (1990).

### 4.2 The GAMS Model Based on Beckman's Formulation

The first component of a GAMS model is **Sets**, which are the indices in the algebraic representations of models. In the case of the sample problem, several different sets are defined below, and then assigned members:

```
SETS
      I  Origins      /A/
      J  Destinations /B/
      L  Links        /1*5/
      P  Paths        /P1*P4/;
```

The sets "I", "J", "L", and "P" are declared to be existent; the members of each set are also defined, separated by commas. These sets are static sets, (i.e., their members will not change). The sets define the domains over which our model will solve. The "*" function allows definition of sets without explicitly listing all of the members. "1*4" declares members 1, 2, 3 and 4. This function simplifies large data set declaration.

The next component of the GAMS model is the **Data**. Data entry was accomplished through the use of Parameters, Tables, and Scalars. For the sample problem, input values are needed for the travel demand between the origins and destinations. In addition, definition and assignment values for capacity and free flow travel time are required. Both the "Table" and "Parameter" functions for the data entry are listed below.

```
TABLE VOLUME(I,J,*)        travel demand from A to B
      TRIPS
A.B    1500;

PARAMETER CAP(L)    capacity of the links
/1  1500
 2  1500
 3  1200
 4  1200
 5  1200/;

PARAMETER SPEED(L)  freeflow speed on the links
/1    50
 2    50
 3    35
 4    35
 5    35/;

PARAMETER DISTANCE(L) distance on the link
/1     5
 2     5
 3     3
 4     3
 5     1/;
```

In this step, the parameters "VOLUME (I,J,*)", "CAP(L)", "SPEED(L)", and "DISTANCE(L)" are defined. Notice that a domain is declared for each data parameter. The ability to define data (and later, equations) over a set domain is a very important feature of the GAMS model, thus allowing the derivation of a large model from a relatively small table of numbers. The capacity and freeflow travel time are link characteristics, so they are defined over the entire set of links "L". "VOLUME" is a three-dimensional table, defined over the set of ordered pairs "(I,J)" and by "trips". The "*" listed in the domain is a wildcard domain value, used here because "trips" is a not a previously defined set.

In addition to these data parameters definintion of mapping sets (or correspondence matrices) is needed under the data function. These data tables, parameters, or scalars, are set up to define a correspondence between the defined SETS. For the example, there are several correspondence matrices needed, as shown below:

```
TABLE ODPATH(I,J,P)       origin-destination to link mapping set
     P1  P2  P3  P4
A.B  1   1   1   1;

TABLE LINKPATH(L,P)       link to path mapping set
     P1  P2  P3  P4
1    1       1
2    1           1
3        1       1
4        1   1
5            1   1;
```

These tables declare a correspondence of origin-destination A.B to paths 1, 2, 3 and 4, via "ODPATH(I,J,P)". This means that paths 1, 2, 3 and 4 travel from origin A to destination B. Table "LINKPATH(L,P)" sets up a correspondence from links to paths. For example, link 1 is included in links that make up path 1, but not in the links for path 2. Link 3 is included in the links that constitute path 2, but not in path 1. A blank value, interpreted as a zero value by GAMS, means that there is no correspondence. These mapping sets are vital when equations are set up, in order to limit the defined domains.

The next component of the GAMS model that needs consideration is the **Variables**. The decision variables (endogenous variables) are declared and assigned over a domain, if pertinent. In addition, every variable must be assigned a type such as FREE (the default value), POSITIVE, NEGATIVE, BINARY, or INTEGER. Keep in mind that the objective function variable must be a scalar quantity and must be "free". Declaration and assignment of the domains for the decision variables are shown below:

```
POSITIVE VARIABLES
F(L)   flow on a link
H(P)   flow on a path;
```

```
VARIABLES
UE      objective function for user equilibrium
SO      objective function for system optimal;
```

The existence of flow over the set of all links, "F(L)", and flow over the set of all paths, "H(P)" is declared. Bounds are assigned to our variables (0 to positive infinity), by defining them as "POSITIVE". The objective functions, "UE" and "SO", are kept free.

The next step in the GAMS model formulation is the **Equations**. This is where GAMS powerful use of relational databases is most apparent. If a group of constraints has the same basic structure, all of the constraints are created by the GAMS software simultaneously, instead of being entered individually by the user. The "EQUATION" function encompasses both equality and inequality constraints, as well as the objective functions. Equations must be declared and defined in separate statements, as shown below:

```
EQUATIONS
DEMAND(I,J) travel demand from origin i to destination j
FLOW(L)   defines flow on a given link
OBJSYS    objective function under system optimization
OBJUSER   objective function under user equilibrium;

DEMAND(I,J) .. SUM(P$ODPATH(I,J,P), H(P)) =E= VOLUME(I,J,'trips');

FLOW(L) .. SUM(P $LINKPATH(L,P), H(P)) =E= F(L);

OBJSYS .. SUM(L, (F(L)* DISTANCE(L)/SPEED(L))* (1 + 0.15*
POWER((F(L)/CAP(L)),4))) =E= SO;

OBJUSER .. SUM(L, (F(L)* DISTANCE(L)/SPEED(L))* 1 + 0.03*POWER
(F(L),4/POWER (CAP(L),4) =E= UE;
```

The first constraint, "DEMAND(I,J)", ensures that the flow which exists between each O-D pair utilizes one or all of the paths which is enumerated for this purpose. In this manner, the conservation of flow from origin I to destination J along the defined paths is ensured. The equation specifically states that the summation of all the flows, H(P), over the set of all paths P which share the same O-D pair as reported in the mapping set ODPATH(I,J,P), must equal the volume which is assigned in the table "Volume". The "$" is called a dollar operator and controls the range of the summation according to the referenced mapping set. This is a very important operator in the GAMS language because it can restrict the set elements which contribute to the total summation.

The next constraint, "FLOW(L)", calculates the total flow on each link. Generally, the equation states that the flow on each link L must be equal to the summation of all the pathflows, H(P), over the set of paths. The dollar operator ensures that, as each path is summed, only links contained in the LINKPATH mapping set are included in the total flow for that path. The traffic flow on each link is needed in the calculation of adjusted impedances for each link.

The "OBJSYS" equation is the objective function for the System Optimized solution. Recalling that the objective of System Optimization is to minimize the total system wide travel cost, this problem is set up as a minimization problem. The objective equation states that the summation, over the set of all links, of the flow on link L multiplied by the impedance of link L is equal to "SO".

The methodology behind the objective function for User Equilibrium, "OBJUSER", is based on Beckman's Formulation of the problem. The objective function is the derivative of the BPR Equation. The equation "OBJUSER" states that the summation, over the set of all links, of the equation listed above is equal to "UE". Although this particular equation has no real economic meaning, it will indicate which values of f(L) minimize user cost.

Finally, after the sets, data, variables and equations are listed, the modeling is begun by calling up the "MODEL" and "SOLVE" functions, below:

```
MODEL SYSTEMOPT/OBJSYS,DEMAND,FLOW/;
SOLVE SYSTEMOPT USING NLP MINIMIZING SO:

MODEL USEREQUIL/OBJUSER,DEMAND,FLOW/;
SOLVE USEREQUIL USING NLP MINIMIZING UE:
```

The output obtained from this run is contained in the Appendix. The first portion of the GAMS output files is merely an echo of the original input file. The remaining portion of the output lists the results. These results are summarized in Table 4.1 below:

**Table 4.1 Summary of Solutions for the Sample Problem Using Beckman's Formulation, Solved Using GAMS**

| | Path 1 | | Path 2 | | Path 3 | | Path 4 | | Total |
|---|---|---|---|---|---|---|---|---|---|
| Model | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | System Cost [min] |
| User | 268 | 12.0 | 1232 | 12.0 | 0 | 12.9 | 0 | 12.9 | 18,000 |
| System | 642 | 12.1 | 858 | 10.7 | 0 | 12.9 | 0 | 12.9 | 16,923 |

### 4.3 The GAMS Model Based on the Author's Formulation

Because of its special features, the GAMS software allows a unique formulation of the User Equilibrium problem, significantly different from Beckman's Model previously discussed. This model is denoted as the "Author's Model".

