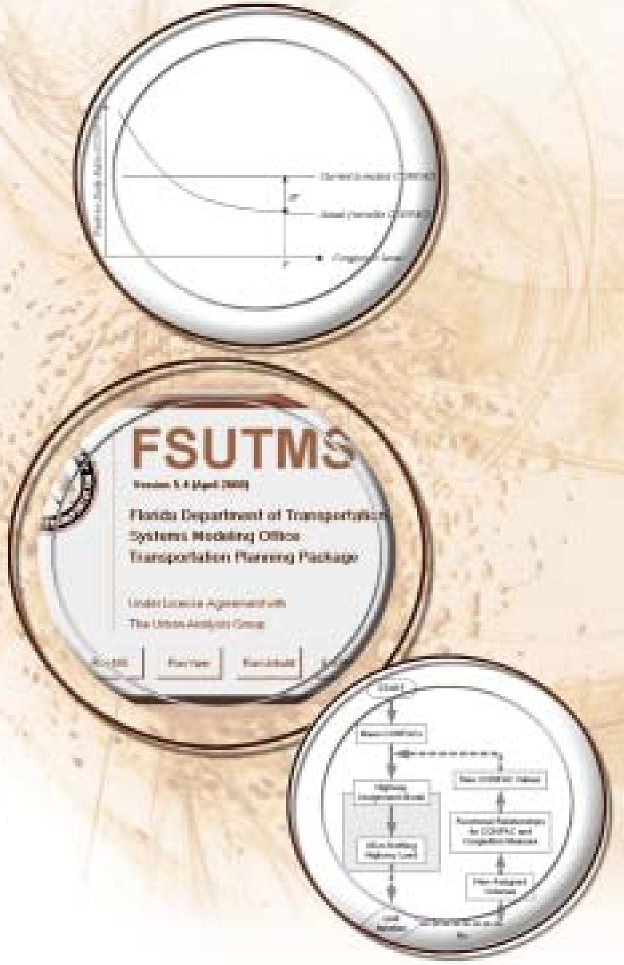
Incorporating Variable Peak-to-Daily Ratios into FSUTMS to Reduce Assignment Errors

Final Report (Volume II)

Contract No. BC-791

September 2003


FSUTMS
Florida Department of Transportatio
Systems Modeling Office
Transportation Planning Package
Under License Agreement with
The Urban Analysis Group

Prepared for

Office of the State Transportation Planner
Systems Planning Office
Florida Department of Transportation

Prepared by

Lehman Center for Transportation Research
Florida International University
and
Department of Civil, Architectural, and Environmental Engineering
University of Miami

Final Report (Volume II)

Incorporating Variable Peak-to-Daily Ratios into FSUTMS to Reduce Assignment Errors

Contract No. BC-791

Prepared for
Office of the State Transportation Planner
Systems Planning Office
State of Florida Department of Transportation
605 Suwannee Street, MS 19
Tallahassee, FL 32399-0450

Prepared by

Albert Gan, Ph.D.
Assistant Professor

Rax Jung, Ph.D.
Research Associate

Min-Tang Li, Ph.D.
Senior Research Associate

Lehman Center for Transportation Research
Department of Civil and Environmental Engineering
Florida International University
10555 W. Flagler Street, EAS 3680
Miami, Florida 33174
Phone: (305) 348-3116; Fax: (305) 348-2802
E-mail: gana@fiu.edu

and

Chang-Jen Lan, Ph.D., P.E.
Assistant Professor
Department of Civil, Architectural, and Environmental Engineering
University of Miami
1251 Memorial Dr., 325 McArthur
Coral Gables, Miami, Florida 33146
Phone: (305) 284-6447; Fax: (305) 284-3492
E-mail: clan@miami.edu

September 2003

Technical Report Documentation Page

1. Report No.

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle

Incorporating Variable Peak-to-Daily Ratios into FSUTMS to Reduce Assignment Errors

5. Report Date

September 2003

6. Performing Organization Code

7. Author(s)

Albert Gan, Rax Jung, Min-Tang Li, and Chang-Jen Lan

8. Performing Organization Report No.

9. Performing Organization Name and Address

Lehman Center for Transportation Research
Florida International University
10555 W. Flagler Street, EAS 3680, Miami, Florida 33174

10. Work Unit No. (TRAIS)

11. Contract or Grant No.

BC-791

12. Sponsoring Agency Name and Address

Office of Research and Development
State of Florida Department of Transportation
605 Suwannee Street, MS 30, Tallahassee, FL 32399-0450

13. Type of Report and Period Covered

Final Report (Volume II)
October 2000–September 2003

14. Sponsoring Agency Code

99700-3596-119

15. Supplementary Notes

Mr. Shi-Chiang Li, AICP, of the Florida Department of Transportation District 4 Office served as the project manager.

16. Abstract

The standard highway assignment model in the Florida Standard Urban Transportation Model Structure (FSUTMS) is based on the equilibrium trip assignment method. The method involves running several iterations of all-or-nothing capacity-restraint assignment with an adjustment for travel time to reflect delays encountered in the associated iteration. This iterative link time adjustment process is accomplished through the widely used Bureau of Public Roads (BPR) volume-delay equation. While the output assignments from traditional forecasting models are usually given in daily traffic, the input capacities are typically given in hourly traffic. A peak-to-daily ratio (CONFAC) is computed as the highest hourly volume of a day divided by the total daily volume. Although the current version of FSUTMS allows the use of different CONFACs for indiviudal facility types, it still assumes that all roadways belonging to a specific type have the same peaking characteristics. The objective of this research is to improve the accuracy of trip assignments in FSUTMS by: (1) calibrating the relationships between CONFAC and a congestion measure for each facility type, using traffic count data, and (2) applying the calibrated relationships in FSUTMS' trip assignment. It was found that the output accuracy between constant and variable CONFAC assignments was not significantly different and the desired improvement in assignment results of the variable CONFAC model was not empirically evident.

17. Key Words

Demand Forecasting, Peak-to-Daily Ratio, Volume-Delay Functions, Traffic Count, Model Calibration

18. Distribution Statement

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 118	22. Price

ACKNOWLEDGEMENTS

This research was funded by the Research Center of the Florida Department of Transportation (FDOT). The authors wish to thank the project manager, Mr. Shi-Chiang Li, AICP, of the Office of Planning and Environmental Management at the FDOT District 4 Office, for his guidance and advice throughout the course of the research. The following individuals are gratefully acknowledged for their assistance in various capacities: Dr. Ling Chen, Associate Professor of Statistics at the Florida International University (FIU), assisted in the statistical modeling; Dr. William T. Olsen, P.E. of the FDOT's Turnpike Enterprise District was especially kind and helpful in providing insights on the toll facilities model; and Dr. Jill Strube of the FIU Institute of Government and Metropolitan Center reviewed the entire draft and provided helpful comments.

DISCLAIMER

The opinions, findings and conclusions expressed in this publication are those of the authors and not necessarily those of the State of Florida Department of Transportation.

TABLE OF CONTENTS

EXECUTIVE SUMMARY

Like other travel demand forecasting models, the Florida Standard Urban Transportation Model Structure (FSUTMS) is not immune to under- and over-assignments of trips to individual links. Although engineers and researchers recognize that demand models are inherently susceptible to assignment errors, various improvements may have the potential to minimize these errors. This research attempts to minimize assignment errors in FSUTMS by better estimating the travel impedances used in its assignment procedure.

The standard highway assignment model in FSUTMS is based on the equilibrium trip assignment method. This method involves running several iterations of all-or-nothing capacity-restraint assignment with an adjustment of travel time to reflect delays encountered in the associated iteration. The first iteration loads trips to the network based on free-flow travel times. All subsequent iterations of assignment utilize the network travel times from the previous iteration as the travel times for the current iteration. The iterative link time adjustment process is accomplished through the widely used Bureau of Public Roads (BPR) volume-delay equation. While the output volumes from traditional forecasting models are usually given in daily traffic, the input capacities are typically given in hourly traffic. Therefore, it is important to convert the corresponding hourly capacity output to its daily equivalent by dividing the hourly capacity by a factor called the peak-to-daily ratio. A peak-to-daily ratio is computed as the highest hourly volume of a day divided by the total daily volume. This ratio, referred to as CONFAC in FSUTMS, carries the same effect on the v/c ratio as a time-of-day factor, which converts a daily volume to its hourly volume equivalent.

Prior to FSUTMS version 5.3, a single CONFAC, typically 0.1, was used for all facility types. This was deemed insufficient to address the different peaking characteristics associated with different types of facilities, and is believed to be one of the culprits of assignment errors. Consequently, FSUTMS version 5.3 adopted a “multiple-CONFAC” structure that allows the use of different CONFACs for different types of facility. This method still assumes that all roadways belonging to a specific type have the same peaking characteristics. While several studies have indicated that the peak-to-daily ratio is a decreasing function of the level of congestion, a constant ratio is used in the current version of FSUTMS. This research attempts to improve the accuracy of trip assignments in FSUTMS by calibrating the relationships between CONFAC and a congestion measure for each facility type, and then applying these calibrated relationships in FSUTMS’ trip assignment.

Traffic counts data from across the state of Florida were used to calibrate the CONFAC functions. The functional relationships clearly demonstrate that the CONFAC is a function of congestion level and facility type. Overall, the functions used to calibrate the CONFAC models are logical and the differences in functional relationships for different facility under different congest levels suggest that the assignment results may be improved by making the CONFAC sensitive to these variables. A major effort was then carried out to implement these calibrated functions in FSUTMS.

Under an ideal, commercial-free environment, the research team would have ready access to the source code for FSUTMS’ highway assignment module and be able to implement the calibrated

functions in the module with ease. However, knowing the difficulty of gaining access to the source code and the potential complications associated with it, the research team decided to incorporate the functions in FSUTMS without involving any modification to any part of the FSUTMS source code. This was accomplished by taking advantage of the iterative process of the FSUTMS assignment procedure. Essentially, this procedure runs the assignment procedure for only one iteration, computes the CONFAC value corresponding to each assigned link volume, updates the link travel times, and performs a new iteration based on the new travel times until the procedure converges.

To be able to conclude that any difference between before and after the implementation of the CONFAC functions is attributed to the use of variable CONFACs, the procedure developed must be shown to adequately replicate the results from the HASSIGN module. The basis for verifying the modified procedure is its ability to reproduce the assignment results from the original FSUTMS model for the exact same network and conditions. Based on the assignment results from the fictitious FTOWN network (under both uncongested and congested conditions) and the real-world Palm Beach network, it was concluded that the procedure adequately replicates the results from the HASSIGN module. The research then proceeded to use the procedure to compare assignment results before and after implementation of the variable CONFACs.

The assignments from each CONFAC option (constant versus variable) were compared against the ground counts for three selected networks for Miami-Dade, Broward, and Palm Beach counties based on the assigned VHT to count VHT ratios, assigned VMT to count VMT ratios, screenline volume-count ratio, and percent root mean square error. It was found that the accuracy between constant and variable CONFAC assignments was not significantly different; the desired improvement in assignment results of the variable CONFAC model is not empirically evident. This result was somewhat expected given the existence of relatively flat CONFAC curves over a wide range of congestion levels in the calibrated models (see Figure 3.7). In other words, in most cases, there was not a significant difference between a constant CONFAC and a CONFAC generated from the calibrated CONFAC functions.

While the results from this research did not empirically support the desired outcome, the concept of applying variable CONFACs should remain encouraging and should not be discounted. It must be recognized that many other factors beyond the scope and control of this study could contribute to its results, including the accuracy of the traffic counts for calibration, the accuracy of ground counts used in model evaluation, the appropriateness of the BPR equation and the associated parameters, etc. It is recommended that future research focus on the following areas:

- The accuracy of the variable CONFAC model might be improved by recalibrating the parameters used in the BPR function. The calibration is an involved task, which is beyond the scope of this research.
- The use of the developed methodology warrants the application of different volume-delay functions. If the other volume-delay functions could be applied, the accuracy might improve. Additional research is needed to determine if the results are generally true for various transportation networks.
- Although the constant CONFAC model can reasonably replicate the FSUTMS assignment results, it is necessary to evaluate the direct integration of the variable CONFAC models into

the FSUTMS source code. Accordingly, more definitive conclusions about the potential advantages of variable CONFAC models can then be drawn.

- The impact of using other assignment algorithm rather than the current equilibrium assignment also needs to be explored.

CHAPTER 1
INTRODUCTION

1.1. Motivation

Like other travel demand forecasting models, the Florida Standard Urban Transportation Model Structure (FSUTMS) is not immune to under- and over-assignments of trips to individual links. Acceptable percentages of these assignment errors for different facility types are given in *Urban Transportation Planning Model Update-Phase II, Task C, Develop Standard Distribution and Assignment Models* (FDOT, 1981). In fact, some of the errors can be so large that the growth in trips for the forecast year is not sufficient to overcome the magnitude of trip under-assignments, resulting in forecast trips that are lower than those of the base year for some links. This often leaves some modelers to wonder if their models were correct (Li *et al.*, 1999). Although it is recognized that demand models are inherently susceptible to assignment errors, various improvements may have the potential to minimize these assignment errors and obtain more accurate trip forecasts. This research attempts to minimize assignment errors in FSUTMS by better estimating the travel impedances used in its assignment procedure.

1.2. FSUTMS' Highway Assignment

The standard highway assignment model in FSUTMS is based on the equilibrium trip assignment method. As shown in Figure 1.1, this method involves running several iterations of all-or-nothing capacity-restraint assignment with an adjustment of travel time to reflect delays encountered in the associated iteration. The first iteration loads trips to the network based on free-flow travel times. All subsequent iterations of assignment utilize the network travel times from the previous iteration as the travel times for the current iteration. The iterative link time adjustment process is accomplished through the widely used Bureau of Public Roads (BPR) volume-delay equation, defined as follows (FDOT, 1981):

$$T_n = T_o \times [1.0 + a(v/c)^b]$$

where T_n = new travel time from the current iteration,
T_o = old travel time from the previous iteration,
a, b = model coefficients, and
v/c = daily volume-to-capacity ratio.

While the output volumes from traditional forecasting models are usually given in daily traffic, the input capacities are typically given in hourly traffic. Therefore, it is important to convert the corresponding hourly capacity to its daily equivalent by dividing the hourly capacity by a factor called the peak-to-daily ratio. A peak-to-daily ratio is computed as the highest hourly volume of a day divided by the total daily volume, as follows:

$$Peak\text{–}to\text{–}Daily\ Ratio = \frac{Highest\ hourly\ volume\ of\ a\ day}{Total\ daily\ volume}$$

This ratio, referred to as CONFAC in FSUTMS, carries the same effect on the v/c ratio as a time-of-day factor, which converts a daily volume to its hourly volume equivalent.

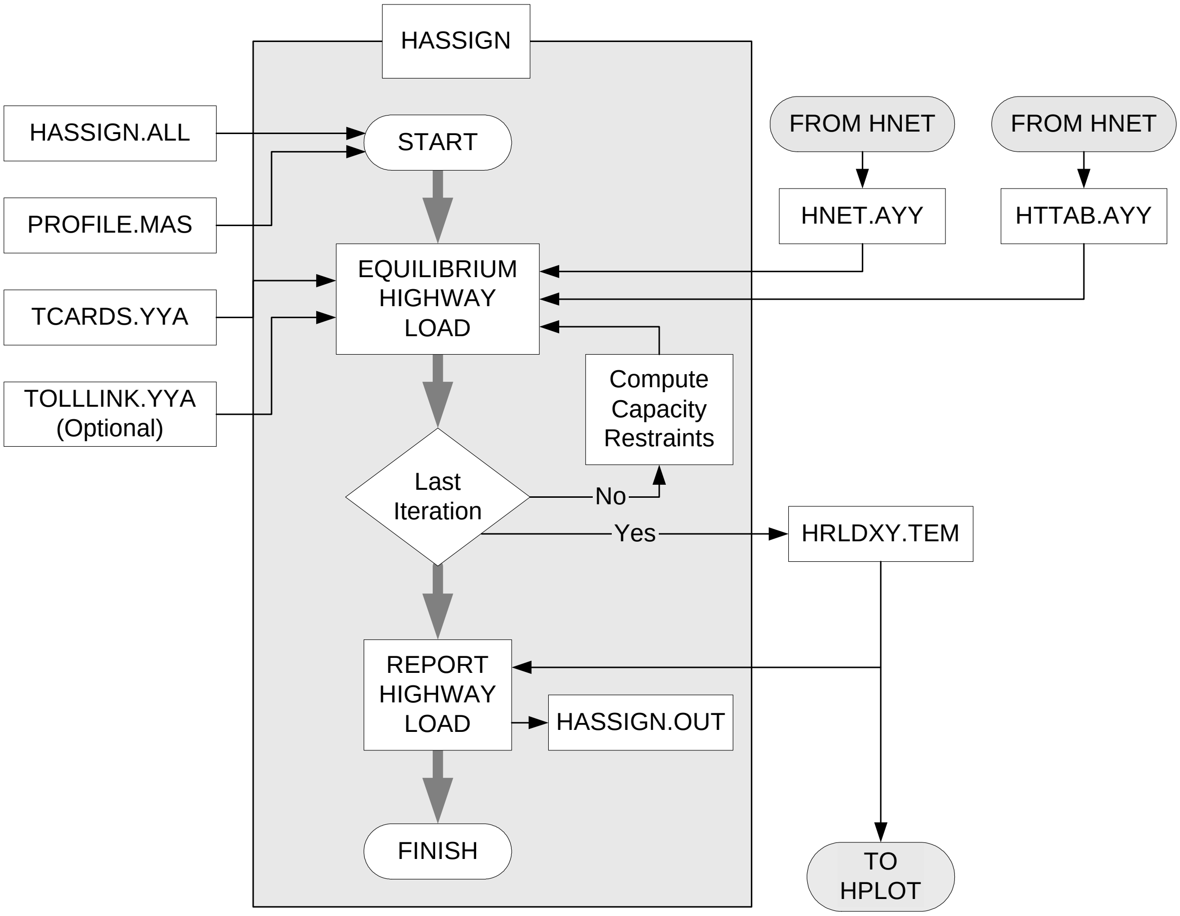


Figure 1.1. FSUTMS' Equilibrium Assignment Method (FDOT, 1981).

1.3. Problem Statement

Prior to FSUTMS version 5.3, a single CONFAC, typically 0.1, was used for all facility types. This was deemed insufficient to address the different peaking characteristics associated with different types of facilities, and is believed to be one of the culprits of assignment errors. Consequently, FSUTMS version 5.3 adopted a "multiple-CONFAC" structure that allows the use of different CONFACs for different types of facility. This method still assumes that all roadways belonging to a specific type have the same peaking characteristics. While several studies (Allaire and Ivan, 1999; Loudon *et al*., 1988; Walters *et al*., 1989) have indicated that the peak-to-daily ratio is a decreasing function of the level of congestion, a constant ratio is used in the current version of FSUTMS.

1.4. Proposed Solution

Figure 1.2 illustrates the current FSUTMS practice and the actual relationship between CONFAC and level of congestion. In this case, the discrepancy, *ΔY,* at congestion level *V*, will cause the link travel times to be overestimated, resulting in trip under-assignments. Clearly, a solution to such assignment errors is to make CONFAC a function of congestion in the "Compute Capacity Restraint" step in Figure 1.1. The functional relationships between CONFAC and level of congestion can be empirically calibrated based on traffic count data and then implemented in FSUTMS' equilibrium assignment. Accordingly, the final assigned volumes obtained in this manner will represent results from applying CONFACs that are not only a function of facility type, but also of congestion level.

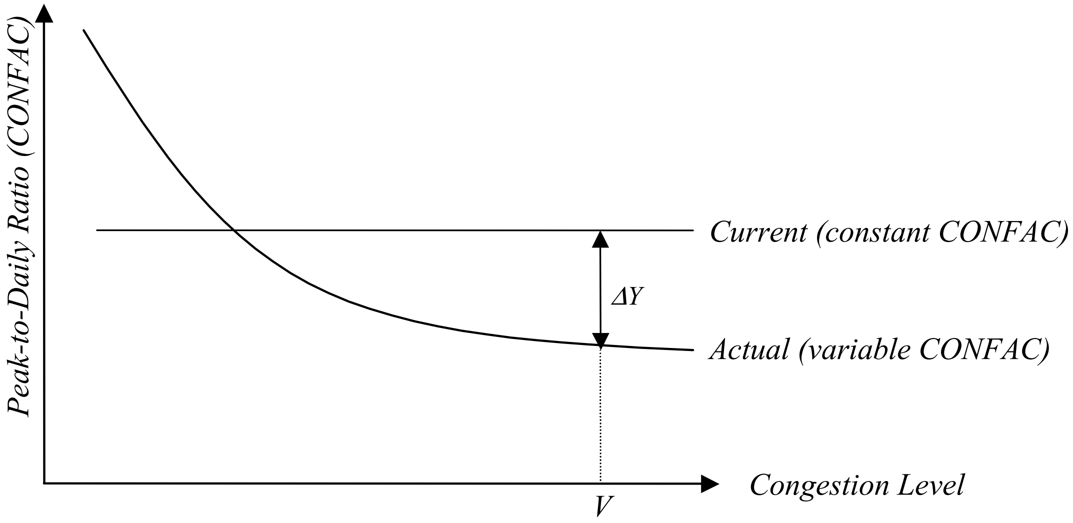


Figure 1.2 General Relationships Between Peak-to-Daily Ratio and Congestion Level.

1.5. Goal of the Study

The goal of this study is to improve the accuracy of trip assignments in FSUTMS by calibrating the relationship between CONFAC and a congestion measure for each facility type, and then applying these calibrated relationships in FSUTMS' trip assignment.

1.6. Report Organization

The rest of this report is organized as follows. Chapter 2 reviews literature on volume-delay functions and the factors affecting these functions. Chapter 3 presents the process of calibrating the CONFAC functions using traffic count data. Chapter 4 documents the process of implementing the calibrated CONFAC functions in FSUTMS. Chapter 5 summarizes and compares results before and after the implementation of the calibrated CONFAC functions. Finally, Chapter 6 summarizes the findings and suggests recommendations for further studies.

CHAPTER 2
VOUME-DELAY FUNCTIONS AND RELATED FACTORS

2.1. Introduction

As discussed in Section 1.2, FSUTMS' equilibrium assignment model uses the Bureau of Public Roads (BPR) volume-delay equation to achieve the effect of capacity restraint. In short, capacity restraint assignment is an iterative process in which the travel impedance is adjusted in terms of a predetermined relationship between the capacity of the link and the volumes assigned to the link. Many functional forms of volume-delay functions have been proposed and used in practice in the past. This chapter provides a review of these volume-delay functions and the factors affecting them. An emphasis is placed on the BPR function since it is the model currently used in FSUTMS.

Unless otherwise stated, the following notation is used throughout the rest of this chapter. The notations denoted by a prime indicate a lane-based measurement.

v = the flow on a link,
T = the travel time per unit distance at flow v,
T_0= the travel time per unit distance at zero flow,
c_p = the practical capacity of a link (LOS C or D),
c_u = the ultimate capacity of a link (LOS E), and
$\alpha, \beta, \gamma, \delta$ = the parameters of the volume-delay function to be estimated from data.

2.2. BPR Function

The BPR volume-delay function, first developed in 1964, has been used extensively in transportation planning studies and in trip assignment algorithms, including in FSUTMS. As given in Section 1.2, the BPR function is defined as:

$$T_n = T_o \times [1.0 + a(v/c_p)^b]$$

where T_n = the new travel time from the current iteration,
T_o = the old travel time from the previous iteration,
a = the coefficient (often set at 0.15)
b = the exponent (often set at 4), and
v/c_p = the ratio of assigned volume-over-practical capacity.

This equation assumes that parameter a is the ratio of travel time per unit distance at practical capacity to that at free flow, and that parameter b determines how fast the curve increases from the free-flow travel time. Uncongested travel time increases by a times $(v/c)^b$, a number which, as the curve in Figure 2.1 shows, grows slowly for v/c ratios less than 1.0 and then rapidly for v/c ratios greater than 1.0. With higher values of b, the onset of congestion effects becomes stronger.

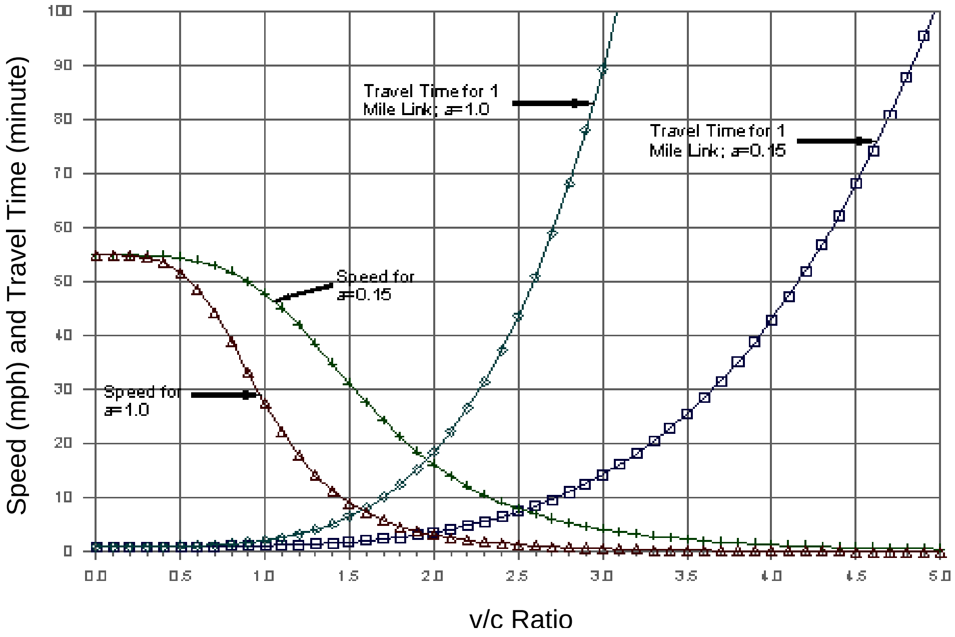


Figure 2.1. Example Travel Times and Speeds From BPR Function (Barton-Aschman and Cambridge Systematics, 1997).

Simplicity and a requirement of minimal data input are two factors that have contributed to the widespread use of the BPR function. The BPR function is also differentiable, which makes it easier to develop efficient routines for finding the equilibrium solution (Skabardonis and Dowling, 1996). However, the BPR function has several shortcomings and other modified functions have been developed in an attempt to improve the performance of the function, which will be discussed in the next two sections.

2.3. Modified Volume-Delay Functions

The following inherent drawbacks are associated with high exponent *b* values of the BPR function (Spiess, 1990):

1. High *b* values reduce the rate of convergence by giving undue penalities to overloaded links during the first few iterations of an equilibrium assignment and can also cause numerical problems, such as overflow conditions and loss of precision.
2. For links with volumes that are far below capacity, the BPR function with high *b* values always yields free-flow times that do not match those of the actual traffic volumes.

To overcome these disadvantages of the BPR function, Spiess (1990) proposed the conical volume-delay function, as defined below:

$$T = T_0\left[2 + \sqrt{\beta^2\left(1 - \frac{v}{c_p}\right)^2 + \alpha^2} - \beta\left(1 - \frac{v}{c_p}\right) - \alpha\right]$$

where $\alpha = \dfrac{2\beta - 1}{2\beta - 2}$, and
$\beta > 1$.

Parameter β corresponds to exponent b of the BPR function. A dramatic improvement in the convergence of the equilibrium assignment was observed when this conical function was used in place of the BPR function in a study for the City of Basel, Switzerland.

Another volume-delay function utilizes a time-dependent form derived from coordinate transformation technique (Akçelik, 1991):

$$T = T_0 + 0.25T_f\left[(x-1) + \sqrt{(x-1)^2 + \frac{8J_A x}{c_u T_f}}\right]$$

where T_f = the flow period (typically one hour),
x = degree of saturation (v/c_u), and
J_A = the delay parameter.

The travel time of Akçelik's equation is determined by the free-flow travel time plus the delay due to queuing. The delay is defined as the average overflow queue divided by capacity. The portion of the equation related to the average overflow queue was developed to take into account the impacts of random variations in the arrival rate. The delay parameter J_A ensures that the equation predicts the desired speed of traffic when demand is equal to capacity. Table 2.1 provides the suggested range of parameter values (Akçelik, 1991). J_A can be computed from the following equation by obtaining the travel time at capacity T_c and substitute $x = 1$:

$$J_A = \frac{2c_u}{T_f}(T_c - T_0)^2$$

Table 2.1. Parameters of Akçelik Function for Different Facility Types (Akçelik, 1991).

Facility Type	Free-Flow Speed, km/h	Capacity per Lane, vphpl	J_A	T_c/T_0
Freeway	120	2000	0.1	1.587
Arterial (uninterrupted)	100	1800	0.2	1.754
Arterial (interrupted)	80	1200	0.4	2.041
Secondary (interrupted)	60	90	0.8	2.272
Secondary (high friction)	40	600	1.6	2.439

Horowitz (1991) updated the parameters used in the BPR function to some of its weaknesses. The coefficient a was calibrated against the speed-volume relationships from the 1985 Highway Capacity Manual (HCM), and the exponent b was determined via nonlinear regression. The updated BPR parameters are shown in Table 2.2 for freeways and arterials. The speeds shown in the table are design speeds of the facility, not the free-flow speeds. Capacities used in the v/c ratio are ultimate capacity rather than practical capacity used in the standard BPR curve. Additionally, the HCM curve has a much lower speed at v/c equals to 1.0 than does the BPR function (Barton-Aschman and Cambridge Systematics, 1997).

Table 2.2. Updated BPR Parameters Using HCM Procedures (Horowitz, 1991).

Parameter	Freeways			Multi-lane		
	70 mph	60 mph	50 mph	70 mph	60 mph	50 mph
a	0.88	0.83	0.56	1.00	0.83	0.71
b	9.80	5.50	3.60	5.40	2.70	2.10

Skabardonis and Dowling (1996) found that the standard BPR curve predicts lower speeds at v/c ratios between 0.8 and 1.0 and higher speeds for over-capacity conditions. They refitted the BPR function to the freeway speed-flow curves and the signalized arterials speed estimation method in the 1994 Highway Capacity Manual (HCM). The updated BPR function was fitted using the ultimate capacity rather than the practical capacity used in the standard BPR function. These authors revised freeway-flow travel time for arterial application in the BRP function as the travel time at the free-flow speed plus the total delay at the signalized intersection. Simulation models were used to develop a hypothetical data set for testing the speed-flow curves against conditions where demand exceeds capacity. Comparisons with field data and simulation results indicated that the updated BPR function provides better accuracy and consistency with the procedures contained in the 1994 HCM; this work yielded parameters of the best-fit updated BPR function as follows:

$a = 0.2$ for freeways, a = 0.05 for arterials; and
$b = 10$.

To account for the 1994 HCM speed-flow relationship, the BPR function used in the San Francisco Metropolitan Transportation Commission (MTC) FCAST Model was also updated. The MTC function is almost identical to the updated BPR curve, except that the coefficient *a* of 0.2 is applied to all facility types rather than just the freeways. However, the higher power exponent (*b*) of the updated BPR function can cause computer overflow and can adversely affect the rate of equilibrium convergence solutions in the traffic assignment process.

Dowling *et al.* (1998) investigated the impact of the standard BPR, updated BPR and Akçelik functions on the performance of equilibrium traffic assignments. The authors compared speed and travel time estimates produced by the three selected functions against simulation results for a freeway and a signalized arterial in the City of Los Angles, California and found that the Akçelik and updated BPR functions had the best performance overall for freeways and arterials. The San Francisco MTC FCAST Model and the Contra Costa Transportation Authority Tri-Valley Model were used to test the effects of the different volume-delay functions on the running times and speed of convergence. These authors concluded that the Akçelik function was as accurate as the updated BPR function and provided more accurate speed estimates than the standard BPR or updated BPR functions. Further, the Akçelik function was also found to significantly improve the rate of convergence to equilibrium assignment.

In addition, Singh's (1999) investigation of the performance tests between the Akçelik and MTC functions show that computing time, equilibrium convergence, and accuracy of prediction for the Akçelik function are approximately the same as the MTC function. However, the Akçelik function has the advantage of predicting the linear impact of congestion on speeds, which results in lower predicted speeds for congested conditions.

Because of the subjective nature of the practical capacity that is not always uniquely defined and is not easily measured, Steenbrink (1974) suggested a volume-delay function similar to the BPR function by replacing practical capacity with ultimate capacity, which is:

$$T = T_0 \times \left(1 + \alpha (v / c_u)^{\beta}\right)$$

The observed data combined with some data generated on the basis of the experience of Dutch traffic engineers were used for the functional calibration. The values of 2.62 and 5 for *α* and *β* for all road types were recommended.

2.4. Other Volume-Delay Functions

A number of other volume-delay functions that were developed independent of the BPR function have been found in the literature. Campbell *et al*. (1959) developed a series of volume-delay functions for use in the Chicago Area Transportation Study (CATS), which involved summing up the accumulative stopped time due to traffic signalization with the free-flow travel time through the section. The accumulative stopped time at any flow was calculated as the mean of maximum and minimum possible stopped times. The mathematical formation of the curves were approximated by two linear sections:

$$\begin{aligned} T &= T_0 && \text{for } \frac{v}{c_u} \le 0.6 \\ T &= T_0 + \alpha\left(\frac{v}{c_u} - 0.6\right) && \text{for } \frac{v}{c_u} > 0.6 \end{aligned}$$

Irwin *et al*. (1961) proposed one of the earlier volume-delay functions for use in an assignment procedure that consists of the two straight-line segments with a similar form to the CATS functions mentioned above:

$$\begin{aligned} T &= T_p + \alpha(v' - c_p') && \text{for } v' < c_p' \\ T &= T_p + \beta(v' - c_p') && \text{for } v' \ge c_p' \end{aligned}$$

where $T_p = T_0 + \alpha c_p'$

This study used different sets of parameters based on the data obtained on streets in Toronto classified by speed limit and estimates of the number of intersections per mile. The authors updated the two-line version to the three-line version in 1962 by introducing a third straight line for the over-saturated region as follows:

$$T = T_p + \alpha(v^{'} - c_p^{'}) \quad \text{for } v^{'} < c_p^{'}$$
$$T = T_p + \beta(v^{'} - c_p^{'}) \quad \text{for } c_p^{'} \le v^{'} \le c_u^{'}$$
$$T = T_u + \gamma(v^{'} - c_u^{'}) \quad \text{for } v^{'} > c_u^{'}$$

where $$T_u = T_p + \beta(c_u^{'} - c_p^{'})$$

The factors for categorizing the street network were further expanded by Branston (1976) to include the presence of different types of transit vehicles to be able to predict transit and vehicle flows. Both techniques were used in a heuristic combined distribution-assignment model that was then applied to the Toronto network. These linear relationships were easily applied in most mathematically assignment procedures and were found to fit the observed data from the results of Irwin *et al.* (1961).

One of the earliest curvilinear volume-delay functions was the exponential curve developed by Smock (1962). The exponential function, which was used in the Detroit Area Transportation Study, has the following form:

$$T = T_0 e^{(v/c_u)}$$

The model was incorporated in a heuristic iterative capacity restraint assignment procedure and was applied to the City of Flint, Michigan. Significantly fewer overloaded assigned volumes were predicted by this model when compared to those predicted by an all-or-nothing assignment procedure.

Based on the desired property that the change in time per unit flow is small when the flow is low, but large when the flow is approaching capacity, Mosher (1963) suggested the following two functions, one logarithmic and another hyperbolic:

Logarithmic function:

$$T = T_0 + \ln(\alpha) - \ln(\alpha - v)$$

Hyperbolic function:

$$T = \beta - \frac{\alpha(T_0 - \beta)}{v - \alpha}$$

where $v \le \alpha$, and α is the saturation flow of a link. To avoid producing infinite travel time, the value of α must be greater than the ultimate capacity (c_u). In addition, there may still remain computational problems because of the possibility of having a predicted flows in the earlier stages of most iterative assignment procedures higher than both c_u and α on some link. This may also result in the subsequent impossibility of handling an infinite travel time. Mosher's volume-delay functions are hence not recommended for use in iterative assignment procedures.

Nevertheless, these two curvilinear functions can be improved by restricting the feasible range to $0 \leq v \leq c_u$. A linear relationship was then used to calculate the travel in the overloaded region. These forms are modified as follows:

Modified logarithmic function:

$$\begin{aligned} T &= T_0 + \beta \ln(\alpha) - \beta \ln(\alpha - v) && \text{for } v \leq c_u \\ T &= T_u + sv && \text{for } v > c_u \end{aligned}$$

where

$$\begin{aligned} T_u &= T_0 + \beta \ln(\alpha) - \beta \ln(\alpha - c_u) \\ s &= \frac{\beta}{\alpha - c_u} \\ \alpha &> c_u \end{aligned}$$

Modified hyperbolic function:

$$\begin{aligned} T &= \beta - \frac{\alpha(T_0 - \beta)}{v - \alpha} && \text{for } v \leq c_u \\ T &= T_u + sv && \text{for } v > c_u \end{aligned}$$

where

$$\begin{aligned} T_u &= \beta - \frac{\alpha(T_0 - \beta)}{c_u - \alpha} \\ s &= \frac{\alpha(T_0 - \beta)}{(c_u - \sigma)^2} \\ \alpha &> c_u, \quad T_0 > \beta \end{aligned}$$

Soltman (1965) proposed another curvilinear function that is similar to Smock's (1962). The function, expressed as follows, was used in the Pittsburgh Area Transportation Study:

$$T = T_0 \times 2^{v/c_p}$$

where $v/c_p \leq 2$. Soltman did not provide information on how this function fits observed flows. Overgaard (1967) proposed the following more general form of the Soltman model:

$$T = T_0 \alpha^{(v/c_p)^\beta}$$

The parameter α is defined as the ratio of the travel time per unit distance at practical capacity to that at zero flow. The value of α was found to vary between 1.0 and 1.7.

