A REGIONAL TRAFFIC SIMULATION/ASSIGNMENT MODEL FOR EVALUTION OF TRANSIT PERFORMANCE AND ASSET UTILIZATION

Project 03 – 03
January 2004

Midwest Regional University Transportation Center
College of Engineering
Department of Civil and Environmental Engineering
University of Wisconsin, Madison
United States Department of Transportation

Authors:
Athanasios Ziliaskopoulos, Northwestern University
Elaine Chang, Northwestern University

Principal Investigator:
Athanasios Ziliaskopoulos, Northwestern University

EXHIBIT B

Technical Report Documentation Page

1. Report No.

2. Government Accession No.

3. Recipient's Catalog No.
CFDA 20.701

4. Title and Subtitle
A Regional Traffic Simulation/Assignment Model for Evaluation of Transit Performance and Asset Utilization: Time Dependent Multimodal and Intermodal Assignment Models

5. Report Date **February 2004**

6. Performing Organization Code

7. Author/s
Athanasios Ziliaskopoulos, Elaine Chang

8. Performing Organization Report No.
MRUTC 03-03

9. Performing Organization Name and Address
Midwest Regional University Transportation Center
University of Wisconsin-Madison
1415 Engineering Drive, Madison, WI 53706

10. Work Unit No. (TRAIS)

11. Contract or Grant No.
DTRS 99-G-0005

12. Sponsoring Organization Name and Address
U.S. Department of Transportation
Research and Special Programs Administration
400 7th Street, SW
Washington, DC 20590-0001

13. Type of Report and Period Covered
Research Report [Dates]

14. Sponsoring Agency Code

15. Supplementary Notes
Project completed for the Midwest Regional University Transportation Center.

16. Abstract

This report presents three approaches to modeling multimodal and intermodal transportation network problems for evaluation of transit performance and asset management. The first approach is an automobile assignment-based multimodal approach, which captures bus movements in the simulator, but assigns only automobile trips to shortest paths under the assumption that the mode split is fixed. For this model, an inner assignment dynamic user equilibrium (IADUE) assignment algorithm was developed to replace the method of successive averages (MSA). Whereas the MSA approach, which has been the state of the art in simulation-based assignment models, assigns vehicles to paths probabilistically, the IADUE approach systematically searches for the equilibrium assignment. Next, a person assignment-based intermodal approach is presented such that mode and path choices are modeled as simultaneous decisions. This approach uses the same multimodal car and bus simulator as in the vehicle assignment approach, and also uses the IADUE algorithm to determine the equilibrium path assignment; however, instead of automobile-only shortest paths, intermodal least cost paths are calculated at each iteration. Third, an integer linear programming formulation of the system optimal intermodal assignment model is presented. Computational results are presented for all three models.

17. Key Words
dynamic traffic assignment
intermodal model
simulation
transit signal priority

18. Distribution Statement
No restrictions. This report is available through the Transportation Research Information Services of the National Transportation Library.

19. Security Classification (of this report)
Unclassified

20. Security Classification (of this page)
Unclassified

21. No. Of Pages
136

22. Price
-0-

DISCLAIMER

This research was funded by the Midwest Regional University Transportation Center. The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof. The contents do not necessarily reflect the official views of the Midwest Regional University Transportation Center, the University of Wisconsin, the Wisconsin Department of Transportation, or the Federal Highway Administration at the time of publication.

The United States Government assumes no liability for its contents or use thereof. This report does not constitute a standard, specification, or regulation.

The United States Government does not endorse products or manufacturers. Trade and manufacturers names appear in this report only because they are considered essential to the object of the document.

Midwest Regional Urban Transportation Center

A REGIONAL TRAFFIC SIMULATION/ASSIGNMENT MODEL FOR EVALUATION OF TRANSIT PERFORMANCE AND ASSET UTILIZATION

Final Draft Report:
Time Dependent Multimodal and Intermodal User Equilibrium Models

2004

Prepared by: Athanasios Ziliaskopoulos
Elaine Chang

Department of Civil Engineering
Northwestern University
Evanston, IL 60208

NORTHWESTERN
UNIVERSITY

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

This report presents three approaches to modeling multimodal and intermodal transportation network problems for evaluation of transit performance and asset management. The first approach is an automobile assignment-based multimodal approach, which captures bus movements in the simulator, but assigns only automobile trips to shortest paths under the assumption that the mode split is fixed. For this model, an inner approximation assignment dynamic user equilibrium (IADUE) assignment algorithm was developed to replace the method of successive averages (MSA). Whereas the MSA approach, which has been the state of the art in simulation-based assignment models, assigns vehicles to paths probabilistically, the IADUE approach systematically searches for the equilibrium assignment. The automobile assignment-based multimodal model has been implemented in the VISTA dynamic traffic assignment (DTA) software and is being used to evaluate the regional impacts of TSP in Chicago.

Results are presented for a test network, as well as for the regional Chicago network to be used in the TSP study. Tests on the small network were done with different variations of the IADUE algorithm with different search phases limited or suppressed in each case. The tests on the small network show that in most cases, an average cost gap per trip of 1-3% was achieved in 17-27 iterations of DTA using IADUE; however, in one test, a limited path set was used and the extreme and Smith direction searches were suppressed. This test terminated in 11 iterations, but achieved a cost gap of only 7%. The results suggest that there is a tradeoff in the number of iterations required and the quality of the solution achieved. Further, computational results are presented for the regional Chicago network. Tests on the regional Chicago network were performed using MSA to generate a limited path set, and then Phase 3 of the IADUE was used to improve the estimate. For the large-scale network, an average cost deviation of 6.4% and 4.9% per trip were achieved for the AM and off-peak base cases and under 10% for the TSP test scenarios. Validation of base case results with with observed counts and travel times showed that estimates were reasonably evenly distributed both above and below the observed values. The objectives of the TSP scenario analyses part of the study were to observe trends in the regional impacts of TSP, as well as to address specific technical concerns, including the spatial impacts of TSP, the effects of TSP along and across coordinated corridors, the effects of conflicting TSP requests, the effects of TSP with far-side stops, and the importance of route choice in the evaluation of TSP. Measures of performance focused on primary travel impacts, including travel time and travel distance of both buses and cars. Two signal priority strategies were implemented in VISTA. The first strategy provides priority by removing green time from the non-priority phases, while the second strategy provides priority by removing green time from the next occurrence of the priority phases. The first strategy requires less recovery time than the second, while the second maintains the original phase split and thus ensures that non-priority phases receive their full green time. Both strategies may be converted to conditional strategies by having them invoked only for certain routes or levels of schedule delay.

Analysis of TSP test cases revealed that in general, TSP strategies 1 and 2 were both more effective at reducing travel time and travel time variability of priority buses during the off-peak period than during the AM peak. Further, it was found that Strategy 1 was more effective at reducing travel times of priority buses than Strategy 2. In addition, Strategy 1 was found to improve travel speed of general traffic along the TSP corridor, while Strategy 2 did not have a significant effect on general traffic along the TSP corridor. At the same time, neither strategy had

a consistent effect on non-priority corridors. In addition, changes in average bus schedule lateness were both positive and negative; however, those changes were not statistically significant in most cases. Conditional strategies resulted in travel times that were faster for express routes than using regular TSP, since there were fewer competing requests. At the same time, the conditional strategies had little effect on the other priority bus routes. In addition, tests of TSP with multiple corridors showed that the addition of parallel and intersecting TSP corridors did not result in consistent nor significant changes in travel times of priority buses, nor of parallel and intersecting non-priority bus routes. Further, minor improvements were seen in average speed for priority corridors, while at the same time speeds along both parallel and intersecting non-priority routes experienced minimal change with addition of TSP corridors. Further, the addition of parallel and intersecting TSP corridors did not result in consistent nor significant changes in speeds for priority corridors. These findings suggest that impacts on TSP corridors are relatively independent of other parallel and intersecting TSP corridors for the configurations tested.

Next, a person assignment-based intermodal approach is presented such that mode and path choices are modeled as simultaneous decisions. This approach uses the same multimodal car and bus simulator as in the vehicle assignment approach, and also uses the IADUE algorithm to determine the equilibrium path assignment; however, instead of automobile-only shortest paths, intermodal least cost paths are calculated at each iteration. This approach was implemented in VISTA, and tests on a small network converged to a reasonable approximation of equilibrium.

Third, an integer linear programming formulation of the system optimal intermodal assignment model was also presented. Computational results on a test network showed that vehicle and person flow through the network abided by cell transmission, as well as boarding and alighting rules. However, it was found that some unrealistic traveler movements occur because the analytical model optimizes the system costs by adjusting both intermodal path choices and traveler movements. As such, no further development of the analytical formulation of the intermodal problem is planned.

Continuing research efforts will focus on the simulation-based approaches, which separate traffic movement calculations from the optimization of the path assignment, and thus avoid the counter-intuitive traffic movements exhibited by the analytical model. More specifically, future research will focus on refinement of the IADUE algorithm, for example, the sensitivity of the algorithm's overall convergence and computational time to the search parameters will be examined to determine how these parameters affect the efficiency of the search. Further, the effects of deviations from the assumptions of monotonic and continuous cost functions on the convergence of the algorithm will be explored.

1.0 INTRODUCTION

This research project addresses route and mode choice issues that arise within the framework of dynamic traffic assignment (DTA) models. Specifically, the state of the art in simulation-based DTA models has typically relied on the method of successive averages (MSA) to determine the path assignment; however, while the MSA algorithm is guaranteed to converge, it does not guarantee equilibrium. As such, this report proposes an inner approximation dynamic user equilibrium (IADUE) algorithm to more accurately estimate the equilibrium route choice and path assignment. In addition, over the past two decades, extensive research has been done in the field of DTA to improve the realism with which cars are propagated through a network, and how cars affect and are affected by time-varying flows within a network. This research project builds on that foundation by exploring methods of modeling multimodal and intermodal networks of cars and buses, such that interactions between buses and cars on the shared time-varying roadway network are captured and the simultaneity of mode and route choice behavior is addressed.

This project was supported by the Midwest Regional Urban Transportation Center (MRUTC), and was carried out in parallel with a study of the regional impacts of transit signal priority (TSP) in Chicago, funded by Chicago's Regional Transportation Authority (RTA). The algorithms and models presented in this report address issues and experiences encountered during the TSP study, but are also applicable in general to the evaluation of transit performance and asset utilization as these are affected by network congestion and travel behavior.

The model used in the TSP study was an extension of Northwestern University's mesoscopic simulation-based DTA model called VISTA (*Visual Interactive System for Transportation Algorithms*), which was modified to include bus movements and signal priority in the simulation. The model's simulator captures the interactions of cars and buses in the shared roadway network, and its routing and assignment algorithms capture full automobile trips and route choice behavior. Traditionally, TSP has been evaluated through microscopic simulation of individual corridors; however, this corridor-based approach assumes fixed intersection flows and thus ignores route choice behavior. In contrast, the regional assignment-simulation approach captures full automobile trips and thus considers path switching behavior that may result in reaction to changes in travel time caused by TSP.

At the beginning of the TSP study, however, the VISTA DTA model, in accordance with the state of the art in DTA modeling, relied on the method of successive averages (MSA) to estimate the vehicle path assignment. While the MSA algorithm is guaranteed to converge, it does not guarantee equilibrium, so subtle TSP-related changes in travel time might not overcome the inaccuracies of MSA to reveal the expected path choice behavior. The support from the MRUTC allowed the researchers to develop an inner approximation dynamic user equilibrium (IADUE) assignment algorithm to more correctly find the equilibrium. Specifically, instead of probabilistically assigning vehicles to paths, as is done with MSA, the IADUE algorithm seeks the assignment pattern that minimizes the equilibrium gap function. The IADUE algorithm was developed in response to the needs of the TSP study to capture path choice behavior for subtle changes in travel time, but is applicable in general to DTA models where an equilibrium assignment is sought.

While the assignment-simulation approach used in the TSP study captures route choice behavior, the VISTA model, again in accordance with the state of the art in DTA modeling, assigns an exogenously defined number of automobile trips, and thus assumes that the mode share remains fixed regardless of the impact of the transit policy. This model is thus considered in this report to be an automobile assignment-based multimodal DTA model, since it assigns automobiles to a multimodal network of cars and buses. Using the automobile assignment-based multimodal DTA model, the simulator captures the effects of congestion and traffic conditions on bus movements. Further, bus travel times and bus travel time variabilities calculated by the simulator can be used to support an evaluation transit performance and asset utilization. In other words, while the automobile assignment-based multimodal DTA approach does not capture mode choice behavior, the multimodal simulator captures transit movements on the roadway network, and thus allows evaluation of TSP impacts to be performed based on travel time and travel time variability measures. These can then be used as inputs to separate models and analyses of operational efficiency and ridership.