The Sets for the problems were the same as previously stated. These are listed below:

```
SETS
     I  Origins       /A/
     J  Destinations /B/
     L  Links         /1*5/
     P  Paths         /P1*P4/;
```

The Data entry was also very similar to the GAMS Model of Beckman's Formulation, with the exception of an additional parameter called "COEFF(L)". This parameter allows the user to specify a value for the coefficient c in the BPR Curve. This gives greater flexibility for solving a problem if all of the links do not use the standard 0.15 as a value for the coefficient. For the sample problem, all of the links use a value for the coefficient of 0.15. The data entry and mapping step for this model is presented below:

```
TABLE VOLUME(I,J,*)       travel demand from A to B
         TRIPS
A.B      1500;

PARAMETER CAP(L)    capacity of links
/1 1500
 2 1500
 3 1200
 4 1200
 5 1200/;

PARAMETER SPEED(L)  freeflow speed on the links
/1    50
 2    50
 3    35
 4    35
 5    35/;

PARAMETER DISTANCE(L)  distance on the link
/1      5
 2      5
 3      3
 4      3
 5      1/;

PARAMETER COEFF(L)  value of c to be used in BPR curve
/1 0.15
 2 0.15
 3 0.15
 4 0.15
 5 0.15/;

TABLE ODPATH(I,J,P)       origin-destination to link mapping set
       P1  P2 P3 P4
A.B    1    1  1   1 ;

TABLE LINKPATH(L,P)       link to path mapping set
       P1  P2 P3 P4
1      1       1
2      1          1
3          1      1
4          1   1
5              1  1 ;
```

The Variables are the next step in the GAMS model Component list. Many of the Variables remained the same, but the Model required the addition of two new positive variables, called "PCOST(P)" and "MCOST(I,J)". The PCOST(P) is a path characteristic, representing the unit time cost of traveling a path. "MCOST(I,J)" is the minimum value of all of the calculated values for the set of PCOST(P), for each I and J (O-D pair). The entire list of variables is presented below:

```
POSITIVE VARIABLES
F(L)      flow on a link
H(P)      flow on a path
PCOST(P)  average time cost of traveling on a path
MCOST(I,J) minimum travel time of all paths for each O-D;

VARIABLES
UE      objective function for user equilibrium
SO      objective function for system optimal;
```

The next step in the GAMS formulation of the model is the equations. Once again, the "DEMAND" equation is used in order to conserve flow along the paths, from the origins to the destinations. Likewise, the "FLOW" equation defines the flow on the links, F(L), and its relationship to the flow along the paths, H(P). Also included are two additional equations which are not found in Beckman's Model formulation. The "PATH COST(P)" equation simply calculates the average travel cost for each path. The impedances are calculated over the set of paths, but the set is restricted to those links that are identified in the mapping set LINKPATH (L,P) as being used in that path. In addition, there is an equation called "MINPATH(I,J)", which utilizes a special GAMS function called "Smin". This operator is used to find the smallest values over the domain of a specific set. In the case of the sample problem, minimum value of the average path cost, PCOST(P), is needed over all of the paths corresponding to a particular O-D pair (I,J). The "MINPATH" equation allows the SMIN operator to find the minimum value for PCOST(P) for each (I,J), over the set of all paths. This set of paths is restricted by the dollar operator, which conducts the search over the paths which correspond to the particular O-D pair, as stated in the mapping set ODPATH(I,J,P). The equations, along with the model statements, are listed below:

```
EQUATIONS
DEMAND(I,J) travel demand from origin i to destination j
FLOW(L)  defines flow on a link as total of its pathflows
PATH COST(P)  average path cost on path p
MINPATH(I,J) finds the min average path cost for a given O-D
OBJSYS      objective function under system optimization
OBJUSER     objective function under user equilibrium;

DEMAND(I,J) .. SUM(P$ODPATH(I,J,P), H(P)) =E= VOLUME(I,J,'trips');

FLOW(L) .. SUM(P $LINKPATH(L,P), H(P)) =E= F(L);

PATH COST(P) .. SUM(L $LINKPATH(L,P), (DISTANCE(L)/SPEED(L)) * (1 +
COEFF(L) *POWER ((F(L)/CAP(L)),4))) =E= PCOST(P);

MINPATH(I,J) .. SMIN(P$ODPATH(I,J,P), PCOST(P)) =E= MCOST(I,J);

OBJSYS .. SUM(L, (F(L)* (DISTANCE(L)/SPEED(L)))* (1 + COEFF(L) *POWER
((F(L)/CAP(L)),4))) =E= SO;

OBJUSER .. SUM((I,J,P),H(P)*(PCOST(P) - MCOST(I,J)))=E= UE;

MODEL SYSTEMOPT/OBJSYS,DEMAND,FLOW/;
SOLVE SYSTEMOPT USING NLP MINIMIZING SO:

MODEL USEREQUIL/OBJUSER,DEMAND,FLOW,PATH COST,MINPATH/;
SOLVE USEREQUIL USING DNLP MINIMIZING UE:
```

The objective function for System Optimization, SYSTEM, is the same as for the previously explained model. This function, which is minimized in the "Model" statement, equates the total sytem wide travel costs to "SO".

The objective function for User Equilibrium, "OBJUSER", differs greatly from Beckman's formulation. The objective function equates the difference between the path cost for a given P, PATH COST(P), and the minimum path cost, MCOST(I,J), for the same O-D pair. The Model statement begins an iterative process of loading the network. After each iteration (new loading), the "MINPATH(I,J)" equation searches the paths for each (I,J) for the minimum path cost. This iterative process continues until the difference between the minimum cost paths and the utilized paths (H(P) > 0) for each O-D pair is at a minimum. The GAMS output file for this case is also contained in the Appendix. The results of the sample problem are summarized in Table 4.2.

**Table 4.2 Summary of Solutions for the Sample Problem Using Author's Formulation, Solved Using GAMS**

| | Path 1 | | Path 2 | | Path 3 | | Path 4 | | Total |
|---|---|---|---|---|---|---|---|---|---|
| Model | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | System Cost [min] |
| User | 268 | 12.0 | 1232 | 12.0 | 0 | 12.9 | 0 | 12.9 | 18,000 |
| System | 642 | 12.1 | 858 | 10.7 | 0 | 12.9 | 0 | 12.9 | 16,923 |

This objective statement models the logic behind Wardrop's First Principle precisely. Each user searches for his minimum cost path, and if it happens that the addition of volume onto this path causes another route to become the minimum path, the user will seek to switch to this route. The ability of the GAMS operator "SMin" to search a selected set for the minimum value allows this type of behavior to be modeled.

## CHAPTER 5

## CONCLUSIONS

Both the GAMS Model of the Beckman's Formulation and the Author's Model calculated the correct User Equilibrium and System Optimal for the small example problem. The results of using all the approaches that were discussed in this paper to solve the sample problem are presented in Table 5.1. Because of the flexibility of the GAMS software, larger networks do not require a reformulation of the models, but just the addition of data.

**Table 5.1 Summary of Solutions for the Sample Problem Using All Approaches**

| | Path 1 | | Path 2 | | Path 3 | | Path 4 | | Total |
|---|---|---|---|---|---|---|---|---|---|
| Model | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | Flow [trips] | Travel Time [min] | System Cost [min] |
| MINUTP | | | | | | | | | |
| AON | 0 | 12.0 | 1500 | 14.1 | 0 | 12.9 | 0 | 12.9 | 21,079 |
| Equil | 266 | 12.0 | 1234 | 12.0 | 0 | 12.9 | 0 | 12.9 | 18,102 |
| TRANPLAN | | | | | | | | | |
| AON | 0 | 12.0 | 1500 | 14.1 | 0 | 12.9 | 0 | 12.9 | 21,079 |
| Equil | 270 | 12.0 | 1230 | 12.0 | 0 | 12.9 | 0 | 12.9 | 17,987 |
| Beckman's Formulation | | | | | | | | | |
| User | 268 | 12.0 | 1232 | 12.0 | 0 | 12.9 | 0 | 12.9 | 18,000 |
| System | 642 | 12.1 | 858 | 10.7 | 0 | 12.9 | 0 | 12.9 | 16,923 |
| Author's Formulation | | | | | | | | | |
| User | 268 | 12.0 | 1232 | 12.0 | 0 | 12.9 | 0 | 12.9 | 18,000 |
| System | 642 | 12.1 | 858 | 10.7 | 0 | 12.9 | 0 | 12.9 | 16,923 |

The convenience of having both types of assignments within the same model formulation allows comparison of results quite easily. Any organization which is considering an IVHS System, in particular an Advanced Traveler Information System (ATIS), must consider both types of assignments- User versus System. By comparing User Equilibrium with System Optimization, a central authority, such as an ATIS authority, may see what kind of results could occur if it attempted to route users so as to optimize the overall system costs. Will the total cost savings be beneficial to the community, if users are routed according to System Optimal solution? Will the benefits be great enough to inconvenience some of the users by assigning them to a longer route? If only a portion of the highway users purchase ATIS, are they going to be penalized by increased travel costs under a System Optimal assignment? These are very pertinent questions which must be answered prior to the development of ATIS for a corridor.