Davidson (1966) proposed the following general-purpose volume-delay curve, which he derived from queuing theory:

$$T = T_0 \left[1 + \frac{J_D x}{1 - x} \right]$$

where J_D = the delay parameter, and
x = degree of saturation (v/c_u).

The parameter J_D accounts for the frequency of factors causing delay along a link (e.g., presence of traffic signal). The J_D is assumed to vary with route type and location in a given metropolitan area. Davidson argued that volume-delay functions should restrain the flow to below capacity as the travel time approaches infinity when link flow approaches capacity in this function. Thus, travel times rapidly increase as volume approaches capacity, which makes the estimation of link capacity critical. However, the meaning of delay parameter J_D is not easily understood, and it is difficult to calibrate the function for specific facilities.

Boyce *et al.* (1981) estimated the J_D parameter by fitting the Davidson curve to general travel time versus traffic relationships presented in the 1965 HCM for freeways and arterial streets. Similar values of J_D parameter for both facilities were found despite the difference in characteristics between facility types. The equilibrium traffic assignments based on the BPR and Davidson curves were applied on a regional network of the Chicago Metropolitan area. The two assignments produced comparable results, but the Davidson function produced 9% more assigned vehicle kilometers for the lower-volume class. The Davidson function was found to slow the convergence of traffic assignment algorithms.

Wardrop (1968) formulated a model in terms of overall journey speed in the network instead of on individual links for the central London area. Based on a more realistic hypothesis than that assumed by Campbell's model, Wardrop derived two linear functions for calculating the queuing times at signalized intersections and the running times between signalized intersections individually, as follows:

$$T = \frac{1}{V_r} + nt_q = \frac{1}{\min\{V_0, \beta(1 - \gamma v)\}} + \frac{n\alpha}{\left(1 - \frac{v}{\lambda S}\right)}$$

$$\gamma < \frac{1}{\lambda S}$$

where V_r = the running speed,
n = the number of signalized intersections per unit distance,
t_q = the queuing time per signalized intersection,
V_0 = the free-flow speed
λ = the proportion of effective green time, and
S = the saturation flow.

The network-based volume-delay function can be readily transformed to a link-based function by replacing n with $1/D$ (the number of signalized intersections per unit distance on a link of length D is $1/D$). Hence, the link-based volume-delay function of Wardrop's model can be modified as:

$$T = \frac{T_0}{1-\gamma v} + \frac{\alpha\beta}{(\alpha - v)D}$$

$$\alpha > c_u, \quad \gamma < \frac{1}{c_u}$$

where c_u is equal to λS. This function has to be restricted to the feasible region ($0 \leq v \leq c_u$) in an iterative assignment procedure. The limitations and solutions of this model are identical to the Mosher model.

In 1966, the Traffic Research Corporation (TRC) proposed the following function for use in a study involving Winnipeg's street network:

$$T = \alpha + \beta(v' - \gamma) + \sqrt{[\beta^2 (v' - \gamma)^2 + \delta]}$$

The function was eye-fitted using the Winnipeg data at low flows and the Toronto data at high flows.

2.5. Time-of-Day Factors

As pointed out in Section 1.2, peak-hour time-of-day factors are usually used to convert daily volumes to their hourly equivalents. By definition, a time-of-day factor is the ratio of vehicle trips made in a peak hour or a peak period to vehicle trips in some given base period, usually a day. Peak-hour volumes typically range from 8 to 12 percent of daily volumes. The percentages (i.e., time-of-day factors) can vary from link to link, depending on the facility and area types with which a link is associated. For example, the peak hour on urban freeways might account for only 6 to 8 percent of the daily traffic because the roads are congested throughout the day. A suburban collector might have as much as 12 to 14 percent of the daily volume during the peak hour. Time-of-day factors can also vary by travel mode, as auto and transit trips can exhibit very different temporal distributions. Transit trips tend to have a more concentrated morning peak with evening trips dropping off substantially compared with auto trips (Barton-Aschman and Cambridge Systematics, 1997).

Time-of-day factors have typically been applied through post-processing assigned trips and are applied to daily traffic volumes based on a daily vehicle trip table. Time-of-day factors can also be applied prior to trip assignment through one of the following three alternatives:

1. Between mode choice and trip assignment,
2. Between trip distribution and mode choice, and
3. Between trip generation and trip distribution.

In each of these alternatives, separate peak period and off-peak period trip tables are created before assignment. The pre-assignment alternatives recognize that the traffic volume on a link is composed of trips with different purposes, each having its own peaking characteristics. For example, work trips have well-defined peaks during the morning and afternoon and shopping trips are more pronounced in the afternoon and on weekends (Barton-Aschman and Cambridge Systematics, 1997). A major study by Cambridge Systematics, Inc. (1997) examined the four methods (pre- and post-assignment) in detail and the major findings are summarized in Table 2.3, which compares the applicability, level of effort and data required, and the limitations and advantages of the four time-of-day implementation methods.

Table 2.3. Comparison of Time-of-Day Implementation Methods.

Method	Applicability	Level of Effort and Data Required	Limitations and Advantages
Assignment after Trip Assignment	Method may be sufficient for smaller MPOs where the duration and intensity of congestion are limited. Most commonly used and simplest method.	Simplest method. Minimal labor and data required. Data required include peak hour factors that reflect peak period link-level travel demand; Directional split factors are also required.	Does not consider peak travel times in assignments. Trip distribution and mode split being done without accounting for congested times. Does not account for localized effects of changes in demand.
Assignment between Mode Choice and Trip Assignment	Method may be applicable in the least congested areas. Widely used method.	Data required include factors representing the percentages of the trips (by purpose and by mode) during each hour and for each direction, production-to attraction or attraction-to-production; Directional split factors are also required.	Trip distribution and mode split being done without accounting for congested times. Lack of sensitivity to general policy changes, increasing congestion levels, and corridor subarea-specific changes.
Assignment between Trip Distribution and Mode Choice	Method may be applicable in the least congested areas. Limited use.	Data required include factors representing the percentages of the trips (by purpose) during each hour and for each direction, production-to-attraction or attraction-to-production directional split factors are also required.	The effects of time of day characteristics such as congestion or transit levels of service are ignored in the way trips are allocated to time periods. Trip distribution and mode split being done without accounting for congested times.
Assignment between Trip Generation and Trip Distribution	Method may be applicable in the least congested areas. Limited use.	Data required include factors representing the percentage trips (by purpose and by mode) during each hour and for each direction, production-to-attraction or attraction-to-production; Directional split factors are also required. This approach can significantly increase model application time.	An advantage of this method is that differences in travel characteristics by time of day can be considered in trip distribution and mode choice. Procedure is not sensitive to increasing levels of congestion, nor is it sensitive to policy changes or congestion-management actions.

2.6. Peak-to-Daily Ratios

As defined in Section 1.2, a peak-to-daily ratio, or CONFAC in FSUTMS, is computed as the highest hourly volume of a day divided by the daily volume, as follows:

$$Peak\text{–}to\text{–}Daily\ Ratio = \frac{Highest\ hourly\ volume\ of\ a\ day}{Total\ daily\ volume}$$

This factor carries the same effect on the v/c ratios as a time-of-day factor, which converts a daily volume to its hourly volume equivalent. A literature search did not identify existing studies on peak-to-daily ratios per se, however, the use and the potential impact of CONFAC have been discussed in Section 1.4. An effort to calibrate this ratio using traffic count data for different facility types is presented in the next chapter.

2.7. Peak Spreading

2.7.1. Peak Spreading Phenomena

Methods considering the effects of peak spreading, instead of the relatively simple factoring methods described in the previous section, have been used in practice. The peak spreading process addresses the problem that projected demand exceeds capacity in certain corridors during peak periods and that failing to account for the excess demand results in a flawed assessment of travel conditions in the future. Trips occur at different rates at different times of the day. When the demand rises to levels at which the quality of service on the road network becomes unacceptable, drivers may have to choose a number of options to another time period, including the following (Cambridge Systematics, Inc., 1997):

- Seek alternative routes to bypass the congestion,
- Change mode of transport,
- Choose a different, more accessible, destination,
- Suppress journey, or
- Divert trip to another time period.

Several studies have proved that peak-hour volume as percentage of the daily volume varies widely throughout a network. One important effect that must be considered is the extended duration of peak times as traffic congestion in general increases and as travelers leave home and work earlier or later in order to avoid the most congested travel conditions, known as "peak spreading."

Two phenomena that are attributed to peak spreading are "active" and "passive" (Hounsell, 1991). As congestion increases in a network with growth in traffic demand, "passive" peak spreading can occur without any change in trip departure times due to increased congestion alone in mid-peak periods affecting traffic flows in subsequent periods. In contrast to this phenomenon, "active" peak spreading occurs when some travelers retime their departure times to avoid unacceptable conditions of congestion in the peak period. Most research on peak spreading has not distinguished between active and passive peak spreading, but on capturing the overall effects of the combination.

According to Allen (1991), although many previous efforts have acknowledged the need to better understand peaking patterns as they affect the accuracy of traffic forecasts, they all fall short of providing a practical approach that could be readily adapted to produce usable results for specific area. He further states that, as a major deficiency, the area of incorporating peak spreading as a result of traffic congestion into large-scale traffic assignment and network equilibrium systems, such as UTPS, requires detailed highway system analysis in major metropolitan areas.

Allen and Schultz (1999) observe that increasing congestion and the growing need for more flexibility in scheduling activities are the key motivators of temporal distribution of traffic, that the greater use of workplace flextime and expanded operating hours of many business permit it to occur. Allaire and Ivan (1999) further note that an important part of developing methods to produce better peak-period and peak-hour forecasts is understanding how travel behavior patterns in a congested transportation system change when capacity or congestion increases.

Most peak spreading models have been specifically developed for the application area. While many of these methods provide a solid basis for further research, the models themselves are not likely to be transferable. Barnes (1998) classified the different methods of implementing peak spreading analysis into the following groups:

1. Four-step process incorporating peak-spreading effects,
2. Peak spreading sub-models, and
3. Stand-alone peak spreading models.

For the purposes of this project, only models categorized under the first two groups are included in the review below.

2.7.2. Four-Step Process Incorporating Peak-Spreading Effects

This group of peak-spreading implementation accommodates the peak spreading methodologies within the confines of the four-step modeling process. In a pioneering study, Loudon *et al.* (1988) provided an estimate of the net effect of traffic congestion without identifying the magnitude of each type of behavioral response. The traffic volumes were collected from freeways and major arterials in Arizona, Texas, and California. The study was based on the constant ratio of total number of trips within each peak period to total daily trips by trip purpose and direction (to or from home) using split factors developed for the region. A functional relationship was then estimated to relate the peak-hour to peak-period volume ratio to the overall volume to capacity ratio during the peak period as follows:

$$P = 1/3 + ae^{b(v/c)}$$

where P = the ratio of peak-hour volume to peak-period (3-hour) volume,
v/c = the volume/capacity ratio for the three-hour period, and
a, b = model parameters.

Based on the derived relationships, a submodel that can reflect the peak-spreading phenomena and predict a peak-hour volume for each link in the highway network in the future year was developed for an UTPS-based system. The results predict peak-hour speeds better than the traditional methods. This method also reduced the error in estimation of link volumes to 2.2% from 16.4%.

An iterative-factoring procedure that applies only to highway trips was implemented in the Boston area for the Central Artery/Tunnel Project (Cambridge Systematics, Inc., 1997). Since the base-year peak-hour volumes were already at or over capacity throughout downtown Boston and daily travel was projected to grow, the resulting forecast year volume estimated with traditional methods exceeded capacities by unrealistic amounts. To solve this problem, Rossi *et al.* (1990) developed a technique to reduce a trip table selectively, according to congestion levels in the corridor corresponding to the OD pair in individual OD cells of the trip table. Since predicted traffic volumes in uncongested corridors were not changed by unrealistic amounts, the selective reduction was superior to global reduction that implies a general decrease in trip generation rates. In addition, the creation of a matrix of interchange-specific peak hour factors to apply to a daily trip table provided an important analysis capability. The major drawbacks to this method include the lack of explicit treatment for the trips being reduced, and the inability to account for changes in traveler behavior in response to congestion.

Goodwin and Coombe (1991) described an effort to adjust a trip matrix for SATURN, an assignment model commonly used in Britain to incorporate the effects of congestion. The following formula is used to model the effect of peak spreading:

$$R = 1 - (k \times V^2)$$

where R = the ratio of the flows in the adjacent two half-hour periods to the flow in the peak hour (the "peakiness" factor),
k = a calibrated coefficient specific to the chosen peak period, and
V = the average peak hour speed.

The end result of an iterative process with this peak spreading formula shows a reduction in the a.m. and p.m. peak hour trip matrices.

DeCorla-Souza *et al.* (1995) adjusted and simplified the four-step process by developing average daily speed determination models for freeways and signalized arterials. This approach was designed to meet the requirement of air quality conformity analysis under the 1990 CAAA. These authors developed and refined empirical functions to estimate hourly link volumes and total daily delay in terms of the ratio of average daily traffic to link capacity (AADT/C) for both freeways and arterials. The travel times were estimated with the resulting speeds to compare with the travel times used for the input of trip distribution. The refined freeway equations were defined with different range of AADT/C value as follows:

$$DR = 0.0797(AADT / C) + 0.00385(AADT / C)^2 \qquad \text{for } AADT/C \leq 8$$

$$DR = 12.1 - 2.95(AADT / C) + 0.193(AADT / C)^2 \qquad \text{for } 8 < AADT/C \leq 12$$

$$DR = 19.6 - 5.36(AADT / C) + 0.0342(AADT / C)^2 \quad \text{for } AADT/C > 12$$

where DR = daily vehicle-hours of delay/1,000 VMT,
$AADT$ = average annual daily traffic, and
C = highway capacity.

The model was insensitive to area type and facility type, however, and the procedure was insensitive to the urban design, demographics, and changes in the levels of traffic congestion.

Allen and Schultz (1996) calibrated a trip-based peak-spreading model for the Washington D.C. region for travel analysis in the Dulles Corridor. It was hypothesized that the total congestion for a trip is a primary reason for peak spreading rather than the congestion of one link. The procedure also assumed that the proportion of the day's travel occurring during the peak 3-hour remains stable over time. A set of curves relating this percentage to the travel time difference for each trip purpose was estimated from the survey data. The following model was developed for each different trip distance range and separate trip purpose:

$$\text{Share}_d = \text{Max}\{[\text{maxshare}_d + \text{slope}_d \times \text{Max}(\text{timediff} - \text{limit}_d, 0)], \text{minshare}_d\}$$

where d = distance range,
maxshare_d = maximum share (the leftmost part of the curve),
slope_d = slope (the drop in peak-hour share per minute of congested time difference),
timediff = congested time – free flow time,
limit_d = limit (the congested time difference at which the line begins to slope downward), and
minshare_d = minimum share (the rightmost part of the curve)

The relationship shows that the peak hour flattens as both independent variables (congestion and the trip length) increase. Because home-based work (HBW) trips constitute the vast majority of a.m. peak-period trips, the HBW model was recommended to represent the general peak-spreading model for all trip purposes. The study found that short trips (less than 5 miles) that are sensitive to congestion levels have a high peak-hour share at almost 50%. However, the peak-hour share of very long trips (greater than 32 miles) appeared to be relatively unaffected by congestion.

A subarea model based on trip-based peak spreading scheme was applied to a major subarea in the San Francisco Bay area (Cambridge Systematics, Inc., 1997). Two major freeways defining four major gateways were identified as the key capacity constraint locations. The following assignment procedure was used:

1. Peak hour volumes were assigned to the highway network and v/c ratios calculated,
2. For gateways with v/c ratios in excess of 1.0, target volumes were estimated so that the v/c ratio would be 1.0 (or slightly higher),
3. A mathematical approach for adjusting trip tables was used to reduce the interchange volumes on the OD pairs using the over-assigned gateways,

4. The revised trip table was assigned and the gateway v/c ratios were checked for reasonableness, and
5. The process was repeated if a close match between the assigned and desired link volumes was not obtained for the gateway links.

In order to address the limitation of link-based models, this model achieved a better consistency from link to link by adjusting the trip tables in response to congestion. However, this approach does not account for changes in traveler behavior due to congestion. Moreover, although the trip reduction approach produces more reasonable peak-hour traffic volume estimations by using the capacity limit, there was no explicit treatment of the trips being reduced. Instead of assuming that the excess trips on each congested interchange were not made, the authors assume that the trips that could not be made in the peak hour were forced to travel outside of the peak hour.

Allaire and Ivan (1999) extend the Loudon *et al.* procedure by incorporating site variables to predict the proportion of the 4-hour peak-period volume on a highway link occurring in the peak hour. The selected freeway locations in Connecticut are widely spread throughout the state to make the model transferable statewide. The basic functional form of this peak-spreading model is:

$$P = 1/4 + ae^{bX}$$

or, in the transformed form:

$$\ln(P - 1/4) = C + bX$$

where P = the ratio of the peak-hour volume to the four-hour peak-period volume,
X = the V/C ratio for the four-hour period,
a, b = parameters to be estimated, and
C = ln a.

The model formulation was adjusted to incorporate site-specific dummy variables, site-specific v/c variables, directional dummy variable, and directional variable. The transformed formulation used for estimation can be extended to the following form:

$$y = C_0 + C_1S + C_2D + b_0X + b_1X_S + b_2X_D$$

where y = ln (P – ¼)
C_0, C_2, b_0, b_2 = scalar parameters to be estimated,
C_1, b_1 = vector parameters to be estimated, with one value for all but one of the analysis sites,
S = vector of site-specific dummy variable having values of 1 or 0,
D = directional dummy variable having a value of 1 if the flow direction is in the reverse-commute direction, 0 otherwise,
X = the V/C ratio for the four-hour period,
X_S = vector of site-specific variables equal to X multiplied by S, and

X_D = directional site-specific variable having the value of the V/C ratio for the 4-hour period if the flow direction is in the reverse-commute direction, 0 otherwise.

A more generic model was run without any site-specific variables before estimating a model using the extended equation. Then the extended equation was written as the original equation ($\ln(P-1/4) = C_i + b_i X$) by grouping the like terms as the following linear functions used to estimate a regression model using data from the ten sites.

$$C_i = C_0 + C_{1i} + C_2$$
$$b_i = b_0 + b_{1i} + b_2$$

where C_i = constant of a given site *i*,
C_{1i} = site-specific constant of a given site *i*,
b_i = slope coefficient of a given site *i*, and
$b_{1\text{i}}$ = site-specific v/c coefficient that acts as a correction to the generic coefficient.

One of the ten sites was chosen to be the base site with both constant (*C*) and slope (*b*) equal to zero. As expected, the negative slope indicates that the ratio of peak-hour to peak-period volume decreases as the v/c ratio on the link increases.

Ivan and Allaire (2000) further examine peak-spreading to estimate models with two different categorizations of freeway study sites: by region within the state, and by location with respect to the origin in which the link is located. The study locations were divided into five regions; considering the diversity of population distribution, characteristics of the city and the economy, together with the observation of original results, one site is significantly different from the others within the same region. The model was estimated using the following formulation:

$$y = C_0+C_RD_R+C_{CP}D_{CP}+C_{SE}D_{SE}+C_WD_W+C_{SL}D_{SL} +b_0X+b_RX_R+b_{CP}X_{CP}+b_{SE}X_{SE}+b_{SW}X_{SW}+b_{SL}X_{SL}$$

where D_R = dummy variable having the value of 1 if the direction of travel is in the reverse-commute direction,
D_{CP} = dummy variable having the value of 1 for Capacity region,
D_{SE} = dummy variable having the value of 1 for the Southeast region,
D_W = dummy variable having the value of 1 for the West region,
D_{SL} = dummy variable having the value of 1 for the Shoreline region,
X = volume to capacity ratio for the peak period,
X_R = volume to capacity ratio for the peak period if the direction of travel is in the reverse-commute direction,
X_{CP} = volume to capacity ratio for the peak period if D_{CP} =1,
X_{SE} = volume to capacity ratio for the peak period if D_{SE} = 1,
X_W = volume to capacity ratio for the peak period if D_W = 1, and
X_{SL} = volume to capacity ratio for the peak period if D_{SL} = 1.

As an alternative to the regional model above, the study locations were also grouped into five categories according to the relative land use densities, dominant travel patterns, and locations within their respective regions. The model formulation is:

$$y = C_0+C_RD_R+C_1D_1+C_2D_2+C_3D_3+C_4D_4+b_0X+b_RX_R+b_1X_1+b_2X_2+b_3X_3+b_4X_4$$

where D_1 = dummy variable having the value of 1 for urban area,
D_2 = dummy variable having the value of 1 for suburban area,
D_3 = dummy variable having the value of 1 for ex-urban area,
D_4 = dummy variable having the value of 1 for shoreline area,
X_1 = volume to capacity ratio for the peak period if $D_1 = 1$,
X_2 = volume to capacity ratio for the peak period if $D_2 = 1$,
X_3 = volume to capacity ratio for the peak period if $D_3 = 1$, and
X_4 = volume to capacity ratio for the peak period if $D_4 = 1$.

By comparing the corresponding constant and slope coefficients among the locations according to each of two models, the difference of spreading level of peak profile at low levels of congestion or sensitivity to increasing congestion respectively can be measured. Nevertheless, the differences in the amount of peak spreading that occurs among these regions was difficult to attribute to specific characteristics within each region. Instead, each of regions in general was considered to have different labor force characteristics or specific attributes that may be affecting the amount of peak spreading.

CHAPTER 3
CALIBRATION OF CONFAC MODELS

3.1. Introduction

In the previous chapter, various volume-delay functions were reviewed. As discussed, a major factor used in the BPR volume-delay function is the peak-to-daily ratio, or CONFAC in the context of FSUTMS. By definition, CONFAC is computed as the highest hourly volume of a day divided by the daily volume, or:

$$CONFAC = \frac{Highest\ hourly\ volume\ of\ a\ day}{Total\ daily\ volume}$$

FSUTMS applies the CONFAC parameter to convert an hourly link capacity to the daily equivalent for the computation of daily v/c ratio in the BPR function, defined as follows:

$$T_n = T_0 \times \left[1.0 + a\left(v/c_p\right)^b\right]$$

where T_n = new travel time from the current iteration,
T_0 = free-flow travel time,
a, b = model parameters, and
v/c_p = ratio of assigned volume-over-practical capacity.

The current version of FSUTMS only allows a constant CONFAC to be applied to a facility type, even though studies (Allaire and Ivan, 1999; Loudon *et al*., 1988; Walters *et al*., 1989) have shown that the peak-to-daily ratios tend to be a decreasing function of congestion level, as depicted in Figure 3.1. This chapter describes the process of calibrating the functional relationships between peak-to-daily ratios and an appropriate congestion measure based on actual traffic count data. The next chapter will describe the implementation of these functional relationships in FSUTMS.

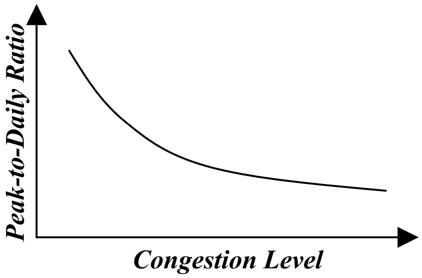


Figure 3.1. General Relationship between Peak-to-Daily Ratio and Congestion Level.

3.2. Effects of Congestion

Figure 3.2 shows three possible traffic profiles under different congestion levels. It can be seen that when a roadway is not congested, the total daily volume will tend to be low, causing the peak-to-daily ratio to be high. On the other hand, when a roadway becomes highly congested, the effect of peak spreading sets in. Under this condition, the excess peak-hour traffic, constrained by capacity, will shift to the peak shoulders. As a result, the peak-to-daily ratio will tend to be low because the total daily volume increases while the highest hourly volume is constrained by the capacity. To improve the prediction of peak hour volumes, the effect of congestion on travel behavior in the peak period needs to be incorporated.

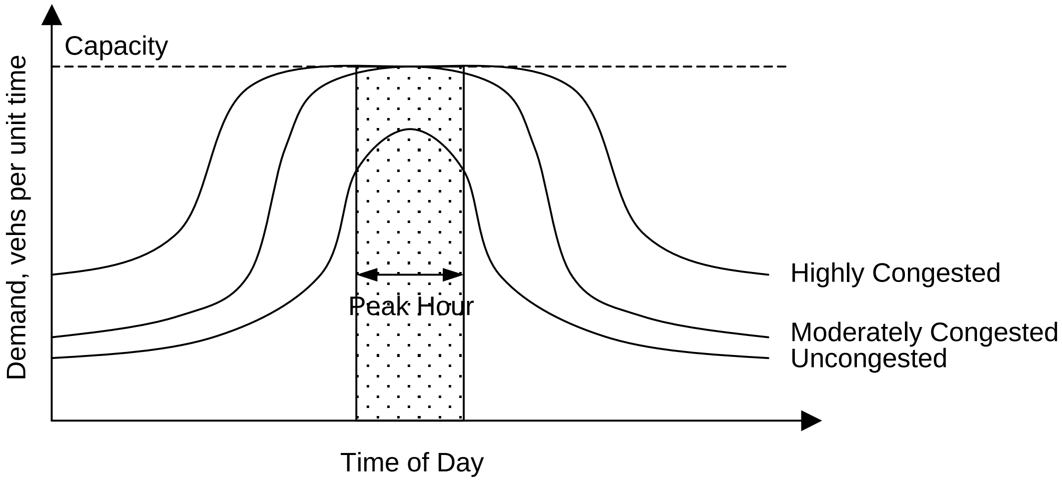


FIGURE 3.2. Peak Spreading under Different Congestion Levels.

3.3. Data Acquisition and Reduction

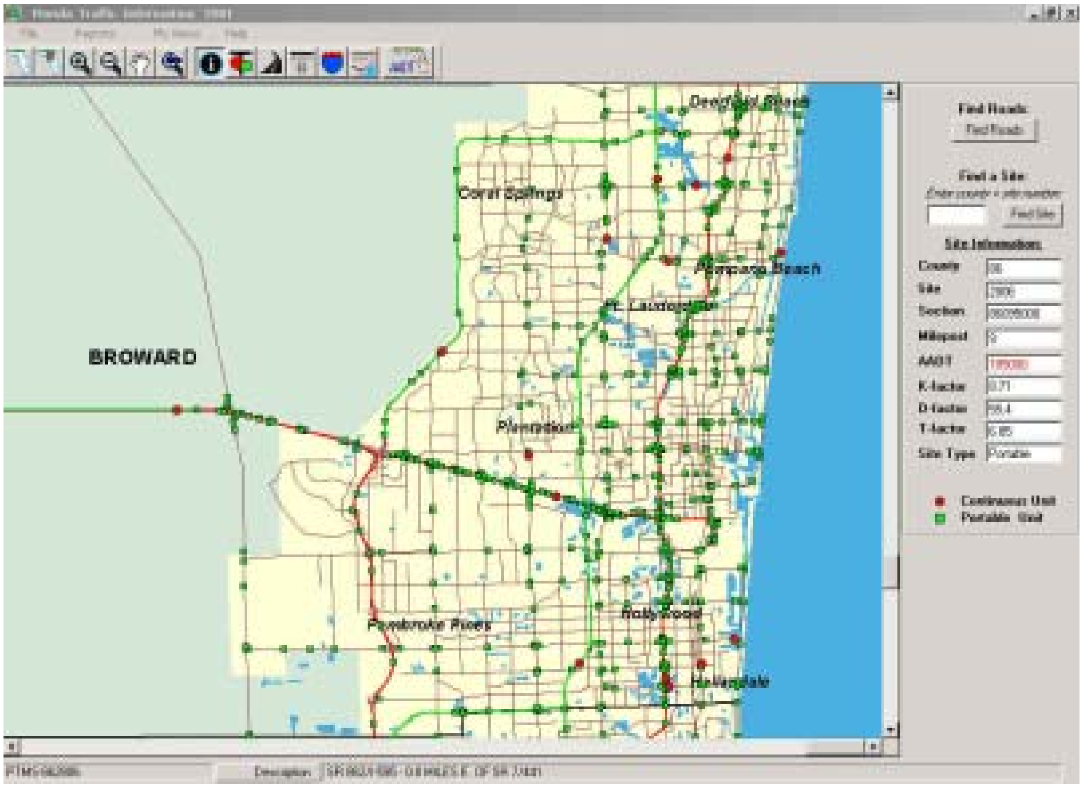
Traffic counts and roadway characteristics are the two categories of data needed in this study.

3.3.1. 15-Minute Directional Traffic Counts

The source of traffic count data is the 2000 Florida Traffic Information (FTI) CD distributed by the Florida Department of Transportation (FDOT). The CD includes traffic counts from a total of 271 Telemetry Traffic Monitoring Sites (TTMSs) and about 7,000 Portable Traffic Monitoring Sites (PTMSs) located throughout the State of Florida. TTMSs record traffic flow continuously for 24 hours a day, 365 days a year, while PTMSs are temporarily placed at strategic locations throughout the state for only a short period of time, typically from one to three days. The 2000 FTI CD includes hourly counts for all TTMS sites and 15-minute counts for all PTMS sites. Figure 3.3 shows a display of the count locations in the Broward County urban area. The PTMS and TTMS sites are represented by square and round dots, respectively.

For the purpose of analyzing the traffic data, an investigation of the relationship between CONFAC and select congestion measures was carried out for all of the TTMSs and PTMSs. Each daily observation is considered a single sample. A plot of the CONFACs derived from the full year of each TTMS site shows that the data points tend to cluster in a small area. This, coupled with the fact that there are only a small number of TTMSs, indicates that the TTMS may not provide a sufficient sample size. Hounsell (1991) stresses the importance of performing the peak spreading analysis based on data from a variety of links located throughout the network. In addition, since roadways in urban areas are often congested, one-hour traffic data may obscure the true peaks and, thus, inhibit the planning parameters as a measure of demand. Jordaan and Van As (1991) found that hourly flows are suitable for studies involving rural traffic, while directional 15-minute flows are better suited for studies involving urban traffic. Accordingly, the PTMS data were thus used in this study for the following two reasons:

- They provide a much larger sample size on the basis of the number of count stations from different roadways.
- The higher data resolution (15-minute intervals) allows the actual peak hour to be captured more accurately than the hourly data.



Site Information

County	86
Site	2906
Section	86095000
Milepost	0
AADT	199000
K-factor	8.71
D-factor	55.4
T-factor	6.85
Site Type	Portable

● Continuous Unit
🟩 Portable Unit

Description: SR 862/I-595 - 0.8 MILES E. OF SR 7/441

Figure 3.3. PTMS and TTMS Sites for the Broward County Urban Area.

The traffic volume during an interval of time shorter than a day must reflect the operating conditions for which it was designed if traffic is to be served properly. Capacity and other traffic analysis focus on the peak hour of traffic volume, because it represents the most critical period for operations, performance evaluation, and has the highest capacity requirements. The peak-hour volume reflects the level of congestion, and in turn, reflects the effect of peak spreading that results from the predicted congestion on the facility.

The highest four successive 15-minute volumes among a period of 24 hours determine the peak hour volume representing the greatest level of traffic activity. The CONFAC is thus calculated in terms of the selected peak hour volume divided by daily volume for all PTMS stations. A program was developed to extract these 15-minute counts from the CD and to estimate the

CONFAC value. All traffic count stations recorded traffic separately in individual directions. Accordingly, a CONFAC is estimated for each direction of traffic for each two-way roadway.

Note that the vehicle counts used in the computation of CONFACs were not adjusted for heavy vehicles. This assumes that the proportion of heavy vehicles in the traffic stream is constant throughout the day. Consequently, the two heavy vehicle factors associated with the peak-hour and daily volumes are the same and do not affect the CONFAC values.

3.3.2. Roadway Characteristics

After the traffic data are extracted, they must be related to geometric and geographic information for model development. Data for variables from the FDOT's Roadway Inventory Characteristics (RCI) database was extracted including number of lanes, functional classification, type of roadway, and traffic count station number. The RCI system is a segmented dataset with many related tables of roadway characteristics and other information for use in the transportation planning process. All the data are referenced to the road system by a section milepost location referencing system. In this system, each highway is assigned an eight-character roadway identifier, which is made up of a two-digit county number, followed by a three-digit section number, and a three-digit subsection number. Data are referenced to the section by milepost, which marks the change of features or attributes on a road section.

The traffic count data were joined with one major RCI table by a unique attribute, two-digit county number followed by four-character traffic count number, to obtain most of necessary characteristics. The other related attributes tables can then be merged with this combined table using the section milepost location reference number. The resulting merged file contains all the data needed for model calibration in this study.

To be consistent with the model chain of FSUTMS, the roadway system under investigation was classified into various groups based on the nine standard facility types in FSUTMS, as listed in Table 3.1. However, the RCI database uses the Federal Functional Classification System to categorize the roadways, which is different than that of FSUTMS. By specifying the respective combination of type of roadway and functional classification in several tables from RCI data as the criteria listed in Table 3.2, five out of nine facility types were found. Note that only count stations on urbanized roadways are included.

Table 3.1. Default Two-Digit Facility Types (FDOT, 1997).

Code	Facility Type
10	Freeways and Expressways
20	Divided Arterials
30	Undivided Arterials
40	Collectors
50	Centroid Connectors
60	One-Way Facilities
70	Ramps
80	HOV Facilities
90	Toll Facilities

Table 3.2. Equivalent Facility Types between RCI System and FSUTMS.

RCI System		FSUTMS
Code/Type of Roadway	Code/Functional Classification	Functional Classification
2 Divided	11 Principal Arterial – Interstate 12 Principal Arterial – Other Freeways and Expressways	Freeways and Expressways
2 Divided	14 Principal Arterial – Other 16 Minor Arterial	Divided Arterials
0 Undivided	14 Principal Arterial – Other 16 Minor Arterial	Undivided Arterials
0 Undivided & 2 Divided	17 Collector	Collectors
4 One-Way	14 Principal Arterial – Other 16 Minor Arterial	One-Way Facilities

3.4. Model Development

This section describes the process of developing the functional relationships for CONFACs and congestion level based on regression analysis.

3.4.1. Congestion Measure

A number of different variables have been used as measures of traffic congestion (Boyce *et al.*, 1981; Hounsell, 1991; and Cottrell, 1998). They include speed, density, volume, travel time, volume to capacity ratio, etc. The variable to be selected must take into account the availability of data and the purpose of application. For this study, it is required that the congestion measure be volume-based so that the model will be sensitive to the assigned volumes from traffic assignment.

Several volume-based measures can be found in the literature, with the more common ones being the average daily traffic (ADT), the v/c ratio, and the volume per lane. ADT is not a suitable measure in this study because the model involves different facility types with varying capacities. Although the v/c ratio has been commonly used as a key indicator of congestion level, it was not used because daily capacity is unknown. Volume per lane was selected as the congestion measure for the base model because the volume is weighted by lane, and that both the lane and volume data are readily available.