An alternate approach to the automobile assignment-based DTA model is to solve the mode split and assignment as a combined problem. It has long been recognized that travelers' mode and route decisions are interrelated, and many static combined mode split and traffic assignment models have been proposed in the literature; however, static combined models cannot provide a realistic representation of time-varying traffic conditions. Moreover, while much research has been done in the field of DTA, the bulk of this work has focused on automobile movements and automobile path choices, under the assumption that mode choice is relatively unaffected by the minute-to-minute dynamics of traffic conditions; however, as information technology matures, it is expected that mode split will become a real-time decision, instead of a long term planning decision assumed by a third planning step separate from the route choice. An intermodal combined mode choice and DTA model would assign person trips, such that mode, route and transfer choices are modeled as simultaneous decisions. As such, changes in congestion resulting from changes in mode choices would be captured. Further, ridership and mode share, which are central issues in transit policy evaluation, would be directly observed in the model. On the other hand, the difficulty with an intermodal model is that the model would require extensive calibration of cost parameters. As such, in addition to the IADUE algorithm, the support from the MRUTC allowed the researchers to explore methods of solving the combined mode split and assignment problem, such that intermodal trip choices can be modeled.

This report explores both simulation-based and analytical approaches to solving the combined dynamic mode choice and traffic assignment problem. The simulation-based approach builds on the automobile assignment model (developed for the Chicago TSP study) by combining the multimodal car and bus simulator with an intermodal least cost path algorithm. The intermodal path assignment is determined using the IADUE algorithm, which was developed for the TSP study. The report also presents an analytical formulation of the combined dynamic mode choice and traffic assignment problem, which assigns person trips to system optimal intermodal paths, with cars and buses propagated based on the logic of the cell transmission model.

In short, this report presents algorithms to address route choice and mode choice in time-dependent traffic assignment problems. Section 2 describes the automobile assignment-based multimodal model developed for the Chicago TSP study. The single-mode version of the IADUE algorithm is presented for equilibrium assignment of automobile trips on a multimodal network.

Computational results for the automobile assignment-based multimodal model applied to the Chicago regional network. Section 3 describes the extensions made to the automobile assignment-based multimodal model to solve the combined time dependent mode choice and traffic assignment model, with a detailed explanation of the intermodal time dependent least cost path algorithm. The intermodal IADUE algorithm was implemented to determine the equilibrium assignment of intermodal paths, and computational results are presented for a small intermodal test network. Section 4 presents the analytical approach to finding the system optimal intermodal paths, along with computational results obtained for a small test network. The report closes with conclusions suggestions for future research in Section 5.

2.0 AUTOMOBILE ASSIGNMENT-BASED MULTIMODAL MODEL

An automobile assignment-based multimodal model has been developed by extending Northwestern University's mesoscopic DTA model, VISTA, such that VISTA's simulator captures bus movements. This approach captures the interactions of cars and buses in the shared roadway network, and will be used as a tool to evaluate the impacts of transit signal priority (TSP) in the Chicago region. Further, an inner approximation dynamic user equilibrium (IADUE) assignment algorithm has been developed to seek the path assignment that minimizes the equilibrium gap function. This section describes the automobile assignment-based multimodal DTA model implemented in VISTA, beginning with an introduction to DTA and VISTA. The major components of VISTA, including the RouteSim traffic simulator, the time dependent shortest path algorithm and the IADUE assignment algorithm are then described. Computational test results are then provided for a small test network, as well as for the regional Chicago network used in the RTA TSP study.

2.1 Implementation in DTA and VISTA

DTA models are used to estimate time-varying network conditions by capturing traffic flow and route choice behavior. DTA models are typically classified as analytical approaches, including mathematical programming, variational inequality and control theory approaches, or as simulation-based heuristic models. This section focuses on simulation-based DTA. An overview of research in the other approaches can be found in Peeta and Ziliaskopoulos (1).

In general, simulation-based DTA models iterate between a traffic simulation module, a time-dependent shortest path module, and a network assignment module. First, given a set of vehicles and their travel paths, the traffic simulation module replicates complex traffic flow dynamics as the vehicles are propagated through the network. The link travel times reported by the simulator are then used to calculate the time dependent shortest paths. Those shortest paths are combined with all previous sets of shortest paths, and the vehicles are loaded onto the network on those paths. A new iteration then begins as the simulator propagates vehicles through the network along the new combination of paths. The process stops when some user-specified convergence criterion is met.

As with other DTA models, VISTA iterates between traffic simulation, shortest path calculations, and network assignment. This section provides an overview of the VISTA iteration procedure, as well as detailed descriptions of the procedure's main components, namely, the RouteSim simulator, the time dependent shortest path algorithm and the equilibrium assignment algorithm. The iterative procedure used in VISTA, along with inputs and information flow in VISTA, are illustrated in Figure 2-1. The ovals represent data, and the rectangles represent processes. Further, the ovals in bold represent data that are entered into the model, whereas the regular ovals represent data that is derived from calculations or processes within the iterative procedure.

The figure shows that network, automobile OD demand, bus routes and schedules, signals and signal priority data are entered into the model. DTA preparation modules, including the cell generator, bus assignment and auto trip profiler, must be used to convert the data sets to the format required by VISTA. Specifically, the network data is typically obtained in terms of nodes and links, and must be converted into a cell network as required by the RouteSim cell transmission-based simulator. The cell transmission model is further explained in Section 2.2.1. In addition, bus schedules are entered in terms of headways for each bus route, and the bus assignment module converts these to specific bus departure times for each bus run, as required by the time dependent routing algorithm. Similarly, the automobile OD trip tables typically indicate the number of automobile trips made during the simulation period, and the auto trip profiler assigns specific departure times to each automobile.

The iterative procedure begins with RouteSim simulation, which simulates the movement of vehicles through the cell network according to the cell transmission model (see Section 2.2.1). By modeling the complex interactions of vehicles and traffic conditions, the simulator determines the travel time for each link. Based on these link travel times, time dependent shortest paths are calculated for each origin, destination and departure time. These shortest paths are then added to the set of all previously calculated paths, and automobile trips are then assigned to paths from the path set.

Traditionally, the assignment pattern has been determined using an MSA procedure, which assigns vehicles to previous paths with an equal probability. This approach converges, but does not guarantee equilibrium, so to address this issue, this report proposes an IADUE algorithm that assigns vehicles to paths that minimize an equilibrium gap function.

When a vehicle paths have been assigned, if the equilibrium condition is satisfied, then the procedure terminates; otherwise, RouteSim is run using the most recent path assignment, updated link travel times are obtained, and the iterative procedure continues until the equilibrium condition is met.

The procedure shown in Figure 2-1 serves a general outline of the relationships between the simulation, path calculation and assignment steps within VISTA DTA procedure. The actual software implementation may vary to save on computational requirements. For example, the time-consuming shortest path calculation may be omitted after several iterations, since it is expected that few paths new paths will be generated. These types of modifications are provided in the more detailed description of the VISTA DTA implementation provided in Section 2.4.5.

The major components of VISTA, including the RouteSim traffic simulator, the time dependent shortest path algorithm and the IADUE assignment algorithm are described in Sections 2.2, 2.3 and 2.4 respectively.

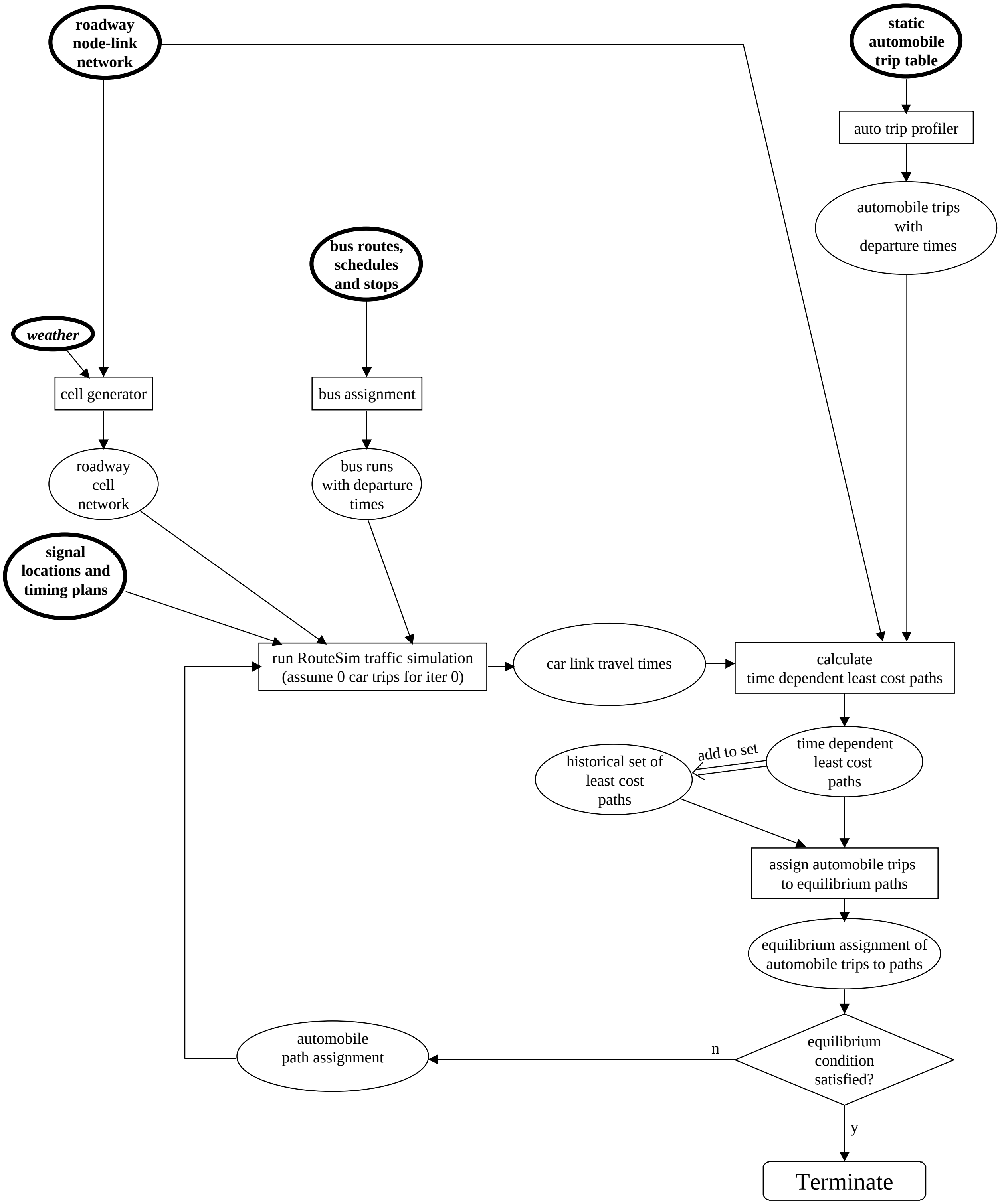

roadway node-link network
static automobile trip table
auto trip profiler
automobile trips with departure times
bus routes, schedules and stops
weather
cell generator
bus assignment
roadway cell network
bus runs with departure times
signal locations and timing plans
run RouteSim traffic simulation (assume 0 car trips for iter 0)
car link travel times
calculate time dependent least cost paths
time dependent least cost paths
add to set
historical set of least cost paths
assign automobile trips to equilibrium paths
equilibrium assignment of automobile trips to paths
equilibrium condition satisfied?
n
y
automobile path assignment
Terminate

Figure 2-1: VISTA Implementation of the Automobile Assignment-based Multimodal Model

2.2 RouteSim Simulator

The RouteSim mesoscopic simulator moves vehicles through the network along predefined paths subject to network phenomena such as capacity limits, signal effects, and bus behavior. This section describes RouteSim's vehicle propagation logic, treatment of signalized intersections and representation of bus movements. In addition, the signal preemption strategies included are described.

2.2.1 Vehicle Propagation

VISTA uses the RouteSim simulator, which propagates vehicles mesoscopically, according to Daganzo's cell transmission model (2), such that link conditions are simulated by evaluating flow at a finite number of intermediate points along each link. The cell transmission model divides a road section into homogeneous cells, so that the length of each cell is equal to the distance traveled by a typical vehicle under free flow traffic conditions in one time step. The state of the system at time *t* is given by the number of vehicles in each cell, $n_i(t)$. Two parameters are used: $N_i(t)$, the maximum number of vehicles that can be present in cell *i* at time *t*, and $Q_i(t)$, the maximum number of vehicles that can flow in cell *i* at time interval *t*. $N_i(t)$ is the product of the cell's length and its jam density, and $Q_i(t)$ is the capacity of cell *i*. The following variables are defined next:

$$S_{i-1}(t) = min\{ n_{i-1}(t),\ Q_{i-1}(t)\} \qquad (2.1)$$

$$R_i(t) = min\{Q_i(t),\ N_i(t)-n_i(t)\} \qquad (2.2)$$

where $S_{i-1}(t)$ is the maximum number of vehicles that can be sent by cell *i-1* to cell *i*, and $R_i(t)$ is the maximum number of vehicles allowed in cell *i*. The number of vehicles that can flow from cell *i-1* to cell *i* when the clock advances from *t* to *t+1*, $y_i(t)$ is defined as the minimum of the two variables:

$$y_i(t) = min\ \{ S_{i-1}(t),\ R_i(t) \} \qquad (2.3)$$

Based on equations (2.1)-(2.3), the recursive relationship of the cell transmission model is expressed as follows:

$$n_i(t+1) = n_i(t) + y_i(t) - y_{i+1}(t) \qquad (2.4)$$

Thus, we can simulate traffic on a network by recursively computing the vehicle occupancy of every cell at each time step during the simulation period. Daganzo showed that the principles of the cell transmission model are consistent with the hydrodynamic theory of traffic flow, but can also capture microscopic effects, such as queuing.