The GAMS Model of Beckman's Formulation solved this problem quite efficiently. The key to this model working so well was the differentiability of the impedance equation, the BPR Curve. Because it was assumed that all of the links were highway links which operated under this standard congestion equation, the minimum value was simple to calculate. As a result of its form, the derivative of this function was calculated to find the minimum. The curve which GAMS was able to optimize was a smooth curve, so the solution to the sample was arrived at by the software very quickly. Larger networks were likewise solved very quickly due to the nature of the objective function.

Beckman's Formulation Model does have its drawbacks, however. If the impedance equation for the links is not differentiable, then this formulation can not be used directly. This would be the case if the paths included a commuter train line; in this case, the impedance is constant. In order to include these links in Beckman's Formulation, the travel time must be put into a form that is differentiable, so that it is no longer constant.

The Author's Model does not rely on any of the constraints or impedances being differentiable. The travel cost equation can be easily modified. If there is a mix of highway links and train links, the coefficient c in the BPR Equation can be set for "0.15" for the highway links, and "0" for the train links. Then, the true path cost can be calculated and used to find User Equilibrium. Instead of relying on the derivative of the impedance equation, the model relies on the GAMS operator "SMin". Moreover, the Author's Formulation is unique because it models Wardrop's First Principle precisely. Each user seeks to minimize his/her individual travel cost. If a lower cost of travel is available by utilizing a different path, the user will switch to this path. The nature of the "Smin" function induces "pathswitching" until the unit path costs on all utilized paths are equal.

The function "SMin" does have a negative point. This function is not a "smooth" curve; it is discontinuous at certain points. The GAMS software does minimize using this function, but the program must go through additional iterations to solve the problem. For the 5 link example problem, GAMS solved the Author's Model for determing User Equilibrium in 2 iterations as compared to 11 iterations for Beckman's Formulation. Therefore, using the PC based software to solve a large problem could be very time consuming.

Data entry into the two models was very simple and straightforward. The GAMS software's database characteristics were very helpful. The biggest drawback to using both the Beckman Model Formulation and the Author's Model was the manual enumeration of the paths. For a large network with multiple O-D pairs, the task of naming many different paths, which can include many links, can become very tedious. MinUTP and TRANPLAN both have "pathbuilding" programs within the software package so that the paths are generated automatically. This type of capability would be invaluable to the users of GAMS models.

Both of the GAMS models that have been described-- Beckman's Formulation and the Author's Model-- provide an interesting outlook on the trip assignment procedure. There are several areas of improvement, however, which must be addressed. Manual enumeration of the paths by the user is a very large drawback to using the models. Future research could include developing a "pathbuilding" program to be used in conjunction with the GAMS software, similar to those used by many UTMS software packages. The GAMS programs are very flexible, and could be incorporated into such a program. In addition, future research could include development of a user friendly program so that a user can work interactively to build the GAMS model database, run the path building program and run the model.

# APPENDIX

## THE GAMS MODEL OUTPUT BASED ON BECKMAN'S FORMULATION

```
_GAMS 2.05/S PC AT/XT                    92/06/22 15:00:41

GENERAL  A L G E B R A I C   M O D E L I N G   S Y S T E M
C O M P I L A T I O N
SETS
I    ORIGINS /A/
J    DESTINATIONS /B/
L     LINKS /1*5/
P     PATHS /P1*P4/ ;

TABLE VOLUME(I,J,*) DEMAND BETWEEN ORIGINS AND DESTINATIONS
        TRIPS
A.B   1500 ;

PARAMETER CAP(L) CAPACITY OF LINKS
/1 1500
 2 1500
 3 1200
 4 1200
 5 1200/;

PARAMETER SPEED(L) FREE FLOW TRAVEL SPEED ON THE LINKS
/1 50
 2 50
 3 35
 4 35
 5 35/;

PARAMETER DISTANCE(L) LENGTH OF EACH LINK IN MILES
/1 5
 2 5
 3 3
 4 3
 5 1/;

PARAMETER COEFF(L) VALUE OF COEFFICIENT TO BE USED IN BPR EQUATION
/1 0.15
 2 0.15
 3 0.15
 4 0.15
 5 0.15/;

TABLE ODPATH(I,J,P) ORIGIN-DESTINATION PATH MATRIX
    P1 P2 P3 P4
```

```
A.B 1 1 1 1;



TABLE LINKPATH(L,P) LINK-PATH MATRIX
  P1 P2 P3 P4
1 1     1
2 1        1
3    1     1
4    1  1
5       1 1;

VARIABLES
F(L)       FLOW ON A LINK
H(P)       FLOW ON A PATH

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
C O M P I L A T I O N

Z          OBJECTIVE FUNCTION
PCOST(P)
MCOST
UCOST
POSITIVE VARIABLES F,H,PCOST,MCOST;

EQUATIONS
OBJUSER      OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM, IN
HOURS PER PATH
OBJSYSTEM    OBJECTIVE FUNCT. UNDER SYSTEM EQUILIBRIUM, IN
USER-HOURS
DEMAND(I,J)   TRAVEL DEMAND
LINKFLOW(L)   FLOW ON EACH LINK
PATHCOST(P)   COST ON EACH PATH
MINPATH
USERCOST;

DEMAND(I,J) .. SUM(P $ODPATH(I,J,P), H(P)) =E= VOLUME(I,J,'TRIPS');

LINKFLOW(L) .. SUM(P $LINKPATH(L,P), H(P)) =E= F(L);

OBJSYSTEM .. SUM(L, F(L)*(DISTANCE(L)/SPEED(L))*(1 +
COEFF(L)*POWER((F(L)
                                   /CAP(L)),4))) =E= Z;

OBJUSER .. SUM(L, F(L)*(DISTANCE(L)/SPEED(L))*(1 + (COEFF(L)/5)*
                  POWER(F(L),4)/POWER(CAP(L),4))) =E= Z;

MODEL TRANSYSTEM
/DEMAND,LINKFLOW,OBJSYSTEM/;

MODEL TRANUSER
/DEMAND,LINKFLOW,OBJUSER/;

```

```
 85 SOLVE TRANSYSTEM USING NLP MINIMIZING Z;
 86 SOLVE TRANUSER USING DNLP MINIMIZING Z;
GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
SYMBOL LISTING

SYMBOL     TYPE   REFERENCES

CAP        PARAM  DECLARED     11  DEFINED     12    REF     75
                  77
COEFF      PARAM  DECLARED     32  DEFINED     33    REF     75
                  77
DEMAND     EQU    DECLARED     65  DEFINED     71 IMPL-ASN     85
                  86    REF    80     83
DISTANCE   PARAM  DECLARED     25  DEFINED     26    REF     75
                  77
F          VAR    DECLARED     54 IMPL-ASN     85    86
                  REF     60     73   2*75   2*77
H          VAR    DECLARED     55 IMPL-ASN     85    86
                  REF     60     71     73
I          SET    DECLARED      2  DEFINED      2    REF      7
                  39     65   2*71  CONTROL    71
IN         EQU    DECLARED     63     REF     64
J          SET    DECLARED      3  DEFINED      3    REF      7
                  39     65   2*71  CONTROL    71
L          SET    DECLARED      4  DEFINED      4    REF     11
                  18     25     32     45     54     66
                2*73   6*75   6*77  CONTROL    73     75
                  77
LINKFLOW   EQU    DECLARED     66  DEFINED     73 IMPL-ASN     85
                  86    REF    80     83
LINKPATH   PARAM  DECLARED     45  DEFINED     45    REF     73
MCOST      VAR    DECLARED     58    REF     60
MINPATH    EQU    DECLARED     68
OBJSYSTEM  EQU    DECLARED     64  DEFINED     75 IMPL-ASN     85
                  REF     80
OBJUSER    EQU    DECLARED     63  DEFINED     77 IMPL-ASN     86
                  REF     83
ODPATH     PARAM  DECLARED     39  DEFINED     39    REF     71
P          SET    DECLARED      5  DEFINED      5    REF     39
                  45     55     57     67   2*71   2*73
              CONTROL     71     73
PATHCOST   EQU    DECLARED     67
PCOST      VAR    DECLARED     57    REF     60
POWER      FUNCT     REF     75   2*77
SPEED      PARAM  DECLARED     18  DEFINED     19    REF     75
                  77
TRANSYSTEM MODEL  DECLARED     79  DEFINED     80    REF     85
TRANUSER   MODEL  DECLARED     82  DEFINED     83    REF     86
UCOST      VAR    DECLARED     59
USERCOST   EQU    DECLARED     69
VOLUME     PARAM  DECLARED      7  DEFINED      7    REF     71
Z          VAR    DECLARED     56 IMPL-ASN     85    86
                  REF     75     77     85     86
```