3.4.2. Selection of Base Model

To obtain good models for describing the relationships, it is important to first identify a base model that has a suitable functional form. Figure 3.4 shows scatter plots of CONFACs as a function of daily volume per lane for different facilities. An earlier attempt to model the functional relationships between these two variables was to apply a peak-spreading model developed by Loudon *et al.* (1988), as follows:

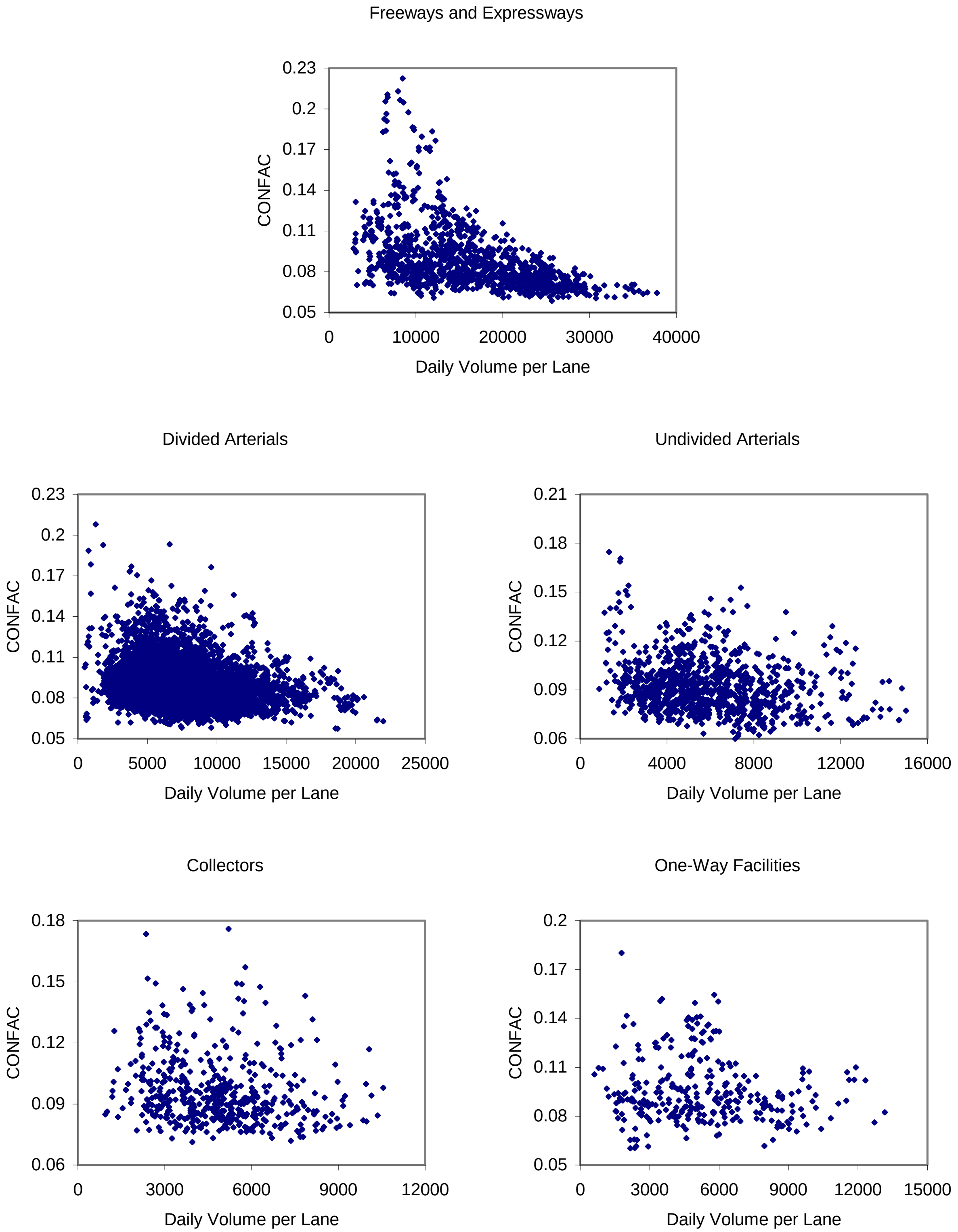


Figure 3.4. Scatter Plots for CONFAC vs. Congestion Level for Different Facilities.

$$P = \frac{1}{3} + \beta_0 e^{\beta_1 (v/c)}$$

where P = the ratio of peak-hour volume to peak-period (three-hour) volume,
v/c = the volume/capacity ratio for the 3-hour period, and
β_0, β_1 = model parameters.

For this model to be applied for the CONFAC functions, it has to be modified from the 3-hour based ratio to the 24-hour based ratio, as follows:

$$P = \frac{1}{24} + \beta_0 e^{\beta_1 C}$$

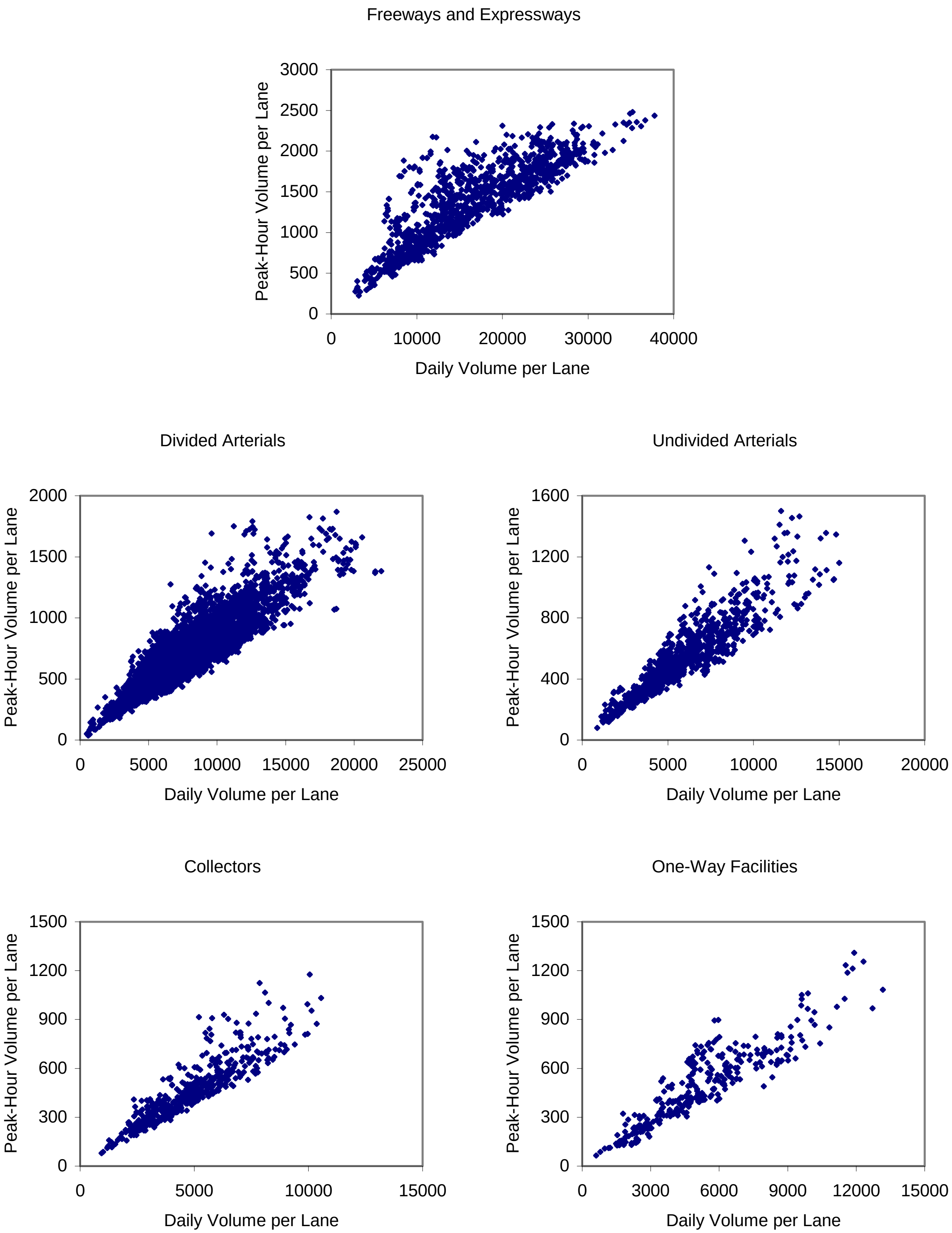
where C is the congestion measure. However, it was found that this model did not produce favorable results. Various other models and approaches were attempted before a final model was adopted. The final model does not describe the relationships between the two variables directly. Instead, it models the relationships between peak-hour volume per lane and congestion level and then uses them to predict the CONFAC values. In other words, the calibrated models predict the value of peak-hour volume instead of CONFAC for a given value of congestion measure. In turn, the predicted CONFAC can be approximated as the predicted peak-hour volume divided by the respective daily volume. The scatter plots in Figure 3.5 strongly suggest a consistent linear relationship between peak-hour volume per lane and congestion measure but also appear that the variance increases with congestion measure. The fact of such notation violates the assumption underlying most process modeling methods that the random errors have constant standard deviation. The analysis was then carried out using weighted least squares regression with the following linear model:

$$V_i = \beta_0 + \beta_1 C_i + \varepsilon_i$$

where V_i = peak-hour volume per lane,
C_i = daily volume per lane,
β_0, β_1 = model parameters, and
ε_i = random error term with mean $E\{\varepsilon_i\} = 0$ and variance $\sigma^2\{\varepsilon_i\} = \sigma^2$; ε_i and ε_j are uncorrelated so that their covariance is zero.

3.4.3. Empirical Model Calibration and Results

The SPSS statistical package was used to calibrate the regression models. The linear regression function was first fitted by ordinary least squares to conduct the preliminary analyses of the residuals. Unconstant error variances were confirmed with plots of the residuals against congestion measure for each facility type. The plots of the absolute residuals against congestion measure suggested that there might be a linear relation between the standard deviation and congestion measure. The estimates of the standard deviation were derived by regressing the absolute residuals against the congestion measure. These values were then used to compute the weights by using the following equation:



Freeways and Expressways

Divided Arterials

Undivided Arterials

Collectors

One-Way Facilities

Figure 3.5. Scatter Plots for Peak-Hour Volume per Lane vs. Daily Volume per Lane.

$$w_i = \frac{1}{\left(\hat{s}_i\right)^2}$$

where w_i and $\hat{s}_i$ denote the weight and estimated standard deviation for each data point respectively. Unlike ordinary least squares that treat every observation equally, the weighted least squares method attempts to give each data point the proper amount of influence over the parameter estimates. The weight w_i determines how much each observation in the data set influences the parameter estimates. The derivation of regression models with weighted least squares may be viewed as ordinary least squares of transformed variables as follows:

$$\sqrt{w_i}V_i = \sqrt{w_i}\left(\beta_0 + \beta_1 C_i + \varepsilon_i\right)$$

which can be expressed as:

$$V_{wi} = \beta_0^{'} + \beta_1 C_{wi} + \varepsilon_{wi}$$

Separate models were developed by segregating all the investigated traffic count stations into groups according to the functional classification of roadways. The great difference between these regression parameters and those obtained with regular least squares indicates that there is a need to reestimate the standard deviation function and the weights based on the residuals for the weighted regression. Estimated parameters change little stopping the procedure using the second weighted least squares regression as compared with those derived from the first weighted least squares regression. The plots of residual e_{wi} $(= V_{wi} - \hat{V}_{wi})$ against the fitted values $\hat{V}_{wi}$ showed that the pattern is slightly better behaved than the associated ones by using ordinary least squares method.

All the calibrated equations were evaluated on the basis of coefficient of determination (R^2) and the proper tendency of trendline. Table 3.3 summaries the modeled regression parameters and the associated statistical output using the linear functional form. The value of R (as high as 0.943) indicates that the selected dependent and independent variables are highly correlated. Thus the fitted regression equations explain the proportion of total variation in the data about the average dependent variable well. The predicted CONFACs derived from dividing the predicted peak-hour volumes by their corresponding daily volumes were then plotted against the congestion measure. Figure 3.6 illustrates a recognizable negative relationship between the two variables for different facility type. In general, the CONFAC value drops abruptly when the congestion measure increases. The decreasing rate in the CONFAC gradually diminishes as the level of congestion increases. As a consequence, the CONFAC asymptotically tends to result in a fairly constant value. This supports the choice of linear functional form for the base model.

Table 3.3. Regression Results.

Statistics	Facility Types				
	Freeways & Expressways	Divided Arterials	Undivided Arterials	Collectors	One-Way Facilities
$\hat{\beta}_0$	476.406	44.313	32.825	23.187	18.957
Std. Error of $\hat{\beta}_0$	17.657	2.836	3.469	6.573	10.487
$\hat{\beta}_1$	5.440E-02	8.242E-02	8.514E-02	9.064E-02	9.098E-02
Std. Error of $\hat{\beta}_1$	8.22E-04	4.31E-04	9.41E-04	1.73E-03	2.29E-03
Sample Size	1227	6898	1019	491	303
Multiple R	0.884	0.917	0.943	0.922	0.917
R^2	0.781	0.841	0.890	0.849	0.840

The model should be able to provide a better estimation of CONFAC than the constant CONAFC method for the iterative equilibrium assignment on the basis of the statistical analyses above; for this reason, the models of linear functional form were selected for incorporation into the BPR formula. The calibrated functional relationships were used for dynamic adjustments of CONFAC in the feedback loop of equilibrium assignment. Based on the derived functional relationships, a submodel that can reflect peak spreading and more accurately predict the travel time for each link in the highway network was developed for an equilibrium-based assignment. For implementation, the calibrated models can be written as the equations shown in Table 3.4. The lower and upper limits of CONFACs were also computed based on the traffic counts for each of the classification of facility types.

Table 3.4. Estimated Regression Functions for the Different Classes of the Categorization.

Facility Types	Regression Equations	CONFAC Limits	
		Lower	Upper
Freeways and Expressways	$\hat{V} = 476.406 + 0.05440C$	0.0670	0.2223
Divided Arterials	$\hat{V} = 44.313 + 0.08242C$	0.0844	0.1734
Undivided Arterials	$\hat{V} = 32.825 + 0.08514C$	0.0873	0.1226
Collectors	$\hat{V} = 23.187 + 0.09064C$	0.0928	0.1152
One-Way Facilities	$\hat{V} = 18.957 + 0.09098C$	0.0924	0.1218

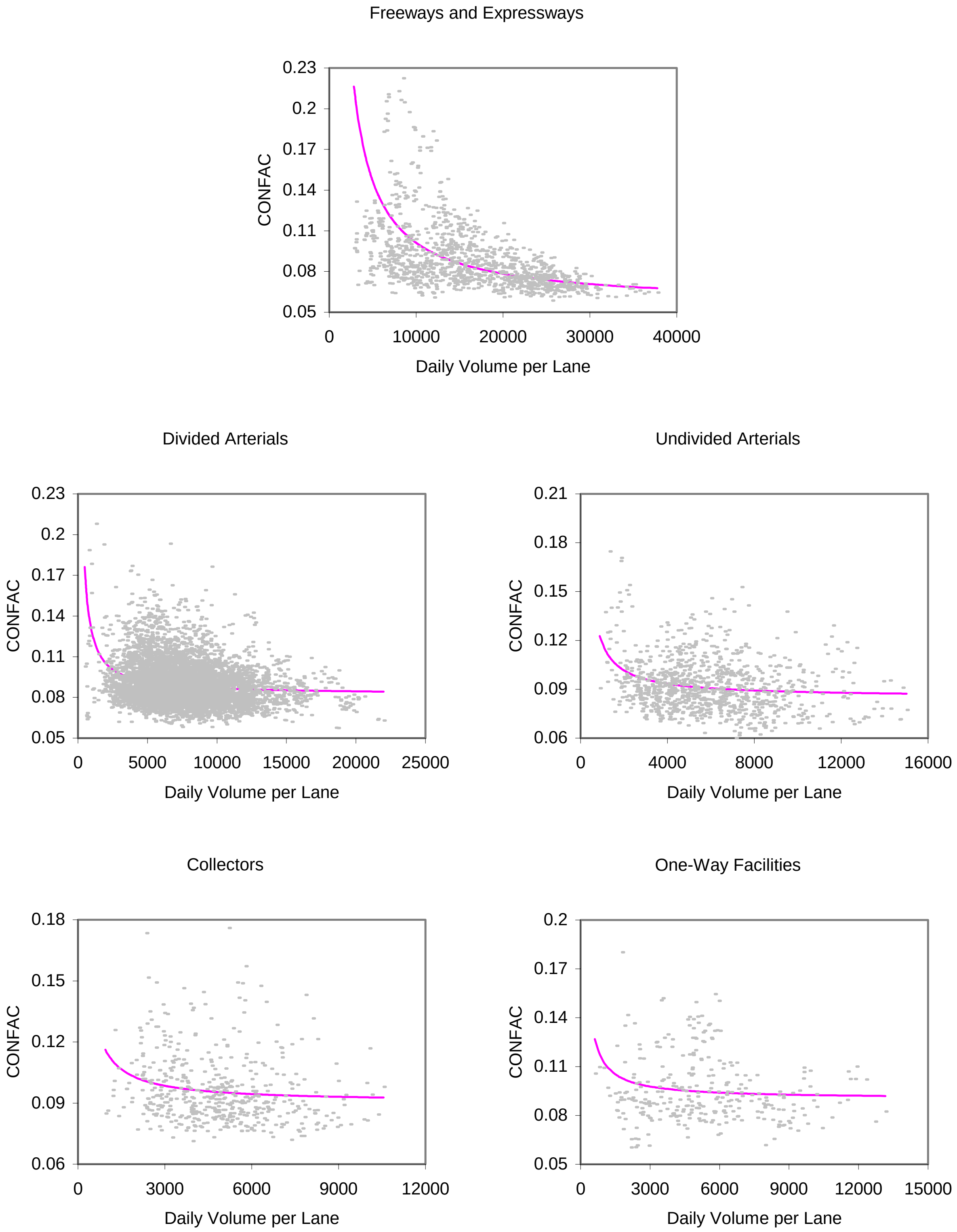


Figure 3.6. Regression Curves for Different Facility Types.

CHAPTER 4
IMPLEMENTATION OF VARIABLE CONFACS IN FSUTMS

4.1. Introduction

This chapter describes the process of implementing in FSUTMS the functional relationships calibrated in the previous chapter. Under an ideal, commercial-free environment, the research team would have ready access to the source code for FSUTMS' highway assignment module and would be able to implement the calibrated functions in the module with ease. However, knowing the difficulty of gaining access to the source code and the potential complications associated with it, the research team decided to incorporate the functions in FSUTMS without involving any modification to any part of the FSUTMS source code. This was accomplished by taking advantage of the iterative process of the FSUTMS assignment procedure. Essentially, this procedure runs the assignment procedure for only one iteration, computes the CONFAC value corresponding to each assigned link volume, updates the link travel times, and performs a new iteration based on the new travel times until the procedure converges.

The theoretical constructs, mathematical description, and algorithm used to implement the equilibrium assignment are described in detail in this chapter. A sample network illustrating the developed procedure is also presented. The calibrated relationships are applied in FSUTMS' equilibrium traffic assignment in an iterative loop to allow different CONFACs to be used for different congestion levels. The iterative loop inside the shaded box in Figure 4.1 depicts the existing iterative process for implementing the equilibrium trip assignment. The functional relationships calibrated in this study are applied in the outer feedback loop, as depicted in the same figure.

4.2. FSUTMS Structure

To implement the iterative equilibrium process in FSUTMS, a thorough understanding of the program structure is required. FSUTMS models configuration is the traditional four-step generation, distribution, model split, and assignment technique. The FSUTMS was customized and adapted for FDOT applications from TRANPLAN, a software developed as part of the URBAN/SYS family of urban planning and related tools to aid transportation planners in analyzing multi-modal transportation systems.

Since FSUTMS is customized on top of the TRANPLAN program framework, its overall program execution procedure in FSUTMS mainly follows that of TRANPLAN. A standard procedure as well as several nonstandard procedures are available for running various applications for different urban areas in FSUTMS. The standard procedure in FSUTMS primarily refers to the control files that are included for general purposes. Several options based on the extent of transit use in a particular area are provided within the standard procedure, including:

- Highway network only,
- Single-path transit network,
- Multi-path/single-period transit network, and

- Multi-path/multi-period transit network.

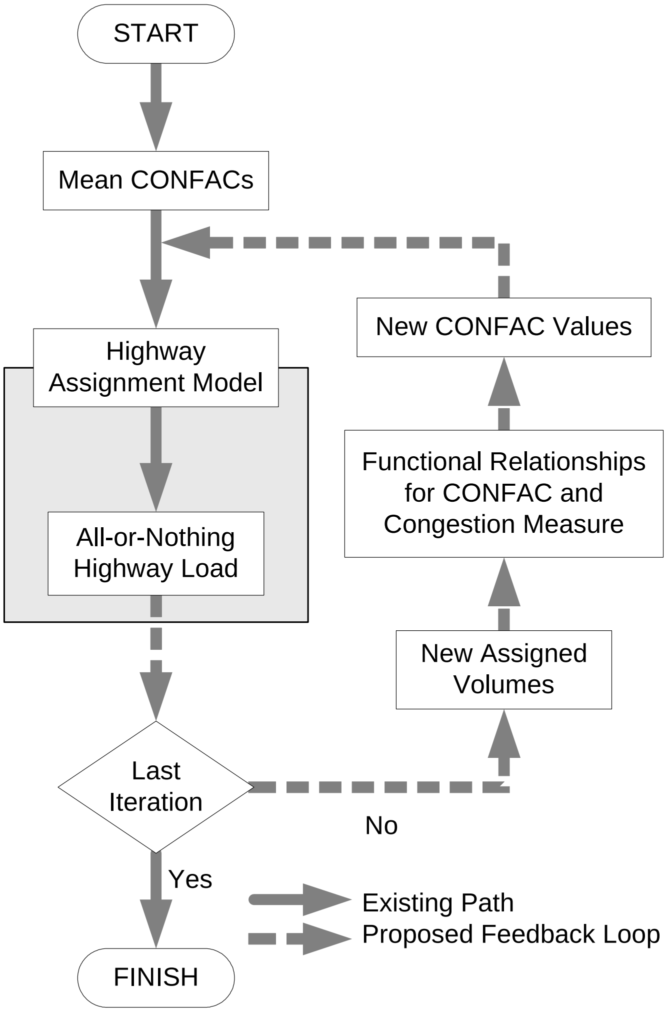


Figure 4.1. Improved Equilibrium Traffic Assignment Method.

On the other hand, the nonstandard application procedures include those used by planning agencies in Tampa Bay, Miami, Orlando, Turnpike, Jacksonville, and Volusia County. They also include those use multiple planning organizations in their joint effort to model multiple urban areas simultaneously. These include such model as the Southeast regional model and the statewide model. The nonstandard procedures were adapted and customized from the standard procedure to serve specific needs and considerations in particular areas.

In spite of the different control file structures used in different models and procedures, the basic modules used and the overall process of TRANSPLAN are largely similar. Figure 4.2 shows the FSUTMS overall highway-only process, which includes the following ten basic modules:

- EXT (External Model), which predicts the travel behavior of trips with at least one end in an external traffic analysis zone.

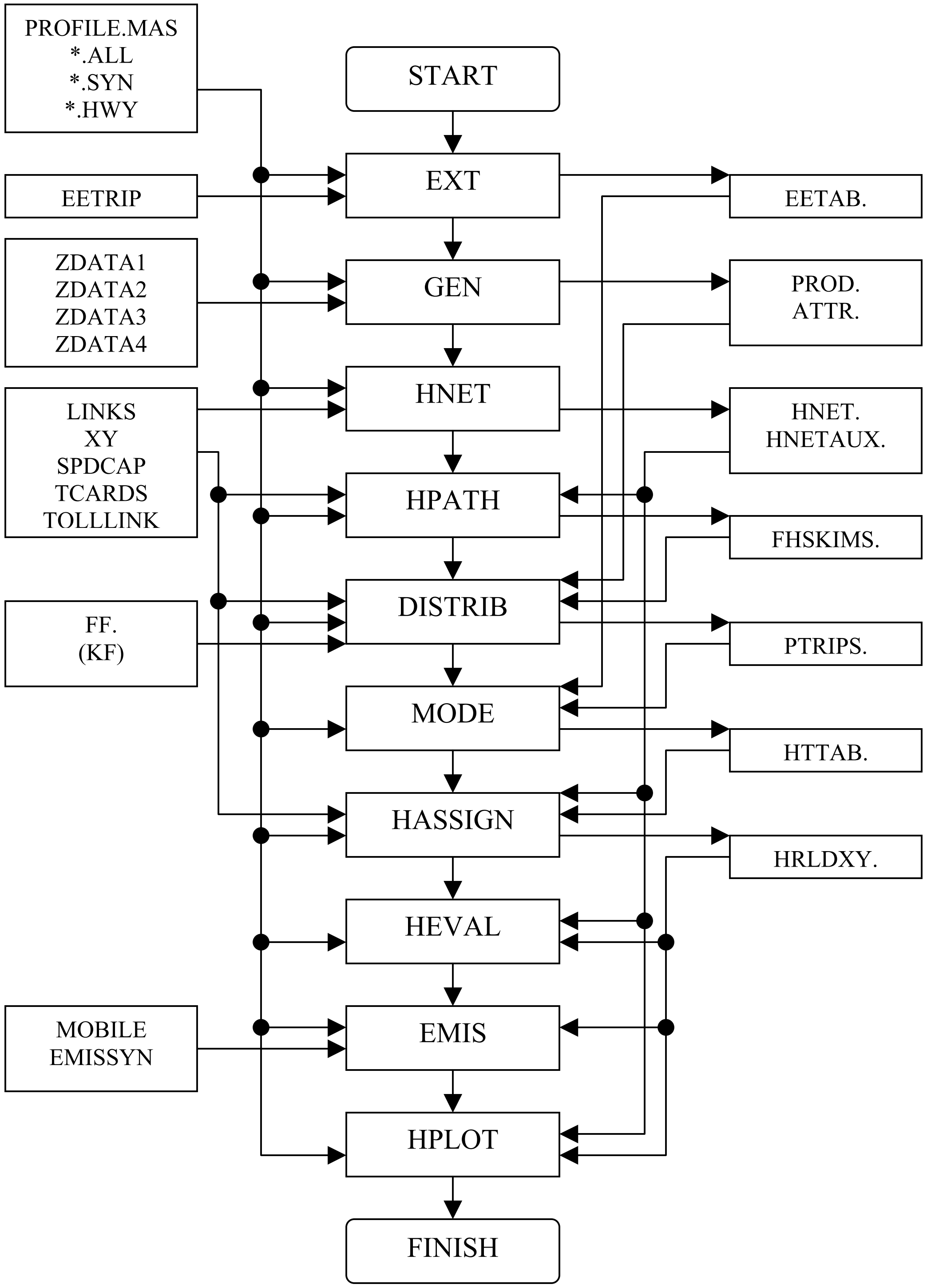


Figure 4.2. FSUTMS Job Process.

- GEN (Trip Generation Model), which defines the number of trips generated by or attracted to the different traffic analysis zones, according to land use, population, and other measures of economic activity.

- HNET (Highway Network Model), which creates a computerized system of nodes and links similar to the highway network, based on all the information provided by the user.

- HPATH (Highway Path-building Model), which calculates distances and travel times (skims) between the different traffic analysis zones.

- DISTRIB (Trip Distribution Model), which distributes the trips originating in one zone to the other zones based on some measures of accessibility and economic activity. It uses a gravity model.

- MODE (Mode Choice Model), which calculates the number of trips that are going to be carried out by the public transportation systems, and the number of trips that are going to be made using private automobiles. It converts from person trips to vehicle trips.

- HASSIGN (Highway Assignment Model), which assigns the trips to the shortest or minimum travel time path between traffic analysis zones.

- HEVAL (Highway Evaluation Model), which summarizes and evaluates all the information produced already. Its purpose can be to validate the model or to analyze it.

- EMIS (Mobile Source Emission), which estimates area-wide vehicle emissions as required by the Clean Air Act Amendments.

- HPLOT (Highway Plotting Model), which produces plot files that graphically represent the entire highway network information produced, validated, and/or analyzed by the software.

Each basic module can be executed individually and separately from each other, or in combination with others, depending on the purpose of an application. When executing FSUTMS, a control file is required to invoke specific TRANPLAN functions. Each module also requires input data and generates output data with a standard FSUTMS file naming convention. The input files have the extension **.yya*, and output files have the extension **.ayy*, where *"yy"* stands for the last 2-digit simulation year, and *"a"* represents the alternative identification. A file named *PROFILE.mas* is also referenced during program execution to identify parameter setting used in each step of the travel demand model. A message/report file is always created to show the results of a module run. Figure 4.3 shows the different types of input and output files used in the program.

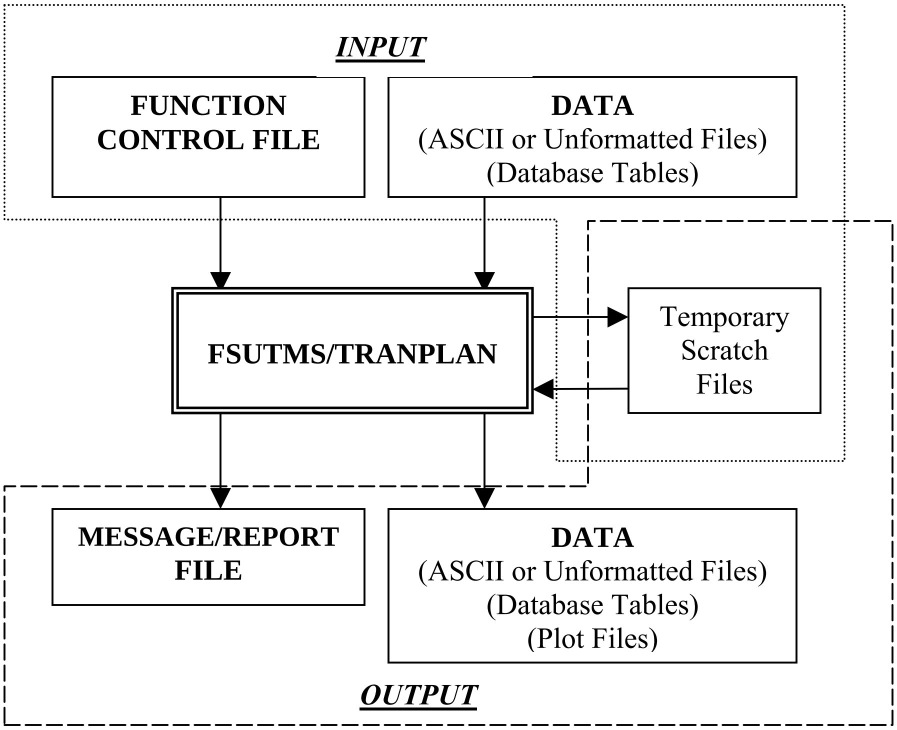


Figure 4.3. FSUTMS Input/Output Structure.

FSUTMS is operated through control files with the extensions **.all* and **.hwy* for the processing of highway modeling steps, and **.all* and **.tr1* through **.tr3* are used during transit modeling. The control files are those needed for controlling FSUTMS job processes. The files contain commands that specify the program(s) to be run, the files, parameters and options to be used by the program(s), as well as any report headers. They are ASCII (text) files produced by the user to provide the program with instructions for performing one or more functions. The general structure of the control files, as follows, is the same for all TRANPLAN functions:

```
$function name
$FILES
    INPUT FILE = ftype, USER ID = $fname$
    OUTPUT FILE = ftype, USER ID = $fname$
$HEADERS
    (Up to three header lines)
$OPTIONS
    (List of options)
$PARAMETERS
    (List of parameters)
$DATA
    (Data)
$END TP FUNCTION
```

Each of these instructions is created by the user to provide FSUTMS with all the specifications needed for the execution of each desired function.

- Function Name: This is the name of the FSUTMS function(s) to be executed and is the first line of any control file. It consists of one or more words that uniquely identify the function.

- Files: Specifies the input and output files and the database tables used and/or created with each function. Each file and database table specification must be located in a separate line.

- Headers: Allows up to three lines of project information such as project name and year to be specified. This input does not affect FSUTMS operations.

- Options: Specifies different options for the execution of the above-mentioned function(s). Each function has a set of allowable options.

- Parameters: Specifies all the parameters that must be followed by the program when executing the above-mentioned function. Each function has a set of allowable parameters.

- Data: Provides supplementary input data. Generally, data are specified in either fixed or free format.

- End TP Function: This is the last entry in all control files. It indicates that FSUTMS has read all of the specifications and data needed to execute the current function.

When FSUTMS is executed, the interface program first generates the *TRNPLXXX.ins* files in sequence based on the abovementioned control files and all the necessary input files for the specific application (see Figure 4.3). The *XXX* defines the index of **.ins* files and begins with three-digit format *001*. It then calls the FSUTMS/TRANPLAN executable programs to run the process, which produces the final output files from the step as well as the input files for the subsequent module.

4.3. Highway Assignment in FSUTMS

The FSUTMS highway assignment model, "HASSIGN," is based on the equilibrium trip assignment, which, by definition, means that no trip can be made by an alternate path without increasing the total travel time of all trips in the network. It involves running several iterations of all-or-nothing capacity-restraint assignment with an adjustment of travel time reflecting delays encountered in the associated iteration. The first iteration utilizes free-flow network times to load all non-transit trips to the network. Given minimum highway paths (skims) between each pair of zones, origin-destination flows are then allocated to the links in their respective minimum routes, accumulating total flows on each link. All subsequent iterations of assignment build on the delays encountered in the previous iterations. This is accomplished by utilizing the network travel times from the previous iteration as the initial travel times for the current iteration (FDOT 1981).

4.4. User-Equilibrium Optimization

The theoretical constructs for equilibrium assignments are commonly referred to as the Wardrop conditions. Wardrop's first principle, also known as the user equilibrium principle, assumes that each user of the congested traffic network seeks to minimize his/her travel time. For a given trip table, the equilibrium assignment of traffic may be found by solving a nonlinear mathematical programming problem. The flow pattern that minimizes the equilibrium program is an equilibrium solution. This program includes a convex objective function, a linear constraint set, and non-negativity conditions on link volumes, as follows (Sheffi, 1985):

$$\min z(\mathbf{x}) = \sum_a \int_0^{x_a} t_a(w)dw$$

subject to

$$\begin{aligned} &\sum_k f_k^{rs} = q_{rs} && \forall r,s \\ &f_k^{rs} \geq 0 && \forall k,r,s \\ &x_a = \sum\nolimits_{rs} \sum\nolimits_k f_k^{rs} \delta_{a,k}^{rs} && \forall a \end{aligned}$$

where x_a = the flow on link a,
$t_a(w)$ = the volume-delay function,
q_{rs} = the demand between origin r and destination s,
f_k^{rs} = the flow on path k connecting origin r with destination s, and

$$\delta_{a,k}^{rs} = \begin{cases} 1 \text{ if } a \text{ belongs to path } k \text{ for O-D pair } (r,s). \\ 0 \text{ otherwise.} \end{cases}$$

The equilibrium assignment in FSUTMS was constructed based on this objective function. The objective function is the sum of the integrals of the link volume-delay functions, while the constraints on the objective function ensure that all solutions are feasible trip assignments. The link flows corresponding to the minimum value of the objective function are those that satisfy the equilibrium conditions. The route times are obtained by computing minimum time paths for all origins to all destinations by using only links that are assigned a non-zero flow. In other words, the travel times along all paths that are used between each OD pair are equal, and no unused path has a lower travel time. These represent the origin to destination accessibility times given the current equilibrium flows.

4.5. Frank-Wolfe Algorithm

Most approaches to minimizing equilibrium programs can be cast as feasible direction method. The Frank-Wolfe algorithm is the standard method for solving the constrained minimization problem since the direction-finding step can be executed relatively efficiently. Assume the current solution is $\mathbf{x}^n$ at the n^th^ iteration. The algorithmic procedure of the descent method is the calculation of $\mathbf{x}^{n+1}$ from $\mathbf{x}^{n}$. The line connecting the current solution points, $\mathbf{x}^{n}$, with the solution of the linearized problem (denoted $\mathbf{y}^{n}$) is the feasible descent direction of the search. The new solution, $\mathbf{x}^{n+1}$, must lie within $\mathbf{x}^{n}$ and $\mathbf{y}^{n}$. The search for a descent direction automatically creates a bound for the line search by accounting for all the aforementioned constraints when the descent direction is determined (Sheffi, 1985). The line search is then performed to compute the weights λ to minimize z($\mathbf{x}$) along ($\mathbf{y}^{n}$ - $\mathbf{x}^{n}$), which is:

$$\min z\left[\mathbf{x}^{\mathbf{n}}+\lambda\left(\mathbf{y}^{\mathbf{n}}-\mathbf{x}^{\mathbf{n}}\right)\right]$$

subject to

$$0 \leq \lambda \leq 1$$

Once the optimal λ_{n} of the line search is found, the next point can be generated in the following form as:

$$\mathbf{x}^{\mathbf{n+1}}=\mathbf{x}^{\mathbf{n}}+\lambda_{n}\left(\mathbf{y}^{\mathbf{n}}-\mathbf{x}^{\mathbf{n}}\right)$$

where ($\mathbf{y}^{n}$ - $\mathbf{x}^{n}$) is a descent direction vector, and λ_{n} is the move size along the direction. Essentially, the iterative capacity restraint assignment procedure starts with an initial feasible solution that satisfies the constraints, determines a feasible direction to move to improve the objective function, and then calculates how far to move in the direction.

Given a network with volume-delay functions for each link and a trip table to be assigned, the Frank-Wolfe method that involves the following steps is then applied to the solution of the system-optimization problem:

1. Perform an all-or-nothing assignment based on free-flow travel impedances to obtain a feasible assignment of trips to the network.
2. Compute the travel impedance on each link corresponding to weighted link volumes from Step 1.
3. Trace minimum path trees from each origin to all destinations based on the link impedances from Step 2.
4. Assign all trips from each origin to each destination to the minimum paths from Step 3 using the all-or-nothing assignment procedure.
5. Conduct a line search to find an optimal λ value that minimizes the objective function of the system-optimization formulation.
6. Linearly combine the current weighted volume and the new all-or-nothing assignment's volume using the optimal λ from Step 5 to obtain a new weighted volume for each link.
7. Stop if the convergence criterion is met; otherwise, return to step 2 for the next iteration.

4.6. Algorithm for Estimating Optimal LAMBDA (*λ)*

The methodology used for determining the iteration weights, λ, is the key of the equilibrium assignment process (Bell *et al.*, 1992). The line search is essentially an approximate onedimensional minimization problem, which deals with a nonlinear function of a single variable. The line search for the optimal λ can be performed with an interval reduction method. It involves an iterative procedure to search for a set of λ bracketed in an interval (a, b). It requires that λ lies within interval (a, b) and the objective function is continuous and uniquely defined in this interval.

This initial interval is (0, 1). The objective function is evaluated at an intermediate λ value at each iteration. A smaller bracketing interval is obtained for the next iteration, either (a, λ) or (λ, b). The interval is reduced at each successive iteration until the bracketing interval is tolerably small. λ is always equal to 1.0 for the first iteration. FSUTMS uses two stopping criteria in the procedure for the following iterations: 1/8 for the second Frank-Wolfe iteration, and $(1/8)^2$ for all subsequent iterations. Figure 4.4 provides the flowchart of the algorithm used for estimating the optimal λ.