Merging and diverging junctions are explicitly modeled. The merging junction consists of a merging cell (*k*) and two upstream cells (*i* and *j*) that feed the merging cell, as depicted in Figure 2-2. The maximum flows that can be sent by the two upstream cells are defined as $S_i(t)$ and $S_j(t)$. The maximum flow received by cell *k* is defined as $R_k(t)$. The flows satisfy the following relationship:

$$y_{ik}(t) \pounds S_i\ (t),\ \ y_{jk}(t) \pounds S_j\ (t) \ \text{ and} \qquad (2.5)$$

$$y_{ik}(t) + y_{jk}(t) \pounds R_k(t) \qquad (2.6)$$

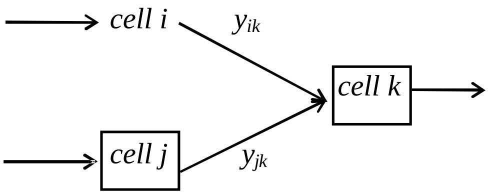


Figure 2-2: Cell connection at a merging junction

In Figure 2-3, a diverging junction is depicted, where the flows from one road section diverge into two road sections. Since our developed simulator is based on assignment traffic, the flows from the diverging cell *k* can be acquired by obtaining the destinations of the vehicles in cell *k*. The sending flows are defined below:

$$S_{k,i}(t) = \min\{\text{ number of vehicles in cell } k \text{ destined to cell } i,\ Q_k(t)\ \} \qquad (2.7)$$

$$S_{k,j}(t) = \min\{\text{ number of vehicles in cell } k \text{ destined to cell } j,\ Q_k(t)\ \} \qquad (2.8)$$

The flow from cell *k* to cell *i* and *j* can be obtained by comparing these values with $R_i(t)$ and $R_j(t)$.

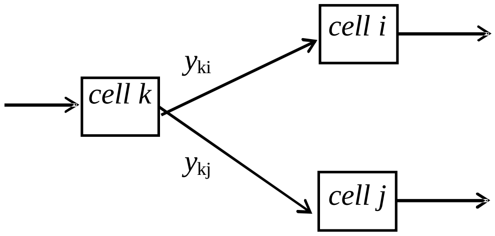


Figure 2-3: Cell connection at a diverging junction

In short, the RouteSim cell transmission-based simulator propagates vehicles through short segments of links based on jam density and maximum flow. Merging and diverging junctions are also explicitly handled.

2.2.2 Signalized Intersections

Modeling the flows at a signalized intersection is more complicated than freeway junctions, since several approaches converge and diverge at a single point and the flows are regulated by a traffic signal. At a four-legged signalized intersection, eight links are connected to a single point, where traffic arrives at and departs from the intersection to four directions. The traffic signal regulates the traffic flow so that conflicting flows are not simultaneously transmitted. Various issues need to be discussed before we can derive the equations that describe the flow transmission at a signalized intersection; i.e., defining the simulation step and the size of the cells close to intersections, modeling turning movements, defining gap acceptance parameters and extending the model to intersections without exclusive turning bays.

2.2.2.1 Modeling Turning Movements

Approaching traffic at a four leg-intersection can move in three directions. We assume that each approach has an exclusive turning lane for each movement (this constraint is later relaxed), which means that we can divide the end cell of the approaching road into three sub-cells representing the three movements as depicted in Figure 2-4.

Sub-cell 1 stores the right turning vehicles, sub-cell 2 the through vehicles and sub-cell 3 the left turning vehicles. The parameters for sub-cells are defined as follows: $sn_{i,k}(t)$ is the number of vehicles in sub-cell k, $SQ_{i,k}(t)$ is the maximum number of vehicles that can flow in sub-cell k, and $SN_{i,k}(t)$ is the maximum number of vehicles that can be present in sub-cell k at time interval t. The variables for the sub-cells are defined as follows:

$$S_{i-1,k}(t) = min\{n_{i-1,k}(t),\ Q_i(t)\} \tag{2.9}$$

$$R_{i,k}(t) = min\{SQ_{i,k}(t), SN_{i,k}(t)-sn_{i,k}(t)\} \tag{2.10}$$

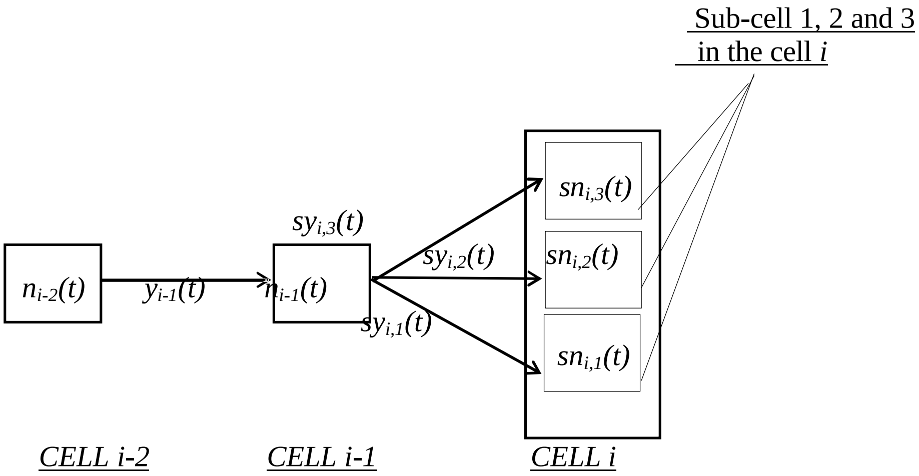


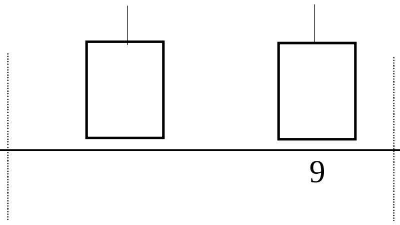
Figure 2-4: Diverging cell connection for approaching roads

where, $n_{i-1,k}(t)$ is the number of vehicles in cell i-1 going to sub-cell k in cell i, $S_{i-1,k}(t)$ is the maximum number of vehicles that can be sent by cell i-1 to sub-cell k, and $R_{i,k}(t)$ is the maximum number of vehicles received by sub-cell k at time interval t. The number of vehicles that can flow from cell $i-1$ to sub-cell k in cell i, $sy_{i,k}(t)$, is given by the following relationship:

$$sy_{i,k}(t) = min\{\ S_{i-1,k}(t), R_{i,k}(t)\ \} \tag{2.11}$$

2.2.2.2 Modeling Intersection Flows

At an intersection, the flows from three directions merge at the first cell of a departing link. Figure 2-5 illustrates the flows of the merging cell for a typical four leg signalized intersection. Cell m is the merging cell for a departing road; cells i, j and k correspond to approaching roads. Sub-cell 1 of cell i, sub-cell 2 of cell j and sub-cell 3 of cell k are transmitting to the merging cell m.



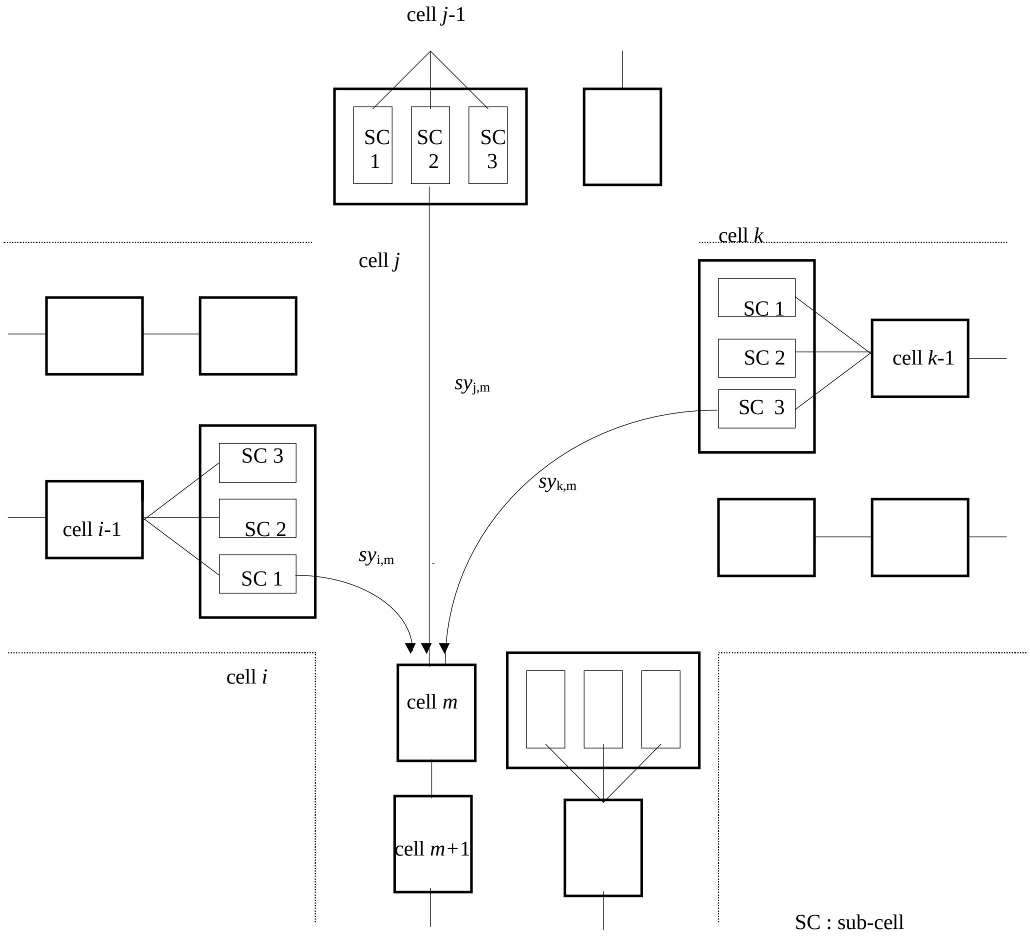


Figure 2-5: Merging cell connections for intersections

Clearly a sub-cell discharges vehicles on green and it does not discharge on red. The signal at an intersection regulates the flow and prevents conflicts. Only the vehicles in a sub-cell on green are allowed to move to cell *m*. The flow from a sub-cell on green to cell *m* is as follows:

$$sy_{x,m}(t)=min\ \{S_{x,y}(t)\ ,\ R_m(t)\} \qquad (2.12)$$

where *x* represents any cell *i*, *j* or *k*, and *y* represents sub-cell 1, 2 or 3. In addition, some portion of the right turning flow in sub-cell 1 of cell *i* is assumed to move to the merging cell *m*, even on a red indication for the movement when *"right turn on red"* is permitted and there is no interference with the vehicles from other directions. Furthermore, some portion of the left turning vehicles are assumed to move to cell *m* when left turn is permitted on green and when turning vehicles do not interfere with those from the opposite through movement.

2.2.2.3 Signal Phasing

Phase sequence is the order that signal phases are displayed during the cycle. Phase sequences may consist of the combinations of the protected and permissive movements depending on the different left turn treatments. Two to six phases make up a typical signal cycle, depending on how the left turns need to be protected. As depicted in Figure 2-6, the phases are generated from the combination of the eight basic NEMA phases, starting at the beginning of each barrier.