SETS

```
I          ORIGINS
J          DESTINATIONS
L          LINKS
P          PATHS
```

PARAMETERS

```
CAP        CAPACITY OF LINKS
COEFF      VALUE OF COEFFICIENT TO BE USED IN BPR EQUATION
DISTANCE   LENGTH OF EACH LINK IN MILES
LINKPATH   LINK-PATH MATRIX
ODPATH     ORIGIN-DESTINATION PATH MATRIX
SPEED      FREE FLOW TRAVEL SPEED ON THE LINKS
VOLUME     DEMAND BETWEEN ORIGINS AND DESTINATIONS
```

VARIABLES

```
F          FLOW ON A LINK
H          FLOW ON A PATH
MCOST      MINIMUM PATH COST FOR O-D
PCOST      PATHCOST
UCOST      USERCOST
Z          OBJECTIVE FUNCTION
```

EQUATIONS

```
DEMAND     TRAVEL DEMAND
LINKFLOW   FLOW ON EACH LINK
MINPATH    FINDS MINIMUM PATH
OBJSYSTEM  OBJECTIVE FUNCT. UNDER SYSTEM EQUILIBRIUM
OBJUSER    OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM
PATHCOST   COST ON EACH PATH
USERCOST   COST FOR EACH USER
```

MODELS

```
TRANSYSTEM
TRANUSER
```

COMPILATION TIME = 0.059 MINUTES

GENERAL ALGEBRAIC MODELING SYSTEM
EQUATION LISTING SOLVE TRANSYSTEM USING NLP FROM LINE 85

---- DEMAND =E=

DEMAND(A,B).. H(P1) + H(P2) + H(P3) + H(P4) =E= 1500 ; (LHS = 0 ***)

```
---- LINKFLOW    =E=  FLOW ON EACH LINK

LINKFLOW(1)..  - F(1) + H(P1) + H(P3) =E= 0 ; (LHS = 0)
LINKFLOW(2)..  - F(2) + H(P1) + H(P4) =E= 0 ; (LHS = 0)
LINKFLOW(3)..  - F(3) + H(P2) + H(P4) =E= 0 ; (LHS = 0)
REMAINING 2 ENTRIES SKIPPED


---- OBJSYSTEM  =E=  OBJECTIVE FUNCT. UNDER SYSTEM EQUILIBRIUM

OBJSYSTEM..  (0.1)*F(1) + (0.1)*F(2) + (0.0857)*F(3) + (0.0857)*F(4) +
(0.0286)*F(5) - Z =E= 0 ; (LHS = 0)

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
COLUMN LISTING      SOLVE TRANSYSTEM USING NLP FROM LINE 85

---- F          FLOW ON A LINK

F(1) (.LO, .L, .UP = 0, 0,  +INF)
     -1       LINKFLOW(1)
     (0.1)    OBJSYSTEM
F(2) (.LO, .L, .UP = 0, 0,  +INF)
     -1       LINKFLOW(2)
     (0.1)    OBJSYSTEM
F(3) (.LO, .L, .UP = 0, 0,  +INF)
     -1       LINKFLOW(3)
     (0.0857) OBJSYSTEM
REMAINING 2 ENTRIES SKIPPED

---- H          FLOW ON A PATH

H(P1)
            (.LO, .L, .UP = 0, 0,  +INF)
      1       DEMAND(A,B)
      1       LINKFLOW(1)
      1       LINKFLOW(2)
H(P2)
            (.LO, .L, .UP = 0, 0,  +INF)
      1       DEMAND(A,B)
      1       LINKFLOW(3)
      1       LINKFLOW(4)
H(P3)
            (.LO, .L, .UP = 0, 0,  +INF)
      1       DEMAND(A,B)
      1       LINKFLOW(1)
      1       LINKFLOW(4)
      1       LINKFLOW(5)
REMAINING ENTRY SKIPPED

---- Z          OBJECTIVE FUNCTION

Z
```

```
        (.LO, .L, .UP =  -INF, 0,  +INF)
   -1    OBJSYSTEM

GENERAL  A L G E B R A I C   M O D E L I N G   S Y S T E M
MODEL STATISTICS    SOLVE TRANSYSTEM USING NLP FROM LINE 85

MODEL STATISTICS

BLOCKS OF EQUATIONS     3    SINGLE EQUATIONS        7
BLOCKS OF VARIABLES     3    SINGLE VARIABLES       10
NON ZERO ELEMENTS      25    NON LINEAR N-Z          5
DERIVATIVE POOL         9    CONSTANT POOL          10
CODE LENGTH           161

GENERATION TIME    =     0.054 MINUTES


EXECUTION TIME     =     0.109 MINUTES
```

```
GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
SOLUTION REPORT    SOLVE TRANSYSTEM USING NLP FROM LINE 85


         S O L V E    S U M M A R Y

   MODEL   TRANSYSTEM        OBJECTIVE  Z
   TYPE    NLP               DIRECTION  MINIMIZE
   SOLVER  MINOS5            FROM LINE  85

**** SOLVER STATUS     1 NORMAL COMPLETION
**** MODEL STATUS      2 LOCALLY OPTIMAL
**** OBJECTIVE VALUE          281.8982

 RESOURCE USAGE, LIMIT         0.069    1000.000
 ITERATION COUNT, LIMIT        4        1000
 EVALUATION ERRORS             0        0

   M I N O S    5.2    (Mar 1988)
   = = = = =

   B. A. Murtagh, University of New South Wales
     and
   P. E. Gill,  W. Murray,  M. A. Saunders and M. H. Wright
   Systems Optimization Laboratory, Stanford University.


 WORK SPACE NEEDED (ESTIMATE) --     654 WORDS.
 WORK SPACE AVAILABLE         --    8100 WORDS.

 EXIT -- OPTIMAL SOLUTION FOUND
 MAJOR ITNS, LIMIT              1      50
 FUNOBJ, FUNCON CALLS          16       0
 SUPERBASICS                    1
 INTERPRETER USAGE            .00
 NORM RG / NORM PI      5.301E-10


---- EQU DEMAND

      LOWER      LEVEL      UPPER    MARGINAL

A.B  1500.000  1500.000  1500.000     0.205


---- EQU LINKFLOW    FLOW ON EACH LINK

    LOWER     LEVEL     UPPER    MARGINAL

1     .         .         .       -0.103
2     .         .         .       -0.103
3     .         .         .       -0.103
4     .         .         .       -0.103
```

```
5    .     .     .    -0.029

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
SOLUTION REPORT    SOLVE TRANSYSTEM USING NLP FROM LINE 85

                LOWER    LEVEL    UPPER   MARGINAL

---- EQU OBJSYSTEM      .       .       .     -1.000

 OBJSYSTEM   OBJECTIVE FUNCT. UNDER SYSTEM EQUILIBRIUM


---- VAR F         FLOW ON A LINK

   LOWER    LEVEL    UPPER   MARGINAL

1    .    641.986   +INF      .
2    .    641.986   +INF      .
3    .    858.014   +INF      .
4    .    858.014   +INF      .
5    .       .      +INF      .

---- VAR H         FLOW ON A PATH

    LOWER    LEVEL    UPPER   MARGINAL

P1    .    641.986   +INF      .
P2    .    858.014   +INF     EPS
P3    .       .      +INF    0.029
P4    .       .      +INF    0.029

                LOWER    LEVEL    UPPER   MARGINAL

---- VAR Z          -INF   281.898   +INF      .

 Z        OBJECTIVE FUNCTION


**** REPORT SUMMARY :       0    NONOPT
                      0 INFEASIBLE
                      0  UNBOUNDED
                      0    ERRORS

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
EQUATION LISTING   SOLVE TRANUSER USING DNLP FROM LINE 86


---- DEMAND      =E=
DEMAND(A,B)..  H(P1) + H(P2) + H(P3) + H(P4) =E= 1500 ; (LHS = 1500)

---- LINKFLOW    =E=  FLOW ON EACH LINK
LINKFLOW(1)..  - F(1) + H(P1) + H(P3) =E= 0 ; (LHS = 0)
LINKFLOW(2)..  - F(2) + H(P1) + H(P4) =E= 0 ; (LHS = 0)
```

```
LINKFLOW(3).. - F(3) + H(P2) + H(P4) =E= 0 ; (LHS = 0)
REMAINING 2 ENTRIES SKIPPED

---- OBJUSER    =E= OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM

OBJUSER.. (0.1005)*F(1) + (0.1005)*F(2) + (0.0891)*F(3) +
(0.0891)*F(4)+(0.0286)*F(5) - Z =E= 0 ; (LHS = -5.1303 ***)

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
COLUMN LISTING    SOLVE TRANUSER USING DNLP FROM LINE 86


---- F       FLOW ON A LINK
F(1)
        (.LO, .L, .UP = 0, 641.9861, +INF)
   -1    LINKFLOW(1)
   (0.1005) OBJUSER
F(2)
        (.LO, .L, .UP = 0, 641.9861, +INF)
   -1    LINKFLOW(2)
   (0.1005) OBJUSER
F(3)
        (.LO, .L, .UP = 0, 858.0139, +INF)
   -1    LINKFLOW(3)
   (0.0891) OBJUSER
REMAINING 2 ENTRIES SKIPPED

---- H       FLOW ON A PATH
H(P1)
        (.LO, .L, .UP = 0, 641.9861, +INF)
    1    DEMAND(A,B)
    1    LINKFLOW(1)
    1    LINKFLOW(2)
H(P2)
        (.LO, .L, .UP = 0, 858.0139, +INF)
    1    DEMAND(A,B)
    1    LINKFLOW(3)
    1    LINKFLOW(4)
H(P3)
        (.LO, .L, .UP = 0, 0, +INF)
    1    DEMAND(A,B)
    1    LINKFLOW(1)
    1    LINKFLOW(4)
    1    LINKFLOW(5)
REMAINING ENTRY SKIPPED

---- Z       OBJECTIVE FUNCTION
Z
        (.LO, .L, .UP = -INF, 281.8982, +INF)
   -1    OBJUSER

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
MODEL STATISTICS    SOLVE TRANUSER USING DNLP FROM LINE 86
```