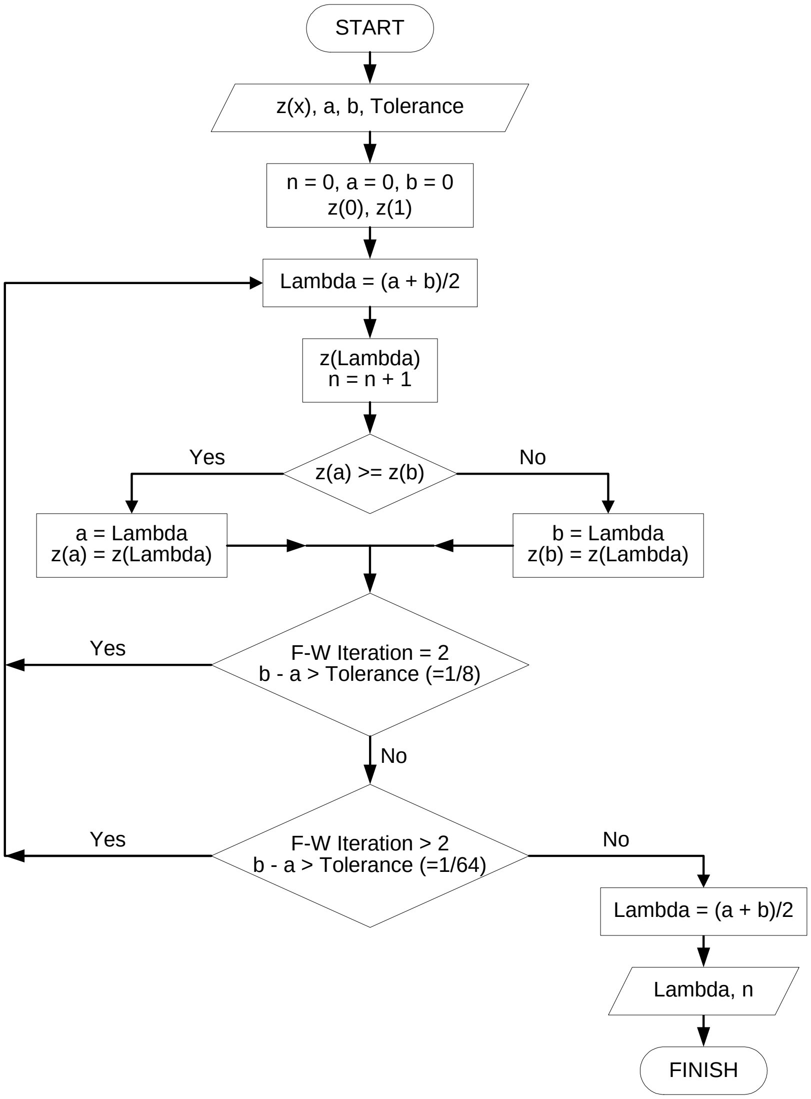


Figure 4.4. Algorithm for Estimating Lambda.

4.7. Toll Facilities Model

The toll facilities model was first developed to increase the sensitivity of FSUTMS to the impact of tolls in the highway assignment process. A toll link is made up of the following three segments of different traffic operational characteristics (Urban Analysis Group, Inc., 1998):

1. Vehicles travel at the free-flow speed and decelerate at a constant deceleration rate to a stop on the upstream link.
2. Vehicle queue at tollbooths to pay tolls according to a multiple-server queuing model.
3. Vehicles accelerate at a constant rate from a full stop back to free-flow speed on the downstream link.

A toll facility can be coded as three one-way segments as one of the configurations in Figure 4.5.

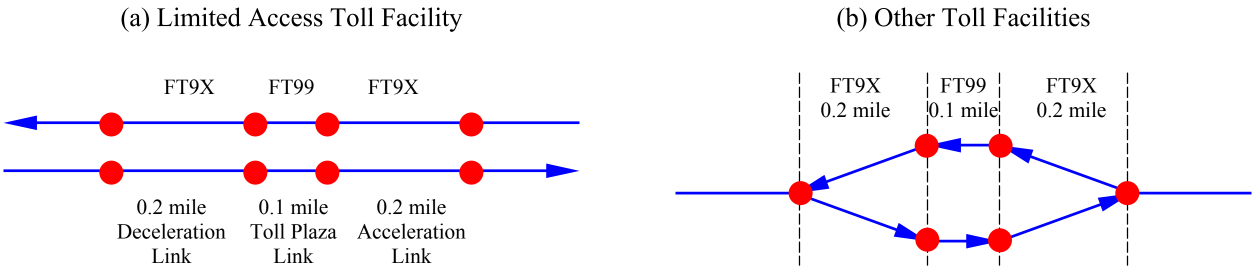


Figure 4.5. Toll Link Configurations and Coding.

The formulas for travel time and distance for the three segments are given as follows:

Deceleration Segment:

$$t_1 = \frac{V_1}{a_1}$$

$$a_1 = 2.35 + 0.055 \times V_1$$

$$d_1 = \frac{V_1 \times t_1}{2}$$

where t_1 = travel time during deceleration,
V_1 = free-flow speed on the upstream segment of toll link,
a_1 = deceleration rate (mph/sec), and
d_1 = travel distance during deceleration.

Queuing Segment: (Multiple-Server Queuing System)

$$P_0 = \frac{1}{\sum_{n=0}^{s-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^s}{s!(1-\rho)}}$$

$$\rho = \frac{\lambda}{s\mu}$$

$$L_q = \frac{P_0\left(\lambda/\mu\right)^s \rho}{s!\left(1-\rho\right)^2}$$

$$t_2 = \frac{L_q}{\lambda} + \frac{1}{\mu}$$

where P_0 = percent of time that all toll plazas are idle,
s = the number of severs,
λ = mean arrival rate,
μ = mean service rate per sever,
ρ = ratio of arrival rate over service rate,
L_q = average queue length, and
t_2 = waiting time including service time in the queue.

Although this multiple-server model is implemented in FSUTMS, a single-server model is documented in the TRANPLAN manual as the model implemented in FSUTMS.

Acceleration Segment:

$$t_3 = \frac{V_3}{a_3}$$

$$a_3 = 2.5 \text{ mph/sec}$$

$$d_3 = \frac{V_3 \times t_3}{2}$$

where t_3 = travel time during acceleration,
V_3 = free-flow speed on the downstream segment of toll link,
a_3 = acceleration rate, and
d_3 = travel distance during acceleration.

The travel time on a toll link is the sum of delay at toll plaza and time penalty representing the imposition of the toll, i.e.,

$$\text{Travel Impedance}_{\text{toll-link}} = \text{Delay at Plaza} + \text{Time Penalty}$$

The toll plaza delay is the total time from the three segments minus the time spent traveling at free-flow speed on the segments of upstream and downstream of the toll link, expressed as follows:

$$\text{Delay at Plaza} = t_1 + t_2 + t_3 - \left(\frac{d_1}{t_1} + \frac{d_3}{t_3}\right)$$

The time penalty is the toll amount (TOLL) multiplied by the time value of toll (CTOLL), as follows:

$$\text{Time Penalty} = \text{CTOLL} \times \text{TOLL}$$

4.8. FSUTMS Implementation

As mentioned in Section 4.1, the CONFAC functions would be implemented in FSUTMS without involving any modification to any part of the FSUTMS source code. FSUTMS is a batch system that executes a series of batch commands based on the coded instructions from control files. Therefore, partial modification of batch command arrangement along with programming of necessary computation steps is an alternative to replicate the algorithm of equilibrium assignment. In essence, this modification will bypass the volume-delay function and the Frank-Wolfe algorithm within the executable file of equilibrium assignment step in the *EQUIL.exe* file; the all-or-nothing assignment will be the only function utilized. The variable CONFAC models are then implemented by the developed executable file (*HASSFB.exe*) that bridges the "missing link" to achieve equilibrium assignment. The FTOWN sample network was used to design and test the procedure.

4.8.1. FTOWN Sample Network

The fictitious "FTOWN" sample network that came with FSUTMS was selected for the sample description of procedure development. FTOWN consists of 15 internal zones, nine external zones, seven types of trip purposes, eight types of roadway facilities, and five types of land usage. The feature of a relatively small but complete network makes it especially suitable for procedure design, testing, and fine-tuning. After the feedback process based on the FTWON network is successfully implemented, it will be extended to the other urban areas, which are based on real-world networks.

4.8.2. Iterative Procedure

The *FSUTMSX.ba*t is a preprocessing batch file provided by FSUTMS. Executing the *FSUTMSX.bat* file generates a series of *TRNPLXXX.ins* files, one *TRNPLNX.con* file, and one *XFSUTMS.bat* file. An iterative capacity restraint assignment procedure is used to modify these **.ins*, *.bat*, and *.con* files to incorporate feedback accordingly.

The basic principle of feedback is to replicate the equilibrium assignment procedure until possible convergence. The procedure begins by running *EQUIL.exe* for only one iteration. The all-or-nothing assigned volume is then extracted to carry out the equilibrium computation procedure by running the developed *HASSFB.exe*. The output from this procedure is finally fed back as the input for the next all-or-nothing assignment to continue the equilibrium process.

There are a total of 18 sequential *TRNPLXXX.ins* files created within the FTOWN network. Since the all-or-nothing assigned volume is the required data, the parameter of the maximum number of iterations in the corresponding control files, *TRNPL017.ins*, was set to 1 as follows:

```
$EQUILIBRIUM HIGHWAY LOAD
$FILES
   INPUT FILE = HWYNET, USER ID = $HNET.A00$
   INPUT FILE = HWYTRIP, USER ID = $HTTAB.A00$
   INPUT FILE = TOLDATA, USER ID = $TOLLLINK.00A$
   OUTPUT FILE = LODHIST, USER ID = $HRLDXY.A00$
$HEADERS
TEST
            HIGHWAY ASSIGNMENT
$OPTIONS
   TOLL FACILITIES MODEL
   MULTIPLE SERVER QUEUES
~    MODEL CAPACITY
$PARAMETERS
   EQUILIBRIUM ITERATIONS = 1 ------------► The equilibrium capacity-restrained method
                                            is forced to terminate after one iteration.
   DAMPING FACTOR = 0.5
   EPS = 0.001
   CTOLL = 0.04
$END TP FUNCTION
```

TRNPLNX.con is a "counter" file that indicates which control file, *TRNPLXXX.ins*, is to be used. After the execution of the TRANPLAN function specified in *TRNPLXXX.ins*, the corresponding control file is deleted and the index is increased by 1. To continue the iterative procedure, the control file, e.g., *TRNPL017.ins* in the FTOWN case, should be retained in the working directory. In addition, if the stopping criteria could not be met, the index decreases by 1 to keep *TRNPL017.ins,* which will then be invoked on the next Frank-Wolfe iteration.

With the modified setting of control files and the "counters," a post-processing batch file *XFSUTMS.bat* can execute the required process. Depicted as follows, the highway assignment procedure, which is extracted from *XFSUTMS.bat*, shows that the DOS commands with bold font are added to implement the iterative procedure. This new arrangement simulates a criterion- or counter-controlled repetition process. After the execution of *HASSFB.exe*, the implied loop-continuation condition is determined by examining the existence of an empty file *HASSIT.txt*. *HASSIT.txt* is created to reflect the convergence criterion or the maximum number of iterations will not be reached. If this is the case, the repetition process returns back to the marked command (:FSU100) and begins another iteration.

```
REM ***** HASSIGN ***** -- START
ECHO OFF
IF EXIST HASSIGN.OUT DEL HASSIGN.OUT
:FSU100 ---------------------------------► A mark stands for the beginning point of
                                           iterative procedure.
DATIME
EQUILB
REM DEL TRNPL017.INS -----------------------► Remark the delete command to retain
                                           TRNPL017.INS.
IF EXIST HASSIGN.OUT GOTO FSU013
REN TRNPL017.OUT HASSIGN.OUT
GOTO FSU014
:FSU013
COPY HASSIGN.OUT +TRNPL017.OUT >NUL
DEL TRNPL017.OUT
:FSU014
IF EXIST TRNPLN.ERR GOTO ERR
START /WAIT HASSFB -----------------------► Start HASSFB.EXE and wait for it to
                                           terminate before running the next command.
```

```
IF EXIST HASSIT.TXT GOTO FSU100
DATIME
RPTLOD
DEL TRNPL018.INS
COPY HASSIGN.OUT +TRNPL018.OUT >NUL
DEL TRNPL018.OUT
IF EXIST TRNPLN.ERR GOTO ERR
DATIME
ECHO ON
REM ***** HASSIGN ***** -- END
```

HASSIT.TXT is created to reflect the need of iterative process.

4.8.3. Equilibrium Computation Procedure

The equilibrium computation procedure was developed to resemble the process of FSUTMS' user-equilibrium program. All the required functions other than shortest routes computation and all-or-nothing assignment were coded in one single program named *HASSFB.exe*. The major functions of this executable file include getting the assigned link volumes from the output file (*HRLDXY.ayy*) of *EQUILB.exe*, performing line search to obtain optimal weight (LAMBDA), computing the weighted average link volume, updating the link travel impedance with the volume-delay function, and feeding the updated link travel impedance back to *HRLDXY.ayy* to proceed to the next Frank-Wolfe iteration until the stopping criteria are met. The flow chart shown in Figure 4.6 represents the implementation logic in the *HASSFB.exe* program. The source code for the program, written in Visual Basic, is listed in Appendix A. The detailed functional description of this procedure is documented below.

4.8.3.1. Initial Trip Assignment

The basic idea of implementing this feedback procedure in FSUTMS is to continuously provide average congested link travel time information after traffic assignment of the current iteration back to the highway network file for the next iteration until the process converges. As depicted in the control file *TRNPL017.ins* above, the file *HNET.ayy* is the input file that provides the network information used to build minimum paths. The output from the highway assignment module (HASSIGN) is a loaded highway network contained in a file, *HRLDXY.ayy*. In comparison, *HRLDXY.ayy* contains additional data fields for storing data related to congested network conditions, but has the machine-language (binary) mode format similar to *HNET.ayy*. These data fields are: congested link impedance, daily capacity, and daily assigned link volume. Presumably, except for congested network conditions, the information is exchangeable between these two files through program manipulation if the binary format is known. Unfortunately, this information is not available.

The machine-generated characters in place of numbers and letters would prohibit direct retrieval and modification to the binary information. Alternatively, one utility program, *LODUNP.exe*, provided as part of the TRANPLAN utility programs, can be used to convert an internal unformatted form to an ASCII file entitled *HRLDXY.txt* (Urban Analysis Group, Inc., 1998). To execute *LODUNP.exe* in DOS environment, type "LODUNP" and respond to the program prompts for input and output files. An example based on the FTOWN network is given below:

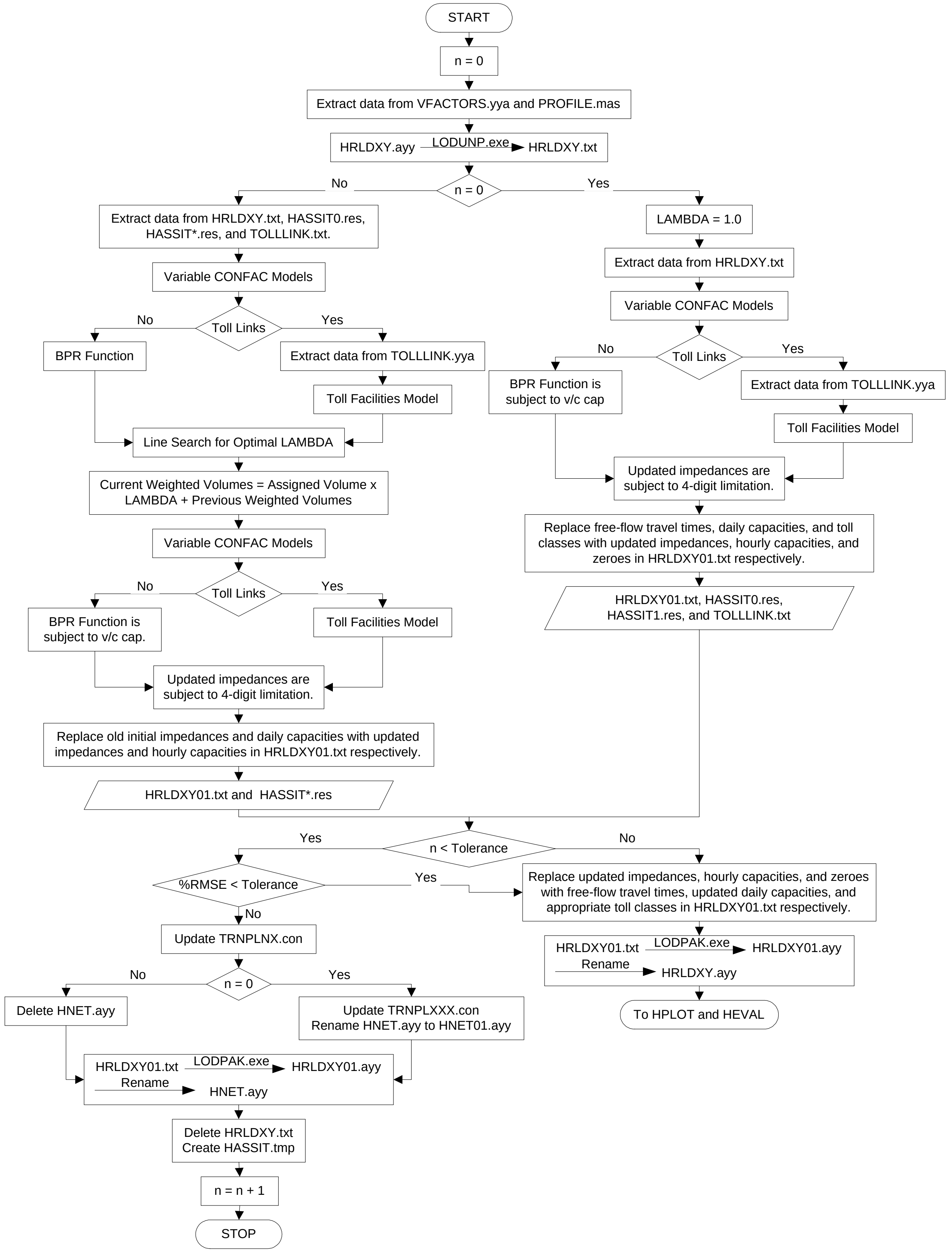

START
n = 0
Extract data from VFACTORS.yya and PROFILE.mas
HRLDXY.ayy
LODUNP.exe
HRLDXY.txt
n = 0
No
Yes
Extract data from HRLDXY.txt, HASSIT0.res, HASSIT*.res, and TOLLLINK.txt.
Variable CONFAC Models
Toll Links
No
Yes
BPR Function
Extract data from TOLLLINK.yya
Toll Facilities Model
Line Search for Optimal LAMBDA
Current Weighted Volumes = Assigned Volume x LAMBDA + Previous Weighted Volumes
Variable CONFAC Models
Toll Links
No
Yes
BPR Function is subject to v/c cap.
Toll Facilities Model
Updated impedances are subject to 4-digit limitation.
Replace old initial impedances and daily capacities with updated impedances and hourly capacities in HRLDXY01.txt respectively.
HRLDXY01.txt and HASSIT*.res
LAMBDA = 1.0
Extract data from HRLDXY.txt
Variable CONFAC Models
Toll Links
No
Yes
BPR Function is subject to v/c cap
Extract data from TOLLLINK.yya
Toll Facilities Model
Updated impedances are subject to 4-digit limitation.
Replace free-flow travel times, daily capacities, and toll classes with updated impedances, hourly capacities, and zeroes in HRLDXY01.txt respectively.
HRLDXY01.txt, HASSIT0.res, HASSIT1.res, and TOLLLINK.txt
n < Tolerance
Yes
No
%RMSE < Tolerance
Yes
No
Update TRNPLNX.con
n = 0
No
Yes
Delete HNET.ayy
Update TRNPLXXX.con
Rename HNET.ayy to HNET01.ayy
HRLDXY01.txt
LODPAK.exe
HRLDXY01.ayy
Rename
HNET.ayy
Delete HRLDXY.txt
Create HASSIT.tmp
n = n + 1
STOP
Replace updated impedances, hourly capacities, and zeroes with free-flow travel times, updated daily capacities, and appropriate toll classes in HRLDXY01.txt respectively.
HRLDXY01.txt
LODPAK.exe
HRLDXY01.ayy
Rename
HRLDXY.ayy
To HPLOT and HEVAL

Figure 4.6. Flow Chart of the Equilibrium Computation Procedure.

```
C:\FSUTMS.V54\FTOWN>LODUNP
 Enter input file name>HRLDXY.A00
 Enter output file name>HRLDXY.TXT
 Delete excessive node coordinates (Y/N)>Y
LODUNP Normal Stop
```

Note that, if the output file exists, the program will stop and generate the fatal error message: "cannot overwrite existing file". One needs to delete the existing file before the program can function correctly. Although *HRLDAY.ayy* and *HNET.ayy* seem to share a similar format, *LODUNP.exe* can only convert *HRLDAY.ayy*. Attempting to convert *HNET.ayy* to text form will generate the error message: "too many bytes read from unformatted record." For these reasons, *HRLDAY.ayy* is used to perform the subsequent data manipulation rather than using *HNET.ayy*.

The example above demonstrates the manual execution of the utility program. To automate the feedback procedure, this program can be executed as follows to minimize unnecessary manual manipulations.

```
C:\FSUTMS.V54\FTOWN>LODUNP < IPLODUNP.tmp
```

where *IPLODUNP.tmp* is a text file containing the following statements.

```
HRLDXY.a00
HRLDXY.txt
Y
```

The *HRLDXY.ayy* file stores the link-based output from trip assignment. The data format is illustrated with one link of the FTOWN network in the following annotated form:

```
 1235 12401 140T 240 240 1504099  4420  31001   0   0T ----► Link attributes
     257 ----► Congested impedance
    3247 -------------------------► Daily assigned volume
```

The configuration shown above was running one equilibrium iteration for trip assignment to meet the needs of programming logic. All link attributes (except congested impedances and daily assigned volumes) are printed in the first line of the link-specific block, leading by the beginning node (ANODE) and the ending node (BNODE). The specific formats and locations of each field for the first line are summarized in Table 4.1. The trip distance, TIME1, and TIME2 are displayed in integer form using units of hundredths of a mile and hundredths of a minute. The daily assigned link volume displayed at the third line is a measure resulting from AON assignment. The second line displays the congested impedance corresponding to the AON assigned volume. This is obtained through the computation procedures of BPR function with mean CONFACs and FSUTMS toll facilities model for non-toll and toll links, respectively.

Several files were created to record the required dataset arranged by order of successive links in the *HRLDXY.txt* for facilitating the computation process. A series of sequential files *HASSIT*.res* were used to record some inputs and outputs during the iterative procedure. The "*" is a numeral in agreement with the number of each iteration of equilibrium assignment. The file *HASSIT0.res* stores the initial network characteristics including mean CONFAC, free-flow travel time, and free-flow speed. The other *HASSIT*.res* files store the resulting output from the process of equilibrium computation procedure for each iteration, which contains AON assigned volume, weighted volume, variable CONFAC, congested impedance, and congested speed. The speed was calculated as the link distance divided by the free-flow/congested travel impedance. Another auxiliary file entitled *TOLLLINK.txt* records the sequence of toll links with the node information for the purpose of the FSUTMS tool facilities model.

Table 4.1. Data Format of Link Attributes in the HRLDXY File.

Column	Field	Description
1-5	ANODE	Node number where the trip is generated
6-10	BNODE	End node number of the trip
11	No. of Lanes	Number of lanes on the link
12-15	Link Distance	Length of the links connecting ANODE with BNODE
16	Field Option	Alphabetic code which indicates both Field 1 and Field 2 are to be interpreted as speed (S) or time (T)
17-20	Field 1	Link free-flow speed or travel time (SPEED1 or TIME1)
21-24	Field 2	Speed at a certain time or travel time, based on capacity and volume (SPEED2 or TIME2)
25-26	Direction Code	Numeric code which represents the direction of the link
27-28	Area Type	Type of areas where the trip is located (residential, rural, fringe, etc.)
29-30	Facility Type	Type of facility used to travel (freeway, collector, frontage road, etc.)
31-32	Screenline	Boundary limits of the traffic analysis area
33-38	Capacity	Daily capacity of the highway link
39-44	Counted Volume	Observed traffic volume
45	Geoloc	-
46-47	Unused	-
48-49	Toll Class	Numeric code which indicates the presence of toll on specific link
50-53	User Code	-

The congested impedances of non-toll links are computed based on the weighted volume; yet the congested impedances of toll links are computed on the basis of AON assigned volume. Since LAMBDA is equal to 1.0 for the first iteration of equilibrium assignment, the current weighted link volumes in the *HRLDXY.txt* file are equal to the daily assigned link volumes. The congestion measure was derived by dividing the current weighted volume by the number of lanes. According to CONFAC models based on facility type, the variable CONFAC corresponding to the respective congestion measure was applied on the v/c ratio in the BPR function for non-toll links as the following equations:

$$v / c_p = \frac{\text{Daily Weighted Volume} \times \text{Variable CONFAC}}{\text{Daily Capacity} \times \text{Mean CONFAC} \times \text{UROADF}} = \frac{\text{Hourly Weighted Volume}}{\text{Hourly Capacity } \times \text{UROADF}}$$

$$T_n = T_0 \times \left[1.0 + a \times \text{DAMPING} \times \left(v / c_p \right)^b \right]$$

where the UROADF parameter is used to adjust level-of-service E capacity to achieve practical capacity (generally at level-of-service C) and the DAMPING parameter is used to lessen the oscillation impact of loadings on an iteration-by-iteration basis.

Another practice minimizes the adjustment of link travel times as the DAMPING parameter is to place a cap on the v/c ratio. A default value of four is used in most FSUTMS models. As a consequence, the congested impedances were recalculated for non-toll links of the network. The resulting congested impedances were used to replace the free-flow travel times in the TIME1 field as the initial travel times for the next iteration through the EQUILIBRIUM HIGHWAY LOAD function.

A technique different from the recalculating process of the BPR function was utilized for toll links. Most of FSUTMS models with toll facilities have replaced the more simplified technique (the UROAD approach) with the comprehensive toll facilities model and the related file *TOLLLINK.yya*. This file contains information on the highway toll plaza characteristics in order to account for the costs and delays associated with using toll facilities. The FSUTMS toll facilities model was applied in the proposed procedure in order to model toll roads. Implementing the newer toll facilities model requires modification of option/parameter settings in *PROFILE.mas* as follows:

```
&TOLLFM
TOLL FACILITIES MODEL -------------------------► Invoke toll facilities model
&MULTSQ
MULTIPLE SERVER QUEUES------------------------► Invoke multiple server queue option
&GMTIME
TIME2-------------------------------------------► TIME2 used with toll mode
```

Toll links have only one approach link and one exit link. No other links can be connected inside the three-link segments. This arrangement indicates that the beginning node of toll link is the ending node of its upstream link; and the ending node of toll link is the beginning node of its downstream link. Accordingly, the free-flow speeds of upstream and downstream links for toll links were acquired from *HASSIT01.txt* that stores the initial network characteristics. The other required information on the network characteristics of each toll plaza was retrieved from the *TOLLLINK.yya* file. The travel impedances were then computed using the equations relating to the three-link segments as referred to above to replace the value of TIME1 in the first line of the link-specific block. The variable CONFAC models are also applied to the ratio of mean arrival rate over mean service rate (ρ), as follows:

$$\rho = \frac{\text{AON Assigned Volume} \times \text{Variable CONFAC}}{\left(3600/\text{SERVT}\right) \times \text{No. of Open Lanes through Toll Plaza}}$$

where the parameter SERVT represents the average uncongested service time at toll facilities associated with each toll link. A queuing system fitting the condition ρ <1 will eventually reach a steady-state condition. The multiple-sever queuing model was derived on the basis of this prerequisite condition. To apply this queuing model on the center segment of toll plaza, the ratio is set as 0.99 if the arrival rate equals or exceeds 0.99 times the service rate.

Note that TIME1 is confined to a four-digit location. The upper limit of the resulting congested impedances was set as 9999 (representing 99.99 minutes). In addition, the capacity in *HRLDXY.txt* is in daily equivalent, while it is in hourly equivalent in *HNET.ayy*. It is necessary to convert this data from daily units to hourly units by dividing it with corresponding mean CONFAC. Finally, the numeric code of the toll class for toll links were replaced with "0" to indicate non-toll links. The transformation is required for duplicating the toll facility modeling procedure in the subsequent equilibrium iterations. The updated information, now in the same format as *HRLDXY.txt,* was then saved in a file *HRLDXY01.txt*.

The stopping criterion for solving the UE program could be based on a convergence threshold of the relevant figures of merit or the maximum number of iterations specified by the user. If the convergence criterion is achieved in less number of iterations than specified, the computation procedure stops the iteration process and produces the output for a final set of congested assignments. Three categories of available information can be used to determine convergence: travel impedance, speed, and assigned link volume. In this study, the Percent Root Mean Square Error (%RMSE) that measures the impedances between two successive iterations is used and calculated as follows:

$$\%RMSE = \frac{\sqrt{\sum_i \left(\text{Congested Impedance}_i - \text{Initial Impedance}_i\right)^2 / \left(\text{Number of Links-1}\right)}}{\sum_i \text{Initial Impedance}_i \,/\, \text{Number of Links}} \times 100$$

Thus, the percent RMSE is a user-specified closure value that is utilized as a convergence criterion. On the other hand, another "counter" file named *HASSCT.con* was created to specify how many numbers of iterations have been performed. Once the convergence criterion is satisfied or the maximum number of iterations is achieved, the iterative process is terminated; otherwise, the next iteration is executed.

If any one of stopping criterion is met, the modification of some data field in *HRLDXY01.txt* is carried out again. The replaced link attributes, including TIME1 and toll class in the first line of the link-specific block, revert back to the original setting. Nonetheless, the hourly capacity should not only be converted back to daily equivalent, but should also reflect the effect of variable CONFAC models. Correspondingly, the updated daily capacity is calculated as the following:

$$\text{Updated Daily Capacity} = \text{Daily Capacity} \times \frac{\text{Variable CONFAC}}{\text{Mean CONFAC}}$$

The all-or-nothing assigned volume and travel impedance at the third and second lines are then replaced with the weighted link volume and corresponding congested impedance, respectively. For use as input to the sequent highway evaluation (HEVAL) and plotting (HPLOT) modules, *HRLDXY01.txt* was converted from ASCII format to binary file *HRLDXY01.ayy* by executing another utility program called *LODPAK.exe*. The *HRLDXY01.ayy* was finally renamed to standard output name: *HRLDXY.ayy*. To use this executable file, the procedure is essentially the same as that for the *LODUNP.exe* program described earlier. The following is an example of how to use this program in the DOS mode:

```
C:\FSUTMS.V54\FTOWN>LODPAK
 Enter input file name>HRLDXY01.txt
 Enter output file name>HRLDXY01.a00
LODPAK Normal Stop
```

Similarly, the manual procedure to execute the program can be automated as follows:

```
C:\FSUTMS.V54\FTOWN>LODPAK < IPLODPAK.tmp
```

where the text file *IPLODPAK.tmp* contains the following statements

```
HRLDXY01.txt
HRLDXY01.a00
```

4.8.3.2. Subsequent Trip Assignments

If the subsequent iteration is to be executed, the *HRLDXY01.txt* file should be converted and renamed to the input binary file *HNET.ayy* to generate a new network skim. The highway path building process will proceed based on this average skim for the equilibrium trip assignment during the next iteration. In spite of that, errors will result by simply performing all-or-nothing assignment with these updated initial travel impedances. This discrepancy may stem from the application of the toll facilities model.

The first iteration of the equilibrium assignment loads all trips to the network based on the predetermined free-flow network times shown in the link attributes TIME1 or TIME2. For toll links, the FSUTMS always specifies these values as zero if the toll facilities model is disabled. When the toll facilities model is implemented, the free-flow travel impedance for toll links is calculated as follows rather than using zero values:

$$\text{Delay at Plaza} = t_1 + t_3 + \text{SERVT} - \left(\frac{d_1}{V_1} + \frac{d_3}{V_3} \right)$$

$$\text{Time Penality} = \text{CTOLL} \times \text{TOLLS}$$

$$\text{Free-Flow Travel Impedance}_{\text{toll-link}} = \text{Delay at Plaza} + \text{Time Penality}$$

Because the proposed methodology requires the HASSIGN module to terminate for each iteration until the stopping criteria is satisfied, the application of the toll facility model assigns the free-flow travel impedance for toll links during every iteration of assignment. Directly replacing the old initial travel time in the TIME2 field to serve as a new initial travel time may double the count of the free-flow travel impedances for toll links. Furthermore, the variable CONFAC should be applied to the computation of congestion impedance for toll links. The toll facilities model is thus required to replicate these steps in order to complete the entire procedure.

Even though the newer toll facilities model has been incorporated into the standard FSUTMS equilibrium assignment procedure, the UROAD approach is still the default procedure when toll links are coded in the network. Unlike toll facilities model, which requires a separate file (*TOLLLINK.yya*) to describe toll facility characteristics, the UROAD approach to toll modeling requires three parameter settings specified in the *PROFILE.mas*: CTOLL, TOLLS, and SERVT.

For properly replicating the FSUTMS toll facilities model, the required option/parameter reference is adjusted. This is accomplished by modifying the entries in the control file *TRNPL017.ins* to execute the EQUILIBRIUM HIGHWAY LOAD function in the next iteration, as follows:

```
$EQUILIBRIUM HIGHWAY LOAD
$FILES
   INPUT FILE = HWYNET, USER ID = $HNET.A00$
   INPUT FILE = HWYTRIP, USER ID = $HTTAB.A00$
   INPUT FILE = TOLDATA, USER ID = $TOLLLINK.00A$
   OUTPUT FILE = LODHIST, USER ID = $HRLDXY.A00$
$HEADERS
TEST
            HIGHWAY ASSIGNMENT
$OPTIONS
~  TOLL FACILITIES MODEL          }
~  MULTIPLE SERVER QUEUES
~  MODEL CAPACITY
$PARAMETERS
   IMPEDANCE = TIME1----------------------->
   EQUILIBRIUM ITERATIONS = 1
   DAMPING FACTOR = 0.5
   EPS = 0.001
   CTOLL = 0.00
   TOLLS = 0.00
           0.00
   SERVT = 0.00
           0.00
$END TP FUNCTION
```

Insert the "~" characters to inactive the execution of the embedded toll facilities model.

TIME1 used with non-toll mode

The required parameters for the UROAD toll model are set as zero.

The embedded toll facilities model can be inactivated by inserting the tilde character (~) immediately before the required option statements. Moreover, the network impedance from which minimum paths will be built is altered in TIME1. Although these practices can disable the function of toll facilities model, the FSUTMS equilibrium assignment may default to the UROAD approach. For this reason, the required user-specified entries for three parameters should be replaced with zeroes to avoid invoking the UROAD toll model.

In general, the equilibrium computation procedure for the subsequent equilibrium iterations is basically similar to the process for the first iteration as described above. The additional process for the equilibrium iterations (other than the first iteration) only involves the replication of the FSUTMS toll facility model and the line search for the optimal value of LAMBDA.

After the output binary file *HRLDXY.ayy* was converted to the ASCII format, the link output was extracted to find a feasible solution to minimize the objective function of equilibrium assignment. The extracted AON assigned volumes for toll links were used directly as an input for the toll facilities model to compute the congested impedances. By contrast, the extracted assigned volumes for non-toll links were combined linearly with the weighted volumes from the previous iteration and were used with the BPR function. The effect of the FSUTMS toll facilities model was incorporated into the minimization problem by adding an extra term to objective function:

$$\min z(\mathbf{x}) = \sum_a \int_0^{x_a} t_a(w)dw + \sum_a t_a x_a$$

where $t_a(w)$ and t_a represent the travel impedances respectively for both non-toll and toll links.

The previous weighted volumes required for computing the "temporary" current weighted volume are stored in a corresponding *HASSIT*.res* file. The temporary current weighted volumes were calculated with λ = 0, 0.5, and 1 for the first iteration of the interval reduction method. The resulting travel impedances applied with the variable CONFAC models were then substituted into the respective objective functions. Following the interval reduction strategy, the algorithm involves the comparison of the objective function values at the interval end points and the interval discarding process in each iterative process. The procedure produces a final interval (a^n, b^n) when the dual stopping criteria are met. The optimal LAMBDA is the midpoint of the final interval. It should be noted that the v/c-ratio cap and the four-digit limitation are not applied to the travel impedance computation in the line search process.

The "true" current weighted volumes are obtained as linear combinations of the extracted assigned volumes and the previous weighed volumes using the optimal value of LAMBDA. The BPR function, integrated with variable CONFAC models, estimates the non-toll link travel impedances resulting from the current weighted volumes. This process utilizes the congested impedances of non-toll links along with the respective values of toll links that are produced during the line search for optimal LAMBDA as updated initial travel times to the subsequent assignment iteration. The whole procedure repeats until a near stable system emerges.

4.9. Procedure Verification

The basis for verifying the modified procedure is its ability to reproduce the assignment results from the original FSUTMS model for the exact same network and conditions. The replicated algorithm of equilibrium assignment, which is hereafter referred to as the constant CONFAC model, was achieved by applying the mean CONFAC values used for the corresponding urban area. Since the objective is to test the accuracy of replication, the assignment characteristics resulting from running the FSUTMS model provide a benchmark for comparison. This section compares the assigned characteristics produced by the constant CONFAC mode against FSUTMS results for two selected urban areas.