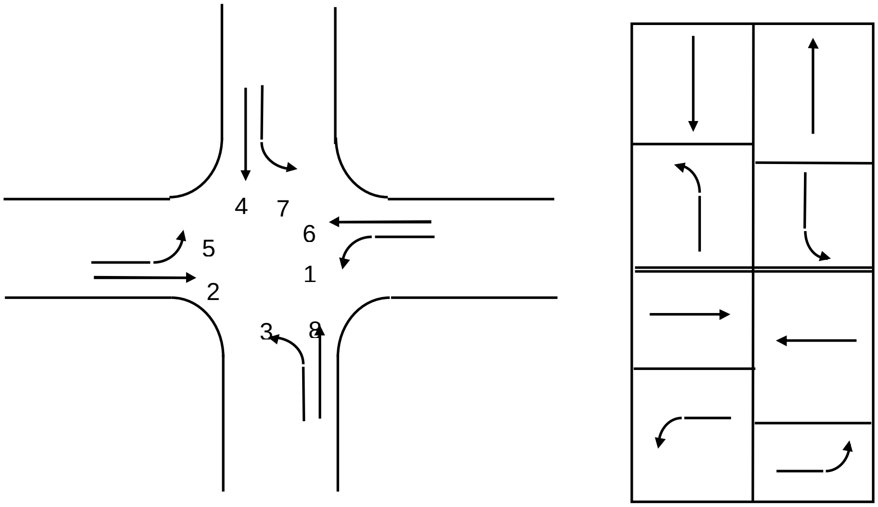


Figure 2-6: NEMA phase numbering system and signal phase definitions

Each phase results in a distinct pattern of flow transmission from a sending sub-cell to a downstream receiving cell. Next, we derive the equations for the receiving cell *m* when phase [2+6] with permissive left turning movements (1 and 5) is set to green; similar equations can be derived for the remaining phases and receiving cells. During phase [2+6], cells *k* and *i* are transmitting, while cell *m* is receiving (Figure 2-5). The flow equations for these transmissions are as follows:

$$y_{i,m}(t) = min\{ S_{i,1}\ (t), R_m(t\} \qquad (2.13)$$

$$y_{j,m}(t) =0 \qquad (2.14)$$

$$y_{k,m}(t) = g(t)min\{ S_{k,3}\ (t), N_m(t)-n_m(t)- y_{i,m}(t)\} \qquad (2.15)$$

where *g(t)* is a *zero-one* binary variable that is set to one if the gap by the left turning vehicles is accepted and zero otherwise. Gap acceptance modeling issues are discussed in the next section. The right turning vehicles from movement 2 have priority over the left turning vehicles, therefore the left turning vehicles have access to the capacity of cell m leftover by the right turning vehicles, *i.e.*, $N_m(t)$-$n_m(t)$- $y_{i,m}(t)$

2.2.2.4 Gap acceptance for turning vehicles

Gap acceptance in RouteSim is handled through the use of conflict definitions. A conflict defines a specific turning movement, the movement that conflicts with it, the priorities and gaps acceptable. This allows for flexible representation of gap acceptance and right turns on red. Driver gap acceptance decisions are mainly affected by the width of the facility to be crossed and the speed of the conflicting traffic. Various critical gap acceptance values for left and right turning vehicles are listed in the Highway Capacity Manual (3). The values range from 5.0 to 6.0 seconds for left turning and 5.5 to 6.5 seconds for right turning vehicles. We use 6 seconds for both right turning and left turning vehicles for simplicity. The gap for turning vehicles can be identified by examining the occupancy of the cell transmitting the conflicting traffic. Since the time step for signalized intersections is two seconds, we consider the occupancy of the last three cells for both left and right turning vehicles.

2.2.2.5 Right turning vehicles on red

Right turns on red are a special case of the conflicts used in general gap acceptance. Vehicles turning right on red should stop before turning and should wait for an appropriate gap, which results in a delay that needs to be accounted for. We assume a stop delay of at least 4 seconds; in other words, if a vehicle is at a right turning sub-cell that is on a red indication and an appropriate gap appears in less than 4 seconds, the vehicle has to wait for at least 4 seconds and accept the next available gap.

2.2.2.6 Intersection without exclusive turning lanes

The end cell of the approaching road is divided into three sub-cells even when there are no right or left turning bays. The values of the sub-cell characteristics for these turning movements are as small as possible, i.e., $SQ_{i,1}(t)$ is set to 1, and $SN_{i,1}(t)$ to 2 so that the flows in or out of the sub-cells are as close as possible to the real flow. We can compute the right turning and through flow more accurately by considering stopped vehicles in sub-cells 2 and 3, respectively. If there is no right turning lane or bay and the through vehicles stop ahead of the right turning vehicles due to a red indication, the right turning vehicles cannot turn even if *"right turn on red"* is allowed. Also, if there is no left turning lane and left turning vehicles are stopped ahead of the through vehicles waiting for a gap, the through vehicles have to wait for the left turning vehicles to clear, *i.e.,* through traffic from sub-cell 2 on green is subject to the occupancy of sub-cell 3.

2.2.3 Bus Movements

Buses differ from cars in size and travel behavior, and realistic simulation of a transportation network requires that buses and their impacts on the network be appropriately represented. First, buses are longer than cars, so they require more roadway capacity than cars. Further, the length of a bus not only affects the amount of space it requires, but also its maneuverability. Specifically, large vehicles require more space to turn and may take more time and cause more of a disruption to traffic when turning at intersections. Bus length is typically modeled using a bus-car equivalence factor, and this captures the added amount of roadway capacity required by buses; however, issues related to maneuverability are not typically captured as this would require data on bus vehicle turning capabilities and roadway geometries, and these are not readily available.

Another difference between buses and cars is that buses stop frequently, at both predefined bus stops and other locations as flag stops. The challenge in modeling bus stops is that buses to not stop at all stops, and the dwell time of each stop varies. Flag stops are even more challenging, as every node and mid-link location is a potential bus stop.

Ideally, dwell times would be determined by the number of boardings and alightings at a stop; however, since the automobile assignment-based approach does not assign person trips, the simulator does not track person movements, so the number of boardings and alightings is not known. Therefore, dwell times can be simulated deterministically or probabilistically. The deterministic approach assumes predetermined dwell times at predetermined bus stops throughout the network. If detailed bus stopping data were collected for each bus stop and route, this approach could approach reality. More often, however, detailed bus stop data is unavailable and an average dwell time is used for each stop. An alternative is to define a probability curve that describes the likelihood of a bus stopping, such that the simulation probabilistically selects the stops and dwells of a bus. This approach also requires stop and dwell time data for development of a probability curve.

The stopping and dwelling of buses affects other vehicles because, assuming no bus bay exists, the stopped bus blocks a lane of traffic for the duration of the dwell. This results in lane-changes to circumvent the stopped bus, which can be captured by microscopic simulators; however mesoscopic and macroscopic do not model individual vehicles nor lanes, so the lane blockage must be modeled as a temporary reduction in roadway capacity.

Buses also tend to accelerate more slowly than smaller vehicles, and this often results in queues behind buses and lane-changing to pass buses. In a microscopic simulation the bus acceleration rate can be calibrated and lane changes can be explicitly modeled, since vehicles are propagated individually; however, mesoscopic and macroscopic simulators don’t explicitly model acceleration. Instead, speed is determined based on the surrounding traffic conditions. Vehicles are moved in platoons, and all vehicles in a platoon are assumed to travel at the same speed. In addition, vehicles are propagated through the network according to first-in-first-out (FIFO) principles, so passing is not modeled.

The RouteSim cell transmission-based simulator has been enhanced to include bus movements. Buses are modeled as longer vehicles; however, the difficulties turning movements have not yet been considered, since data on bus turning radii and intersection geometries have not been obtained. Bus dwells can be modeled either deterministically or probabilistically; however, since people movements are not captured in the simulation, dwells are not associated with boardings and alightings. Buses are held for the duration of the dwell, and the lane blockage resulting from a stopped bus is modeled by temporarily reducing the maximum flow and storage of the cell in which the bus stop is located. Acceleration is not captured in the cell transmission model, so the slower acceleration rate of buses is not simulated. On congested networks, where speeds are limited by capacity more so than by acceleration constraints, this approximation may be reasonable; however in other cases, bus travel times may appear slightly faster than in reality.

2.2.4 Transit Signal Priority

Four signal preemption strategies have been coded into the simulator. The first two strategies provide phase extension or truncation upon detection of a bus, but differ in the phases that follow preemption. The third strategy provides green with compensation to the non-priority phase. The fourth phase provides conditional priority based on schedule adherence and bus priority levels. The strategies are described below along with flow charts showing detailing the logic used in the code.

Strategy 1: If the current phase coincides with the phase requested, then the phase is extended, otherwise the current phase is truncated, so that the bus approach receives a green. With phase extension, the priority phase is held until the bus traverses the intersection. The regular signal timing plan is then recovered by shortening the each of non-priority phases proportionally throughout the following cycle. As such, non-priority phases experience a reduced amount of green time; however, the regular timing plan is recovered within one cycle following the extension. With phase truncation, the non-priority phase is truncated, and the priority phase is invoked and held until its regularly scheduled phase change. As a result, the non-priority phase experiences a reduced amount of green time; however, the original timing plan and coordination with other signals is recovered immediately after the priority phase is served.

Strategy 2: As with Strategy 1, if the current phase coincides with the phase requested, then the phase is extended, otherwise the current phase is truncated, so that the bus approach receives a green; however, Strategies 1 and 2 differ in their recovery approaches. With phase extension, the priority phase is held until the bus traverses the intersection. The regular signal timing plan is then recovered by reducing the next occurrence of the priority phase by the amount of the previous extension. As such, priority and non-priority phases experience their regular total green times; however, the regular timing plan is recovered only after the next occurrence of the priority phase. With phase truncation, the non-priority phase is truncated, and the priority phase is invoked. All phases in the next cycle are then held for their regular phase lengths until the next occurrence of the truncated phase. The recovery is then completed when that phase is extended by the amount by which it was previously truncated. As a result, all phases experience their regular total green times; however, the original timing plan and coordination with other signals is recovered only after the next occurrence of the truncated phase.

Strategy 3: The third strategy is a conditional version of Strategy 1. Specifically, the extension, truncation and recovery logic for Strategy 3 is the same as that described for Strategy 1; however, using Strategy 3, priority treatment is provided only for buses on specified priority routes and/or are late by a predetermined amount. Further, if two conflicting priority requests are received at the same time, the higher priority request (based on route priority and schedule adherence) is selected and served. No new requests are considered during extension, truncation or recovery from an existing priority treatment.

Strategy 4: Just as Strategy 3 is a conditional version of Strategy 1, Strategy 4 is a conditional version of Strategy 2. Specifically, the extension, truncation and recovery logic for Strategy 4 is the same as that described for Strategy 2; however, using Strategy 4, priority treatment is provided only for buses on specified priority routes and/or are late by a predetermined amount. Further, if two conflicting priority requests are received at the same time, the higher priority request (based on route priority and schedule adherence) is selected and served. No new requests are considered during extension, truncation or recovery from an existing priority treatment.

These four strategies will form the basis of the TSP scenarios to be tested and evaluated in the Chicago regional TSP study. Strategies 1 and 2 represent simple priority, providing only phase extension or truncation to reduce any intersection delay to the priority vehicle; however, it is expected that strategy 1 will have less of a negative impact on any coordinated corridors (both along and across the TSP corridor) because the strategy requires less time to return to the original phase schedule following priority. On the other hand, it is expected that Strategy 2 will result in lower delays to uncoordinated cross-street traffic, since the total amount of green time allotted to non-priority phases is not reduced. Further, Strategies 3 and 4 are conditional versions of Strategies 1 and 2 respectively, where the extension, truncation and recovery logic of the conditional strategies is the same as the logic for Strategies 1 and 2; however, priority is only provided to buses on priority routes and exhibiting a specified amount of route delay. Further, if two priority requests are received at the same time, the higher priority request is selected and served. As with Strategies 1 and 2, the conditional strategies do not consider new requests during extension, truncation or recovery from an existing priority treatment.

2.3 Time Dependent Shortest Path Algorithm

Ziliaskopoulos and Mahmassani (4) introduced a time-dependent shortest path algorithm that computes the shortest path from every node and departure time to a destination node without assuming FIFO links. It is built on Equation 2.16, which is a modified version of Cooke and Halsey's (5) functional equation.

$$d_i(t) = \min\{d_i(t), c_{ij}(t) + d_j(t + c_{ij}(t))\} \qquad \forall i \in \Gamma^{-1}(j) \tag{2.16}$$

where

$d_i(t)$ = the minimum time path from node i to destination node D
$c_{ij}(t)$ = the travel time on arc (i,j) at time departing from node i at time t
N = set of all nodes
s = the destination node
T = set of discrete time steps

Initially, the destination node is entered into the scan eligible list. Further, distance labels for the destination node, $d_s(t)$, are set to 0 for all t, and the distance labels for each time step of the remaining nodes are set to ∞. The iterative process begins with removal of the destination node from the scan eligible list. For the node j being scanned, each upstream node i is examined for every time step, $t \in T$. If Equation 2.17 holds for nodes i and j at time t, then the distance label can be improved, so $d_i(t)$ is updated with the improved distance.

$$d_i(t) > c_{ij}(t) + d_j(t + c_{ij}(t)) \tag{2.17}$$

If $d_i(t)$ is changed for at least one time step, then node i is entered into the scan eligible list. The iterative process continues with selection of a new node from the scan eligible list. The process terminates when no nodes remain in the scan eligible list.

2.4 Dynamic User Equilibrium Assignment Algorithm

Simulation-based DTA models typically use the MSA approach to assign vehicles to paths; however, this algorithm does not guarantee an equilibrium solution. For the VISTA model, an IADUE algorithm has been developed to calculate the equilibrium path assignment. This section describes the user equilibrium path assignment problem and the proposed IADUE algorithm, beginning with an introduction to the notation.

2.4.1 Notation

On a network G(V,A,T) where V is the set of nodes, A the set of arcs and [0,T] an assignment period, let d^t_{rs} be the number of vehicles generated at node r and destined to node s ($r,s \in V$) at time t∈[0,T]. Let P be the set of all spatiotemporal paths from all origins to all destinations, i.e. P={p^1, p^2, … p^π}. Each path p^k, $1 \leq k \leq \pi$ belongs to a set P(r,s,t), which contains all paths departing at time t in [0,T] from node r to node s ∈ V.