MODEL STATISTICS

| | | | |
|---|---|---|---|
| BLOCKS OF EQUATIONS | 3 | SINGLE EQUATIONS | 7 |
| BLOCKS OF VARIABLES | 3 | SINGLE VARIABLES | 10 |
| NON ZERO ELEMENTS | 25 | NON LINEAR N-Z | 5 |
| DERIVATIVE POOL | 9 | CONSTANT POOL | 8 |
| CODE LENGTH | 176 | | |

GENERATION TIME = 0.057 MINUTES

EXECUTION TIME = 0.167 MINUTES

GENERAL A L G E B R A I C M O D E L I N G S Y S T E M
SOLUTION REPORT SOLVE TRANUSER USING DNLP FROM LINE 86

S O L V E S U M M A R Y

```
MODEL   TRANUSER        OBJECTIVE  Z
TYPE    DNLP            DIRECTION  MINIMIZE
SOLVER  MINOS5          FROM LINE  86

**** SOLVER STATUS     1 NORMAL COMPLETION
**** MODEL STATUS      2 LOCALLY OPTIMAL
**** OBJECTIVE VALUE           271.8410

RESOURCE USAGE, LIMIT        0.069     1000.000
ITERATION COUNT, LIMIT       2         1000
EVALUATION ERRORS            0         0

M I N O S   5.2   (Mar 1988)
= = = = =

B. A. Murtagh, University of New South Wales
 and
P. E. Gill,  W. Murray,  M. A. Saunders and M. H. Wright
Systems Optimization Laboratory, Stanford University.

WORK SPACE NEEDED (ESTIMATE) --     660 WORDS.
WORK SPACE AVAILABLE         --    8100 WORDS.

EXIT -- OPTIMAL SOLUTION FOUND
MAJOR ITNS, LIMIT               1     50
FUNOBJ, FUNCON CALLS           10      0
SUPERBASICS                     1
INTERPRETER USAGE             .00
NORM RG / NORM PI       2.403E-08

---- EQU DEMAND

       LOWER     LEVEL     UPPER    MARGINAL

A.B  1500.000  1500.000  1500.000     0.200

---- EQU LINKFLOW    FLOW ON EACH LINK

       LOWER     LEVEL     UPPER    MARGINAL

1        .         .         .       -0.100
2        .         .         .       -0.100
3        .         .         .       -0.100
4        .         .         .       -0.100
5        .         .         .       -0.029
```

GENERAL A L G E B R A I C M O D E L I N G S Y S T E M

```
SOLUTION REPORT     SOLVE TRANUSER USING DNLP FROM LINE 86

                LOWER    LEVEL    UPPER    MARGINAL

---- EQU OBJUSER        .        .        .      -1.000

 OBJUSER    OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM

---- VAR F         FLOW ON A LINK

    LOWER    LEVEL     UPPER    MARGINAL

1    .    267.644    +INF      .
2    .    267.644    +INF      .
3    .    1232.356   +INF      .
4    .    1232.356   +INF      .
5    .       .       +INF      .

---- VAR H         FLOW ON A PATH

    LOWER    LEVEL     UPPER    MARGINAL

P1   .    267.644    +INF      .
P2   .    1232.356   +INF     EPS
P3   .       .       +INF     0.029
P4   .       .       +INF     0.029

                LOWER    LEVEL    UPPER    MARGINAL

---- VAR Z          -INF    271.841    +INF      .

 Z        OBJECTIVE FUNCTION


**** REPORT SUMMARY :       0    NONOPT
                            0 INFEASIBLE
                            0 UNBOUNDED
                            0   ERRORS

**** FILE SUMMARY

INPUT   C:\GAMS205\GAMSDATA\BECKMAN.GMS
OUTPUT  C:\GAMS205\GAMSDATA\BECKMAN.LST

EXECUTION TIME     =      0.068 MINUTES
```

**THE GAMS MODEL OUTPUT BASED ON THE AUTHOR'S FORMULATION**

```
_GAMS 2.05/S PC AT/XT                    92/06/22 15:02:41
GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
C O M P I L A T I O N

SETS
I    ORIGINS /A/
J    DESTINATIONS /B/
L     LINKS /1*5/
P     PATHS /P1*P4/ ;

TABLE VOLUME(I,J,*) DEMAND BETWEEN ORIGINS AND DESTINATIONS
        TRIPS
A.B    1500  ;

PARAMETER CAP(L) CAPACITY OF LINKS
/1 1500
 2 1500
 3 1200
 4 1200
 5 1200 /;

PARAMETER SPEED(L) FREE FLOW TRAVEL SPEED ON THE LINKS
/1 50
 2 50
 3 35
 4 35
 5 35/;

PARAMETER DISTANCE(L) LENGTH OF EACH LINK
/1 5
 2 5
 3 3
 4 3
 5 1 /;

PARAMETER COEFF(L) VALUE OF COEFFICIENT TO BE USED IN BPR EQUATION
/1 0.15
 2 0.15
 3 0.15
 4 0.15
 5 0.15/;

TABLE ODPATH(I,J,P) ORIGIN-DESTINATION PATH MATRIX
     P1 P2 P3 P4
A.B  1  1  1  1;

TABLE LINKPATH(L,P) LINK-PATH MATRIX
  P1 P2 P3 P4
```

```
1 1    1
2 1      1
3   1    1
4   1  1
5      1 1 ;

POSITIVE VARIABLES
F(L)      FLOW ON A LINK
H(P)      FLOW ON A PATH
PCOST(P)  UNIT TRAVEL COST OF EACH PATH
MCOST(I,J) MINIMUM VALUE OF UNIT TRAVEL COST;
```