4.9.1. Test Networks

The modified procedure was tested in an equilibrium traffic assignment of the fictitious FTOWN network. The uncongested network of the FTOWN original setting could reach the stopping threshold within a few iterations. In order to better test the sensitivity of the developed methodology to the congested scenario, the original trip production data contained in *ZDATA1.yya* was increased. The population inflation was accomplished by uniformly factoring the dwelling unit and population in each of Traffic Analysis Zones (TAZs) by factors five and ten times for multi-family and single family, respectively.

The FSUTMS model was first executed to determine the number of iterations needed for equilibrium assignment. An insignificant assignment improvement was obtained after ten iterations for the modified network setting. The closure criterion (EPS) of HASSIGN module was set at 0.01 (the ratio of hours of travel during the weighted iteration over the hours of travel during the current iteration). The following tests were limited to ten iterations for each technique.

The procedure was also tested on a real-world network, Palm Beach County, Florida, to demonstrate the accuracy of the methodology developed here. This roadway network includes a total of 9661 directed links, 3909 nodes, and 1172 TAZs. The EPS 0.04 for the stopping criteria of equilibrium assignment was used, which corresponds to run four iterations.

4.9.2. Similarity Assessment

The HEVAL module of FSUTMS was developed to perform highway assignment analysis. The information of the loaded highway network was then retrieved from *HEVAL.out* produced by the HEVAL program. The assignment results consist of tabular summaries categorized by facility type, area types, and number of lanes. The evaluation criteria selected for analyzed and comparing two assignments are summarized below (FDOT, 1997):

- Total Volume on All Links – assignment volumes by category summed for all links in the network.
- Total VMT and Total VHT for All Links – vehicle miles traveled and vehicle hours traveled by category calculated from link volumes summed for all links (including those without traffic counts).
- Screenline Volume – screenline summaries of total assignment volumes.

The similarity tests of both assignment techniques as measured by different characteristics are shown in Tables 4.2-4.5. For the case of FTOWN, the discrepancy percentages for Total Volume on All Links and Total VMT for All Links Using Volumes are less than 0.001% and 0.01% for total area and all scales of different categories, respectively. Except the outlying business district (OBD) area type that underestimates by as much as 0.2241%, the discrepancy percentage for the other scales of Total VHT for All Links Using Volumes are less than 0.1%. The Screenline Volume shows that the discrepancy percentages for all six srceenlines are overestimated by less than 0.01%. All these small differences between the two techniques are believed to be the result of rounding errors and the use of different precisions for variables in the two procedures.

In the case of Palm Beach County, the assignment characteristics generally have higher divergence against the results of FSUTMS model as expected because the network is much larger than FTOWN's. Regardless of the measures of evaluation, however, the majority of discrepancy percentages are smaller than 1%. A breakdown by facility types indicates that the undivided arterials tend to be over-assigned by less than 2%, but toll facilities tend to be under-assigned by no greater than 7%. This relatively large discrepancy was produced by the four-digit limitation on the updated initial travel impedances. The resulting congested impedances were extremely large for some links, especially toll facilities, during the equilibrium iterative procedure. To deal with the problem of convergence, the Miami model set a cap 320 minutes for the computation of link-congested impedance. The area type breakdown indicates that the rural area tends to be under-assigned by about 2%.

The counted volumes are additionally used to test the closeness between two assignment techniques. The assigned volumes of FSUTMS model were separated into the groups based on counted volumes, and the percent RMSE was made for each group. The statistical measure was calculated by the formula below:

$$\%RMSE = \frac{\sqrt{\sum_i \left(\text{Assigned Volume}_i - \text{Counted Volume}_i\right)^2 / \left(\text{Number of Links-1}\right)}}{\sum_i \text{Counted Volume}_i \, / \text{ Number of Links}} \times 100$$

Table 4.6 was compiled to show a comparison of percent RMSE stratified by volume group. The corresponding values are depicted in Figure 4.7. The FSUTMS and CONFAC models produced almost identical errors because the two distributions essentially overlap. The differences between both procedures are negligibly small.

Table 4.2. Comparison of Total Volume on All Links by Scale.

Scale of Computation	**FTOWN**			**Palm Beach County**		
	FSUTMS	Constant CONFAC	Discrep.%	FSUTMS	Constant CONFAC	Discrep.%
Area	2199409	2199416	+0.0003	67851496	67986720	+0.1993
Facility Type						
- Freeways	573213	573216	+0.0005	11577671	11607814	+0.2604
- Divided Art.	481320	481317	-0.0006	34084276	34217368	+0.3905
- Undivided Art.	988715	988721	+0.0006	8068589	8163273	+1.1735
- Collectors	127187	127189	+0.0016	9046176	9062910	+0.1850
- One-way Streets	28974	28973	-0.0035	1028593	1037920	+0.9068
- Ramps	-	-	-	1583887	1543181	-2.5700
- HOV	-	-	-	536581	538037	+0.2713
- Toll	-	-	-	1925727	1816219	-5.6866
Area Type						
- CBD	162971	162971	+0.0000	2436177	2439474	+0.1353
- CBD Fringe	477326	477324	-0.0004	1458864	1469076	+0.7000
- Residential	1004262	1004260	-0.0002	44148168	44306712	+0.3591
- OBD	474780	474788	+0.0017	15313417	15363390	+0.3263
- Rural	80070	80073	+0.0037	4494872	4408073	-1.9311
Number of Lanes						
- 1	96521	96522	+0.0010	1803643	1794262	-0.5201
- 2	237283	237285	+0.0008	11668250	11613414	-0.4700
- 3	415447	415449	+0.0005	12375073	12414458	+0.3183
- 4	602575	602575	0.0000	22186624	22243852	+0.2579
- 6	454884	454887	+0.0007	18595346	18697896	+0.5515
- 8	392699	392698	-0.0003	1222561	1222842	+0.0230

Table 4.3. Comparison of Total VMT for All Links by Scale.

Scale of Computation	FTOWN			Palm Beach County		
	FSUTMS	Constant CONFAC	Discrep.%	FSUTMS	Constant CONFAC	Discrep.%
Area	2598213	2598227	+0.0005	24498994	24506568	+0.0309
Facility Type						
- Freeways	423742	423742	+0.0000	6012896	6026661	+0.2289
- Divided Art.	798679	798674	-0.0006	10476552	10518919	+0.4044
- Undivided Art.	1165317	1165329	+0.0010	2739024	2784178	+1.6485
- Collectors	181501	181508	+0.0039	2907868	2911475	+0.1240
- One-way Streets	28974	28973	-0.0035	167911	170257	+1.3972
- Ramps	-	-	-	351413	347664	-1.0668
- HOV	-	-	-	277033	277664	+0.2278
- Toll	-	-	-	1566294	1469751	-6.1638
Area Type						
- CBD	438117	438118	+0.0002	369105	369482	+0.1021
- CBD Fringe	695360	695358	-0.0003	311049	313299	+0.7234
- Residential	874624	874622	-0.0002	16958176	17019232	+0.3600
- OBD	480977	480985	+0.0017	3738983	3750934	+0.3196
- Rural	109135	109144	+0.0082	3121678	3053623	-2.1801
Number of Lanes						
- 1	16067	16068	+0.0062	607303	606073	-0.2025
- 2	255637	255643	+0.0023	5026935	4965541	-1.2213
- 3	373173	373174	+0.0003	6277081	6293491	+0.2614
- 4	789730	789735	+0.0006	7011089	7033795	+0.3239
- 6	358018	358021	+0.0008	5358426	5389240	+0.5751
- 8	805588	805587	-0.0001	218159	218429	+0.1238

Table 4.4. Comparison of Total VHT for All Links by Scale.

Scale of Computation	FTOWN			Palm Beach County		
	FSUTMS	Constant CONFAC	Discrep.%	FSUTMS	Constant CONFAC	Discrep.%
Area	115112	115073	-0.0339	579456	580767	+0.2262
Facility Type						
- Freeways	9586	9586	+0.0000	122382	122704	+0.2631
- Divided Art.	41904	41892	-0.0286	266042	267237	+0.4492
- Undivided Art.	56230	56201	-0.0516	68405	69674	+1.8551
- Collectors	6245	6248	+0.0480	78882	78927	+0.0570
- One-way Streets	1146	1146	+0.0000	4656	4725	+1.4820
- Ramps	-	-	-	10527	10457	-0.6650
- HOV	-	-	-	4929	4940	+0.2232
- Toll	-	-	-	23633	22103	-6.4740
Area Type						
- CBD	25103	25104	+0.0040	12580	12718	+1.0970
- CBD Fringe	31093	31070	-0.0740	8550	8634	+0.9825
- Residential	34338	34365	+0.0786	408796	410629	+0.4484
- OBD	20976	20929	-0.2241	95614	95574	-0.0418
- Rural	3603	3605	+0.0555	53916	53211	-1.3076
Number of Lanes						
- 1	653	653	0.0000	15184	15153	-0.2042
- 2	8404	8407	+0.0357	117383	116870	-0.4370
- 3	8304	8304	0.0000	127603	128006	+0.3158
- 4	36947	36915	-0.0866	175226	175928	+0.4006
- 6	21157	21143	-0.0662	138221	138959	+0.5339
- 8	39647	39652	+0.0126	5839	5851	+0.2055

Table 4.5. Comparison of Screenline Volume.

Screenline	FTOWN			Palm Beach County		
	FSUTMS	Constant CONFAC	Discrep.%	FSUTMS	Constant CONFAC	Discrep.%
- #1	144814	144814	+0.0000	447932	447947	+0.0033
- #2	143816	143818	+0.0014	320243	321783	+0.4809
- #3	73378	73378	+0.0000	196638	195633	-0.5111
- #4	16332	16333	+0.0061	331825	332366	+0.1630
- #5	67140	67140	+0.0000	297893	298041	+0.0497
- #6	-	-	-	176250	177281	+0.5850
- #7	-	-	-	140124	140129	+0.0036
- #8	-	-	-	82470	82568	+0.1188
- #9	-	-	-	61537	61823	+0.4648
- #10	-	-	-	102375	102550	+0.1709
- #11	-	-	-	101619	101572	-0.0463
- #12	-	-	-	206867	206950	+0.0401
- #13	-	-	-	112486	111829	-0.5841
- #14	-	-	-	46637	47120	+1.0357
- #15	-	-	-	245484	245915	+0.1756
- #16	-	-	-	15285	15286	+0.0065
- #17	-	-	-	172906	173921	+0.5870
- #18	-	-	-	207902	208037	+0.0649
- #95	-	-	-	2136766	2139366	+0.1217
- #96	-	-	-	27086	27087	+0.0037
- #99	1105971	1105973	+0.0002	14295492	14341290	+0.3204

Table 4.6. Comparison of Percent RMSE at Different Counted Link Flow Levels.

Counted Volume	Links	%RMSE	
		FSUTMS	Constant CONFAC
1 - 1999	25	85.46	87.83
2000 - 3999	130	77.28	76.77
4000 - 5999	173	52.68	52.67
6000 - 7999	182	50.71	50.73
8000 - 9999	140	43.65	43.35
10000 - 11999	160	36.12	36.22
12000 - 13999	139	31.86	31.55
14000 - 15999	104	26.77	27.57
16000 - 17999	121	28.10	28.31
18000 - 19999	94	25.54	25.79
20000 - 24999	90	25.86	25.85
25000 - 29999	34	18.20	18.68
30000 - 49999	28	12.46	12.66
50000 - 99999	42	8.77	8.56

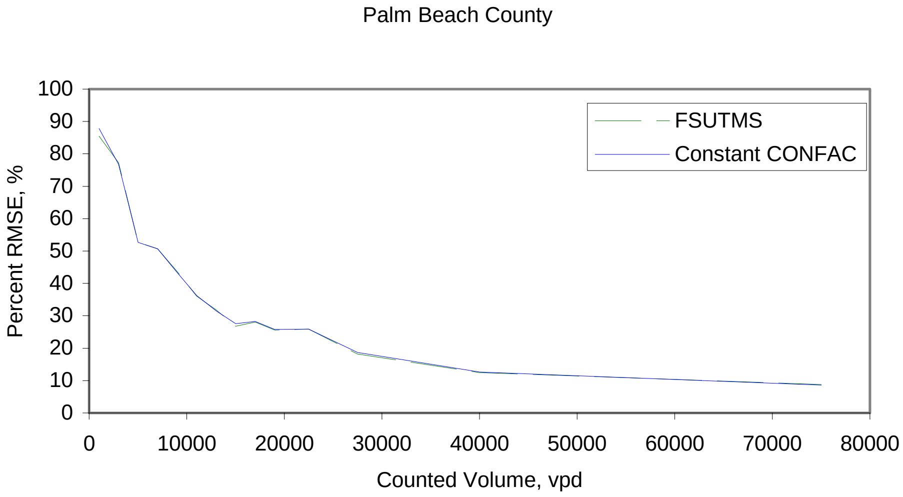


Figure 4.7. Comparison of %RMSE between FSUTMS Model and Constant CONFAC Mode.

CHAPTER 5
CASE STUDIES FOR VARIABLE CONFAC MODELS

5.1. Introduction

This chapter compares the assignment results before and after the implementation of the CONFAC functions (i.e., variable CONFACs) calibrated in Chapter 3. The original CONFACs, which were developed from a local travel characteristics survey and found in the *VFACTORS.yya* file that comes with each urban model data set, is hereafter referred to as the "constant CONFACs." The procedure developed and described in the previous chapter has been shown to adequately replicate the results from the HASSIGN module. Thus, any difference between before and after the implementation of the CONFAC functions can be attributed to use of variable CONFACs. These results are compared against the ground counts for three selected networks: Miami-Dade, Broward, and Palm Beach. In general, comparisons can be made based on aggregate and disaggregate measures. This study uses the following four measures:

- Assigned VHT to count VHT ratios.
- Assigned VMT to count VMT ratios.
- Screenline volume-count ratio.
- Percent root mean square error.

These measures are evaluated at different assessment levels, including areawide, screenline, facility type, area type, number of lanes, and volume level. Table 5.1 defines the different levels of assessment for each of the measures considered.

Table 5.1. Traffic Assignment Accuracy Levels (FDOT, 1981).

Measure	Assessment Level
Assigned VMT/Count VMT	Areawide, facility type, area type, number of lanes
Assigned VHT/Count VHT	Areawide, facility type, area type, number of lanes
Volume-Count Ratios	Screenline
Percent Root Mean Square Error	Areawide, volume level

5.2. Analysis and Comparison Results

The HEVAL module of FSUTMS for highway evaluation was run to generate the measurement values. Note that the extracted results only include the calculated and summed values for links with traffic counts coded in the network, not all links of the network. The assignment volumes for constant CONFAC scenario are those obtained after the default number of iterations of the equilibrium assignment for each urban area. The closure criterion of relative gap of travel impedance was then determined for the corresponding variable CONFAC scenario. The results of the analysis are shown in Table 5.2. The performance was then evaluated to examine the impact of different approaches on the assignment results. The network characteristics of the three urban areas are shown in Table 5.3.

Table 5.2. Closure Criteria Required to Reach Convergence.

Closure Criteria	Broward County	Miami-Dade County	Palm Beach County
%RMSE between Two Successive Travel Impedances	0.35	0.19	0.28
Number of Iteration needed for Constant CONFAC	6	12	4
Number of Iteration needed for Variable CONFAC	10	17	4

Table 5.3. Highway Network Parameters by Urban Area.

Characteristic	Broward County	Miami-Dade County	Palm Beach County
One-Way Links	11175	19304	9661
Nodes	4311	7473	3909
Internal Zones	892	1500	1150
External Zones	41	21	22

5.2.1. Assigned VMT to Count VMT Ratios

The ratio of assigned VMT to count VMT (AVMT/CVMT) measures the net error by summing over- and under-assignment errors. Table 5.4 presents the errors obtained for the comparison of the AVMT/CVMT ratio between the different approaches for each of the three urban areas. The percent error was computed as follows:

$$Percent\ Error = \frac{AVMT - CVMT}{CVMT} \times 100$$

A zero percent indicates no bias between the assigned and observed VMT. In this case, the positive errors obtained from over-assignments are canceled out by adding the negative errors obtained from under-assignments.

As illustrated in Figure 5.1, there is no significant bias in total assigned VMT for constant and variable CONFAC methods for all three urban models. Figures 5.2-5.4 show a variation of over- and under-assignments for different facility types, area types, and numbers of lanes. The notations used in the plots are defined below:

F1: Freeways and Expressways
F2: Divided Arterials
F3: Undivided Arterials
F4: Collectors
F5: Centroid Connectors
F6: One-Way Facilities
F7: Ramp
F8: HOV Facilities
F9: Toll Facilities

A1: CBD Areas
A2: CBD Fringe Areas
A3: Residential Areas
A4: OBD Areas
A5: Rural Areas

Table 5.4. Accuracy of Ratio of Assigned VMT to Count VMT by Scale.

Scale of Computation	**Broward County**		**Miami-Dade County**		**Palm Beach County**	
	Constant CONFAC	Variable CONFAC	Constant CONFAC	Variable CONFAC	Constant CONFAC	Variable CONFAC
Area	-0.78%	-1.75%	-1.71%	**+1.01%**	+0.60%	+0.81%
Facility Type						
- Freeways	-4.83%	**-1.89%**	-5.12%	+8.84%	-1.02%	+3.80%
- Divided Art.	+1.84%	+2.20%	+2.71%	**-0.40%**	+1.74%	**+0.79%**
- Undivided Art.	+0.50%	+1.14%	+1.33%	-7.43%	-0.60%	-4.98%
- Collectors	+3.02%	-5.43%	+10.61%	**-1.23%**	+7.38%	**+3.51%**
- One-way Streets	-24.47%	-26.33%	-8.32%	-14.90%	+1.83%	-4.02%
- Ramps	-46.17%	-60.78%	-	-	-	-
- HOV	-2.02%	-4.01%	-13.60%	**-10.37%**	+0.44%	+0.81%
- Toll	-3.26%	-10.84%	-5.46%	-17.10%	-12.50%	-13.11%
Area Type						
- CBD	+4.62%	-5.85%	+9.39%	+13.40%	-4.01%	-6.55%
- CBD Fringe	+2.95%	**-0.66%**	-	-	-2.67%	-3.88%
- Residential	+0.08%	-3.55%	-2.73%	**+0.07%**	+0.66%	+1.07%
- OBD	-1.38%	**-0.65%**	+0.60%	+6.03%	+2.49%	+2.64%
- Rural	+1.03%	+1.03%	-11.56%	-29.00%	-1.82%	-2.19%
Number of Lanes						
- 1	-2.02%	-4.01%	-6.70%	-11.90%	+0.44%	+0.81%
- 2	-8.08%	-14.94%	+3.79%	**-1.41%**	-2.34%	-4.75%
- 3	-3.50%	-4.97%	-3.99%	**-0.64%**	-0.96%	+3.74%
- 4	+0.79%	**-0.25%**	-5.26%	+9.37%	+2.84%	**+1.56%**
- 5	-4.32%	**+0.28%**	-	-	-	-
- 6	+1.62%	+1.88%	+22.40%	+36.51%	+2.02%	**-0.12%**
- 8	+21.75%	+23.22%	-	-	-5.55%	**-3.12%**
- 9+	-0.82%	-0.82%	-	-	-	-

In general, the biases are larger for the variable CONFAC model than for the constant CONFAC model (see Figures 5.2 – 5.4), but a number of exceptions exist. With respect to facility type and area type, the variable CONFAC model does not improve consistently for different study networks (see Figures 5.2 and 5.3). However, the bias tends to be lower for the higher number of lanes when the variable CONFAC model is applied for Broward and Palm Beach, but higher for Miami (see Figure 5.4).

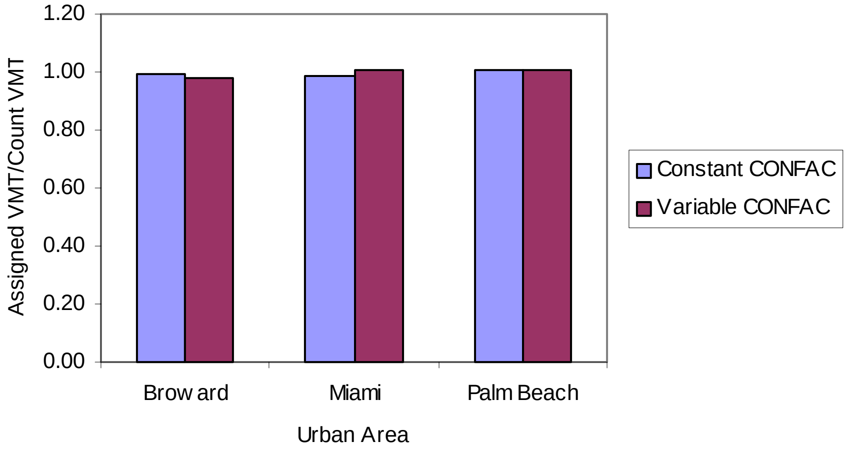
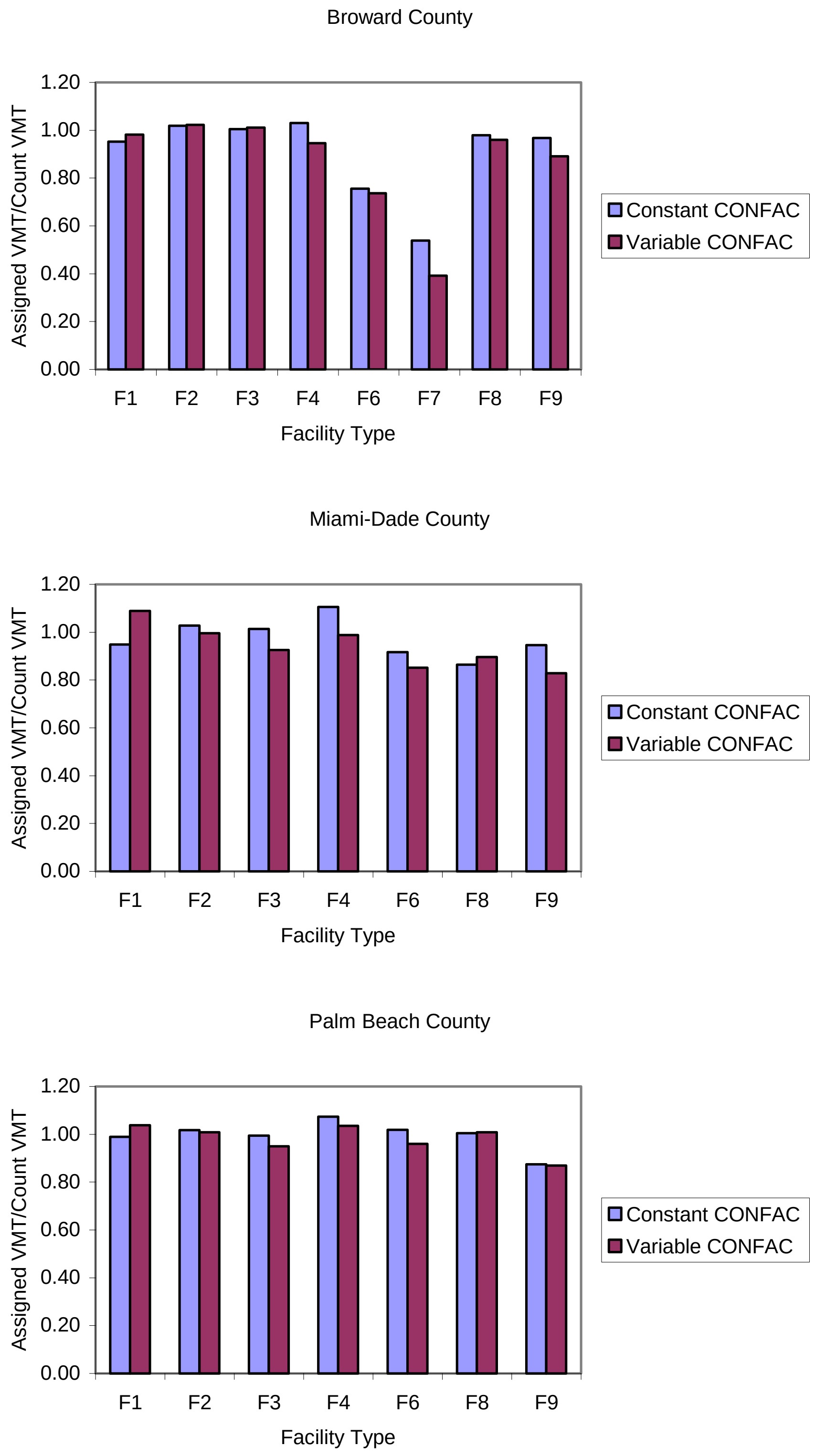


Figure 5.1. Comparison of Area-wide Assigned VMT to Count VMT Ratios.


Broward County

Miami-Dade County

Palm Beach County

Figure 5.2. Comparison Assigned VMT to Count VMT Ratios by Facility Type.

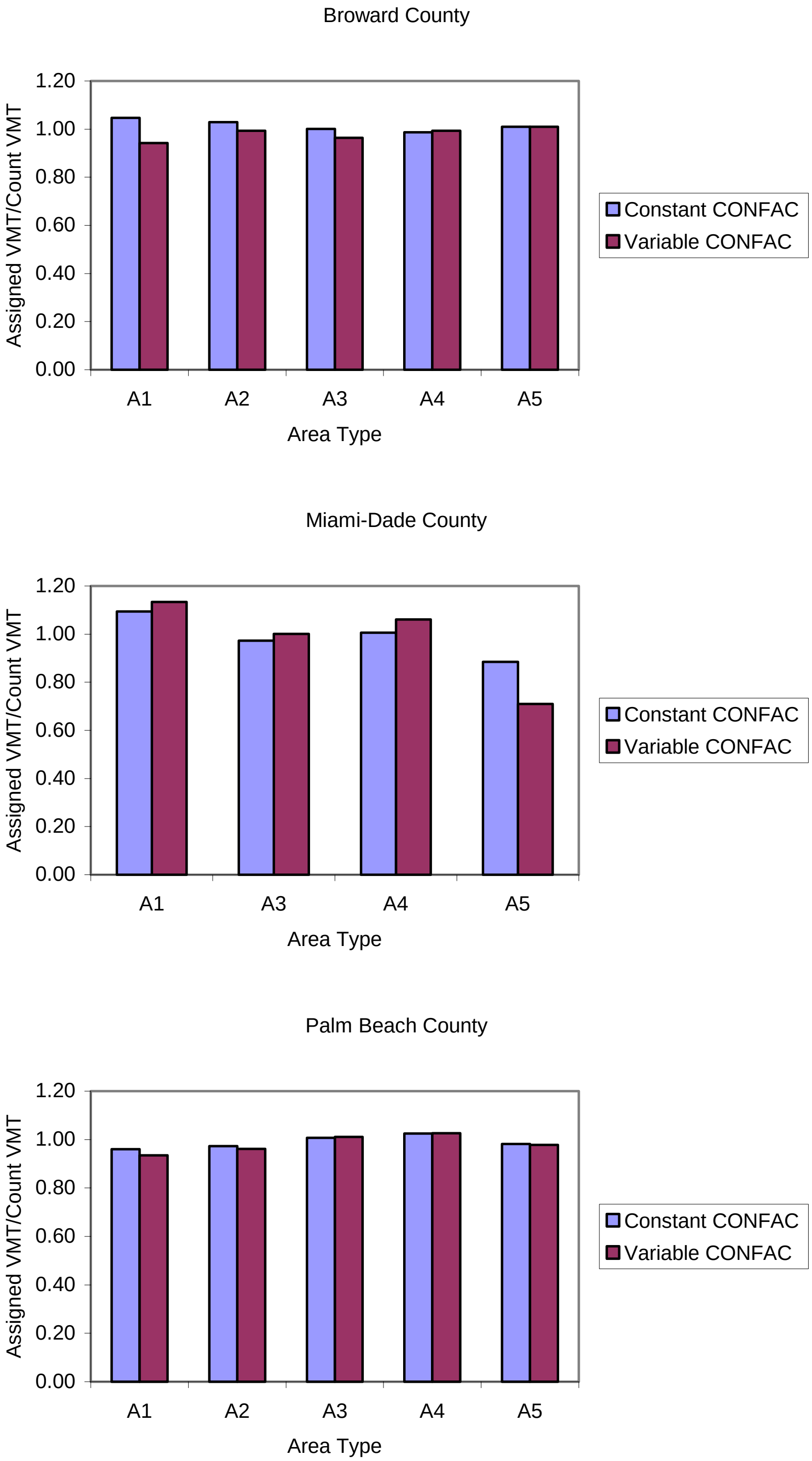


Figure 5.3. Comparison of Assigned VMT to Count VMT Ratios by Area Type.

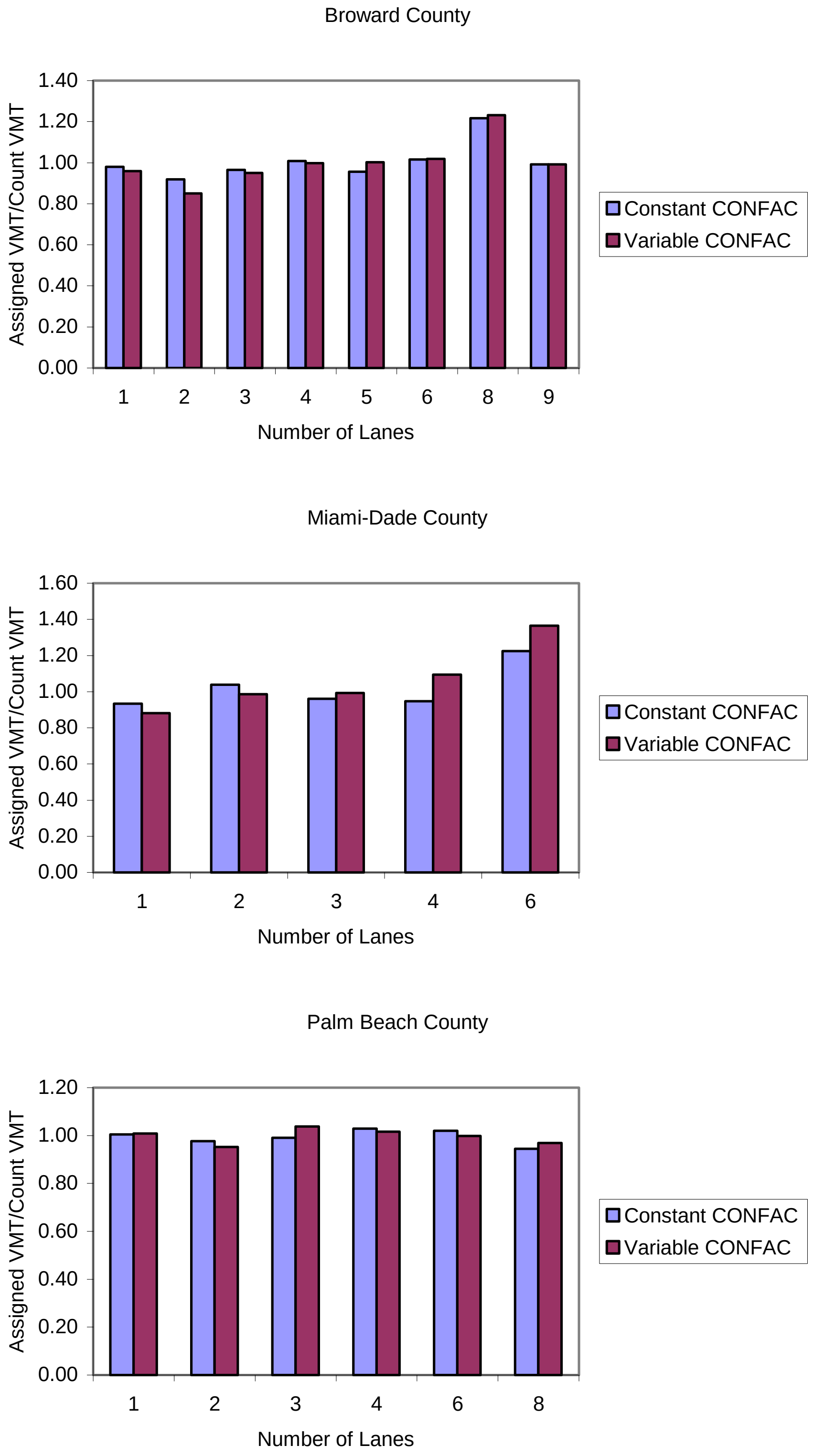


Figure 5.4. Comparison of Assigned VMT to Count VMT Ratios by Number of Lanes.

5.2.2. Assigned VHT to Count VHT Ratios

The ratio of assigned to counted vehicle hours of travel (AVHT/CVHT) provides another measure of the net over- or under-assignments. The results of area-wide and stratification by different assessment levels are given in Table 5.5. As illustrated in Figure 5.5, the two models produce comparable total AVHT/CVHT ratio for Broward and Palm Beach, but the variable CONFAC model tends to over-assign traffic for the Miami network. The pattern for assigned VHT for three urban areas is similar to that for VMT bias by all assessment levels as shown in Figures 5.6 through 5.8.

Table 5.5. Accuracy of Assigned VHT to Count VHT Ratios.

Scale of Computation	**Broward County**		**Miami-Dade County**		**Palm Beach County**	
	Constant CONFAC	Variable CONFAC	Constant CONFAC	Variable CONFAC	Constant CONFAC	Variable CONFAC
Area	+0.93%	**-0.41%**	+2.91%	+4.48%	+1.11%	**+0.80%**
Facility Type						
- Freeways	-5.30%	**-2.10%**	-2.74%	+11.95%	-0.75%	+4.20%
- Divided Art.	+2.67%	**+2.64%**	+6.34%	**+2.50%**	+1.66%	**+0.44%**
- Undivided Art.	+1.60%	+2.16%	+11.10%	**+0.72%**	-2.11%	-6.59%
- Collectors	+4.06%	-4.81%	+22.33%	+10.22%	+9.32%	**+4.74%**
- One-way Streets	-1.15%	-8.26%	-2.86%	-11.20%	+2.04%	-3.75%
- Ramps	-46.00%	-60.67%	-	-	-	-
- HOV	-1.71%	-3.75%	+3.71%	**+2.12%**	+0.49%	+0.84%
- Toll	-2.55%	-10.17%	-3.08%	-13.02%	-12.51%	-13.10%
Area Type						
- CBD	+3.52%	-6.90%	+13.60%	+13.89%	-3.58%	-5.38%
- CBD Fringe	+2.65%	**-2.21%**	-	-	-4.56%	-5.86%
- Residential	+1.87%	-2.44%	+1.46%	+3.06%	+1.14%	**+0.93%**
- OBD	+0.26%	+1.04%	+4.01%	+6.96%	+3.08%	**+2.80%**
- Rural	+0.99%	+0.99%	-0.33%	-16.75%	-0.92%	-1.30%
Number of Lanes						
- 1	-1.71%	-3.75%	+5.87%	**-0.45%**	+0.49%	+0.84%
- 2	-0.79%	-8.60%	+10.02%	**+4.21%**	+0.31%	-2.58%
- 3	-5.24%	-5.59%	-1.60%	**+0.97%**	-0.68%	+4.12%
- 4	+1.89%	**-0.20%**	-0.26%	+14.26%	+2.45%	**+0.79%**
- 5	-4.08%	**+0.47%**	-	-	-	-
- 6	+2.11%	+2.44%	+22.40%	+36.51%	+2.06%	**-0.39%**
- 8	+22.55%	+24.37%	-	-	-4.69%	**-2.54%**
- 9+	-0.87%	-0.87%	-	-	-	-

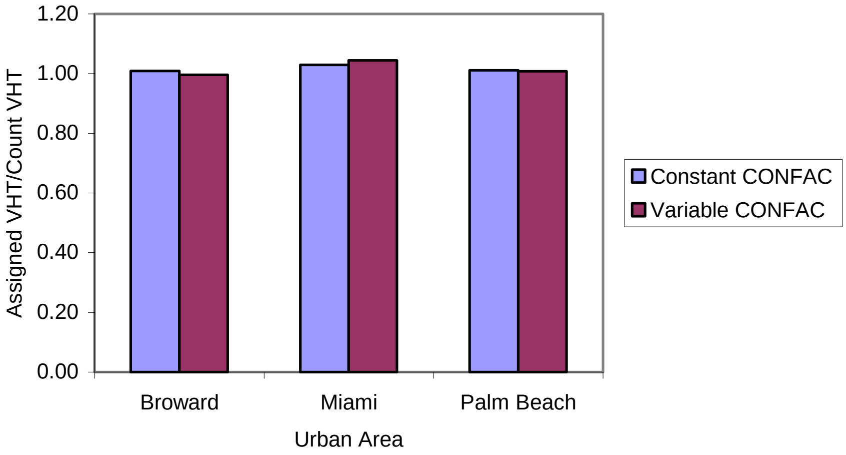


Figure 5.5. Comparison of Area-wide Assigned VHT to Count VHT Ratios.