We denote with

x^{p^k} the number of vehicles choosing to follow path p^k -- Ξ in vector notation

$y^{p^k}(\Xi)$ the travel time on path p^k -- Ψ(Ξ) in vector notation

The demand relationships $\sum_{p^k \in P(r,s,t)} x^{p^k} = d_{rs}^t$ form a closed, bounded, convex space D⊂R^π. As such, any assignment Ξ in D is feasible, given that the traffic flow propagation law adopted prevents gridlock, and thus allows all vehicles to complete their trips within time T. Practically, this means that the traffic simulator can perform any assignment in D given a sufficient assignment period T, so that all assigned vehicles exit the network. Thus, given a Ξ in D, the travel times, Ψ(Ξ), can be computed using a simulator to move vehicles through the network.

The space D is compact and Ψ(Ξ) further is assumed to be continuous; therefore, there is at least one solution to the above VI problem. Uniqueness is more difficult to show since it requires that the cost function Ψ(Ξ) is monotonic. Mathematically, monotonicity is stated as follows:

$$\left(\Psi(\Xi_1) - \Psi(\Xi_2)\right) \cdot (\Xi_1 - \Xi_2) \geq 0 \ \forall \Xi_1, \Xi_2 \in \mathrm{D} \qquad (2.18)$$

In fact, traffic network realities, such as signals prevent the travel time cost function from being truly continuous. Further, due to temporal interactions of the vehicles assigned earlier with those assigned later, monotonicity may not hold. Lack of monotonicity precludes proof of solution uniqueness, as well as algorithm convergence. However, in spite of these assumptions, in practice, the algorithm gives good results in a reasonable amount of time.

2.4.2 Equilibrium Conditions

The proposed IADUE algorithm seeks the Wardrop user equilibrium path assignment Ξ* in D. The Wardrop equilibrium occurs when no driver has a less costly alternative route. Further, it is assumed that a drivers' selection of an alternative path is a unilateral decision based on the current traffic conditions. The equilibrium can be defined mathematically in terms of the demand gap, as well as the cost gap, as described in this section.

2.4.2.1 Equilibrium Demand Gap

According to the Wardrop principle,

$$y^{p^x}\left(\Xi^*\right) > y^{p^y}\left(\Xi^*\right) \text{ implies that } x^{p^x} = 0 \ \ \forall\, p^x, p^y \in P(r,s,t) \ \forall r,s,t \qquad (2.19)$$

In other words, at the equilibrium path assignment, any path that is more costly than the minimum cost path remains unused. Alternatively, it can be stated that

$$\Psi\left(\Xi^*\right)^T \Xi > \Psi\left(\Xi^*\right)^T \Xi^* \ \ \forall \Xi \in \mathrm{D} \qquad (2.20)$$

This formulation is less strict than the previous one as it allows some drivers to choose more costly routes, as long as the total route cost is reduced. As such, it is a necessary condition for equilibrium, but not sufficient.

Suppose we have a solution $\Xi' \in \mathrm{D}$ such that

$$\psi^{p^x}(\Xi') > \psi^{p^y}(\Xi') \text{ for some } p^x, p^y \in P(r,s,t) \text{ and } r,s,t \qquad (2.21)$$

but that $\xi^{p^x} > 0$ for some p^x, where $p^x \neq p^y$ $\xi^{\text{rstp'}} > 0$

This suggests that switching vehicles along p^x to the cheaper route p^y will reduce the total cost by

$$\psi^{p^x}(\Xi')\xi^{p^x} - \psi^{p^y}(\Xi')\xi^{p^y} > 0 \qquad (2.22)$$

Therefore, if the resulting path assignment is Ξ'', then

$$\Psi(\Xi')^T\, \Xi'' < \Psi(\Xi')^T\, \Xi' \;\; \forall \Xi'' \in D \qquad (2.23)$$

As such, we have shown that

$$\psi^{p^x}(\Xi^*) > \psi^{p^y}(\Xi^*) \text{ implies that } \xi^{p^x} = 0 \;\; \forall\; p^x, p^y \in P(r,s,t) \;\; \forall r,s,t \qquad (2.24)$$

if and only if

$$\Psi(\Xi^*)^T\, \Xi < \Psi(\Xi^*)^T\, \Xi^* \;\; \forall \Xi \in D \qquad (2.25)$$

Therefore, an equilibrium solution, Ξ*, exists where

$$\Psi(\Xi^*)^T\, (\Xi - \Xi^*) \geq 0 \quad \forall \Xi \in D \qquad (2.26)$$

This formulation is known as the variational inequality (VI) formulation of the user equilibrium traffic assignment problem.

2.4.2.2 Equilibrium Cost Gap

Let $\psi^{p_{\min rst}}(\Xi) = \min\limits_{p \in P(r,s,t)} \left\{ \psi^{p}(\Xi^*) \right\}$

and let $\Psi_{\min}(\Xi)$ be the matrix of $\psi^{p_{\min rst}}(\Xi)$ for all paths.

It is clear that

$$\xi^{p^x}\left(\psi^{p^x}(\Xi) - \psi^{p_{\min rst}}(\Xi)\right) \geq 0 \quad \forall p^x \in P(r,s,t) \quad \forall r,s,t \quad \forall \Xi \in D \qquad (2.27)$$

At a Wardrop equilibrium solution, Ξ^*, no driver has a less costly alternative route, or in other words, only the minimum cost paths are used. This condition can be expressed as

$$\xi^{p^x *}\left(\psi^{p^x}(\Xi^*) - \psi^{p_{\min rst}}(\Xi^*)\right) = 0 \quad \forall p^x \in P(r,s,t) \quad \forall r,s,t \qquad (2.28)$$

For the whole system, the equilibrium can be expressed as follows:

$$\sum_{r \in V} \sum_{s \in V} \sum_{t \in [0,T]} \sum_{p^x \in P(r,s,t)} \xi^{p^x *}\left(\psi^{p^x}(\Xi^*) - \psi^{p_{\min rst}}(\Xi^*)\right) = 0 \qquad (2.29)$$

or in matrix form as

$$\Xi^{*T}\left(\Psi(\Xi^*) - \Psi_{\min}(\Xi^*)\right) = 0 \qquad (2.30)$$

The IADUE algorithm uses the concepts of both the demand gap and the cost gap in order to find the dynamic user equilibrium path assignment.

2.4.3 Minimization of Gap Functions

The equilibrium conditions defined in Section 3 are difficult to solve directly, so instead gap or merit functions are formulated such that the optimal solutions of these functions coincide with points that satisfy the equilibrium conditions. In mathematical terms, if we let Ω be the set of equilibrium solutions that satisfy the demand and cost gap conditions in Section 2.4.2, then V is a gap function for those conditions if

$$\begin{aligned} &\text{i.}\quad \mathrm{V}(\Xi)=0 \ \ \forall\Xi\in\Omega \\ &\text{ii.}\quad \mathrm{V}(\Xi)\geq 0 \ \ \forall\Xi\in D \end{aligned} \tag{2.31}$$

Translation of the equilibrium conditions into gap functions allows us to take advantage of numerical search approaches, which are commonly used for optimization problems. More specifically, in the proposed IADUE algorithm, the assignment is found by selecting a descent direction (in the assignment space D) and step length that minimize the gap function. As such, the general procedure for the DTA model with IADUE assignment comprises iteration through simulation, path generation, selection of a search direction and step length, and assignment to the updated solution.

The proposed DTA-IADUE algorithm includes three different search phases, each consistent with the procedure outlined above, but differing in their gap functions and descent directions. The different phases take advantage of different convergence properties of each gap function and descent direction, and will thus be used in different stages of the search for the equilibrium solution. The gap functions and descent directions associated with each phase are described next.

2.4.3.1 Phase 1

The first phase of the DTA-IADUE algorithm uses an extreme direction search approach to minimize the gap function,

$$V_{extreme}(\Xi)=\sum_{i=1}^{N}\max\left\{0,-\Psi(\Xi)^T(P_i-\Xi)\right\} \tag{2.32}$$

where P_i is an extreme point of the convex hull D, and
N is the number of extreme points defined.

Each extreme point P_i represents an all-or-nothing path assignment, and any point in D, including the equilibrium solutions, $\Xi^*\in\Omega\subseteq D$, can be defined as a convex combination of the extreme points P_i.

As required by the definition of the gap function (2.31)(ii), it is clear that $\mathrm{V}_{extreme}(\Xi)\geq 0 \ \ \forall\Xi\in D$. Further, to show that it is consistent with (2.31)(i), we recall from (2.26) that the equilibrium point, Ξ^*, occurs if and only if

$$\Psi(\Xi^*)^T\,(\Xi-\Xi^*)\geq 0 \quad \forall\Xi\in D$$

Since it would be infeasible to check that this condition holds for all points $\Xi\in D$, we take advantage of the property that D is a convex hull, and accept that if the condition holds for a set of extreme points of D, then it must also hold for D. The condition is expressed in terms of extreme points as (2.33).

$$\Psi(\Xi^*)^T\,(P_i-\Xi^*)\geq 0 \quad \forall i\in[1,N] \tag{2.33}$$

Next, by taking the negative of the cost and reversing the inequality, we get (2.34).

$$-\Psi\left(\Xi^*\right)^T \left(P_i - \Xi^*\right) \le 0 \quad \forall i \in [1, N] \tag{2.34}$$

Therefore, at an equilibrium solution, $\Xi^* \in \Omega$, it is clear that

$$\max\left\{0, -\Psi\left(\Xi\right)^T \left(P_i - \Xi\right)\right\} = 0 \quad \forall i \in [1, N]$$

and thus that

$$V_{extreme}\left(\Xi\right) = \sum_{i=1}^{N} \max\left\{0, -\Psi\left(\Xi\right)^T \left(P_i - \Xi\right)\right\} = 0 \tag{2.35}$$

On the other hand, since (2.26) holds only at equilibrium Ξ^*, then at some non-equilibrium solution Ξ'

$$\Psi\left(\Xi'\right)^T \left(\Xi - \Xi'\right) < 0 \quad \text{for some } \Xi \in D$$

Since D is a convex hull, this suggests that

$$\Psi\left(\Xi'\right)^T \left(P_i - \Xi'\right) < 0 \quad \text{for some } i \in [1, N] \tag{2.36}$$

and $$-\Psi\left(\Xi'\right)^T \left(P_i - \Xi'\right) \le 0 \quad \text{for some } i \in [1, N] \tag{2.37}$$

Therefore, at a non-equilibrium solution, $\Xi' \notin \Omega$, it is clear that

$$\max\left\{0, -\Psi\left(\Xi\right)^T \left(P_i - \Xi\right)\right\} > 0 \quad \text{for some } i \in [1, N]$$

and thus that

$$V_{extreme}\left(\Xi\right) = \sum_{i=1}^{N} \max\left\{0, -\Psi\left(\Xi\right)^T \left(P_i - \Xi\right)\right\} > 0 \tag{2.38}$$

Therefore, we have shown that the gap function (2.32) is indeed consistent with the definition of the gap function listed in (2.31), and thus that the minimum point of (2.32) occurs only at an equilibrium path assignment. Moreover, it is worth noting that the value of the gap function provides a measure of the extent to which the current solution deviates from equilibrium.

To find the minimum point of (2.32), a numerical search approach is used, such that a descent direction (in the assignment space D) and step length are selected, a new solution is found and the search procedure is repeated. In Phase 1, the algorithm selects, as the descent direction, an extreme point composed of an all-or-nothing assignment to the shortest paths. The new solution thus becomes

$$\Xi_{n+1} = \left(1 - \lambda\right)\Xi_n + \lambda P_n \tag{2.39}$$

where n is the iteration number,
P_n is the extreme point descent direction selected for iteration n, and
λ is the step length ($0 \le \lambda \le 1$).

As such, higher values of λ correspond to longer steps toward the extreme point, which corresponds to loading more travelers on the shortest paths. In other words, travelers are switched from higher cost paths to lower cost paths.

While Phase 1 is structured as a search procedure, the main purpose of the phase is not so much to find the equilibrium solution, but rather to generate paths and extreme points for Phases 2 and 3. When sufficient numbers of paths and extreme points have been generated, Phase 1 is terminated.