GENERAL ALGEBRAIC MODELING SYSTEM
COMPILATION

```

VARIABLES
SO       OBJECTIVE FUNCTION UNDER SYSTEM OPTIMIZATION
UE       OBJECTIVE FUNCTION UNDER USER EQUILIBRIUM;

EQUATIONS
OBJUSER      OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM, IN
HOURS PER PATH
OBJSYSTEM    OBJECTIVE FUNCT. UNDER SYSTEM EQUILIBRIUM, IN
USER-HOURS
DEMAND(I,J)  TRAVEL DEMAND FORM ORIGINS TO DESTINATIONS
LINKFLOW(L)  FLOW ON EACH LINK
PATHCOST(P)  UNIT COST ON EACH PATH
MINPATH(I,J) FIND MINIMUM UNIT COST AMONG THE PATHS;

DEMAND(I,J) .. SUM(P $ODPATH(I,J,P), H(P)) =E= VOLUME(I,J,'TRIPS');

LINKFLOW(L) .. SUM(P $LINKPATH(L,P), H(P)) =E= F(L);

PATHCOST(P) .. SUM(L $LINKPATH(L,P), (DISTANCE(L)/SPEED(L))*(1 +
COEFF(L)*POWER((F(L)/CAP(L)), 4))) =E= PCOST(P);

MINPATH(I,J) .. SMIN(P $ODPATH(I,J,P), PCOST(P)) =E= MCOST(I,J) ;

OBJSYSTEM .. SUM(L, F(L)*(DISTANCE(L)/SPEED(L))*(1 +
COEFF(L)*POWER((F(L)/CAP(L)),4))) =E= SO;

OBJUSER .. SUM((I,J,P),H(P)*(PCOST(P) - MCOST(I,J))) =E= UE;

MODEL SYSTEMOPT
/DEMAND,LINKFLOW,PATHCOST,OBJSYSTEM/;

MODEL USEREQUIL
/DEMAND,LINKFLOW,PATHCOST,MINPATH,OBJUSER/;

SOLVE SYSTEMOPT USING NLP MINIMIZING SO;
SOLVE USEREQUIL USING DNLP MINIMIZING UE;
```

```
GENERAL  A L G E B R A I C   M O D E L I N G   S Y S T E M
SYMBOL LISTING

SYMBOL     TYPE   REFERENCES

CAP        PARAM  DECLARED     11  DEFINED    12   REF     73
                  77
COEFF      PARAM  DECLARED     32  DEFINED    33   REF     73
                  77
DEMAND     EQU    DECLARED     64  DEFINED    69  IMPL-ASN    87
                  88   REF    82    85
DISTANCE   PARAM  DECLARED     25  DEFINED    26   REF     73
                  77
F          VAR    DECLARED     52 IMPL-ASN    87    88
                  REF    71    73   2*77
H          VAR    DECLARED     53 IMPL-ASN    87    88
                  REF    69    71    79
I          SET    DECLARED      2  DEFINED     2   REF      7
                  39    55    64    67   2*69   2*75
                  79  CONTROL    69    75    79
IN         EQU    DECLARED     62   REF     63
J          SET    DECLARED      3  DEFINED     3   REF      7
                  39    55    64    67   2*69   2*75
                  79  CONTROL    69    75    79
L          SET    DECLARED      4  DEFINED     4   REF     11
                  18    25    32    43    52    65
                2*71  6*73  6*77  CONTROL    71    73
                  77
LINKFLOW   EQU    DECLARED     65  DEFINED    71  IMPL-ASN    87
                  88   REF    82    85
LINKPATH   PARAM  DECLARED     43  DEFINED    43   REF     71
                  73
MCOST      VAR    DECLARED     55 IMPL-ASN    88   REF     75
                  79
MINPATH    EQU    DECLARED     67  DEFINED    75  IMPL-ASN    88
                  REF    85
OBJSYSTEM  EQU    DECLARED     63  DEFINED    77  IMPL-ASN    87
                  REF    82
OBJUSER    EQU    DECLARED     62  DEFINED    79  IMPL-ASN    88
                  REF    85
ODPATH     PARAM  DECLARED     39  DEFINED    39   REF     69
                  75
P          SET    DECLARED      5  DEFINED     5   REF     39
                  43    53    54    66   2*69   2*71
                2*73  2*75  2*79  CONTROL    69    71
                  73    75    79
PATHCOST   EQU    DECLARED     66  DEFINED    73  IMPL-ASN    87
                  88   REF    82    85
PCOST      VAR    DECLARED     54 IMPL-ASN    87    88
                  REF    73    75    79
POWER      FUNCT  REF     73    77
SO         VAR    DECLARED     58 IMPL-ASN    87   REF     77
                  87
```

```
SPEED      PARAM  DECLARED     18  DEFINED    19    REF    73
              77
SYSTEMOPT  MODEL  DECLARED     81  DEFINED    82    REF    87
UE         VAR    DECLARED     59  IMPL-ASN   88    REF    79
              88
GENERAL  A L G E B R A I C   M O D E L I N G   S Y S T E M
SYMBOL LISTING

SYMBOL     TYPE   REFERENCES

USEREQUIL  MODEL  DECLARED     84  DEFINED    85    REF    88
VOLUME     PARAM  DECLARED      7  DEFINED     7    REF    69
```

SETS

```
I        ORIGINS
J        DESTINATIONS
L        LINKS
P        PATHS
```

PARAMETERS

```
CAP        CAPACITY OF LINKS
COEFF      VALUE OF COEFFICIENT TO BE USED IN BPR EQUATION
DISTANCE   LENGTH OF EACH LINK
LINKPATH   LINK-PATH MATRIX
ODPATH     ORIGIN-DESTINATION PATH MATRIX
SPEED      FREE FLOW TRAVEL SPEED ON THE LINKS
VOLUME     DEMAND BETWEEN ORIGINS AND DESTINATIONS
```

VARIABLES

```
F        FLOW ON A LINK
H        FLOW ON A PATH
MCOST    MINIMUM VALUE OF UNIT TRAVEL COST
PCOST    UNIT TRAVEL COST OF EACH PATH
SO       OBJECTIVE FUNCTION UNDER SYSTEM OPTIMIZATION
UE       OBJECTIVE FUNCTION UNDER USER EQUILIBRIUM
```

EQUATIONS

```
DEMAND     TRAVEL DEMAND FORM ORIGINS TO DESTINATIONS
IN         HOURS PER PATH
LINKFLOW   FLOW ON EACH LINK
MINPATH    FIND MINIMUM UNIT COST AMONG THE PATHS
OBJSYSTEM  OBJECTIVE FUNCT. UNDER SYSTEM EQUILIBRIUM
OBJUSER    OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM
PATHCOST   UNIT COST ON EACH PATH
```

MODELS

SYSTEMOPT

```
USEREQUIL

COMPILATION TIME     =       0.064 MINUTES

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
EQUATION LISTING    SOLVE SYSTEMOPT USING NLP FROM LINE 87


---- DEMAND     =E=  TRAVEL DEMAND FORM ORIGINS TO DESTINATIONS
DEMAND(A,B)..  H(P1) + H(P2) + H(P3) + H(P4) =E= 1500 ; (LHS = 0 ***)

---- LINKFLOW   =E=  FLOW ON EACH LINK
LINKFLOW(1)..  - F(1) + H(P1) + H(P3) =E= 0 ; (LHS = 0)
LINKFLOW(2)..  - F(2) + H(P1) + H(P4) =E= 0 ; (LHS = 0)
LINKFLOW(3)..  - F(3) + H(P2) + H(P4) =E= 0 ; (LHS = 0)
REMAINING 2 ENTRIES SKIPPED

---- PATHCOST   =E=  UNIT COST ON EACH PATH
PATHCOST(P1)..  (0)*F(1) + (0)*F(2) - PCOST(P1) =E= -0.2 ; (LHS = 0 ***)
PATHCOST(P2)..  (0)*F(3) + (0)*F(4) - PCOST(P2) =E= -0.1714 ; (LHS = 0 ***)
PATHCOST(P3)..  (0)*F(1) + (0)*F(4) + (0)*F(5) - PCOST(P3) =E= -0.2143 ;(LHS =
0 ***)
REMAINING ENTRY SKIPPED

---- OBJSYSTEM  =E=  OBJECTIVE FUNCT. UNDER SYSTEM EQUILIBRIUM

OBJSYSTEM..  (0.1)*F(1) + (0.1)*F(2) + (0.0857)*F(3) + (0.0857)*F(4)+
(0.0286)*F(5) - SO =E= 0 ; (LHS = 0)

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
COLUMN LISTING      SOLVE SYSTEMOPT USING NLP FROM LINE 87

---- F          FLOW ON A LINK
F(1)
          (.LO, .L, .UP = 0, 0,  +INF)
     -1     LINKFLOW(1)
     (0)    PATHCOST(P1)
     (0)    PATHCOST(P3)
     (0.1)  OBJSYSTEM
F(2)
          (.LO, .L, .UP = 0, 0,  +INF)
     -1     LINKFLOW(2)
     (0)    PATHCOST(P1)
     (0)    PATHCOST(P4)
     (0.1)  OBJSYSTEM
F(3)
          (.LO, .L, .UP = 0, 0,  +INF)
     -1     LINKFLOW(3)
     (0)    PATHCOST(P2)
     (0)    PATHCOST(P4)
     (0.0857) OBJSYSTEM
REMAINING 2 ENTRIES SKIPPED
```

```
---- H       FLOW ON A PATH
H(P1)
         (.LO, .L, .UP = 0, 0, +INF)
     1    DEMAND(A,B)
     1    LINKFLOW(1)
     1    LINKFLOW(2)
H(P2)
         (.LO, .L, .UP = 0, 0, +INF)
     1    DEMAND(A,B)
     1    LINKFLOW(3)
     1    LINKFLOW(4)
H(P3)
         (.LO, .L, .UP = 0, 0, +INF)
     1    DEMAND(A,B)
     1    LINKFLOW(1)
     1    LINKFLOW(4)
     1    LINKFLOW(5)
REMAINING ENTRY SKIPPED

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
COLUMN LISTING     SOLVE SYSTEMOPT USING NLP FROM LINE 87

---- PCOST    UNIT TRAVEL COST OF EACH PATH
PCOST(P1)
         (.LO, .L, .UP = 0, 0, +INF)
    -1    PATHCOST(P1)
PCOST(P2)
         (.LO, .L, .UP = 0, 0, +INF)
    -1    PATHCOST(P2)
PCOST(P3)
         (.LO, .L, .UP = 0, 0, +INF)
    -1    PATHCOST(P3)
REMAINING ENTRY SKIPPED

---- SO      OBJECTIVE FUNCTION UNDER SYSTEM OPTIMIZATION

SO
         (.LO, .L, .UP = -INF, 0, +INF)
    -1    OBJSYSTEM

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
MODEL STATISTICS   SOLVE SYSTEMOPT USING NLP FROM LINE 87

MODEL STATISTICS

BLOCKS OF EQUATIONS    4   SINGLE EQUATIONS    11
BLOCKS OF VARIABLES    4   SINGLE VARIABLES    14
NON ZERO ELEMENTS     39   NON LINEAR N-Z      15
DERIVATIVE POOL        9   CONSTANT POOL       10
CODE LENGTH          371

GENERATION TIME   =     0.066 MINUTES
```