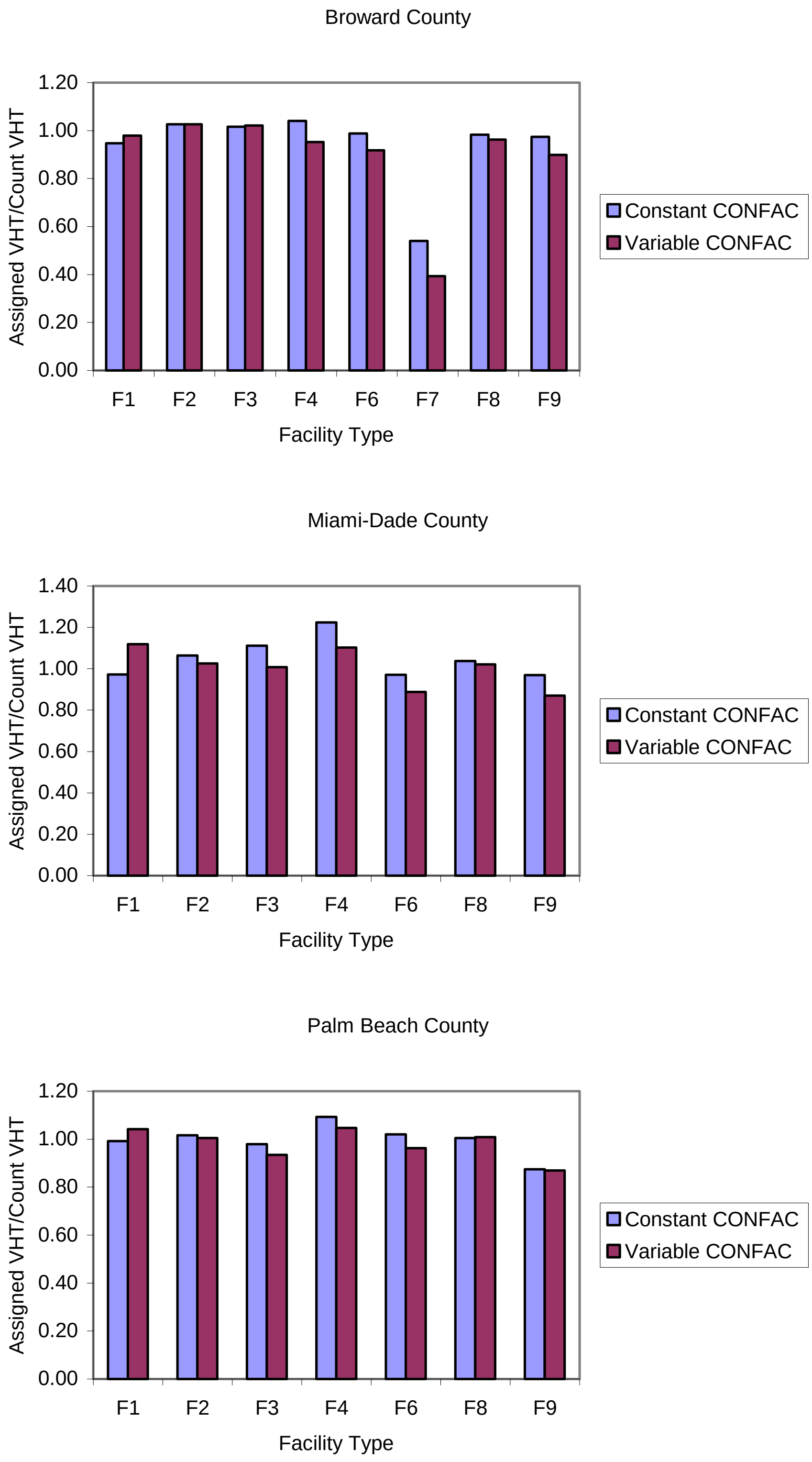


Figure 5.6. Comparison of Assigned VHT to Count VHT Ratios by Facility Type.

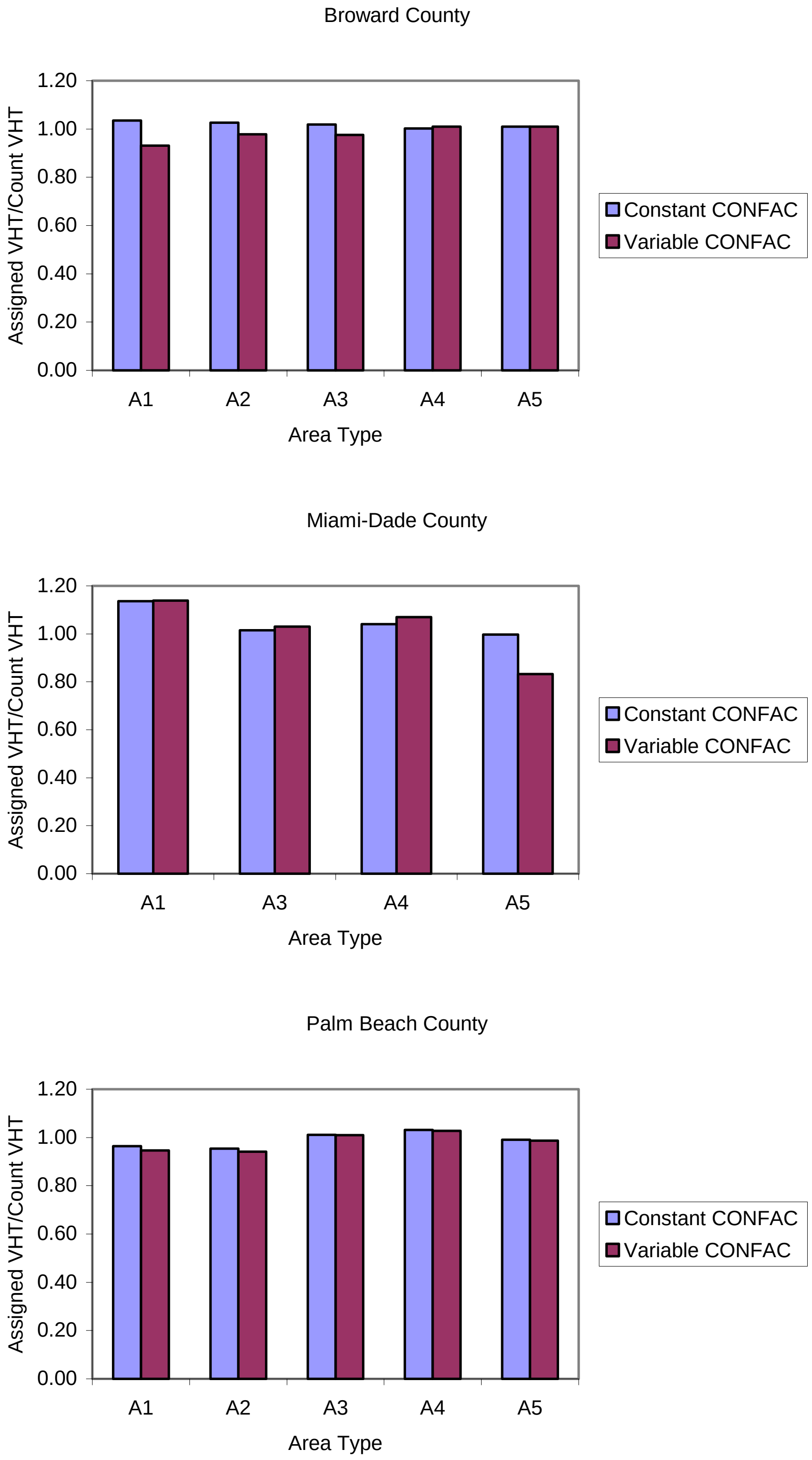
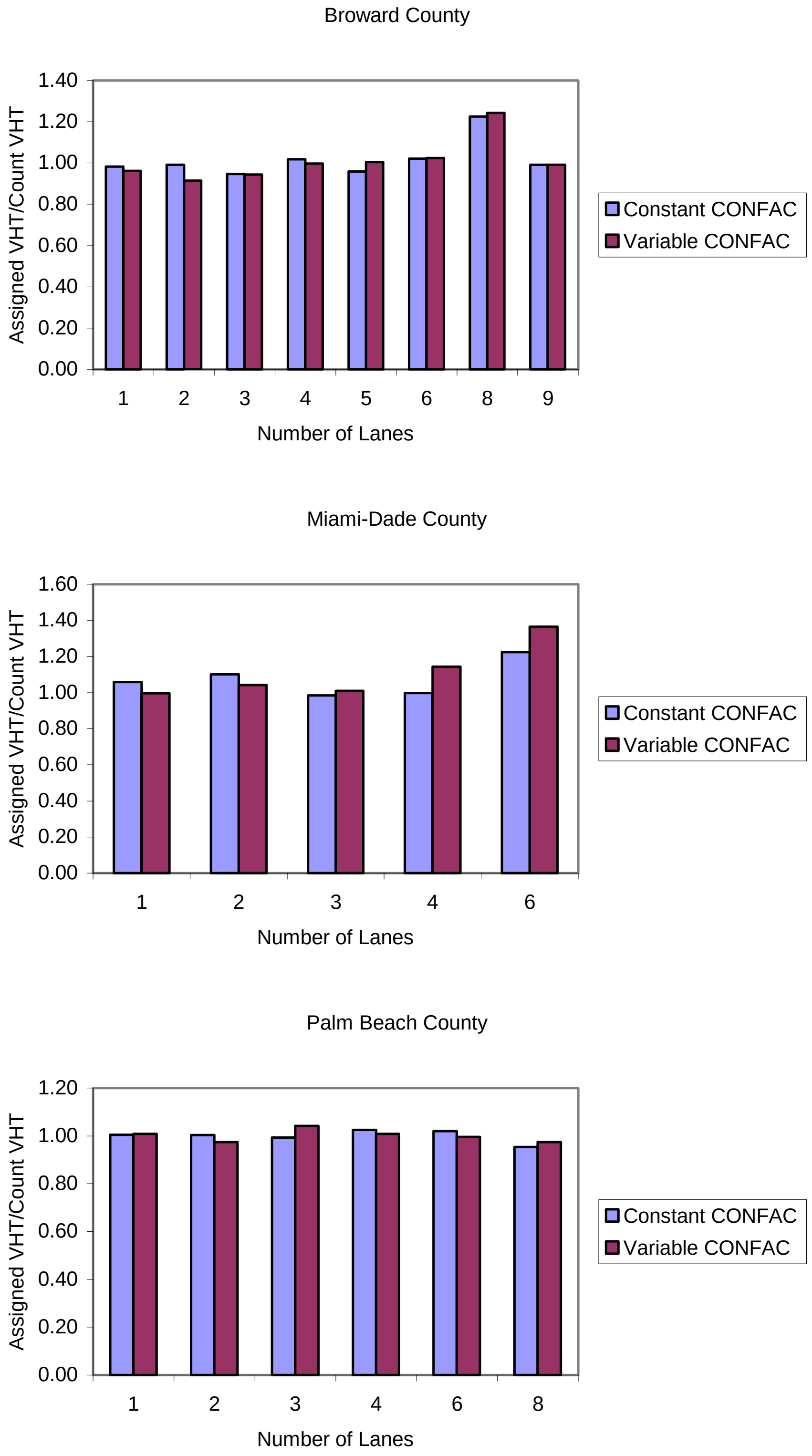


Figure 5.7. Comparison of Assigned VHT to Count VHT Ratios by Area Type.



Broward County

Miami-Dade County

Palm Beach County

Figure 5.8. Comparison of Assigned VHT to Count VHT Ratios by Number of Lanes.

5.2.3. Screenline Volume-Count (v/c) Ratios

The volume-count (v/c) ratios for screenlines are given in Table 5.6. For each screenline, the assigned volumes should be compared to the ground counts to check the compatibility of highway network. Even though errors are reduced for some screenlines, Figure 5.9 shows some increases in errors when the variable CONFAC model is applied. It should be noted that ten of the 14 ratios failed to meet the suggested accuracy range (±10% for assigned volume more than 50,000 vpd) in the constant CONFAC model for Miami network.

Table 5.6. Accuracy of Volume-Count Ratios for Screenline.

Screenline	**Broward County**		**Miami-Dade County**		**Palm Beach County**	
	Constant CONFAC	Variable CONFAC	Constant CONFAC	Variable CONFAC	Constant CONFAC	Variable CONFAC
#1	-4.54%	-4.59%	-48.85%	-49.03%	-0.35%	-0.35%
#2	-5.77%	-5.79%	-26.20%	-28.77%	+4.97%	**+4.49%**
#3	+7.64%	**+6.82%**	-10.03%	-17.00%	+14.34%	**+7.65%**
#4	-10.96%	-11.08%	+7.82%	**-1.16%**	+0.31%	+4.12%
#5	+8.96%	**+8.52%**	+3.49%	-4.49%	+4.98%	**+4.87%**
#6	-2.19%	-2.35%	+4.30%	**-4.17%**	+4.28%	+4.48%
#7	-0.94%	-0.94%	+19.12%	**+8.58%**	+0.22%	+0.22%
#8	-0.95%	-0.95%	-26.83%	-29.89%	+41.83%	+41.98%
#9	+0.04%	+0.04%	-20.88%	-24.74%	-24.80%	-26.12%
#10	-	-	-10.34%	-15.27%	-5.50%	**-3.94%**
#11	-	-	-1.72%	-3.42%	-2.08%	-5.01%
#12	+19.72%	**+16.52%**	-28.43%	**-23.87%**	+3.82%	+3.83%
#13	-18.11%	-19.33%	-53.28%	**-52.08%**	-10.21%	**-9.30%**
#14	+0.33%	-0.62%	-	-	-0.53%	-1.42%
#15	-	-	-	-	+15.99%	**+15.46%**
#16	-	-	-	-	-0.24%	-0.24%
#17	-	-	-	-	-15.46%	-16.17%
#18	-	-	-	-	-10.99%	-12.18%
#59	-2.72%	**-2.00%**	-	-	-	-
#75	-3.59%	-4.26%	-	-	-	-
#82	+22.80%	**+3.98%**	-	-	-	-
#86	-13.67%	-25.50%	-	-	-	-
#91	+5.23%	**+0.00%**	-	-	-	-
#95	-7.99%	**-3.51%**	-	-	+0.44%	+5.95%
#96	-	-	-	-	+0.32%	+0.32%
#99	+3.75%	**+2.21%**	-16.74%	-19.39%	-0.72%	-2.32%

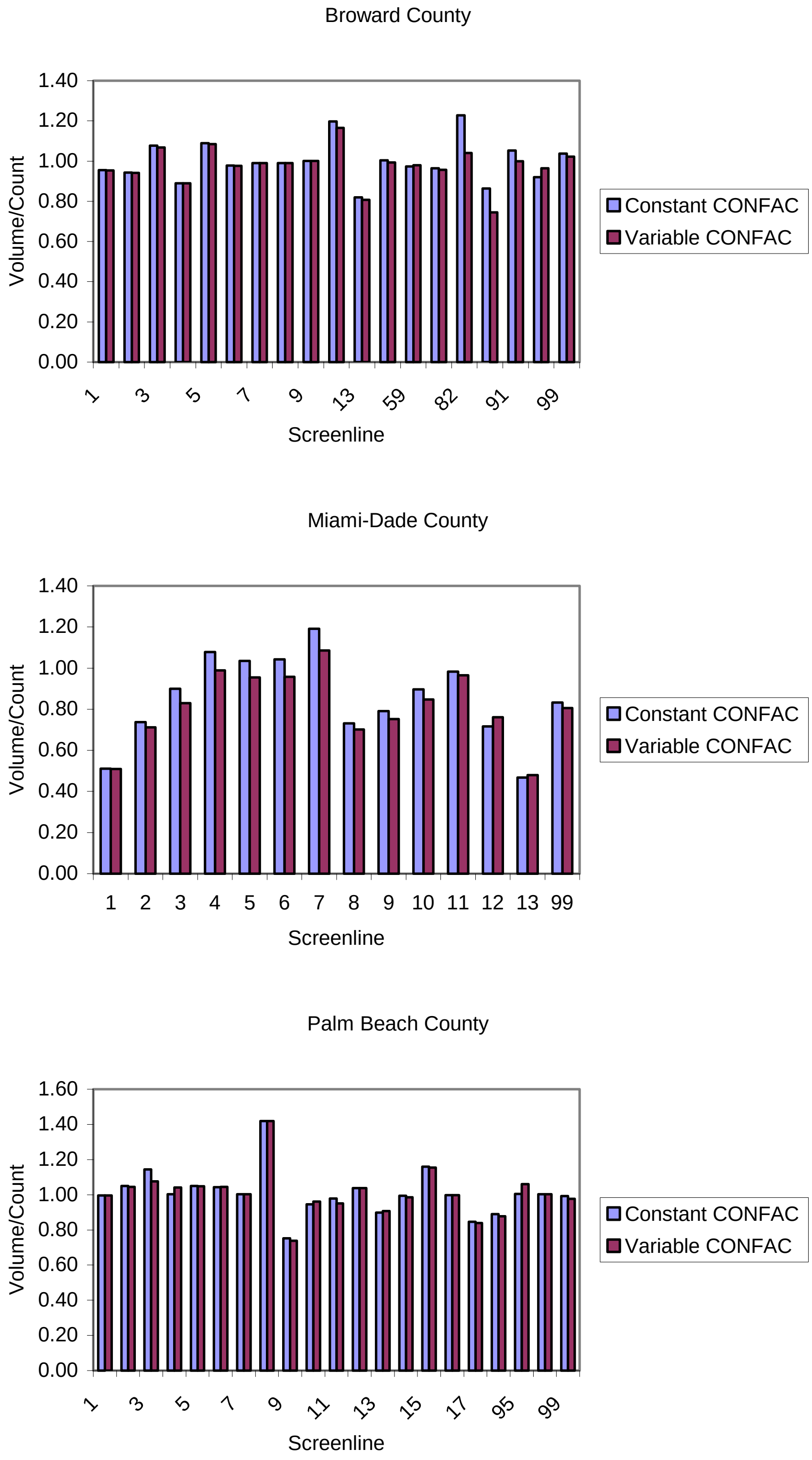


Figure 5.9. Comparison of Volume-Count Ratio.

5.2.4. Percent Root Mean Square Error

The Percent Root Mean Square variable (%RMSE) measures the total error by summing up the square of each of over- and under-assignment errors. The results of %RMSE of the predicted volumes from the two approaches are compared in Table 5.7. A lower value indicates a better match of the assigned volumes with the ground counts. The %RMSE of total area was first examined as a single measure of accuracy. Figure 5.10 shows that the variable CONFAC model and the constant CONFAC model produce similar values for %RMSE for Broward and Palm Beach, but the %RMSE for the variable CONFAC model is higher for Miami. As Figure 5.11 shows, the %RMSE decreases as the estimated count volume increases. This indicates that a better fit between assigned volumes and ground counts occurs at higher observed volumes, which is expected because the variance of volume group tends to be smaller with increasing count volume. It was also observed that both approaches overestimated under the low ground-count condition while underestimated under the middle and high ground-count conditions. This indicates that the BPR function or the equilibrium assignment algorithm itself tended to obliquely assign the trips.

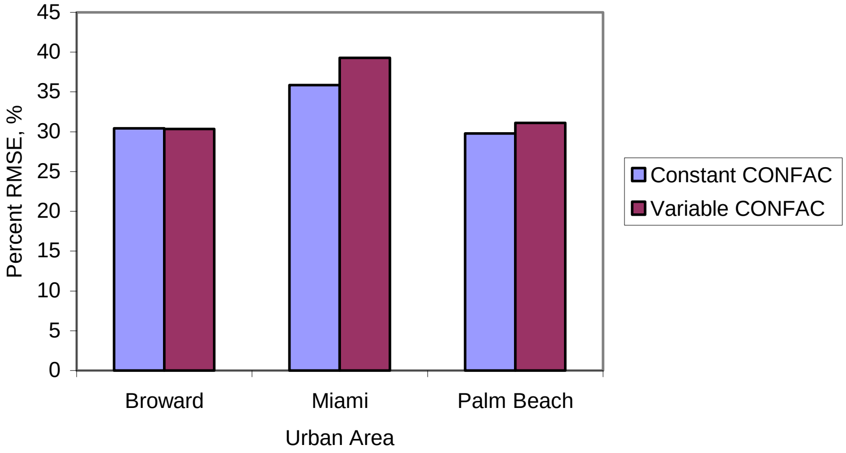
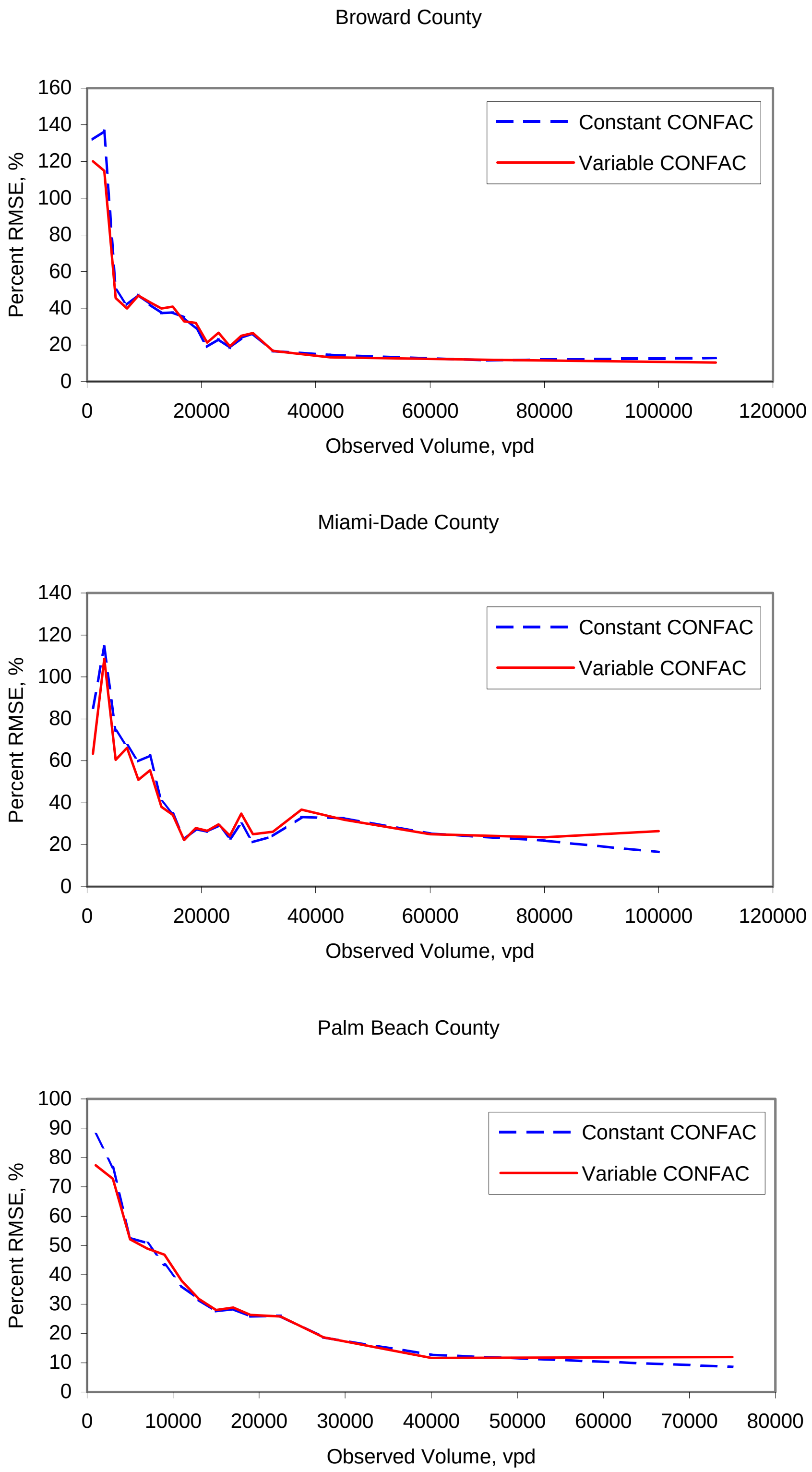


Figure 5.10. Comparison of Areawide Percent Root Mean Square Errors.

Table 5.7. Comparison of Percent Root Mean Square Errors by Detailed Assessment Level.

Broward County				Miami				Palm Beach County			
Assessment Level	Links	Constant CONFAC	Variable CONFAC	Assessment Level	Links	Constant CONFAC	Variable CONFAC	Assessment Level	Links	Constant CONFAC	Variable CONFAC
Area	1944	30.42%	**30.33%**	Area	1498	35.86%	39.30%	Area	1462	29.77%	31.11%
Volume Group				Volume Group				Volume Group			
1 - 1999	48	131.97%	**120.05%**	1 - 1999	50	85.20%	**63.34%**	1 - 1999	25	87.83%	**77.36%**
2000 - 3999	143	136.76%	**114.92%**	2000 - 3999	77	114.35%	**108.66%**	2000 - 3999	130	76.77%	**72.78%**
4000 - 5999	193	49.94%	**45.53%**	4000 - 5999	87	74.20%	**60.46%**	4000 - 5999	173	52.67%	**52.07%**
6000 - 7999	202	41.81%	**39.86%**	6000 - 7999	106	67.22%	**66.18%**	6000 - 7999	182	50.73%	**48.95%**
8000 - 9999	195	47.40%	**46.90%**	8000 - 9999	125	59.82%	**51.03%**	8000 - 9999	140	43.35%	46.86%
10000 - 11999	117	42.26%	43.15%	10000 - 11999	134	62.47%	**55.50%**	10000 - 11999	160	36.22%	37.93%
12000 - 13999	121	37.34%	39.84%	12000 - 13999	101	40.61%	**38.03%**	12000 - 13999	139	31.55%	31.73%
14000 - 15999	142	37.76%	40.89%	14000 - 15999	114	34.98%	**34.26%**	14000 - 15999	104	27.57%	27.97%
16000 - 17999	116	34.94%	**32.75%**	16000 - 17999	104	22.60%	**22.24%**	16000 - 17999	121	28.31%	28.80%
18000 - 19999	129	28.96%	31.98%	18000 - 19999	116	27.52%	28.02%	18000 - 19999	94	25.79%	26.35%
20000 - 21999	116	18.96%	21.28%	20000 - 21999	73	26.18%	26.65%	20000 - 24999	90	25.85%	**25.83%**
22000 - 23999	90	23.37%	26.63%	22000 - 23999	68	29.19%	29.75%	25000 - 29999	34	18.68%	**18.65%**
24000 - 25999	88	18.45%	19.25%	24000 - 25999	53	23.20%	24.27%	30000 - 49999	28	12.66%	**11.62%**
26000 - 27999	46	23.72%	24.92%	26000 - 27999	36	29.94%	34.84%	Over 50000	42	8.56%	11.95%
28000 - 29999	52	26.29%	26.45%	28000 - 29999	34	21.19%	25.06%				
30000 - 34999	59	16.66%	16.70%	30000 - 34999	42	24.22%	26.23%				
35000 - 49999	32	14.62%	**13.20%**	35000 - 39999	31	33.29%	36.79%				
50000 - 89999	23	11.64%	11.84%	40000 - 49999	39	32.69%	**31.86%**				
Over 90000	32	12.82%	**10.35%**	50000 - 69999	42	25.30%	**25.04%**				
				70000 - 89999	28	21.96%	23.56%				
				Over 90000	38	16.58%	26.45%				


Broward County

Miami-Dade County

Palm Beach County

Figure 5.11. Comparison of Percent RMSE by Volume Group.

CHAPTER 6
SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Like other travel demand forecasting models, the Florida Standard Urban Transportation Model Structure (FSUTMS) is not immune to under- and over-assignments of trips to individual links. Although researchers recongnize that demand models are inherently susceptible to assignment errors, certain improvements have the potential to minimize these errors. This research attempts to minimize assignment errors in FSUTMS by better estimating the travel impedances used in its assignment procedure.

The standard highway assignment model in FSUTMS is based on the equilibrium trip assignment method. This method involves running several iterations of all-or-nothing capacity-restraint assignment with an adjustment of travel time to reflect delays encountered in the associated iteration. The first iteration loads trips to the network based on free-flow travel times. All subsequent iterations of assignment utilize the network travel times from the previous iteration as the travel times for the current iteration. The iterative link time adjustment process is accomplished through the widely used Bureau of Public Roads (BPR) volume-delay equation. While the output volumes from traditional forecasting models are usually given in daily traffic, the input capacities are typically given in hourly traffic. Therefore, it is important to convert the corresponding hourly volume output to its daily equivalent by dividing the hourly capacity by a factor called the peak-to-daily ratio. A peak-to-daily ratio is computed as the highest hourly volume of a day divided by the total daily volume. This ratio, referred to as CONFAC in FSUTMS, carries the same effect on the v/c ratio as a time-of-day factor, which converts a daily volume to its hourly volume equivalent.

Prior to FSUTMS version 5.3, a single CONFAC, typically 0.1, was used for all facility types. This was deemed insufficient to address the different peaking characteristics associated with different types of facilities, and is believed to be one of the culprits of assignment errors. Consequently, FSUTMS version 5.3 adopted a "multiple-CONFAC" structure that allows the use of different CONFACs for different types of facility. This method still assumes that all roadways belonging to a specific type have the same peaking characteristics. While several studies have indicated that the peak-to-daily ratio is a decreasing function of the level of congestion, a constant ratio is used in the current version of FSUTMS. This research attempts to improve the accuracy of trip assignments in FSUTMS by calibrating the relationships between CONFAC and a congestion measure for each facility type, and then applying these calibrated relationships in FSUTMS' trip assignment.

Traffic count data from across the state of Florida were used to calibrate the CONFAC functions. The functional relationships clearly demonstrate that the CONFAC is a function of congestion level and facility type. Overall, the functions used to calibrate the CONFAC models are logical and the differences in functional relationships for different facility under different congest levels suggest that the assignment results may be improved by making the CONFAC sensitive to these variables. A major effort was then carried out to implement these calibrated functions in FSUTMS.

Under an ideal, commercial-free environment, the research team would have ready access to the source code for FSUTMS' highway assignment module and be able to implement the calibrated functions in the module with ease. However, knowing the difficulty of gaining access to the source code and the potential complications associated with it, the research team decided to incorporate the functions in FSUTMS without involving any modification to any part of the FSUTMS source code. This was accomplished by taking advantage of the iterative process of the FSUTMS assignment procedure. Essentially, this procedure runs the assignment procedure for only one iteration, computes the CONFAC value corresponding to each assigned link volume, updates the link travel times, and performs a new iteration based on the new travel times until the procedure converges.

To be able to conclude that any difference between before and after the implementation of the CONFAC functions is attributed to use of variable CONFACs, the procedure developed must be shown to adequately replicate the results from the HASSIGN module. The basis for verifying the modified procedure is its ability to reproduce the assignment results from the original FSUTMS model for the exact same network and conditions. Based on the assignment results from the fictitious FTOWN network (under both uncongested and congested conditions) and the real-world Palm Beach network, it was concluded that the procedure adequately replicates the results from the HASSIGN module. The research then proceeded to use the procedure to compare assignment results before and after implementation of the variable CONFACs.

The assignments from each CONFAC option (constant versus variable) were compared against the ground counts for three selected networks for Miami-Dade, Broward, and Palm Beach counties based on the assigned VHT to count VHT ratios, assigned VMT to count VMT ratios, screenline volume-count ratio, and percent root mean square error. The accuracy between constant and variable CONFAC assignments was not significantly different, that desired improvement in assignment results of the variable CONFAC model is not empirically evident. This result was somewhat expected given the existence of a relatively flat CONFAC curves over a wide range of congestion levels in the calibrated models (see Figure 3.7). In other words, in most cases, there was not a significant difference between a constant CONFAC and a CONFAC generated from the calibrated CONFAC functions.

While the results from this research did not empirically support the desired outcome, the concept of applying variable CONFACs should not be discounted. It must be recognized that many other factors beyond the scope and control of this study could have contributed to the results of this study. This may include accuracy of the traffic counts for calibration, the accuracy of ground counts used in model evaluation, the appropriateness of the BPR equation and the associated parameters, etc. It is recommended that future research focus on the following areas:

- The accuracy of the variable CONFAC model might be improved by recalibrating the parameters used in the BPR function. The calibration is an involved task, which is beyond the scope of this research.
- The use of the developed methodology warrants the application of different volume-delay functions, which may improve the model accuracy. Additional research is needed to determine if the results are generally true for various transportation networks.

- Although the constant CONFAC model can reasonably replicate the FSUTMS assignment results, it is necessary to evaluate the direct integration of the variable CONFAC models into the FSUTMS source code. Accordingly, more definitive conclusions about the potential advantages of variable CONFAC models can then be drawn.
- The impact of using other assignment algorithms rather than the current equilibrium assignment also needs to be explored.

REFERENCES

Akçelik, R., “Travel Time Functions for Transport Planning Purpose: Davidson’s Function, Its Time-Dependent Form and an Alternative Travel Time Function,” *Australian Road Research*, Vol. 21, No. 3, 1991, pp. 49-59.

Allaire, S. A. and J. N. Ivan, “Factor Influencing Peak Spreading on Connecticut Freeways,” Presented at Transportation Research Board Annual Meeting, Washington, D.C., Paper 990201, January 1999.

Allen, W. G., “Analysis of Corridor Traffic Peaking,” *Transportation Research Record 1305*, Transportation Research Board, National Academic Council, Washington, D.C., 1991, pp. 50-60.

Allen, W. G. and G. W. Schultz, “Congestion-Based Peak Spreading Model,” *Transportation Research Record 1556*, Transportation Research Board, National Academic Council, Washington, D.C., 1996, pp. 8-15.

Barnes, J., *Peak Spreading Analysis Review of Relevant Issues and Synthesis of Current Practice, Phase I*, Research Report Prepared for Washington State Transportation Commission, Department of Transportation, July 1998.

Barton-Aschman Associates, Inc. and Cambridge Systematics, Inc., *Model Validation and Reasonableness Checking Manual*, Prepared for Travel Model Improvement Program, Federal Highway Administration, February 1997.

Bell *et al.*, *Implementation of an Equilibrium Capacity Restraint Model in the Texas Large Network Assignment Package*, Research Report Prepared for Texas Department of Transportation, November 1992.

Boyce *et al.*, “Effect on Equilibrium Trip Assignment of Different Link Congestion Functions,” *Transportation Research, Part A*, Vol. 15A, No. 3, 1981, pp. 223-232.

Branston, D., “Link Capacity Functions: A Review,” *Transportation Research*, Vol. 10, No. 4, 1976, pp. 223-236.

Cambridge Systematics, Inc., *Time-of-Day Modeling Procedures*, Final Report Prepared for The Travel Model Improvement Program (TMIP), Federal Highway Administration, http://www.bts.gov/tmip/papers/triptabl/Time-Day/toc.htm, February 1997.

Campbell, R. B., “Design Hour Volume Relationships,” Prepared for Ontario Planning Division, Department of Highway, July 1957.

Cottrell, W. D., “Estimating the Probability of Freeway Congestion Recurrence,” *Transportation Research Record 1634*, Transportation Research Board, National Academic Council, Washington, D.C., 1998, pp. 19-27.

DeCorla-Souza *et. al.*, “Simplied and Rational Approach to Address New Modeling Requirements for Conformity Analysis,” *Transportation Research Record 1472*, Transportation Research Board, National Academic Council, Washington, D.C., 1995, pp. 9-16.

Dowling *et al.*, “Accuracy and Performance of Improved Speed-Flow Curves,” *Road and Transport Research*, Vol. 7, No. 2, 1998, pp. 36-51.

Florida Department of Transportation, *Urban Transportation Planning Model Update-Phase II, Task C, Develop Standard Distribution and Assignment Models*, 1981.

Florida Department of Transportation, *Documentation and Procedural Updates to The Florida Standard Urban Transportation Model Structure*, *Final Technical Report No. 8*, *FSUTMS Highway Trip Assignment Model*, 1997.

Florida Department of Transportation, *Documentation and Procedural Updates to The Florida Standard Urban Transportation Model Structure*, *Final Technical Report No. 9*, *FSUTMS Highway Evaluation Model*, 1997.

Goodwin, R. P. and R. D. Coombe, “Dealing with the National Road Traffic forecasts in Urban Areas,” Planning and Transport Research and Computation (International) Co. Meeting, Vol. P347, 1991, pp. 241-250.

Hounsell, N. B., “Modelling of Traffic Growth in Congested Urban Networks,” Prepared for the Planning and Transport Research and Computation International Association, Vol. P347, September 1991, pp. 229-240.

Hounsell, N. B., “Peak Spreading and Congestion: Techniques for Distinguishing ‘Passive’ from ‘Active’ Responses by Road Users,” *Transportation Planning Systems*, Vol. 1, No. 3, 1991, pp. 39-46.

Ivan, J. N. and S. A. Allaire, *Estimating the Temporal Distribution of Traffic Within the Peak Period*, Final Report Prepared for the Joint Highway Research Advisory Council (JHRAC) of the University of Connecticut and the Connecticut Dept. of Transportation, January 2000.

Jordaan, P. W. and A. C. Van As, “Peaking Characteristics of Urban Traffic,” *The Civil Engineer in South Africa*, Vol. 33, No. 7, 1991, pp. 237-241.

Li, S. C. and A. Lapkowski, “Improving the Accuracy of Demand Forecast Models,” *Florida Transportation Modeling Newsletter*, December 1999, pp. 3-4.

Loudon *et al.*, “Predicting Peak-Spreading under Congested Conditions,” *Transportation Research Record 1203*, Transportation Research Board, National Academic Council, Washington, D.C., 1988, pp. 1-9.

Sheffi, Y., *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods,* Prentice-Hall, Inc., 1985.