2.4.3.2 Phase 2

The gap function used in Phase 2 is similar to that of Phase 1, except that the demand gap values are squared, as shown in (2.40). This gap function is labeled V_{Smith}, since it is based on a gap function that was proposed by Smith for the static path flow equilibrium problem (6).

$$V_{Smith}\left(\Xi\right)=\sum_{i=1}^{N}\max^{2}\left\{0,-\Psi\left(\Xi\right)^{T}\left(P_i-\Xi\right)\right\} \qquad (2.40)$$

As with the gap function from Phase 1, it is clear that the non-negativity requirement (2.31)(ii) holds. Further, using the same arguments as those in Section 2.4.3.1, given that (2.34) holds at the equilibrium solution, it is clear that

$$\max^{2}\left\{0,-\Psi\left(\Xi\right)^{T}\left(P_i-\Xi\right)\right\}=0 \quad \forall i\in[1,N]$$

and thus that

$$V_{Smith}\left(\Xi\right)=\sum_{i=1}^{N}\max^{2}\left\{0,-\Psi\left(\Xi\right)^{T}\left(P_i-\Xi\right)\right\}=0 \qquad (2.41)$$

Further, given that (2.37) holds for a non-equilibrium assignment $\Xi'\notin\Omega$, it is clear that

$$\max^{2}\left\{0,-\Psi\left(\Xi\right)^{T}\left(P_i-\Xi\right)\right\}>0 \quad \text{for some } i\in[1,N]$$

and thus that

$$V_{Smith}\left(\Xi\right)=\sum_{i=1}^{N}\max^{2}\left\{0,-\Psi\left(\Xi\right)^{T}\left(P_i-\Xi\right)\right\}>0 \qquad (2.42)$$

Therefore, as with the gap function from Phase 1, that gap function for Phase 2 (2.40), is indeed consistent with the definition of the gap function listed in (2.31). In other words, the minimum point of (2.40) occurs only at an equilibrium path assignment. The primary difference between the $V_{extreme}$ and V_{Smith} is that V_{Smith} is differentiable. Smith took advantage of this property to prove the convergence rate of his algorithm.

To find the minimum point of (2.40), a descent direction (in the assignment space D) and step length are selected, a new solution is found and the search procedure is repeated. In Phase 2, the descent direction is composed of a weighted average of the extreme directions that deviate from equilibrium, as shown in (2.43). This direction is labeled Δ_{Smith} since it was proposed by Smith (6).

$$\Delta_{Smith\ n}=\frac{\sum_{i=1}^{N}\max\left(0,-\Psi\left(\Xi_n\right)^{T}\left(P_i-\Xi_n\right)\right)\cdot\left(P_i-\Xi_n\right)}{\sum_{i=1}^{N}\max\left(0,-\Psi\left(\Xi_n\right)^{T}\left(P_i-\Xi_n\right)\right)} \qquad (2.43)$$

The new solution thus becomes

$$\Xi_{n+1} = \Xi_n + \mathrm{l}\Delta_{Smith\ n} \qquad (2.44)$$

where n is the iteration number,
$\Delta_{\text{Smith n}}$ is the descent direction for iteration n, and
λ is the step length ($0 \leq \lambda \leq 1$).

Smith showed that assuming a monotone differentiable cost function, the $\Delta_{\text{Smith n}}$ direction guarantees converge a rate of

$$\frac{d}{dt} V_{Smith}\left(\Xi(t)\right) \leq -2V_{Smith}\left(\Xi(t)\right) \qquad (2.45)$$

where F=F(t) is a differentiable function of t defined for all t≥0.

In computational tests, this approach has made efficient progress in seeking out the region of the equilibrium solution; however, as the solution converges, the rate of convergence drops. As such, a third search procedure is invoked. The gap function and descent directions used in the third phase are described next.

2.4.3.3 Phase 3

Whereas the V_{extreme} and V_{Smith} gap functions were based on the demand gap formulations of equilibrium, the gap function used in the third phase is based on the cost gap formulation of the equilibrium. This gap function is labeled V_{cost} and is shown in (2.46).

$$V_{cost}\left(\Xi\right) = \sum_{r\in V}\sum_{s\in V}\sum_{t\in[0,T]}\sum_{p\in P(r,s,t)} \mathrm{x}^{p}\left(\mathrm{y}^{p}\left(\Xi\right) - \mathrm{y}^{p_{\min rst}}\left(\Xi\right)\right) \qquad (2.46)$$

As required by the definition of the gap function (2.31)(ii), it is clear that

$$\mathrm{y}^{p}\left(\Xi\right) - \mathrm{y}^{p_{\min rst}}\left(\Xi\right) \geq 0 \text{ and } \mathrm{x}^{p} \geq 0 \quad \forall p \in P(r,s,t) \quad \forall r,s,t$$

so
$$V_{cost}\left(\Xi\right) = \sum_{r\in V}\sum_{s\in V}\sum_{t\in[0,T]}\sum_{p\in P(r,s,t)} \mathrm{x}^{p}\left(\mathrm{y}^{p}\left(\Xi\right) - \mathrm{y}^{p_{\min rst}}\left(\Xi\right)\right) \geq 0 \qquad (2.47)$$

Further, to show that it is consistent with (2.31)(i), we recall from (2.27) and (2.28) that

$$\mathrm{x}^{p^x}\left(\mathrm{y}^{p^x}\left(\Xi\right) - \mathrm{y}^{p_{\min rst}}\left(\Xi\right)\right) \geq 0 \quad \forall p^x \in P(r,s,t) \quad \forall r,s,t \quad \forall \Xi \in D$$

and that at the equilibrium point, Ξ^*,

$$\mathrm{x}^{p^x *}\left(\mathrm{y}^{p^x}\left(\Xi^*\right) - \mathrm{y}^{p_{\min rst}}\left(\Xi^*\right)\right) = 0 \quad \forall p^x \in P(r,s,t) \quad \forall r,s,t \qquad (2.48)$$

As such, it is clear that $V_{cost}\left(\Xi\right) = 0$ only for $\Xi \in \Omega$, as required by (2.31)(i).

In this phase, the descent direction is X_n, where each element of X_n is set to the values shown in (2.49).

$$x^{p^x n} = \begin{cases} \mathsf{x}^{p^x n} & \text{if } p \text{ is neither the min nor max cost path for rst n} \\ \mathsf{x}^{p^x n} + \mathsf{x}^{p_{\max rst} n} & \text{if } p \text{ is the min cost path for rst n} \\ 0 & \text{if } p \text{ is the max cost path for rst n} \end{cases} \qquad (2.49)$$

The new solution thus becomes

$$\Xi_{n+1} = (1 - \mathrm{l})\,\Xi_n + \mathrm{l}\, X_n \qquad (2.39)$$

where n is the iteration number,
X_n is the descent direction selected for iteration n, and
λ is the step length (0≤λ≤1).

This descent direction shifts travelers from the highest cost path to the lowest cost path, while leaving travelers on mid-cost paths unchanged for each origin-destination-departure time. In computational tests, this approach has made efficient progress in the later stages of a search; when the solution is near convergence. As such, this phase is the last phase in the algorithm and is continued until the equilibrium assignment is obtained.

2.4.4 Algorithm

As with traditional DTA models, which typically rely on MSA, the DTA model with IADUE uses simulation to determine link costs and a TDSP algorithm to calculate new paths, and in general follows the procedure outlined in Figure 2-1. However, rather than assigning vehicles to paths using a random averaging approach, the DTA-IADUE model uses a search strategy to determine the path assignment. Specifically, the IADUE assignment algorithm numerically optimizes the gap function to find the solution that comes closest to the equilibrium. The optimization of the non-linear gap function is done by first selecting a descent direction in D, and then selecting a step length in that direction. The proposed algorithm includes three phases (outside of the initialization phase, Phase 0) where each phase differs by the gap function defined and the descent direction selected. Detailed descriptions of the gap functions and descent directions used in each phase are given in Section 2.4.3. The DTA-IADUE algorithm is shown in Figure 2-7.

Phase 0 – Initialize
Set n=0.
Set link travel time to free flow.
Calculate least cost paths for each rst.
Set Ξ_0 to all-or-nothing assignment of d^t_{rs} to least cost path for rst.

Phase 1 – Search in Extreme Point Direction
Simulate traffic conditions with assignment Ξ_n.
Update link travel times.
If (n=0) or ((|path set|$_n$ - |path set|$_{n-1}$) > **routing-stop-precentage***|path set|$_{n-1}$)
    Calculate least cost paths for each rst.
    Add new paths to path set.
Set descent direction (a-o-n assignment to least cost paths, extreme point P_n).

```
Select step length $\lambda_n = \text{argmin}_\lambda\ (V_{demand}((1-\lambda)\Xi_n + \lambda P_n))$ using golden section search.
Assign demand to $\Xi_{n+1} = (1-\lambda_n)\Xi_n + \lambda_n P_n$ .
If n < assignment-max-extremes
   Repeat Phase 1.
Else
   Go to Phase 2.
Set n=n+1.

Phase 2 – Search in Smith Direction
Simulate traffic conditions with assignment $\Xi_n$.
Update link travel times.
Choose descent direction $\Delta_n$ .
Select step length $\lambda_n = \text{argmin}_\lambda\ (V_{smith}(\Xi_n + \lambda\Delta_n))$ using golden section search.
Assign demand to $\Xi_{n+1} = \Xi_n + \lambda_n\Delta_n$ .
If $V_{smith}(\Xi_{n+1})\ /\ V_{smith}(\Xi_n)$ > assignment-switch-ratio
   Repeat Phase 2
Else
   Go to Phase 3
Set n=n+1

Phase 3 – Search in Non-extreme Direction
Simulate traffic conditions with assignment $\Xi_n$.
Update link travel times.
Choose descent direction $X_n$.
Select step length $\lambda_n = \text{argmin}_\lambda\ (V_{cost}((1-\lambda)\Xi_n + \lambda P_n))$ using golden section search.
Assign demand to $\Xi_{n+1} = \Xi_n + \lambda X_n$ .
If ( $V_{cost}(\Xi_{n+1})\ /\ \Sigma_{rstp}\psi_{rstp}(\Xi_{n+1})$ > cost-gap-percentage)
   Repeat Phase 3.
   Set n=n+1.
Else
   Terminate.
```

Note:
The routing-stop-precentage, assignment-switch-threshold, assignment-switch-ratio and cost-gap-percentage are convergence parameters that can be set within the VISTA implementation of DTA-IADUE.

Figure 2-7: The DTA-IADUE algorithm

The algorithm begins with Phase 0, which generates an initial assignment Ξ_0 based on all-or-nothing assignment to the least cost paths under free flow network conditions. The algorithm then begins Phase 1 by simulating vehicle flow along the paths defined by Ξ_0. Link travel times are then updated according to the simulation results. Next, shortest paths are calculated based on the updated link travel times, and an extreme point, P_0, is selected as the all-or-nothing assignment of vehicles to those least cost paths. P_0 represents the descent direction for this iteration, such that the next solution, Ξ_1, will be a linear combination of Ξ_0 and P_0. The step length, λ_0, is selected through a line search process to find the value of λ_0 that minimizes the gap function, $V_{demand}((1-\lambda)\Xi_n + \lambda P_n))$. A new assignment is then set to $\Xi_1 = (1-\lambda_0)\Xi_0 + \lambda_0 P_0$. Phase 1 is repeated until a pre-specified number of extreme points is created (number of maximum extreme points is named the **assignment-max-extremes** in the VISTA implementation). The extreme points generated in Phase 1 are then used in Phase 2.

Shortest paths are calculated only at the beginning of Phase 1. As more iterations of Phase 1 are completed, fewer and fewer new paths are generated. When the number of new paths generated in a single iteration drops below a selected threshold (named the **routing-stop-percentage** in VISTA), path generation is stopped, and the rest of the IADUE algorithm continues, attempting to equilibrate trips among only the paths already generated. This saves time by eliminating the potentially time-consuming TDSP calculation.

Typically, the **assignment-switch-threshold** is set to 0.1, indicating that if the new assignment does not move more than 10% away from the previous solution toward the extreme point, then Phase 1 has converged. This does not mean that the equilibrium solution has been found. Rather, it simply means that the descent direction and gap function have brought the solution close enough to the equilibrium that Phase 2 can be invoked. In other words, Phase 1 is appropriate for moving quickly from a distant point toward the region of the equilibrium solution, but then converges slowly when it gets close to the solution. As such, Phase 2, using the Smith gap function and Smith descent direction, is invoked to improve the speed of convergence as the algorithm approaches the equilibrium solution.

Phase 2 is similar to Phase 1 in that it begins with simulation of the vehicles on an initial assignment, Ξ_n, passed down from the previous phase and then updates link travel times based on the simulation results. However, no shortest paths nor extreme points are calculated in this phase. Instead, Phase 2 relies on a descent direction, $\Delta_{Smith\ n}$, which is described in Section 2.4.3.2. The step length, λ_n, is selected through a line search process to find the value of λ_n that minimizes the Smith gap function, $V_{smith}(\Xi_n + \lambda\Delta_{Smith\ n}))$. A new assignment is then set to $\Xi_{n+1} = \Xi_n + \lambda_n\Delta_{Smith\ n}$. If the ratio of value of $V_{smith}(\Xi_{n+1})$ to $V_{smith}(\Xi_n)$ is less than a pre-specified convergence threshold (named the **assignment-switch-ratio** in the VISTA implementation), then Phase 2 is repeated. Otherwise, the algorithm enters Phase 2. In either case, the iteration counter, n, is incremented before the next iteration or phase is begun.