EXECUTION TIME = 0.121 MINUTES

GENERAL A L G E B R A I C M O D E L I N G S Y S T E M
SOLUTION REPORT SOLVE SYSTEMOPT USING NLP FROM LINE 87

S O L V E S U M M A R Y

MODEL SYSTEMOPT OBJECTIVE SO
TYPE NLP DIRECTION MINIMIZE
SOLVER MINOS5 FROM LINE 87

**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS 2 LOCALLY OPTIMAL
**** OBJECTIVE VALUE 281.8982

| | | |
|---|---|---|
| RESOURCE USAGE, LIMIT | 0.121 | 1000.000 |
| ITERATION COUNT, LIMIT | 6 | 1000 |
| EVALUATION ERRORS | 0 | 0 |

M I N O S 5.2 (Mar 1988)
= = = = =

B. A. Murtagh, University of New South Wales
and P. E. Gill, W. Murray, M. A. Saunders and M. H.
Wright Systems Optimization Laboratory, Stanford
University.

WORK SPACE NEEDED (ESTIMATE) -- 1004 WORDS.
WORK SPACE AVAILABLE -- 8100 WORDS.

EXIT -- OPTIMAL SOLUTION FOUND

| | | |
|---|---|---|
| MAJOR ITNS, LIMIT | 11 | 50 |
| FUNOBJ, FUNCON CALLS | 27 | 34 |
| SUPERBASICS | 1 | |
| INTERPRETER USAGE | .02 | |
| NORM RG / NORM PI | 7.527E-08 | |

---- EQU DEMAND TRAVEL DEMAND FORM ORIGINS TO DESTINATIONS

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| A.B | 1500.000 | 1500.000 | 1500.000 | 0.205 |

---- EQU LINKFLOW FLOW ON EACH LINK

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| 1 | . | . | . | -0.103 |
| 2 | . | . | . | -0.103 |
| 3 | . | . | . | -0.103 |
| 4 | . | . | . | -0.103 |
| 5 | . | . | . | -0.029 |

GENERAL A L G E B R A I C M O D E L I N G S Y S T E M

SOLUTION REPORT SOLVE SYSTEMOPT USING NLP FROM LINE 87

---- EQU PATHCOST UNIT COST ON EACH PATH

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| P1 | -0.200 | -0.200 | -0.200 | EPS |
| P2 | -0.171 | -0.171 | -0.171 | EPS |
| P3 | -0.214 | -0.214 | -0.214 | EPS |
| P4 | -0.214 | -0.214 | -0.214 | EPS |

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| ---- EQU OBJSYSTEM | . | . | . | -1.000 |

OBJSYSTEM OBJECTIVE FUNCT. UNDER SYSTEM EQUILIBRIUM

---- VAR F FLOW ON A LINK

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| 1 | . | 641.986 | +INF | . |
| 2 | . | 641.986 | +INF | . |
| 3 | . | 858.014 | +INF | . |
| 4 | . | 858.014 | +INF | . |
| 5 | . | . | +INF | . |

---- VAR H FLOW ON A PATH

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| P1 | . | 641.986 | +INF | . |
| P2 | . | 858.014 | +INF | EPS |
| P3 | . | . | +INF | 0.029 |
| P4 | . | . | +INF | 0.029 |

---- VAR PCOST UNIT TRAVEL COST OF EACH PATH

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| P1 | . | 0.201 | +INF | . |
| P2 | . | 0.178 | +INF | . |
| P3 | . | 0.218 | +INF | . |
| P4 | . | 0.218 | +INF | . |

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| ---- VAR SO | -INF | 281.898 | +INF | . |