Singh, R., “Improved Speed-Flow Relationships: Application to Transportation Planning Models,” Paper Presented at the 7th TRB Conference, Boston, Massachusetts, March 1999.

Skabardonis, A. and R. Dowling, “Improved Speed-Flow Relationships for Planning Application,” *Transportation Research Record 1572*, Transportation Research Board, National Academic Council, Washington, D.C., 1996, pp. 18-23.

Spiess, H., “Conical Volume-Delay Functions,” *Transportation Science*, Vol. 24, No. 2, 1990, pp. 153-158.

Transportation Research Board, *Highway Capacity Manual*, *Special Report 209*, 1998.

Urban Analysis Group, Inc., *URBAN/SYS Version 9.0 User Manuals for TRANPLAN and NIS*, *General Description*, 1998.

Urban Analysis Group, Inc., *URBAN/SYS Version 9.0 User Manuals for TRANPLAN and NIS*, *Loading*, 1998.

Urban Analysis Group, Inc., *URBAN/SYS Version 9.0 User Manuals for TRANPLAN and NIS*, *Miscellaneous Utilities*, 1998.

Walters, C. C. Poe, and R. Stokes, “Development of Planning Values for Urban Freeways in Large Texas cities,” *Transportation Research Record 1320*, Transportation Research Board, National Academic Council, Washington, D.C., 1989, pp. 32-39.