Typically, the **assignment-switch-ratio** is set to 0.9, indicating that if the new assignment does not improve the gap value by more than 10% of the previous gap value, then Phase 2 has converged. This does not mean that the equilibrium solution has been found. Rather, it simply means that the Smith direction and Smith gap function have brought the solution close enough to the equilibrium that Phase 3 can be invoked. In other words, Phase 2 is appropriate as in intermediate search for the equilibrium solution, but then converges slowly when it gets close to the solution. As such, Phase 3, using a cost-based descent direction and gap function, is invoked to improve the speed of convergence as the algorithm approaches the equilibrium solution.

Again, as with Phases 1 and 2, Phase 3 begins with simulation of the vehicles on an initial assignment, Ξ_n, passed down from the previous phase and then updates link travel times based on the simulation results. Phase 3 then calculates a descent direction, X_n, as described in (2.49).

The step length, λ_n, is selected through a line search process to find the value of λ_n that minimizes the cost gap function, $V_{cost}((1-\lambda)\Xi_n + \lambda P_n)$. A new assignment is then set to $\Xi_{n+1} = (1-$

$\lambda_n)\Xi_n + \lambda P_n$. If the ratio of value of $V_{cost}(\Xi_{n+1})$ to the total system travel time exceeds a pre-specified convergence threshold (named the **cost-gap-percentage** in the VISTA implementation), then the iteration counter, n, is incremented and Phase 3 is repeated. Otherwise, the algorithm terminates.

Typically, the **cost-gap-percentage** is set to 0.1, indicating that if the gap function value is less then 10% of the total system travel time, then the algorithm has converged to a solution that reasonably approximates the equilibrium solution. More specifically, a cost gap of 0 suggests that all travelers are indeed assigned to the minimum cost paths, and any positive value of the cost gap divided by the total system travel time essentially indicates the amount of inefficiency in an average trip in the system. As such, value of the cost gap divided by the total system travel time also gives a measure of how far off the equilibrium our solution is.

Each phase includes a line search for the optimal step length. Tests of the algorithm have shown that the golden section search provides relatively rapid convergence. The golden section search procedure is carried out as shown in Figure 2-8.

```
Step 1
Compute $F(\lambda)$ for the following values of $\lambda$ (run RouteSim for each test):
        (i) $\lambda_a = 0$
        (ii) $\lambda_b = \lambda_a$ + assignment-search-low($\lambda_d - \lambda_a$)
        (iii) $\lambda_c = \lambda_a$ + assignment-search-high($\lambda_d - \lambda_a$)
        (iv) $\lambda_d = \min\{1.5*\lambda_{n-1},0.8\}$, where $\lambda_{n-1}=1$ for n=0

Step 2
If $F(\lambda_a)$ or $F(\lambda_b)$ is the minimum then set
$\lambda_d=\lambda_c$
$\lambda_c=\lambda_b$
$\lambda_a=\lambda_a$
$\lambda_b=\lambda_a+$ assignment-search-high($\lambda_{c\ new} - \lambda_a$)
and go to Step 3.
Else if $F(\lambda_c)$ or $F(\lambda_d)$ is the minimum then set
$\lambda_a=\lambda_b$
$\lambda_b=\lambda_c$
$\lambda_d=\lambda_d$
$\lambda_c=\lambda_{b\ new}+$ assignment-search-low ($\lambda_d - \lambda_{b\ new}$)
        and go to Step 3.

Step 3
If $(\lambda_{last} - \lambda_{last-1})$ < assignment-stop-difference
   or $(V(\lambda_{last}) / V(\lambda_{last-1}))$ > assignment-previous-gap-ratio
   or $(V(\lambda_{best}) / V(\lambda_{worst}))$ < assignment-worst-gap-ratio
Stop line search and select best $\lambda$.
        Else go to Step 1.
```

Note:
The assignment-search-high, assignment-search-low, assignment-stop-difference, assignment-previous-gap-ratio and assignment-worst-gap-ratio are parameters that can be set within the VISTA implementation of DTA-IADUE.

Figure 2-8: Golden section search procedure for step length selection

Essentially, the golden section search tests the gap values at different points, or λ values, along the search direction. With each iteration, it tests four λ values and narrows it's search by dropping one of the outside values, and adding and testing a new λ value in the middle. The locations of the middle values can be specified by parameters named **assignment-search-low** and **assignment-search-high**.

The test of each λ value requires simulation of traffic conditions resulting from the assignment using the test value of λ, updating of the link travel times according to the test assignment, and calculation of the resulting gap function value.

The line search is terminated if one of three convergence conditions is met. The first condition requires that the difference the last two lambda values tested is less than a specified **assignment-stop-difference**. According to the golden search method the most recently tested lambda values are always in the middle, so these are essentially the two closest values. If these are extremely close together, then the search is considered to have converged. A typical value for the assignment-stop-difference is 0.15.

The second convergence criterion requires that the ratio of the gap values produced by the last two test lambda values is greater than a specified **assignment-previous-gap-ratio**, which is typically set to 0.9. This condition is based on the idea that if the gap values produced by the last two tests are essentially the same, then the search procedure has found a flat spot along the curve, which is thus taken as the minimum point along the descent direction. This condition can only invoke termination of the line search after the first full line search iteration. This limit ensures that lambda values which are very far apart cannot be interpreted as a flat spot.

The third convergence criterion requires that the ratio of the best and worst gap values produced by the any of the previously tested lambda values is less than a specified **assignment-worst-gap-ratio**, which is typically set to 0.1. This condition suggests that if sufficient improvement has been found in the gap value then the search procedure can be terminated.

2.4.5 Algorithm Convergence Issues

In general, the search procedure assumes that a path's cost increases with the number of vehicles loaded on that path; however, as previously explained, due to path interactions, the cost function may not behave this way in reality. As such, the convergence of the algorithm cannot be proven, and the algorithm must thus be considered heuristic; however, in practice preliminary results show reasonably close approximations to equilibrium. Detailed examinations of the behavior of the cost function with demand, as well as factors that affect convergence rate in practice are left to future research. In addition, the applicability of global optimization algorithms to overcome these difficulties may be explored.

It should be noted that while the rate of convergence is not guaranteed, it is certain that the algorithm will not diverge. This is true because should it be found that the solution cannot be improved in a particular direction, then the step length will be set to λ=0, such that $\Xi_{n+1} = \Xi_n$. The gap function is thus limited to $V_{n+1} \leq V_n$. This holds true for the gap functions of all three phases.

With three different phases defined, and different convergence properties associated with each, an important question is that of when to switch from phase to phase. In other words, it remains unclear what values of **assignment-switch-threshold** and **assignment-switch-ratio** would provide the most efficient convergence, and how those values should be selected. Similarly, it remains unclear how the convergence of the line search (**assignment-stop-difference**, **assignment-previous-gap-ratio**, **assignment-worst-gap-ratio**) affects the progress of the following iterations. These issues require further attention.

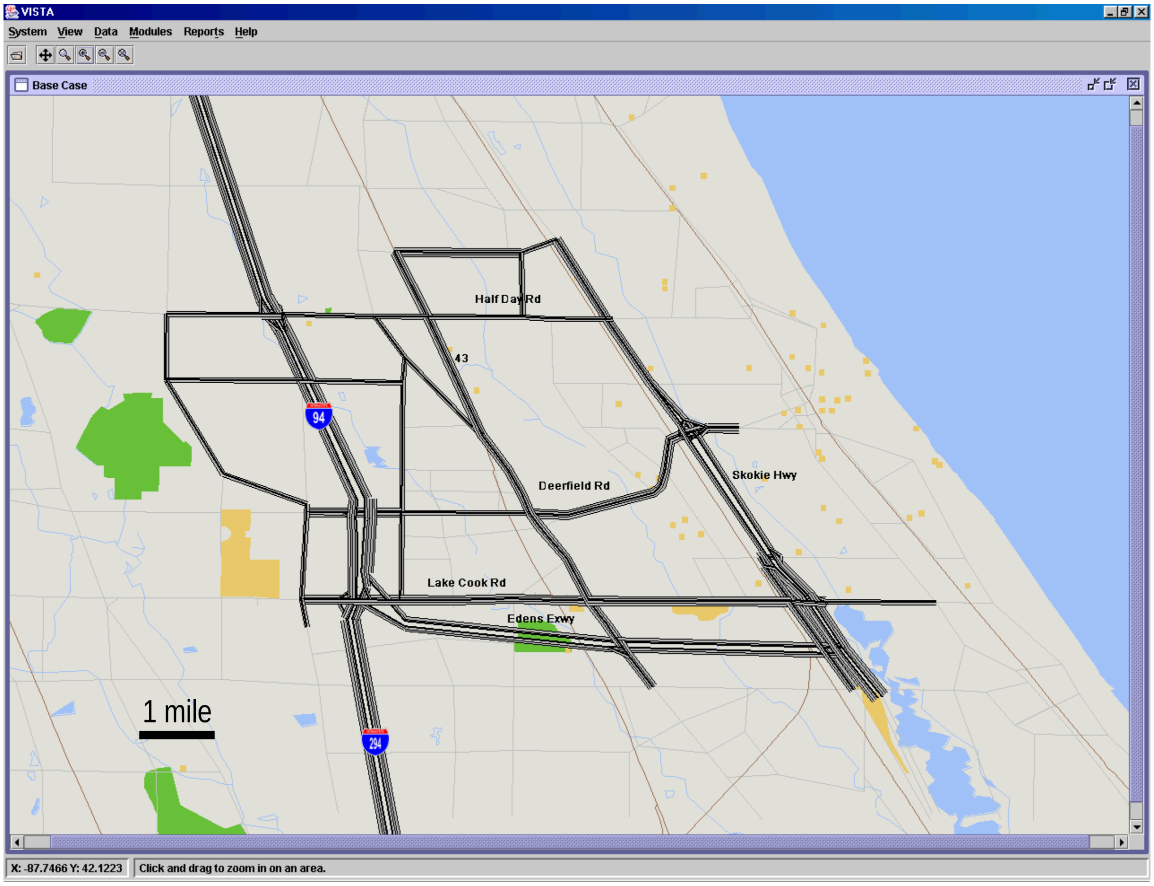
2.5 Computational Results

The automobile assignment-based multimodal DTA model with IADUE assignment was implemented in the VISTA framework, and computational tests were performed on two networks of different sizes. The first network is relatively small network, created as an extraction from the large Chicago network. The second network is the full regional Chicago network used for the TSP study.

The focus of this section is on the behavior of the DTA-IADUE algorithm, and does not attempt to analyse the traffic conditions predicted by the model. More specifically, this section presents a detailed examination of the smaller test network to explore the convergence properties of the DTA-IADUE algorithm with limits imposed on different phases of the algorithm. These limits are similar to those that will necessarily be imposed when running the algorithm on large-scale regional networks, so the conclusions drawn from tests on the smaller network will be used to guide implementation of the algorithm on larger networks. Further, in this section, discussion of the regional Chicago network is limited to the topic of computational requirements of the DTA-IADUE algorithm when implemented on a large-scale network. Evaluation and analysis of the model results will be available in the final report for the TSP study.

2.5.1 Test Network 1 – I-94 and Lake Cook Rd

The first test network represents a small area around the junction of I-94 and Lake-Cook Road north of Chicago (see Figure 2-9). The network includes 82 nodes, 169 links and 24 zones. For the test, 32,337 automobile trips between 199 origin-destination pairs were loaded in the first 1.5 hours of the 2-hour simulation period. The cell transmission-based simulation is discretized into 6-second time steps. The tests were run on this network were performed to observe the behavior of the DTA-IADUE algorithm, rather than the multimodal aspect of the model. As such, no buses were included in this network. Test runs on this network were performed on dual 2GHz Athlon servers with 4GB RAM.



X: -87.7466 Y: 42.1223

Click and drag to zoom in on an area.

Figure 2-9: Test network 1 – I-94 and Lake-Cook Road

2.5.1.1 Test Run A

The first test, Test Run A, included all three phases, with each phase was permitted to run until it was unable to improve its solution. More specifically, Phase 1 was permitted to generate paths until the number of new paths generated was negligible (less than 1% of the path set, in this case). Following the path generation iterations, Phase 1 was permitted to generate extreme points, until it was unable to reduce the demand gap. This state was reflected by a selection of $\lambda=0$ during the line search. Similarly, Phase 2 was terminated when the Smith search yielded a step size $\lambda=0$, indicating that the Smith direction would not longer reduce the Smith gap. Further, while the algorithm was set to terminate when the average cost gap per vehicle was less than 1% of the average trip travel time, the algorithm actually terminated before that condition was satisfied, by selecting $\lambda=0$ in Phase 3. This indicated that the cost gap search was no longer able to improve the solution.

Figure 2-10 shows the value of the cost gap and average cost gap per trip with each iteration through all phases of the algorithm. As expected, the cost gap fluctuated throughout Phase 1. This occurred because the addition of new lower-cost paths and extreme points resulted in increases in the demand and cost gap values. The plot shows that, despite the fluctuations, the extreme direction search of Phase 1 brought the cost gap down to about 2.5%. Phases 2 and Phase 3 reduced the cost gap down to the final value of 1.1%, or approximately 5.5 seconds per trip.