SO OBJECTIVE FUNCTION UNDER SYSTEM OPTIMIZATION
GENERAL A L G E B R A I C M O D E L I N G S Y S T E M
SOLUTION REPORT SOLVE SYSTEMOPT USING NLP FROM LINE 87

```
**** REPORT SUMMARY :      0     NONOPT
                         0 INFEASIBLE
                         0  UNBOUNDED
                         0     ERRORS

GENERAL   A L G E B R A I C   M O D E L I N G   S Y S T E M
EQUATION LISTING    SOLVE USEREQUIL USING DNLP FROM LINE 88

---- DEMAND      =E=  TRAVEL DEMAND FORM ORIGINS TO DESTINATIONS

DEMAND(A,B)..  H(P1) + H(P2) + H(P3) + H(P4) =E= 1500 ; (LHS = 1500)

---- LINKFLOW    =E=  FLOW ON EACH LINK

LINKFLOW(1)..  - F(1) + H(P1) + H(P3) =E= 0 ; (LHS = 0)
LINKFLOW(2)..  - F(2) + H(P1) + H(P4) =E= 0 ; (LHS = 0)
LINKFLOW(3)..  - F(3) + H(P2) + H(P4) =E= 0 ; (LHS = 0)
REMAINING 2 ENTRIES SKIPPED

---- PATHCOST    =E=  UNIT COST ON EACH PATH

PATHCOST(P1)..  (3.1359122E-6)*F(1) + (3.1359122E-6)*F(2) - PCOST(P1) =E= -
0.2 ; (LHS = -0.2)

PATHCOST(P2)..  (1.5666134E-5)*F(3) + (1.5666134E-5)*F(4) - PCOST(P2) =E=  -
0.1714 ; (LHS = -0.1714)

PATHCOST(P3)..  (3.1359122E-6)*F(1) + (1.5666134E-5)*F(4) + (0)*F(5)  -
PCOST(P3) =E= -0.2143 ; (LHS = -0.2143)

REMAINING ENTRY SKIPPED


---- MINPATH     =E=  FIND MINIMUM UNIT COST AMONG THE PATHS

MINPATH(A,B)..  (0)*PCOST(P1) + (1)*PCOST(P2) + (0)*PCOST(P3) +
(0)*PCOST(P4)  - MCOST(A,B) =E= 0 ; (LHS = 0.1781 ***)


---- OBJUSER     =E=  OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM

OBJUSER..  (0.201)*H(P1) + (0.1781)*H(P2) + (0.2181)*H(P3) + (0.2181)*H(P4) +
(641.9865)*PCOST(P1) + (858.0135)* PCOST(P2) + (0)*PCOST(P3) +
(0)*PCOST(P4) - (1500)* MCOST(A,B) - UE =E= 0 ; (LHS = 281.8982 ***)

GENERAL   A L G E B R A I C   M O D E L I N G   S Y S T E M
EQUATION LISTING    SOLVE USEREQUIL USING DNLP FROM LINE 88

   OBJUSER     =E=  OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM

GENERAL   A L G E B R A I C   M O D E L I N G   S Y S T E M
COLUMN LISTING      SOLVE USEREQUIL USING DNLP FROM LINE 88
```

```
---- F        FLOW ON A LINK
F(1)
            (.LO, .L, .UP = 0, 641.9865,  +INF)
    -1      LINKFLOW(1)
 (3.1359122E-6) PATHCOST(P1)
 (3.1359122E-6) PATHCOST(P3)

F(2)
            (.LO, .L, .UP = 0, 641.9865,  +INF)
    -1      LINKFLOW(2)
 (3.1359122E-6) PATHCOST(P1)
 (3.1359122E-6) PATHCOST(P4)

F(3)
            (.LO, .L, .UP = 0, 858.0135,  +INF)
    -1      LINKFLOW(3)
 (1.5666134E-5) PATHCOST(P2)
 (1.5666134E-5) PATHCOST(P4)

REMAINING 2 ENTRIES SKIPPED

---- H        FLOW ON A PATH

H(P1)
            (.LO, .L, .UP = 0, 641.9865,  +INF)
     1      DEMAND(A,B)
     1      LINKFLOW(1)
     1      LINKFLOW(2)
    (0.201)  OBJUSER
H(P2)
            (.LO, .L, .UP = 0, 858.0135,  +INF)
     1      DEMAND(A,B)
     1      LINKFLOW(3)
     1      LINKFLOW(4)
    (0.1781) OBJUSER
H(P3)
            (.LO, .L, .UP = 0, 0,  +INF)
     1      DEMAND(A,B)
     1      LINKFLOW(1)
     1      LINKFLOW(4)
     1      LINKFLOW(5)
    (0.2181) OBJUSER
REMAINING ENTRY SKIPPED

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
COLUMN LISTING     SOLVE USEREQUIL USING DNLP FROM LINE 88

---- PCOST     UNIT TRAVEL COST OF EACH PATH

PCOST(P1)
            (.LO, .L, .UP = 0, 0.201,  +INF)
    -1      PATHCOST(P1)
    (0)     MINPATH(A,B)
```

```
   (641.9865) OBJUSER

PCOST(P2)
          (.LO, .L, .UP = 0, 0.1781,  +INF)
    -1     PATHCOST(P2)
    (1)    MINPATH(A,B)
   (858.0135) OBJUSER

PCOST(P3)
          (.LO, .L, .UP = 0, 0.2181,  +INF)
    -1     PATHCOST(P3)
    (0)    MINPATH(A,B)
    (0)    OBJUSER

REMAINING ENTRY SKIPPED

---- MCOST     MINIMUM VALUE OF UNIT TRAVEL COST
MCOST(A,B)
          (.LO, .L, .UP = 0, 0,  +INF)
    -1     MINPATH(A,B)
  (-1500)    OBJUSER


---- UE        OBJECTIVE FUNCTION UNDER USER EQUILIBRIUM
UE
          (.LO, .L, .UP =  -INF, 0,  +INF)
    -1     OBJUSER

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
MODEL STATISTICS   SOLVE USEREQUIL USING DNLP FROM LINE 88

MODEL STATISTICS

BLOCKS OF EQUATIONS     5    SINGLE EQUATIONS      12
BLOCKS OF VARIABLES     5    SINGLE VARIABLES      15
NON ZERO ELEMENTS      48    NON LINEAR N-Z        23
DERIVATIVE POOL        13    CONSTANT POOL         11
CODE LENGTH           324

GENERATION TIME    =     0.075 MINUTES


EXECUTION TIME     =     0.192 MINUTES
```

```
GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
SOLUTION REPORT    SOLVE USEREQUIL USING DNLP FROM LINE 88

        S O L V E    S U M M A R Y

   MODEL  USEREQUIL         OBJECTIVE  UE
   TYPE   DNLP           DIRECTION  MINIMIZE
   SOLVER  MINOS5            FROM LINE  88

**** SOLVER STATUS     1 NORMAL COMPLETION
**** MODEL STATUS     2 LOCALLY OPTIMAL
**** OBJECTIVE VALUE              0.0000

 RESOURCE USAGE, LIMIT          0.274    1000.000
 ITERATION COUNT, LIMIT         11         1000
 EVALUATION ERRORS              0          0

   M I N O S   5.2   (Mar 1988)
   = = = = =

   B. A. Murtagh, University of New South Wales
   and P. E. Gill,  W. Murray,  M. A. Saunders and M.
       H. Wright Systems Optimization Laboratory, Stanford
       University.


 WORK SPACE NEEDED (ESTIMATE) --   1343 WORDS.
 WORK SPACE AVAILABLE         --   8100 WORDS.

 EXIT -- OPTIMAL SOLUTION FOUND
 MAJOR ITNS, LIMIT           5     50
 FUNOBJ, FUNCON CALLS          194    194
 SUPERBASICS                  1
 INTERPRETER USAGE            .11
 NORM RG / NORM PI     4.248E-05

---- EQU DEMAND     TRAVEL DEMAND FORM ORIGINS TO DESTINATIONS

     LOWER     LEVEL     UPPER    MARGINAL

A.B  1500.000  1500.000  1500.000      EPS

---- EQU LINKFLOW    FLOW ON EACH LINK

    LOWER     LEVEL     UPPER    MARGINAL

1     .         .         .         EPS
2     .         .         .         EPS
3     .         .         .        0.012
4     .         .         .        0.012
5     .         .         .         EPS

GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
```

SOLUTION REPORT SOLVE USEREQUIL USING DNLP FROM LINE 88

---- EQU PATHCOST UNIT COST ON EACH PATH

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| P1 | -0.200 | -0.200 | -0.200 | -267.630 |
| P2 | -0.171 | -0.171 | -0.171 | 267.630 |
| P3 | -0.214 | -0.214 | -0.214 | EPS |
| P4 | -0.214 | -0.214 | -0.214 | EPS |

---- EQU MINPATH FIND MINIMUM UNIT COST AMONG THE PATHS

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| A.B | . | . | . | 1500.000 |

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| ---- EQU OBJUSER | . | . | . | -1.000 |

OBJUSER OBJECTIVE FUNCT. UNDER USER EQUILIBRIUM

---- VAR F FLOW ON A LINK

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| 1 | . | 267.630 | +INF | . |
| 2 | . | 267.630 | +INF | . |
| 3 | . | 1232.370 | +INF | . |
| 4 | . | 1232.370 | +INF | . |
| 5 | . | . | +INF | . |

---- VAR H FLOW ON A PATH

| | LOWER | LEVEL | UPPER | MARGINAL | |
|---|---|---|---|---|---|
| P1 | . | 267.630 | +INF | . | |
| P2 | . | 1232.370 | +INF | -0.025 | NOPT |
| P3 | . | . | +INF | 0.016 | |
| P4 | . | . | +INF | 0.016 | |

---- VAR PCOST UNIT TRAVEL COST OF EACH PATH

| | LOWER | LEVEL | UPPER | MARGINAL |
|---|---|---|---|---|
| P1 | . | 0.200 | +INF | -1500.000 |
| P2 | . | 0.200 | +INF | 1500.000 |
| P3 | . | 0.229 | +INF | . |
| P4 | . | 0.229 | +INF | . |

```
GENERAL  A L G E B R A I C  M O D E L I N G  S Y S T E M
SOLUTION REPORT    SOLVE USEREQUIL USING DNLP FROM LINE 88

---- VAR MCOST     MINIMUM VALUE OF UNIT TRAVEL COST

     LOWER   LEVEL   UPPER   MARGINAL

A.B    .      0.200   +INF     .

               LOWER    LEVEL    UPPER    MARGINAL

---- VAR UE          -INF      .      +INF     .

 UE      OBJECTIVE FUNCTION UNDER USER EQUILIBRIUM


**** REPORT SUMMARY :      1    NONOPT ( NOPT)
                     0 INFEASIBLE
                     0  UNBOUNDED
                     0    ERRORS

**** FILE SUMMARY

INPUT   C:\GAMS205\GAMSDATA\5LINKS.GMS
OUTPUT  C:\GAMS205\GAMSDATA\5LINKS.LST

EXECUTION TIME      =      0.071 MINUTES
```

## REFERENCES


Beckman, H.M., MacGuire, and Winsten. 1956. *Studies in the Economies of Transportation*, Yale University Press.

Brooke, A., D. Kendrick, and A. Meeraus. 1988. *GAMS: A User's Guide*.

Comsis. July 1992. *MinUTP Technical User Manual, 92 B-Version*.

Florian, M., R. Chapleau, S. Nguyen, C. Achim, L. James-Lefebvre, S. Garneau, J. Lefebvre, and C. Fisk. 1979. *Validation and Application of EMME: An Equilibrium Based Two-Mode Urban Transportation Planning Method*, Centre for Transportation Research Publication No. 103, University of Montreal.

LeBlanc, L.J., E.K. Morlok, and W.P. Pierskalla. 1975. "An Efficient Approach to Solving the Road Network Equilibrium Traffic Assignment Problem." *Transportation Research*, vol.9, pp. 309-318.

Meyer, M., and E. Miller. 1984. *Urban Transportation Planning: A Decision-Oriented Approach*.

The Urban Analysis Group. 1990. *TRANPLAN: User Manual, Version 7.0*.

U.S. Bureau of Public Roads and U.S. Department of Commerce. 1964. *Traffic Assignment Manual*.

Wardrop, J. G. 1952. "Some Theoretical Aspects of Road Traffic Research." *Proceedings of the Institute of Civil Engineers*, vol. 1, pp. 325-378.