Appendix

Codes for Equilibrium Trip Assignment (HASSFB.exe)

```
' The program was developed to resemble the process solving the user-equilibrium program. All the required
' functions other than shortest routes computation and AON assignment were coded. The major functions include
' getting the assigned link volumes from output file (HRLDXY.ayy) of EQUILB.exe, performing line search to
' obtain optimal weight (LAMBDA), computing the weighted average link volume, updating the link travel
' impedance with volume-delay function, and feeding the updated link travel impedance back to HRLDXY.ayy to
' proceed the next Frank-Wolfe iteration until reaching the stopping criteria.

Option Explicit
Dim mFileSysObj As New FileSystemObject
Dim mFile1 As File, mFile2 As File, mFile3 As File, mFile4 As File, mFile5 As File, mFile6 As File
Dim mFile7 As File, mFile8 As File, mFile9 As File, mFile10 As File, mFile11 As File, mFile12 As File
Dim mFile13 As File, mFile14 As File, mFile15 As File, mFile16 As File, mFile17 As File, mFile18 As File
Dim mFile19 As File, mFile20 As File, mFile21 As File, mFile22 As File, mFile23 As File, mFile24 As File
Dim mFile25 As File, mFile26 As File, mFile27 As File, mFile28 As File, mFile29 As File
Dim mTxtStream1 As TextStream, mTxtStream2 As TextStream, mTxtStream3 As TextStream
Dim mTxtStream4 As TextStream, mTxtStream5 As TextStream, mTxtStream6 As TextStream
Dim mTxtStream7 As TextStream, mTxtStream8 As TextStream, mTxtStream9 As TextStream
Dim mTxtStream10 As TextStream, mTxtStream11 As TextStream, mTxtStream12 As TextStream
Dim mTxtStream13 As TextStream, mTxtStream14 As TextStream, mTxtStream15 As TextStream
Dim mTxtStream16 As TextStream, mTxtStream17 As TextStream, mTxtStream18 As TextStream
Dim mTxtStream19 As TextStream, mTxtStream20 As TextStream, mTxtStream21 As TextStream
Dim mTxtStream22 As TextStream, mTxtStream23 As TextStream, mTxtStream24 As TextStream
Dim mTxtStream25 As TextStream, mTxtStream26 As TextStream, mTxtStream27 As TextStream
Dim mTxtStream28 As TextStream, mTxtStream29 As TextStream

Dim aryUROADF(99) As Double, aryDFCONFAC(99) As Double
Dim aryBPRLOS(99) As Double, aryBPREXP(99) As Double
Dim aryFreeFlowImpd() As Double, aryPreImpedance() As Double, aryUpdatedImpedance() As Double
Dim aryDailyCapacity() As Double, aryHourlyCapacity() As Double, aryCONFAC() As Double
Dim aryCountedVolume() As Double, aryUpdatedDailyVolume() As Double
Dim aryTollLinkStr() As String, aryAdjLinkStr() As String
Dim aryTollClass() As Integer, blnTollLink() As Boolean
Dim aryANODE() As String, aryBNODE() As String
Dim aryAreaType() As Integer, aryFacilityType() As Integer
Dim aryNoLanes() As Double, aryDistance() As Double
Dim aryDailyVolume() As Double, aryPreUpdatedDailyVolume() As Double

Dim strNoIterations As String, intNoIterations As Integer
Dim dblInitialImpedance As Double, dblAssignedImpedance As Double, dblSumInitialImpedance As Double
Dim dblSumOfCount As Double, dblSubNoLinks As Double
Dim blnFTMatch As Boolean, blnVGMatch As Boolean
Dim dblSSE As Double, dblRMSE As Double, dblConvergenceCriteria As Double
Dim dblNoLinks As Double, dblTolerance As Double
Dim strTextLine As String, strTextLine1 As String, strTextLine2 As String, strTextLine3 As String
Dim CurIndex As Integer
Dim OldName, NewName
Dim dblLowerLimit As Double, dblUpperLimit As Double, dblLambda As Double
Dim dblLowerUpdatedDV As Double, dblUpperUpdatedDV As Double
Dim dblLowerObjFun As Double, dblUpperObjFun As Double, dblObjFun As Double
Dim strANode As String, strBNode As String, strSpeed As String, strTollLink As String
Dim strTollANODE As String, strTollBNODE As String
Dim dblCTOLL As Double, dblMaxLanes As Double, dblToll As Double
Dim dblServtMin As Double, dblServtSec As Double, dblServt As Double
Dim dblSpeed As Double, dblSpeed1 As Double, dblSpeed3 As Double
Dim dblSpdFactor As Double, dblRho As Double, dblP0 As Double
Dim dblT1 As Double, dblT2 As Double, dblT3 As Double
Dim dblD1 As Double, dblD3 As Double
Dim dblDecelTravelTime As Double, dblAccelTravelTime As Double
Dim dblDenominator As Double, dblQueueLength As Double
Dim intFT1 As Integer, intFT2 As Integer
Dim dblHourlyCapacity As Double, dblUpdatedDailyCapacity As Double
Dim dblDampingFactor As Double, dblVCMax As Double, dblVC As Double
Dim i As Long, j As Integer, h As Integer, k As Integer, m As Integer, n As Integer
Dim dblCM As Double, dblCONFAC As Double, dblAnalyzedDV As Double, dblUpdatedDV As Double
Dim strINS As String, strOUT As String, intINSNo As Integer, intOUTNo As Integer

Private Sub Form_Load()
  ' Change the working path to current working path.
  ChDir (App.Path)

  ' Read mean CONFACs from "VFACTORS.00A".
  Set mFile1 = mFileSysObj.GetFile(App.Path + "\VFACTORS.00A")
  Set mTxtStream1 = mFile1.OpenAsTextStream(ForReading)

  Do While Not mTxtStream1.AtEndOfStream
    strTextLine = mTxtStream1.ReadLine
    intFT2 = CInt(Mid$(strTextLine, 5, 3))
    aryUROADF(intFT2) = CDbl(Mid$(strTextLine, 18, 7))
    aryDFCONFAC(intFT2) = CDbl(Mid$(strTextLine, 35, 7))
    aryBPRLOS(intFT2) = CDbl(Mid$(strTextLine, 53, 7))
    aryBPREXP(intFT2) = CDbl(Mid$(strTextLine, 71, 7))
  Loop
  Call mTxtStream1.Close  ' Close the text stream.

  ' Call the sunroutine "Comparison" to check the stopping criteria of convergence.
  Call Comparison
End Sub

Private Sub Comparison()   ' Check the stopping criteria of convergence.
  ' Read the number of iteration from counter file HASSCT.con.
  Set mFile2 = mFileSysObj.GetFile(App.Path + "\HASSCT.con")
  Set mTxtStream2 = mFile2.OpenAsTextStream(ForReading)
  strNoIterations = mTxtStream2.ReadLine
  intNoIterations = CInt(Trim$(strNoIterations))
  Call mTxtStream2.Close

  ' Delete files HASSIT*.res storing assignment results of each iteration during the 1st equilibrium
  ' iteration.
  If intNoIterations = 0 Then
    If mFileSysObj.FileExists(App.Path + "\HASSIT0.RES") Then
      Call mFileSysObj.DeleteFile(App.Path + "\HASSIT*.RES")
    End If
  End If

  ' Read the number of links of the study urban area from HASSIGN.OUT.
  Set mFile3 = mFileSysObj.GetFile(App.Path + "\HASSIGN.OUT")
  Set mTxtStream3 = mFile3.OpenAsTextStream(ForReading)
  Do While Not mTxtStream3.AtEndOfStream
    strTextLine = mTxtStream3.ReadLine

    If Mid$(strTextLine, 30, 15) = "NUMBER OF LINKS" Then
      dblNoLinks = CDbl(LTrim$(Mid$(strTextLine, 49, 6)))
```

```
        Exit Do
    End If
Loop
Call mTxtStream3.Close

' Call the subroutine "DataProcess" to calculate the updated initial travel impednaces for the next
' equilibrium iteration.
Call DataProcess

' Compute the stopping criterion based on the percent RMSE of link travel impedances between two successive
' iterations.
dblSumInitialImpedance = 0
dblSSE = 0

For i = 0 To (CLng(dblNoLinks) - 1)
    If intNoIterations = 0 Then
        dblInitialImpedance = aryFreeFlowImpd(i) / 100
    Else
        dblInitialImpedance = aryPreImpedance(i) / 100
    End If

    dblAssignedImpedance = aryUpdatedImpedance(i) / 100

    dblSumInitialImpedance = dblSumInitialImpedance + dblInitialImpedance
    dblSSE = dblSSE + (dblAssignedImpedance - dblInitialImpedance) ^ 2
Next i

dblRMSE = Sqr(dblSSE / (dblNoLinks - 1)) / (dblSumInitialImpedance / dblNoLinks)

' Create a text file of HASSRMSE.TXT for writing the value of the percent RMSE for each iteration.
If intNoIterations = 0 Then
    If Not mFileSysObj.FileExists(App.Path + "\HASSRMSE.TXT") Then
        Call mFileSysObj.CreateTextFile(App.Path + "\HASSRMSE.TXT")
    End If
End If

Set mFile22 = mFileSysObj.GetFile(App.Path + "\HASSRMSE.TXT")
Set mTxtStream22 = mFile22.OpenAsTextStream(ForAppending)
Call mTxtStream22.WriteLine(CStr(dblRMSE))
Call mTxtStream22.Close

' Set up the user-specified value of convergence criterion.
dblConvergenceCriteria = 0.001

' Check the stopping criteria of convergence.
' 1. Assignment stops on the 1st iteration.
If dblRMSE < dblConvergenceCriteria And intNoIterations = 0 Then
    ' Call the subroutine "FinalRMSE" to calculate the percent RMSE between the predicted volumes and count
    ' volumes by category for assessing the performance of procedure integrated with variable CONFAC models.
    Call FinalRMSE

    ' Call the subroutine "FinalRecords" to update the final output of "HRLDXY.ayy" for use as input to the
    ' sequent highway evaluation (HEVAL) and plotting (HPLOT) modules.
    Call FinalRecords

    Call mFileSysObj.DeleteFile(App.Path + "\TollLink.TXT")
```

```
  End

' 2. Assignment reaches the convergence threshold with less than the maximum number of iterations specified
' by the user or exceeds the maximum number of iterations.
ElseIf (dblRMSE < dblConvergenceCriteria And intNoIterations < 9) Or intNoIterations >= 9 Then
  Call FinalRMSE

  Call FinalRecords

  Call mFileSysObj.DeleteFile(App.Path + "\HASSIT.TXT")
  Call mFileSysObj.DeleteFile(App.Path + "\TollLink.TXT")
  Call mFileSysObj.DeleteFile(App.Path + "\HNET.A00")

  ' Revert back to the original input file HNET.ayy of equilibrium assignment (HASSIGN) module.
  OldName = "HNET01.A00": NewName = "HNET.A00"   ' Define file names.
  Name OldName As NewName   ' Rename file.

  ' Set the value of the number of iterations with zero.
  Set mFile4 = mFileSysObj.GetFile(App.Path + "\HASSCT.con")
  Set mTxtStream4 = mFile4.OpenAsTextStream(ForWriting)
  Call mTxtStream4.WriteLine("0")
  Call mTxtStream4.Close

  End

' The assignment does not meet the stopping criteria.
Else
  ' Call the subroutine "Lodpak" to convert the updated input file HRLDXY01.txt to binary file HRLDXY.ayy.
  Call Lodpak

  Call mFileSysObj.DeleteFile(App.Path + "\HRLDXY.TXT")
  Call mFileSysObj.DeleteFile(App.Path + "\HRLDXY01.TXT")

  ' Reserve the original input file HNET.ayy of equilibrium assignment (HASSIGN) module on the 1st
  ' iteration.
  If intNoIterations = 0 Then
    OldName = "HNET.A00": NewName = "HNET01.A00"
    Name OldName As NewName
  Else
    Call mFileSysObj.DeleteFile(App.Path + "\HNET.A00")
  End If

  ' Rename the updated input binary file HRLDXY01.ayy to HNET.ayy for generate a new network skim on the
  ' next iteration.
  OldName = "HRLDXY01.A00": NewName = "HNET.A00"
  Name OldName As NewName

  If mFileSysObj.FileExists(App.Path + "\HRLDXY.A00") Then
    Call mFileSysObj.DeleteFile(App.Path + "\HRLDXY.A00")
  End If

  ' Call the subroutine "UpdateTRNPLNX" to decrease the index by 1 to retain the corresponding control file
  ' TRNPLXXX.ins to be invoked on the next iteration.
  Call UpdateTRNPLNX
```

```
    ' Call the subroutine "UpdateTRNPLXXX" to update the specification within TRNPLXXX.ins on the 1st
    ' iteration.
    If intNoIterations = 0 Then
      Call UpdateTRNPLXXX
    End If

    ' Update the value of the number of iterations.
    intNoIterations = intNoIterations + 1
    Set mFile5 = mFileSysObj.GetFile(App.Path + "\HASSCT.con")
    Set mTxtStream5 = mFile5.OpenAsTextStream(ForWriting)
    Call mTxtStream5.WriteLine(CStr(intNoIterations))
    Call mTxtStream5.Close

    ' An empty file HASSIT.txt is created to reflect that the convergence criterion or the maximum number of
    ' iterations is not reached.
    If Not mFileSysObj.FileExists(App.Path + "\HASSIT.TXT") Then
      Call mFileSysObj.CreateTextFile(App.Path + "\HASSIT.TXT")
    End If

    End
  End If
End Sub

' Calculate the updated initial travel impednaces for the next equilibrium iteration.
Private Sub DataProcess()
  ReDim aryANODE(CLng(dblNoLinks) - 1)
  ReDim aryBNODE(CLng(dblNoLinks) - 1)
  ReDim aryNoLanes(CLng(dblNoLinks) - 1)
  ReDim aryDistance(CLng(dblNoLinks) - 1)
  ReDim aryFreeFlowImpd(CLng(dblNoLinks) - 1)
  ReDim aryAreaType(CLng(dblNoLinks) - 1)
  ReDim aryFacilityType(CLng(dblNoLinks) - 1)
  ReDim aryDailyCapacity(CLng(dblNoLinks) - 1)
  ReDim aryCountedVolume(CLng(dblNoLinks) - 1)
  ReDim aryTollClass(CLng(dblNoLinks) - 1)
  ReDim aryDailyVolume(CLng(dblNoLinks) - 1)
  ReDim aryUpdatedDailyVolume(CLng(dblNoLinks) - 1)
  ReDim aryCONFAC(CLng(dblNoLinks) - 1)
  ReDim aryUpdatedImpedance(CLng(dblNoLinks) - 1)

  ' Extract the damping factor and the cap of v/c ratio from PROFILE.mas.
  Set mFile6 = mFileSysObj.GetFile(App.Path + "\PROFILE.MAS")
  Set mTxtStream6 = mFile6.OpenAsTextStream(ForReading)
  Do While Not mTxtStream6.AtEndOfStream
    strTextLine = mTxtStream6.ReadLine

    If Trim$(Mid$(strTextLine, 1, 8)) = "&DAMPING" Then
      dblDampingFactor = CDbl(Trim$(mTxtStream6.ReadLine))
    End If

    If Trim$(Mid$(strTextLine, 1, 6)) = "&VCMAX" Then
      strTextLine = mTxtStream6.ReadLine

      If Trim$(Mid$(strTextLine, 1, 1)) = "~" Then
        dblVCMax = 4
      Else
      dblVCMax = CDbl(Trim$(strTextLine))
    End If
    Exit Do
  End If
Loop
Call mTxtStream6.Close

' Call the subroutine "Lodunp" to convert the binary file HRLDXY.A00 to text file HRLDXY.TXT using
' LODUNP.exe.
Call Lodunp

' Call the subroutine "Lambda" to compute the iteration weights (LAMBDA).
Call Lambda

' Get the file HRLDXY.txt for reading.
Set mFile7 = mFileSysObj.GetFile(App.Path + "\HRLDXY.TXT")
Set mTxtStream7 = mFile7.OpenAsTextStream(ForReading)

' Create a text file of HRLDXY01.txt for writing the updated information.
Call mFileSysObj.CreateTextFile(App.Path + "\HRLDXY01.TXT")
Set mFile8 = mFileSysObj.GetFile(App.Path + "\HRLDXY01.TXT")
Set mTxtStream8 = mFile8.OpenAsTextStream(ForWriting)

' Compute the congested travel impedances in terms of the optimal LAMBDA.
' 1. Computation for the initial trip assignment.
If intNoIterations = 0 Then
  ' Create a text file HASSIT0.res for storeing the initial network characteristics.
  Call mFileSysObj.CreateTextFile(App.Path + "\HASSIT0.RES")
  Set mFile17 = mFileSysObj.GetFile(App.Path + "\HASSIT0.RES")
  Set mTxtStream17 = mFile17.OpenAsTextStream(ForWriting)

  ' Writing the heading for HASSIT0.res.
  strTextLine = "  ANODE  BNODE  NOLANES  LINKDIST  AREATYPE  FACTYPE" & _
  "     CONFAC0  IMPEDANCE0  SPEED0"
  Call mTxtStream17.WriteLine(strTextLine)

  ' Create a text file TollLink.txt for marking down the toll links in a sequence.
  Call mFileSysObj.CreateTextFile(App.Path + "\TollLink.TXT")
  Set mFile28 = mFileSysObj.GetFile(App.Path + "\TollLink.TXT")
  Set mTxtStream28 = mFile28.OpenAsTextStream(ForWriting)

  ' Extract the link-specific block information from HRLDXY.txt.
  Do While Not mTxtStream7.AtEndOfStream
    strTextLine = mTxtStream7.ReadLine

    If RTrim$(Mid$(strTextLine, 1)) = "E" Then
      For i = 0 To (CLng(dblNoLinks) - 1)
        strTextLine1 = mTxtStream7.ReadLine

        aryANODE(i) = Trim$(Mid$(strTextLine1, 1, 5))
        aryBNODE(i) = Trim$(Mid$(strTextLine1, 6, 5))
        aryNoLanes(i) = CDbl(Mid$(strTextLine1, 11, 1))
        aryDistance(i) = CDbl(Mid$(strTextLine1, 12, 4))
        aryFreeFlowImpd(i) = CDbl(Mid$(strTextLine1, 17, 4))
        aryAreaType(i) = CInt(Mid$(strTextLine1, 27, 2))
        aryFacilityType(i) = CInt(Mid$(strTextLine1, 29, 2))
      aryDailyCapacity(i) = CDbl(Trim$(Mid$(strTextLine1, 33, 6)))
      aryCountedVolume(i) = CDbl(Trim$(Mid$(strTextLine1, 39, 6)))
      aryTollClass(i) = CInt(Trim$(Mid$(strTextLine1, 48, 2)))

      mTxtStream7.SkipLine

      strTextLine3 = mTxtStream7.ReadLine
      aryDailyVolume(i) = CDbl(Trim$(Mid$(strTextLine3, 1, 8)))

      ' Write the from node and end node of toll links to TollLink.txt.
      If aryTollClass(i) <> 0 Then
        strTextLine = Space(7 - Len(aryANODE(i))) + aryANODE(i) + Space(7 - Len(aryBNODE(i))) + _
        aryBNODE(i)
        Call mTxtStream28.WriteLine(strTextLine)
      End If

      If aryFreeFlowImpd(i) <> 0 Then
        dblSpeed = aryDistance(i) / (aryFreeFlowImpd(i) / 60)
        strSpeed = CStr(Format$(dblSpeed, "###0.000"))
      Else
        strSpeed = ""
      End If

      ' Write the extracted initial network characteristics to HASSIT0.res.
      strTextLine = Space(7 - Len(aryANODE(i))) + aryANODE(i) + Space(7 - Len(aryBNODE(i))) + _
      aryBNODE(i) + Space(9 - Len(CStr(aryNoLanes(i)))) + CStr(aryNoLanes(i)) + _
      Space(10 - Len(CStr(aryDistance(i)))) + CStr(aryDistance(i)) + _
      Space(10 - Len(CStr(aryAreaType(i)))) + CStr(aryAreaType(i)) + _
      Space(9 - Len(CStr(aryFacilityType(i)))) + CStr(aryFacilityType(i)) _
      + Space(2) + CStr(Format$(aryDFCONFAC(aryFacilityType(i)), "0.00000000")) + _
      Space(12 - Len(CStr(Format$(aryFreeFlowImpd(i), "###0.0000")))) + _
      CStr(Format$(aryFreeFlowImpd(i), "###0.0000")) + Space(8 - Len(strSpeed)) + strSpeed
      Call mTxtStream17.WriteLine(strTextLine)
    Next i
  End If
Loop

Call mTxtStream17.Close
Call mTxtStream28.Close

' Read the strings from TOLLLINK.yya and HASSIT0.res to arrays for the computation of multiple-server
' queues.
Call TollFacility

' Get the file HRLDXY.txt for reading.
Set mFile10 = mFileSysObj.GetFile(App.Path + "\HRLDXY.TXT")
Set mTxtStream10 = mFile10.OpenAsTextStream(ForReading)

' Create a text file HASSIT1.res for storing the results of the 1st trip assignment.
Call mFileSysObj.CreateTextFile(App.Path + "\HASSIT1.RES")
Set mFile18 = mFileSysObj.GetFile(App.Path + "\HASSIT1.RES")
Set mTxtStream18 = mFile18.OpenAsTextStream(ForWriting)

' Writing the heading for HASSIT1.res.
strTextLine = "  ANODE  BNODE  NOLANES  LINKDIST  AREATYPE  FACTYPE      VOL1" & _
"    W_VOL1     CONFAC1     IMPEDANCE1  SPEED1"
```

```
Call mTxtStream18.WriteLine(strTextLine)

' Read the link-specific block information from HRLDXY.txt and update the link-specific block
' information based on the assigned volumes.
Do While Not mTxtStream10.AtEndOfStream
  strTextLine = mTxtStream10.ReadLine
  Call mTxtStream8.WriteLine(strTextLine)

  If RTrim$(Mid$(strTextLine, 1)) = "E" Then
    For i = 0 To (CLng(dblNoLinks) - 1)
      strTextLine1 = mTxtStream10.ReadLine
      mTxtStream10.SkipLine
      mTxtStream10.SkipLine

      ' The weighted volume is equal to the daily assigned volume on the 1st iteration.
      aryUpdatedDailyVolume(i) = aryDailyVolume(i) * dblLambda

      ' Variable CONFAC is computed based on the emperical models.
      dblCM = aryUpdatedDailyVolume(i) / aryNoLanes(i)
      dblAnalyzedDV = aryUpdatedDailyVolume(i)
      Call FunCONFAC
      aryCONFAC(i) = dblCONFAC

      ' Hourly capacity is computed based on the mean CONFAC extracted from VAFCTORS.yya.
      dblHourlyCapacity = aryDailyCapacity(i) * aryDFCONFAC(aryFacilityType(i))

      ' The v/c ratio is subject to an upper bound.  The default value for the maximum v/c ratio is 4.
      dblVC = (aryUpdatedDailyVolume(i) * aryCONFAC(i)) / (dblHourlyCapacity * _
      aryUROADF(aryFacilityType(i)))
      If dblVC > dblVCMax Then
        dblVC = dblVCMax
      End If

      ' The BPR function is used to calculate the updated initial travel impedance for non-toll
      ' facilities.
      If aryTollClass(i) = 0 Then
        aryUpdatedImpedance(i) = aryFreeFlowImpd(i) * (1 + aryBPRLOS(aryFacilityType(i)) * _
        dblDampingFactor * ((dblVC) ^ aryBPREXP(aryFacilityType(i))))

      ' The toll facilities model is used to calculate the updated initial travel impedance for toll
      ' facilities.
      Else
        Call MultipleServers
      End If

      If aryUpdatedImpedance(i) <> 0 Then
        strSpeed = CStr(CInt(aryDistance(i) / (aryUpdatedImpedance(i) / 60)))
      Else
        strSpeed = ""
      End If

      ' Write the updated assignment characteristics to HASSIT1.res.
      strTextLine = Space(7 - Len(aryANODE(i))) + aryANODE(i) + Space(7 - Len(aryBNODE(i))) + _
      aryBNODE(i) + Space(9 - Len(CStr(aryNoLanes(i)))) + CStr(aryNoLanes(i)) + _
      Space(10 - Len(CStr(aryDistance(i)))) + CStr(aryDistance(i)) + _
      Space(10 - Len(CStr(aryAreaType(i)))) + CStr(aryAreaType(i)) + _
```

```
        Space(9 - Len(CStr(aryFacilityType(i)))) + CStr(aryFacilityType(i)) _
        + Space(10 - Len(CStr(CLng(aryDailyVolume(i))))) + CStr(CLng(aryDailyVolume(i))) + _
        Space(10 - Len(CStr(CLng(aryUpdatedDailyVolume(i))))) + CStr(CLng(aryUpdatedDailyVolume(i))) + _
        Space(2) + CStr(Format$(aryCONFAC(i), "0.00000000")) + _
        Space(15 - Len(CStr(Format$(aryUpdatedImpedance(i), "###0.0000")))) + _
        CStr(Format$(aryUpdatedImpedance(i), "###0.0000")) + Space(8 - Len(strSpeed)) + strSpeed
        Call mTxtStream18.WriteLine(strTextLine)

        ' The updated travle impedance is subject to the limitation of four-digit location.
        If aryUpdatedImpedance(i) >= 9999 Then
          aryUpdatedImpedance(i) = 9999
        End If

        ' The updated link-specific block information is then writen to HRLDXY01.txt.
        strTextLine = Mid$(strTextLine1, 1, 16) + Space(4 - Len(CStr(CLng(aryUpdatedImpedance(i))))) _
        + CStr(CLng(aryUpdatedImpedance(i))) + Mid$(strTextLine1, 21, 12) _
        + Space(6 - Len(CStr(CLng(dblHourlyCapacity)))) + CStr(CLng(dblHourlyCapacity)) _
        + Mid$(strTextLine1, 39, 9) + " 0" + Mid$(strTextLine1, 50)
        Call mTxtStream8.WriteLine(strTextLine)
        Call mTxtStream8.WriteLine("")
        Call mTxtStream8.WriteLine("")
      Next i
    End If
  Loop

  Call mTxtStream10.Close
  Call mTxtStream18.Close

' 2. Computation for the subsequent trip assignment.
ElseIf intNoIterations >= 1 Then
  ' Create a text file HASSIT*.res for storing the results of the subsequent trip assignment.
  Call mFileSysObj.CreateTextFile(App.Path + "\HASSIT" & CStr(intNoIterations + 1) & ".RES")
  Set mFile18 = mFileSysObj.GetFile(App.Path + "\HASSIT" & CStr(intNoIterations + 1) & ".RES")
  Set mTxtStream18 = mFile18.OpenAsTextStream(ForWriting)

  ' Writing the heading for HASSIT*.res.
  If ((intNoIterations + 1) - 10) < 0 Then
    strTextLine = "  ANODE  BNODE  NOLANES  LINKDIST  AREATYPE  FACTYPE" & _
    "      VOL" & CStr(intNoIterations + 1) & "    W_VOL" & CStr(intNoIterations + 1) & "     CONFAC" & _
    CStr(intNoIterations + 1) & "     IMPEDANCE" & CStr(intNoIterations + 1) & "  SPEED" & _
    CStr(intNoIterations + 1)
  Else
    strTextLine = "  ANODE  BNODE  NOLANES  LINKDIST  AREATYPE  FACTYPE" & "     VOL" & _
    CStr(intNoIterations + 1) & "   W_VOL" & CStr(intNoIterations + 1) & "    CONFAC" & _
    CStr(intNoIterations + 1) & "    IMPEDANCE" & CStr(intNoIterations + 1) & " SPEED" & _
    CStr(intNoIterations + 1)
  End If
  Call mTxtStream18.WriteLine(strTextLine)

  ' Read the link-specific block information from HRLDXY.txt and update the link-specific block
  ' information based on the weighted volumes.
  Do While Not mTxtStream7.AtEndOfStream
    strTextLine = mTxtStream7.ReadLine
    Call mTxtStream8.WriteLine(strTextLine)

    If RTrim$(Mid$(strTextLine, 1)) = "E" Then
```

```
For i = 0 To (CLng(dblNoLinks) - 1)
  strTextLine1 = mTxtStream7.ReadLine

  mTxtStream7.SkipLine
  mTxtStream7.SkipLine

  ' The current weighted volume is obtained by linearly combining the assigned volume and the
  ' weighted volumes from the previous iteration using optimal LAMBDA.
  aryUpdatedDailyVolume(i) = aryDailyVolume(i) * dblLambda + aryPreUpdatedDailyVolume(i) * _
  (1 - dblLambda)

  ' Apply variable CONFAC models with the weighted volume.
  dblCM = aryUpdatedDailyVolume(i) / aryNoLanes(i)
  dblAnalyzedDV = aryUpdatedDailyVolume(i)
  Call FunCONFAC
  aryCONFAC(i) = dblCONFAC

  ' The v/c ratio is subject to an upper bound.
  dblVC = (aryUpdatedDailyVolume(i) * aryCONFAC(i)) / (aryHourlyCapacity(i) * _
  aryUROADF(aryFacilityType(i)))
  If dblVC > dblVCMax Then
    dblVC = dblVCMax
  End If

  ' The updated initial travel impedance for non-toll facilities is computed using the BPR
  ' function, while the one for toll facilities is computed using toll facilities models during
  ' the subroutine "LAMBDA".
  If Not blnTollLink(i) Then
    aryUpdatedImpedance(i) = aryFreeFlowImpd(i) * (1 + aryBPRLOS(aryFacilityType(i)) * _
    dblDampingFactor * ((dblVC) ^ aryBPREXP(aryFacilityType(i))))
  End If

  If aryUpdatedImpedance(i) <> 0 Then
    strSpeed = CStr(CInt(aryDistance(i) / (aryUpdatedImpedance(i) / 60)))
  Else
    strSpeed = ""
  End If

  ' Write the updated assignment characteristics to HASSIT*.res.
  strTextLine = Space(7 - Len(aryANODE(i))) + aryANODE(i) + Space(7 - Len(aryBNODE(i))) + _
  aryBNODE(i) + Space(9 - Len(CStr(aryNoLanes(i)))) + CStr(aryNoLanes(i)) + _
  Space(10 - Len(CStr(aryDistance(i)))) + CStr(aryDistance(i)) + _
  Space(10 - Len(CStr(aryAreaType(i)))) + CStr(aryAreaType(i)) + _
  Space(9 - Len(CStr(aryFacilityType(i)))) + CStr(aryFacilityType(i)) _
  + Space(10 - Len(CStr(CLng(aryDailyVolume(i))))) + CStr(CLng(aryDailyVolume(i))) + _
  Space(10 - Len(CStr(CLng(aryUpdatedDailyVolume(i))))) + CStr(CLng(aryUpdatedDailyVolume(i))) + _
  Space(2) + CStr(Format$(aryCONFAC(i), "0.00000000")) + _
  Space(15 - Len(CStr(Format$(aryUpdatedImpedance(i), "###0.0000")))) + _
  CStr(Format$(aryUpdatedImpedance(i), "###0.0000")) + Space(8 - Len(strSpeed)) + strSpeed
  Call mTxtStream18.WriteLine(strTextLine)

  ' The updated travle impedance is subject to the limitation of four-digit location.
  If aryUpdatedImpedance(i) >= 9999 Then
    aryUpdatedImpedance(i) = 9999
  End If
```

```
          ' The updated link-specific block information is then writen to HRLDXY01.txt.
          strTextLine = Mid$(strTextLine1, 1, 16) + Space(4 - Len(CStr(CLng(aryUpdatedImpedance(i))))) _
          + CStr(CLng(aryUpdatedImpedance(i))) + Mid$(strTextLine1, 21, 12) _
          + Space(6 - Len(CStr(CLng(aryHourlyCapacity(i))))) + CStr(CLng(aryHourlyCapacity(i))) _
          + Mid$(strTextLine1, 39)
          Call mTxtStream8.WriteLine(strTextLine)
          Call mTxtStream8.WriteLine("")
          Call mTxtStream8.WriteLine("")
        Next i
      End If
    Loop

    Call mTxtStream18.Close
  End If

  Call mTxtStream7.Close
  Call mTxtStream8.Close
End Sub

' Convert the binary file HRLDXY.ayy to text file HRLDXY.txt using LODUNP.exe.
Private Sub Lodunp()
  ' Create a redirecting file IPLODUNP.tmp for executing the LODUNP.exe
  Call mFileSysObj.CreateTextFile(App.Path + "\IPLODUNP.TMP")
  Set mFile9 = mFileSysObj.GetFile(App.Path + "\IPLODUNP.TMP")
  Set mTxtStream9 = mFile9.OpenAsTextStream(ForWriting)
  Call mTxtStream9.WriteLine("HRLDXY.A00")
  Call mTxtStream9.WriteLine("HRLDXY.TXT")
  Call mTxtStream9.WriteLine("Y")
  Call mTxtStream9.Close

  ' Create a file DELAY1.tmp for deferring the time to prevent a run-time error.
  Open App.Path & "\DELAY1.TMP" For Output As #1
  Close #1

  ' Delete the existing file HRLDXY.txt to avoid the fatal error.
  If mFileSysObj.FileExists(App.Path + "\HRLDXY.TXT") Then
    Call mFileSysObj.DeleteFile(App.Path + "\HRLDXY.TXT")
  End If

  Open App.Path & "\IPLODUNP.BAT" For Output As #1
  Print #1, "LODUNP < IPLODUNP.TMP"
  Print #1, "echo > DELAY1.TMP"
  Close #1

  Shell (App.Path + "\IPLODUNP.BAT")

  ' The function is to ensure the conversion with LODUNP.exe finishes before the following code is executed.
  Do While FileLen("DELAY1.TMP") = 0
    DoEvents
  Loop

  Call mFileSysObj.DeleteFile(App.Path + "\IPLODUNP.TMP")
  Call mFileSysObj.DeleteFile(App.Path + "\DELAY1.TMP")
  Call mFileSysObj.DeleteFile(App.Path + "\IPLODUNP.BAT")
End Sub
```

```
Private Sub TollFacility() ' Read strings from TOLLLINK.yya and HASSIT0.res to arrays.
  ReDim aryTollLinkStr(0)
  ReDim aryAdjLinkStr(0)

  'Read strings from TOLLLINK.yya
  Set mFile24 = mFileSysObj.GetFile(App.Path + "\TOLLLINK.00A")
  Set mTxtStream24 = mFile24.OpenAsTextStream(ForReading)

  Do While Not mTxtStream24.AtEndOfStream
    CurIndex = UBound(aryTollLinkStr)
    ReDim Preserve aryTollLinkStr(CurIndex + 1)
    aryTollLinkStr(CurIndex) = mTxtStream24.ReadLine
  Loop
  Call mTxtStream24.Close

  'Read strings from HASSIT0.res
  Set mFile19 = mFileSysObj.GetFile(App.Path + "\HASSIT0.RES")
  Set mTxtStream19 = mFile19.OpenAsTextStream(ForReading)

  Do While Not mTxtStream19.AtEndOfStream
    CurIndex = UBound(aryAdjLinkStr)
    ReDim Preserve aryAdjLinkStr(CurIndex + 1)
    aryAdjLinkStr(CurIndex) = mTxtStream19.ReadLine
  Loop
  Call mTxtStream19.Close
End Sub

' Calculate the travel impedance for toll links using toll facilities model.
Private Sub MultipleServers()
  ' Call the subroutine "TollLink" to retrieve toll model parameters from the string array.
  Call TollLink

  ' Extract the speed of upstream and downstream links for toll link from the string array aryAdjLinkStr() to
  ' compute the travel times corresponding to the uniform speed for deceleration ans accerleration.
  strBNode = aryANODE(i)
  strANode = aryBNODE(i)
  For k = LBound(aryAdjLinkStr) To UBound(aryAdjLinkStr)
    strTextLine = aryAdjLinkStr(k)

    If Trim$(Mid$(strTextLine, 8, 7)) = strBNode Then
      dblSpeed1 = CDbl(Trim$(Mid$(strTextLine, 77, 8)))

      ' Calculate the travel time and travel lenght for deceleration segment.
      dblT1 = dblSpeed1 / (2.35 + 0.055 * dblSpeed1) / 60
      dblD1 = 0.5 * dblSpeed1 * dblT1 / 60

      ' Calculate the travle time corresponding to the uniform speed for deceleration.
      dblDecelTravelTime = dblD1 / dblSpeed1 * 60
    ElseIf Trim$(Mid$(strTextLine, 1, 7)) = strANode Then
      dblSpeed3 = CDbl(Trim$(Mid$(strTextLine, 77, 8)))

      ' Calculate the travel time and travel lenght for acceleration.
      dblT3 = dblSpeed3 / 2.5 / 60
      dblD3 = 0.5 * dblSpeed3 * dblT3 / 60

      ' Calculate the travle time corresponding to the uniform speed for acceleration.
      dblAccelTravelTime = dblD3 / dblSpeed3 * 60
    End If
  Next k

  ' Calculate the queuing time using multiple-server queuing model based on the assigned volume for the
  ' queuing segment.
  dblCM = aryDailyVolume(i) / aryNoLanes(i)
  dblAnalyzedDV = aryDailyVolume(i)
  Call FunCONFAC

  ' Compute the ratio of arrival rate over service rate, which is subject to a maximum value 0.99.
  dblRho = aryDailyVolume(i) * dblCONFAC / ((3600 / dblServt) * dblMaxLanes)
  If dblRho >= 0.99 Then
    dblRho = 0.99
  End If

  dblDenominator = 1
  For h = 1 To Int(dblMaxLanes)
    If h = CInt(dblMaxLanes) Then
      dblDenominator = dblDenominator + ((dblRho * dblMaxLanes) ^ h) / (Factorial(h) * (1 - dblRho))
    Else
      dblDenominator = dblDenominator + ((dblRho * dblMaxLanes) ^ h) / Factorial(h)
    End If
  Next h

  ' Compute the percent of time that all toll plazas are idle.
  dblP0 = 1 / dblDenominator

  ' Compute the average queue length.
  dblQueueLength = dblP0 * ((dblRho * dblMaxLanes) ^ dblMaxLanes) * dblRho /  _
  (Factorial(CInt(dblMaxLanes)) * ((1 - dblRho) ^ 2))

  ' The waiting time of queuing segment includes service time in the queue.
  If aryDailyVolume(i) = 0 Then
    dblT2 = (1 / (3600 / dblServt)) * 60
  Else
    dblT2 = (dblQueueLength / (aryDailyVolume(i) * dblCONFAC) + 1 / (3600 / dblServt)) * 60
  End If

  ' The toll plaza delay is the total time from three segments minus the time spent traveling at free-flow
  ' speed on the segments of upstream and downstream of the toll link. The travel impedance on the toll link
  ' is the sum of delay at toll plaza and time penalty representing the imposition of the toll.
  aryUpdatedImpedance(i) = (dblCTOLL * dblToll * 60 + dblT1 + dblT2 + dblT3 - _
  (dblDecelTravelTime + dblAccelTravelTime)) * 100
End Sub

' Read toll model parameters from the string array aryTollLinkStr().
Private Sub TollLink()
  For k = LBound(aryTollLinkStr) To UBound(aryTollLinkStr)
    strTextLine = aryTollLinkStr(k)

    If Mid$(strTextLine, 1, 5) = "CTOLL" Then
      dblCTOLL = CDbl(Trim$(Mid$(strTextLine, 9)))
    ElseIf aryANODE(i) = Trim$(Mid$(strTextLine, 6, 5)) And _
    aryBNODE(i) = Trim$(Mid$(strTextLine, 12, 5)) Then
      aryTollClass(i) = CInt(Trim$(Mid$(strTextLine, 1, 2)))
      dblMaxLanes = CDbl(Trim$(Mid$(strTextLine, 47, 2)))
      dblToll = CDbl(Trim$(Mid$(strTextLine, 50, 5)))
      If Mid$(strTextLine, 56, 1) = " " Then
        dblServtMin = 0
      Else
        dblServtMin = CDbl(Mid$(strTextLine, 56, 1))
      End If
      dblServtSec = CDbl(Mid$(strTextLine, 58, 2))
      dblServt = dblServtMin * 60 + dblServtSec
      Exit For
    End If
  Next k
End Sub

' A factorial used in the toll facilities model.
Private Function Factorial(h) As Double
  Factorial = 1
  For j = 1 To h
    Factorial = Factorial * j
  Next j
End Function

Private Sub Lambda() ' Estimate the optimal LAMBDA.
  If intNoIterations = 0 Then
    ' Create a text file LAMBDA.txt for writing the value of lamda for each iteration.
    If Not mFileSysObj.FileExists(App.Path + "\LAMBDA.TXT") Then
      Call mFileSysObj.CreateTextFile(App.Path + "\LAMBDA.TXT")
    End If

    ' The LAMBDA is equal to 1 on the 1st iteraation.
    Set mFile21 = mFileSysObj.GetFile(App.Path + "\LAMBDA.TXT")
    Set mTxtStream21 = mFile21.OpenAsTextStream(ForWriting)
    dblLambda = 1
    Call mTxtStream21.WriteLine(CStr(dblLambda))
    Call mTxtStream21.Close

  ' Perform line search to obtain optimal LAMBDA when the number of iterations is more than one.
  ElseIf intNoIterations >= 1 Then
    ReDim blnTollLink(CLng(dblNoLinks) - 1)
    ReDim aryANODE(CLng(dblNoLinks) - 1)
    ReDim aryBNODE(CLng(dblNoLinks) - 1)
    ReDim aryAreaType(CLng(dblNoLinks) - 1)
    ReDim aryNoLanes(CLng(dblNoLinks) - 1)
    ReDim aryDistance(CLng(dblNoLinks) - 1)
    ReDim aryCountedVolume(CLng(dblNoLinks) - 1)
    ReDim aryDailyVolume(CLng(dblNoLinks) - 1)
    ReDim aryPreUpdatedDailyVolume(CLng(dblNoLinks) - 1)
    ReDim aryCONFAC(CLng(dblNoLinks) - 1)
    ReDim aryUpdatedImpedance(CLng(dblNoLinks) - 1)
    ReDim aryHourlyCapacity(CLng(dblNoLinks) - 1)
    ReDim aryFreeFlowImpd(CLng(dblNoLinks) - 1)
    ReDim aryFacilityType(CLng(dblNoLinks) - 1)
    ReDim aryDailyCapacity(CLng(dblNoLinks) - 1)
    ReDim aryPreImpedance(CLng(dblNoLinks) - 1)

    ' Read the strings from TOLLLINK.yya and HASSIT0.res to arrays for the computation of multiple-server
    ' queues.
    Call TollFacility

    ' Open LAMBDA.txt for appending the values of LAMBDA.
    Set mFile21 = mFileSysObj.GetFile(App.Path + "\LAMBDA.TXT")
    Set mTxtStream21 = mFile21.OpenAsTextStream(ForAppending)

    dblLowerLimit = 0
    dblUpperLimit = 1
    dblLowerObjFun = 0
    dblUpperObjFun = 0
    dblObjFun = 0
    n = 0

    ' The algorithm for estimating the optimal LAMBDA.
    Do
      ' Select LAMBDA to be the midpoint of lower and upper bounds of the line search interval.
      dblLambda = (dblLowerLimit + dblUpperLimit) / 2

      ' The 1st loop of the algorithm for estimating LAMBDA.
      If n = 0 Then
         ' Get the file HRLDXY.txt for reading.
         Set mFile10 = mFileSysObj.GetFile(App.Path + "\HRLDXY.TXT")
         Set mTxtStream10 = mFile10.OpenAsTextStream(ForReading)

         ' Get the file HASSIT0.res for reading.
         Set mFile19 = mFileSysObj.GetFile(App.Path + "\HASSIT0.RES")
         Set mTxtStream19 = mFile19.OpenAsTextStream(ForReading)

         ' Get the file of the last HASSIT*.res for reading.
         Set mFile20 = mFileSysObj.GetFile(App.Path + "\HASSIT" & CStr(intNoIterations) & ".RES")
         Set mTxtStream20 = mFile20.OpenAsTextStream(ForReading)

         ' Read out the first line of heading in HASSIT0.res and HASSIT*.res
         mTxtStream19.SkipLine
         mTxtStream20.SkipLine

         ' Get the file TollLink.txt to read the from node and end node of toll links.
         Set mFile29 = mFileSysObj.GetFile(App.Path + "\TollLink.TXT")
         Set mTxtStream29 = mFile29.OpenAsTextStream(ForReading)
         strTollLink = mTxtStream29.ReadLine
         strTollANODE = Trim$(Mid$(strTollLink, 1, 7))
         strTollBNODE = Trim$(Mid$(strTollLink, 8, 7))

         ' Read the link-specific block information from HRLDXY.txt to find a feasible solution to minimize
         ' the objective function of equilibrium assignment.
         Do While Not mTxtStream10.AtEndOfStream
            strTextLine = mTxtStream10.ReadLine

            If RTrim$(Mid$(strTextLine, 1)) = "E" Then
               For i = 0 To (CLng(dblNoLinks) - 1)
                  strTextLine1 = mTxtStream10.ReadLine

                  aryANODE(i) = Trim$(Mid$(strTextLine1, 1, 5))
                  aryBNODE(i) = Trim$(Mid$(strTextLine1, 6, 5))
                  aryNoLanes(i) = CDbl(Mid$(strTextLine1, 11, 1))
```

```
aryDistance(i) = CDbl(Mid$(strTextLine1, 12, 4))
aryAreaType(i) = CInt(Mid$(strTextLine1, 27, 2))
aryFacilityType(i) = CInt(Mid$(strTextLine1, 29, 2))
aryDailyCapacity(i) = CDbl(Trim$(Mid$(strTextLine1, 33, 6)))
aryCountedVolume(i) = CDbl(Trim$(Mid$(strTextLine1, 39, 6)))

mTxtStream10.SkipLine

strTextLine3 = mTxtStream10.ReadLine
aryDailyVolume(i) = CDbl(Trim$(Mid$(strTextLine3, 1, 8)))

' Extract the weighted volume and travle impedance from the HASSIT*.res for the previous
' iteration.
strTextLine = mTxtStream20.ReadLine
aryPreUpdatedDailyVolume(i) = CDbl(Trim$(Mid$(strTextLine, 63, 10)))
aryPreImpedance(i) = CDbl(Trim$(Mid$(strTextLine, 85, 15)))

' Calculate the weighted volume using the lower bound of interval.
dblLowerUpdatedDV = aryDailyVolume(i) * dblLowerLimit + _
aryPreUpdatedDailyVolume(i) * (1 - dblLowerLimit)

' Apply the variable CONFAC models.
dblCM = dblLowerUpdatedDV / aryNoLanes(i)
dblAnalyzedDV = dblLowerUpdatedDV
Call FunCONFAC
aryCONFAC(i) = dblCONFAC

' Hourly capacity is computed based on the mean CONFAC extracted from VAFCTORS.yya.
aryHourlyCapacity(i) = aryDailyCapacity(i) * aryDFCONFAC(aryFacilityType(i))

dblVC = (dblLowerUpdatedDV * aryCONFAC(i)) / _
(aryHourlyCapacity(i) * aryUROADF(aryFacilityType(i)))

' Extract the free-flow travel time from the HASSIT0.res.
strTextLine = mTxtStream19.ReadLine
aryFreeFlowImpd(i) = CDbl(Trim$(Mid$(strTextLine, 65, 12)))

' Compute the tentative travel impedance for toll and non-toll links using the toll
' facilities model and the BPR function respectively.
If aryANODE(i) = strTollANODE And aryBNODE(i) = strTollBNODE Then
  Call MultipleServers

  blnTollLink(i) = True

  If Not mTxtStream29.AtEndOfStream Then
    strTollLink = mTxtStream29.ReadLine
    strTollANODE = Trim$(Mid$(strTollLink, 1, 7))
    strTollBNODE = Trim$(Mid$(strTollLink, 8, 7))
  End If
Else
  aryUpdatedImpedance(i) = aryFreeFlowImpd(i) * _
  (1 + aryBPRLOS(aryFacilityType(i)) * dblDampingFactor * _
  (1 / (aryBPREXP(aryFacilityType(i)) + 1)) * ((dblVC) ^ aryBPREXP(aryFacilityType(i))))

  blnTollLink(i) = False
End If
```

```
      ' Calculate the objective function using the lower bound of interval.
      dblLowerObjFun = dblLowerObjFun + dblLowerUpdatedDV * aryUpdatedImpedance(i) / 100

      ' Calculate the weighted volume using the upper bound of interval.
      dblUpperUpdatedDV = aryDailyVolume(i) * dblUpperLimit + _
      aryPreUpdatedDailyVolume(i) * (1 - dblUpperLimit)

      dblCM = dblUpperUpdatedDV / aryNoLanes(i)
      dblAnalyzedDV = dblUpperUpdatedDV
      Call FunCONFAC
      aryCONFAC(i) = dblCONFAC

      dblVC = (dblUpperUpdatedDV * aryCONFAC(i)) / _
      (aryHourlyCapacity(i) * aryUROADF(aryFacilityType(i)))

      ' The travel impedance of toll links is computed on the basis of assigned volume.  Only
      ' the travel impedance of non-toll links is subject to the change of weighted volume.
      If Not blnTollLink(i) Then
        aryUpdatedImpedance(i) = aryFreeFlowImpd(i) * (1 + aryBPRLOS(aryFacilityType(i)) _
        * dblDampingFactor * (1 / (aryBPREXP(aryFacilityType(i)) + 1)) * _
        ((dblVC) ^ aryBPREXP(aryFacilityType(i))))
      End If

      ' Calculate the objective function using the upper bound of interval.
      dblUpperObjFun = dblUpperObjFun + dblUpperUpdatedDV * aryUpdatedImpedance(i) / 100

      ' Calculate the weighted volume using the average LAMBDA.
      dblUpdatedDV = aryDailyVolume(i) * dblLambda + _
      aryPreUpdatedDailyVolume(i) * (1 - dblLambda)

      dblCM = dblUpdatedDV / aryNoLanes(i)
      dblAnalyzedDV = dblUpdatedDV
      Call FunCONFAC
      aryCONFAC(i) = dblCONFAC

      dblVC = (dblUpdatedDV * aryCONFAC(i)) / _
      (aryHourlyCapacity(i) * aryUROADF(aryFacilityType(i)))

      If Not blnTollLink(i) Then
        aryUpdatedImpedance(i) = aryFreeFlowImpd(i) * (1 + aryBPRLOS(aryFacilityType(i)) * _
        dblDampingFactor * (1 / (aryBPREXP(aryFacilityType(i)) + 1)) * _
        ((dblVC) ^ aryBPREXP(aryFacilityType(i))))
      End If

      ' Calculate the objective function using the average LAMBDA.
      dblObjFun = dblObjFun + dblUpdatedDV * aryUpdatedImpedance(i) / 100
    Next i
  End If
Loop

Call mTxtStream10.Close
Call mTxtStream19.Close
Call mTxtStream20.Close
Call mTxtStream29.Close
```

```
  ' The subsequent loop of the algorithm for estimating LAMBDA.
  ElseIf n >= 1 Then
    dblObjFun = 0

    For i = 0 To (CLng(dblNoLinks) - 1)
      ' Calculate the weighted volume using the average LAMBDA.
      dblUpdatedDV = aryDailyVolume(i) * dblLambda + _
      aryPreUpdatedDailyVolume(i) * (1 - dblLambda)

      dblCM = dblUpdatedDV / aryNoLanes(i)
      dblAnalyzedDV = dblUpdatedDV
      Call FunCONFAC
      aryCONFAC(i) = dblCONFAC

      dblVC = (dblUpdatedDV * aryCONFAC(i)) / _
      (aryHourlyCapacity(i) * aryUROADF(aryFacilityType(i)))

      If Not blnTollLink(i) Then
        aryUpdatedImpedance(i) = aryFreeFlowImpd(i) * (1 + aryBPRLOS(aryFacilityType(i)) * _
        dblDampingFactor * (1 / (aryBPREXP(aryFacilityType(i)) + 1)) * _
        ((dblVC) ^ aryBPREXP(aryFacilityType(i))))
      End If


      ' Calculate the objective function using the average LAMBDA.
      dblObjFun = dblObjFun + dblUpdatedDV * aryUpdatedImpedance(i) / 100
    Next i
  End If

  ' If the objective function for lower bound is greater than or equal to the objective function for
  ' upper bound, the lower bound and its respective objective function are replaced with LAMBDA and
  ' its objective function.
  If dblLowerObjFun >= dblUpperObjFun Then
    dblLowerLimit = dblLambda
    dblLowerObjFun = dblObjFun
  Else
    dblUpperLimit = dblLambda
    dblUpperObjFun = dblObjFun
  End If

  n = n + 1

  ' The stopping criteria for the algorithm of estimating LAMBDA is 1/8 for the second equilibrium
  ' iteration, and (1/8)^2 for all subsequent iterations. These are corresponding to 4 and 7 loops in
  ' the procedure.
  If intNoIterations = 1 Then
    dblTolerance = 3
  Else
    dblTolerance = 6
  End If
Loop Until n = dblTolerance

' The optimal LAMBDA is the average of final lower bound and upper bound.
dblLambda = (dblLowerLimit + dblUpperLimit) / 2

Call mTxtStream21.WriteLine(CStr(dblLambda))
```

```
        Call mTxtStream21.Close
    End If

End Sub

Private Sub FunCONFAC() ' Variable CONFAC models.
    intFT1 = CInt(Mid$(CStr(aryFacilityType(i)), 1, 1))

    If dblAnalyzedDV = 0 Then
        dblCONFAC = aryDFCONFAC(aryFacilityType(i))
    Else
        ' Facility type 1: freeways and expressways.
        If intFT1 = 1 Then
            dblCONFAC = (402.787 + 0.05867 * dblCM) * aryNoLanes(i) / dblAnalyzedDV

            If dblCONFAC > 0.2006 Then
                dblCONFAC = 0.2006
            ElseIf dblCONFAC < 0.0693 Then
                dblCONFAC = 0.0693
            End If

        ' Facility type 2: divided arterials.
        ElseIf intFT1 = 2 Then
            dblCONFAC = (60.714 + 0.08014 * dblCM) * aryNoLanes(i) / dblAnalyzedDV

            If dblCONFAC > 0.2048 Then
                dblCONFAC = 0.2048
            ElseIf dblCONFAC < 0.0829 Then
                dblCONFAC = 0.0829
            End If

        ' Facility type 3: undivided arterials.
        ElseIf intFT1 = 3 Then
            dblCONFAC = (36.433 + 0.08456 * dblCM) * aryNoLanes(i) / dblAnalyzedDV

            If dblCONFAC > 0.1261 Then
                dblCONFAC = 0.1261
            ElseIf dblCONFAC < 0.087 Then
                dblCONFAC = 0.087
            End If

        ' Facility type 4: collectors.
        ElseIf intFT1 = 4 Then
            dblCONFAC = (33.731 + 0.08823 * dblCM) * aryNoLanes(i) / dblAnalyzedDV

            If dblCONFAC > 0.124 Then
                dblCONFAC = 0.124
            ElseIf dblCONFAC < 0.0914 Then
                dblCONFAC = 0.0914
            End If

        ' Facility type 6: one-way facilities.
        ElseIf intFT1 = 6 Then
            dblCONFAC = (40.85 + 0.08655 * dblCM) * aryNoLanes(i) / dblAnalyzedDV

            If dblCONFAC > 0.153 Then
        dblCONFAC = 0.153
      ElseIf dblCONFAC < 0.0897 Then
        dblCONFAC = 0.0897
      End If

    ' Other facility types.
    Else
      dblCONFAC = aryDFCONFAC(aryFacilityType(i))
    End If
  End If
End Sub

' Calculate the percent RMSE between the predicted volumes and count volumes in terms of area-wide, facility
' type, and volume group for assessing the performance of procedure integrated with variable CONFAC models.
Private Sub FinalRMSE()
  ' Create or get the file EVALRMSE.txt to write the results of the percent RMSE.
  If Not mFileSysObj.FileExists(App.Path + "\EVALRMSE.TXT") Then
    Call mFileSysObj.CreateTextFile(App.Path + "\EVALRMSE.TXT")
  End If

  Set mFile23 = mFileSysObj.GetFile(App.Path + "\EVALRMSE.TXT")
  Set mTxtStream23 = mFile23.OpenAsTextStream(ForWriting)

  Call mTxtStream23.WriteLine("RMSE -- FTOWN")
  Call mTxtStream23.WriteBlankLines(1)

  ' Calculate the percent RMSE area-wide.
  dblSumOfCount = 0
  dblSSE = 0
  dblSubNoLinks = 0

  ' Only the link with count volume is included.
  For i = 0 To (CLng(dblNoLinks) - 1)
    If aryCountedVolume(i) <> 0 Then
      dblSumOfCount = dblSumOfCount + aryCountedVolume(i)
      dblSSE = dblSSE + (aryUpdatedDailyVolume(i) - aryCountedVolume(i)) ^ 2
      dblSubNoLinks = dblSubNoLinks + 1
    End If
  Next i

  dblRMSE = Sqr(dblSSE / (dblSubNoLinks - 1)) / (dblSumOfCount / dblSubNoLinks) * 100

  strTextLine = " Area     " & CStr(Format$(dblRMSE, "###0.00")) & "%"
  Call mTxtStream23.WriteLine(strTextLine)
  Call mTxtStream23.WriteBlankLines(1)

  ' Calculate the percent RMSE classified by facility type.
  For intFT1 = 1 To 9
    dblSumOfCount = 0
    dblSSE = 0
    dblSubNoLinks = 0
    blnFTMatch = False

    For i = 0 To CLng(dblNoLinks - 1)
      If CInt(Mid$(CStr(aryFacilityType(i)), 1, 1)) = intFT1 Then
        If aryCountedVolume(i) <> 0 Then
          blnFTMatch = True
          dblSumOfCount = dblSumOfCount + aryCountedVolume(i)
          dblSSE = dblSSE + (aryUpdatedDailyVolume(i) - aryCountedVolume(i)) ^ 2
          dblSubNoLinks = dblSubNoLinks + 1
        End If
      End If
    Next i

    If blnFTMatch Then
      dblRMSE = Sqr(dblSSE / (dblSubNoLinks - 1)) / (dblSumOfCount / dblSubNoLinks) * 100
      strTextLine = " F" & CStr(intFT1) & "       " & CStr(Format$(dblRMSE, "###0.00")) & "%"
      Call mTxtStream23.WriteLine(strTextLine)
    End If
  Next intFT1

  Call mTxtStream23.WriteBlankLines(1)

  ' Calculate the percent RMSE for count volumes greater than 50,000 vpd.
  dblSumOfCount = 0
  dblSSE = 0
  dblSubNoLinks = 0
  blnVGMatch = False

  For i = 0 To CLng(dblNoLinks - 1)
    If aryCountedVolume(i) > 50000 Then
      blnVGMatch = True
      dblSumOfCount = dblSumOfCount + aryCountedVolume(i)
      dblSSE = dblSSE + (aryUpdatedDailyVolume(i) - aryCountedVolume(i)) ^ 2
      dblSubNoLinks = dblSubNoLinks + 1
    End If
  Next i

  If blnVGMatch Then
    dblRMSE = Sqr(dblSSE / (dblSubNoLinks - 1)) / (dblSumOfCount / dblSubNoLinks) * 100
    strTextLine = " >50,000vpd  " & CStr(Format$(dblRMSE, "###0.00")) & "%"
    Call mTxtStream23.WriteLine(strTextLine)
  End If

  ' Calculate the percent RMSE for count volumes less than 50,000 vpd
  dblSumOfCount = 0
  dblSSE = 0
  dblSubNoLinks = 0
  blnVGMatch = False

  For i = 0 To CLng(dblNoLinks - 1)
    If aryCountedVolume(i) < 50000 And aryCountedVolume(i) <> 0 Then
      blnVGMatch = True
      dblSumOfCount = dblSumOfCount + aryCountedVolume(i)
      dblSSE = dblSSE + (aryUpdatedDailyVolume(i) - aryCountedVolume(i)) ^ 2
      dblSubNoLinks = dblSubNoLinks + 1
    End If
  Next i

  If blnVGMatch Then
    dblRMSE = Sqr(dblSSE / (dblSubNoLinks - 1)) / (dblSumOfCount / dblSubNoLinks) * 100
    strTextLine = " <50,000vpd  " & CStr(Format$(dblRMSE, "###0.00")) & "%"
```

```
        Call mTxtStream23.WriteLine(strTextLine)
    End If

    Call mTxtStream23.Close
End Sub

' Convert the updated input file HRLDXY01.txt to binary file HRLDXY01.ayy using LODPAK.exe.
Private Sub Lodpak()
    ' Create a redirecting file IPLODPAK.tmp for executing the LODPAK.exe
    Call mFileSysObj.CreateTextFile(App.Path + "\IPLODPAK.TMP")
    Set mFile11 = mFileSysObj.GetFile(App.Path + "\IPLODPAK.TMP")
    Set mTxtStream11 = mFile11.OpenAsTextStream(ForWriting)
    Call mTxtStream11.WriteLine("HRLDXY01.TXT")
    Call mTxtStream11.WriteLine("HRLDXY01.A00")
    Call mTxtStream11.Close

    ' Create a file DELAY2.tmp for deferring the time to prevent a run-time error.
    Open App.Path & "\DELAY2.TMP" For Output As #1
    Close #1

    If mFileSysObj.FileExists(App.Path + "\HRLDXY01.A00") Then
        Call mFileSysObj.DeleteFile(App.Path + "\HRLDXY01.A00")
    End If

    Open App.Path & "\IPLODPAK.bat" For Output As #1
    Print #1, "LODPAK < IPLODPAK.TMP"
    Print #1, "echo > DELAY2.TMP"
    Close #1

    Shell (App.Path + "\IPLODPAK.bat")

    ' The function is to ensure the conversion with LODPAK.exe finishes before the following code is executed.
    Do While FileLen("DELAY2.TMP") = 0
        DoEvents
    Loop

    Call mFileSysObj.DeleteFile(App.Path + "\IPLODPAK.TMP")
    Call mFileSysObj.DeleteFile(App.Path + "\DELAY2.TMP")
    Call mFileSysObj.DeleteFile(App.Path + "\IPLODPAK.bat")
End Sub

' Reduce the index by 1 to retain the corresponding control file TRNPLXXX.ins to be invoked on the next
' iteration.
Private Sub UpdateTRNPLNX()
    ' Get the counter file TRNPLNX.con to read the index.
    Set mFile12 = mFileSysObj.GetFile(App.Path + "\TRNPLNX.CON")
    Set mTxtStream12 = mFile12.OpenAsTextStream(ForReading)

    strTextLine = mTxtStream12.ReadLine
    strINS = Trim$(strTextLine)
    intINSNo = CInt(Mid$(strINS, 6, 3))
    intINSNo = intINSNo - 1

    strTextLine = mTxtStream12.ReadLine
    strOUT = Trim$(strTextLine)
    intOUTNo = CInt(Mid$(strOUT, 6, 3))
```

```
  intOUTNo = intOUTNo - 1

  Call mTxtStream12.Close

  ' The updated index is written back to TRNPLNX.con.
  Set mFile13 = mFileSysObj.GetFile(App.Path + "\TRNPLNX.CON")
  Set mTxtStream13 = mFile13.OpenAsTextStream(ForWriting)

  strTextLine = Mid$(strINS, 1, 5) + CStr(Format(intINSNo, "000")) + Mid$(strINS, 9, 4)
  Call mTxtStream13.WriteLine(strTextLine)

  strTextLine = Mid$(strOUT, 1, 5) + CStr(Format(intOUTNo, "000")) + Mid$(strOUT, 9, 4)
  Call mTxtStream13.WriteLine(strTextLine)

  Call mTxtStream13.Close
End Sub

' Adjust the required option/parameter reference in control file TRNPLXXX.ins for the equilibrium highway load.
Private Sub UpdateTRNPLXXX()
  ' Get the file TRNPLXXX.ins for reading.
  Set mFile26 = mFileSysObj.GetFile(App.Path + "\TRNPL" & CStr(Format(intINSNo, "000")) & ".INS")
  Set mTxtStream26 = mFile26.OpenAsTextStream(ForReading)

  ' Create a text file TRNPLXXX.txt to write the updated option/parameter reference.
  Call mFileSysObj.CreateTextFile(App.Path + "\TRNPL" & CStr(Format(intINSNo, "000")) & ".TXT")
  Set mFile27 = mFileSysObj.GetFile(App.Path + "\TRNPL" & CStr(Format(intINSNo, "000")) & ".TXT")
  Set mTxtStream27 = mFile27.OpenAsTextStream(ForWriting)

  ' Modify the entries to inactive the execution of the toll facilities model and disable the default UROAD
  ' approach.
  Do While Not mTxtStream26.AtEndOfStream
    strTextLine = mTxtStream26.ReadLine

    If Right$(Trim$(strTextLine), 21) = "TOLL FACILITIES MODEL" Then
      strTextLine = "~  TOLL FACILITIES MODEL"
    ElseIf Right$(Trim$(strTextLine), 22) = "MULTIPLE SERVER QUEUES" Then
      strTextLine = "~  MULTIPLE SERVER QUEUES"
    ElseIf Trim$(strTextLine) = "$PARAMETERS" Then
      Call mTxtStream27.WriteLine(strTextLine)
      strTextLine = "   IMPEDANCE = TIME1"
    ElseIf Left$(Trim$(strTextLine), 5) = "CTOLL" Then
      strTextLine = "   CTOLL = 0.00"
    ElseIf Left$(Trim$(strTextLine), 5) = "TOLLS" Then
      strTextLine = "   TOLLS = 0.00"
      Call mTxtStream27.WriteLine(strTextLine)

      For m = 1 To 18
        mTxtStream26.SkipLine
        strTextLine = "           0.00"
        Call mTxtStream27.WriteLine(strTextLine)
      Next m

      mTxtStream26.SkipLine
      strTextLine = "           0.00"
    ElseIf Left$(Trim$(strTextLine), 5) = "SERVT" Then
      strTextLine = "   SERVT = 0.00"
      Call mTxtStream27.WriteLine(strTextLine)

      For m = 1 To 18
        mTxtStream26.SkipLine
        strTextLine = "          0.00"
        Call mTxtStream27.WriteLine(strTextLine)
      Next m

      mTxtStream26.SkipLine
      strTextLine = "          0.00"
    End If

    Call mTxtStream27.WriteLine(strTextLine)
  Loop

  Call mTxtStream26.Close
  Call mTxtStream27.Close

  ' Delete the original TRNPLXXX.ins file.
  Call mFileSysObj.DeleteFile(App.Path + "\TRNPL" & CStr(Format(intINSNo, "000")) & ".INS")

  ' Define file names.
  OldName = "TRNPL" & CStr(Format(intINSNo, "000")) & ".TXT": NewName = "TRNPL" &
CStr(Format(intINSNo, "000")) & ".INS"
  Name OldName As NewName
End Sub

' Update the final output of "HRLDXY.ayy" for use as input to the sequent highway evaluation (HEVAL) and
' plotting (HPLOT) modules.
Private Sub FinalRecords()
  ' Get the file HRLDXY.txt for reading.
  Set mFile14 = mFileSysObj.GetFile(App.Path + "\HRLDXY.TXT")
  Set mTxtStream14 = mFile14.OpenAsTextStream(ForReading)

  ' Get the file HRLDXY01.txt for writing the original free-flow impedance and the updated daily capacity in
  ' terms of final weighted volume.
  Set mFile15 = mFileSysObj.GetFile(App.Path + "\HRLDXY01.TXT")
  Set mTxtStream15 = mFile15.OpenAsTextStream(ForWriting)

  ' Create the file HRLDXY02.txt for writing the original free-flow impedance and original daily capacity.
  Call mFileSysObj.CreateTextFile(App.Path + "\HRLDXY02.TXT")
  Set mFile16 = mFileSysObj.GetFile(App.Path + "\HRLDXY02.TXT")
  Set mTxtStream16 = mFile16.OpenAsTextStream(ForWriting)

  ' Read the link-specific block information from HRLDXY.txt and update the link-specific block information
  ' based on the weighted volumes.
  Do While Not mTxtStream14.AtEndOfStream
    strTextLine = mTxtStream14.ReadLine
    Call mTxtStream15.WriteLine(strTextLine)
    Call mTxtStream16.WriteLine(strTextLine)

    If RTrim$(Mid$(strTextLine, 1)) = "E" Then
      For i = 0 To (CLng(dblNoLinks) - 1)
        strTextLine = mTxtStream14.ReadLine
        mTxtStream14.SkipLine
        mTxtStream14.SkipLine
        dblUpdatedDailyCapacity = aryDailyCapacity(i) * aryDFCONFAC(aryFacilityType(i)) / aryCONFAC(i)

        ' The TIME1 field in the 1st line of link-specific block is replaced with the original free-flow
        ' impedance in the HRLDXY02.TXT.
        If aryTollClass(i) = 0 Then
          strTextLine1 = Mid$(strTextLine, 1, 16) + Space(4 - Len(CStr(CLng(aryFreeFlowImpd(i))))) _
          + CStr(CLng(aryFreeFlowImpd(i))) + Mid$(strTextLine, 21)
        Else
          strTextLine1 = Mid$(strTextLine, 1, 16) + Space(4 - Len(CStr(CLng(aryFreeFlowImpd(i))))) _
          + CStr(CLng(aryFreeFlowImpd(i))) + Mid$(strTextLine, 21, 27) + _
          Space(2 - Len(CStr(aryTollClass(i)))) + CStr(aryTollClass(i)) + Mid$(strTextLine, 50)
        End If
        Call mTxtStream16.WriteLine(strTextLine1)

        ' The 2nd line of link-specific block is replaced with the final congested impedance.
        strTextLine2 = Space(8 - Len(CStr(CLng(aryUpdatedImpedance(i))))) _
        + CStr(CLng(aryUpdatedImpedance(i)))
        Call mTxtStream16.WriteLine(strTextLine2)

        ' The 3rd line of link-specific block is replaced with the final weighted volume.
        strTextLine3 = Space(8 - Len(CStr(CLng(aryUpdatedDailyVolume(i))))) _
        + CStr(CLng(aryUpdatedDailyVolume(i)))
        Call mTxtStream16.WriteLine(strTextLine3)

        ' The TIME1 and daily capacity fields in the 1st line of link-specific block are replaced with the
        ' original free-flow impedance and updated daily capacity in the HRLDXY01.TXT.
        If aryTollClass(i) = 0 Then
          strTextLine1 = Mid$(strTextLine, 1, 16) + Space(4 - Len(CStr(CLng(aryFreeFlowImpd(i))))) _
          + CStr(CLng(aryFreeFlowImpd(i))) + Mid$(strTextLine, 21, 12) + _
          Space(6 - Len(CStr(CLng(dblUpdatedDailyCapacity)))) + _
          CStr(CLng(dblUpdatedDailyCapacity)) + Mid$(strTextLine, 39)
        Else
          strTextLine1 = Mid$(strTextLine, 1, 16) + Space(4 - Len(CStr(CLng(aryFreeFlowImpd(i))))) _
          + CStr(CLng(aryFreeFlowImpd(i))) + Mid$(strTextLine, 21, 12) + _
          Space(6 - Len(CStr(CLng(dblUpdatedDailyCapacity)))) + CStr(CLng(dblUpdatedDailyCapacity)) + _
          Mid$(strTextLine, 39, 9) + Space(2 - Len(CStr(aryTollClass(i)))) _
          + CStr(aryTollClass(i)) + Mid$(strTextLine, 50)
        End If
        Call mTxtStream15.WriteLine(strTextLine1)

        ' The 2nd line of link-specific block is replaced with the final congested impedance.
        strTextLine2 = Space(8 - Len(CStr(CLng(aryUpdatedImpedance(i))))) + _
        CStr(CLng(aryUpdatedImpedance(i)))
        Call mTxtStream15.WriteLine(strTextLine2)

        ' The 3rd line of link-specific block is replaced with the final weighted volume.
        strTextLine3 = Space(8 - Len(CStr(CLng(aryUpdatedDailyVolume(i))))) + _
        CStr(CLng(aryUpdatedDailyVolume(i)))
        Call mTxtStream15.WriteLine(strTextLine3)
      Next i
    End If
  Loop

  Call mTxtStream14.Close
  Call mTxtStream15.Close
```

```
  Call mTxtStream16.Close

  ' Call the subroutine "Lodpak" to convert the updated input file HRLDXY01.txt to binary file HRLDXY.ayy.
  Call Lodpak

  ' HRLDXY.A00 is used for the following HEVAL purpose.
  If mFileSysObj.FileExists(App.Path + "\HRLDXY.A00") Then
    Call mFileSysObj.DeleteFile(App.Path + "\HRLDXY.A00")
  End If

  OldName = "HRLDXY01.A00": NewName = "HRLDXY.A00"
  Name OldName As NewName

  ' HRLDXY.TXT is used for the Equilibrium Feedback.
  If mFileSysObj.FileExists(App.Path + "\HRLDXY.TXT") Then
    Call mFileSysObj.DeleteFile(App.Path + "\HRLDXY.TXT")
  End If

  OldName = "HRLDXY02.TXT": NewName = "HRLDXY.TXT"
  Name OldName As NewName

  If mFileSysObj.FileExists(App.Path + "\HRLDXY01.TXT") Then
    Call mFileSysObj.DeleteFile(App.Path + "\HRLDXY01.TXT")
  End If
End Sub
```