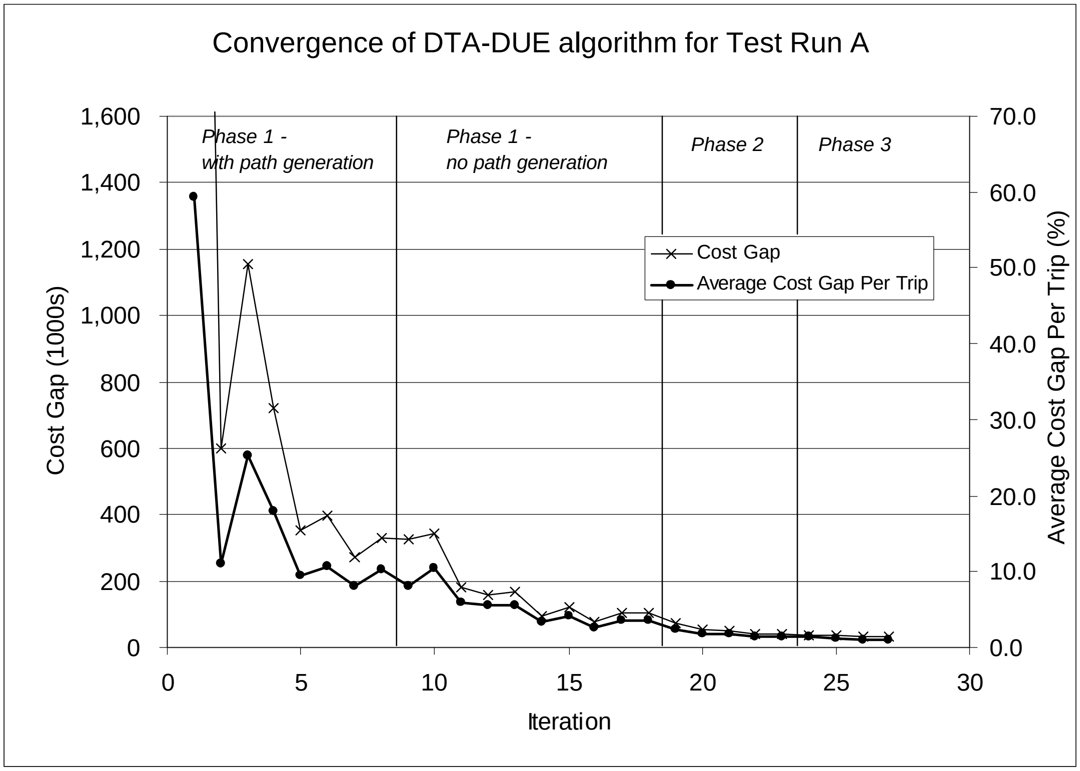


Figure 2-10: Convergence of the Cost Gap and Cost Gap per Trip for Test Run A

The total computational time required for Test Run A was 00:53:04. Table 2-1 shows a breakdown of the number of iterations, the number of simulations required, and the computational time required for each phase. In each phase, each iteration includes a line search during which several step lengths are tested. For each step length test, a simulation run is performed. With the line search parameters set to the values shown in Table 2-2, each line search required 5-8 simulations, with each simulation taking an average of 11 seconds. Simulations accounted for the majority of the computational time expended, so future tests would benefit from a clearer understanding of how the line search parameters affect the number of simulations required and the convergence rate of the overall algorithm. More specifically, if the line search convergence criteria are set loosely, then fewer simulations are required for each line search, but more overall assignment search iterations may be required. A clear understanding of these tradeoffs may save significant computational time.

Table 2-1: Computational Time for Test Run A

Phase	Number of Iterations	Number of Simulations	Computational Time

1 – path generation	8	47	00:13:51
1 – no path generation	10	58	00:15:10
2	5	39	00:12:25
3	4	33	00:11:38
Total	27	177	00:53:04

Table 2-2: Line Search Parameters

Parameter	Value
Assignment-search-low	0.38
Assignment-search-high	0.62
Assignment-stop-difference	0.05
Assignment-previous-gap-ratio	0.99
Assignment-worst-gap-ratio	0.05

Outside of simulation, the remainder of the computational time is devoted to updating of link and path travel times following each simulation, calculation of assignment patterns and calculation of time dependent shortest paths. Typically, calculation of time dependent shortest paths is also a relatively time-consuming task, especially in larger networks; however, in this test, each path generation required on average only 2 seconds.

In Phase 1, the routing-stop-percentage was set to 1%, such that when the number of new paths generated was less than 1% of the total number of paths in the path set, the time dependent shortest path calculation was no longer invoked for subsequent iterations. Figure 2-11 shows the growth of the path set through the eight iterations of Phase 1 during which paths were generated. The plot shows that many paths were added in the early iterations, but that the growth of the path set leveled off in later iterations to a final path set size of 659 for 199 OD pairs. Therefore it seems reasonable to omit the path generation step in later iterations, since the path generation can be computationally time-consuming for realistic large-scale networks.

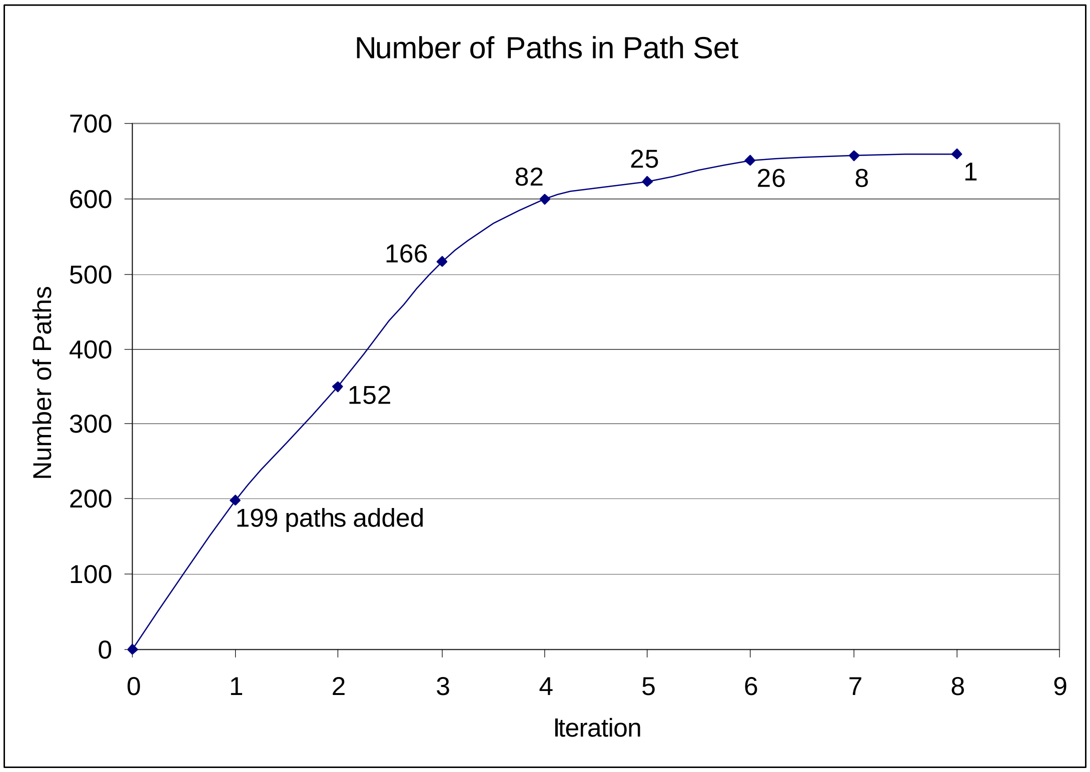


Figure 2-11: Growth of the Path Set

2.5.1.2 Test Runs B and C

Test Runs B and C were performed to examine the effect of omitting Phase 2, as well as omitting the end of Phase 1 (following path generation) along with Phase 2. More specifically, Test Run B was performed with a full extreme search, followed by the cost gap search of Phase 3. Test Run C was performed such that on the path generation portion of Phase 1 was performed, and then Phase 3 was invoked immediately following that. These tests were performed to examine the convergence properties of the search algorithm under implementation variations that might be made to accommodate computational time constraints. In these tests, the line search parameters were as shown in Table 2-2.

The change in the average cost gap per trip throughout the DTA-IADUE procedure is shown in Figure 2-12 for Test Runs A, B and C. The plot shows that Test Run B, without the Smith search, followed almost the exact same convergence pattern as Test Run A, but stopped two iterations earlier. While Test Run B terminated earlier, it stopped with an average cost gap per trip of 1.3%, which was not as close to equilibrium as Test Run A’s 1.1%. As with Test Run B, Test Run C required only 25 iterations to terminate; however, it achieved an average cost gap per trip of 2.1%, which was much higher than Test Runs A and B. These results are summarized in Table 2-3.

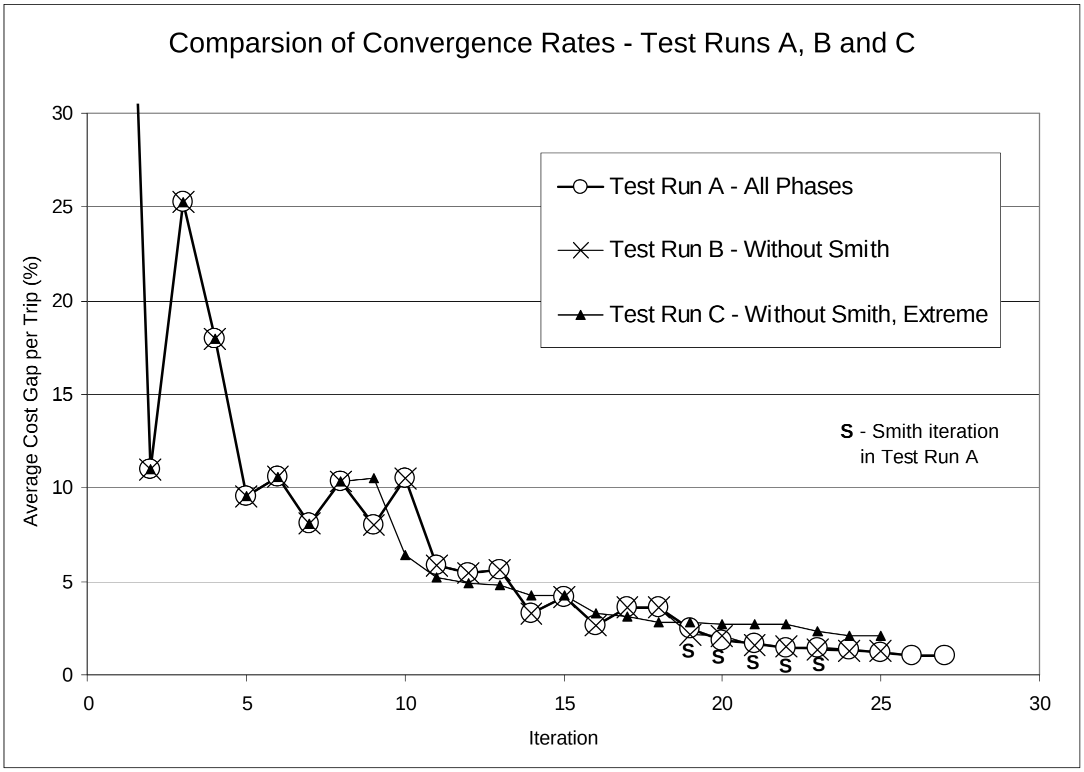


Figure 2-12: Convergence of Test Runs A, B and C

Table 2-3: Comparison of Results for Test Runs A, B and C

Test Run	Number of Iterations	Average Cost Gap per Trip (%)	Average Cost Gap per Trip (seconds)
A	27	1.1	5.5
B	25	1.3	6.7
C	25	2.1	10.8

The results indicate that use of the Smith search (Phase 2) did indeed result in a better solution; however, a larger number of iterations were needed. As such, this suggests a tradeoff exists between the amount of computational time required and the quality of the solution. In this case, since the difference between the average cost gaps of Test Runs A and B is effectively only about 1 second per vehicle, the difference in the quality of the solution may be considered negligible, since travelers may not be sensitive to such a small difference.

Test Run C produced a solution that was farther from equilibrium than the first two tests, without providing any savings in computational time. This suggests that the use of the extreme and Smith searches in the early stages of the search may indeed provide improved solutions. On the other hand, since each test took a different search path toward its final solution, it may simply be a coincidence that Test Run C finished at a solution (a local minimum of the cost gap function) that was less ideal than the solutions found by Test Runs A and B. Tests on different networks and under different demand conditions would be required before a conclusive statement can be made about the effectiveness of the extreme and Smith search phases; however, these initial tests suggest that these may indeed result in better solutions.

As for Test Run A, Test Runs B and C were performed with a routing-stop-percentage of 1%, and the path generation phase yielded the same results as shown in Figure 2-11, with 659 paths generated in 12 iterations.

2.5.1.3 Test Runs D and E

Test Runs D and E were performed to examine the effect of limiting the number of extreme points generated. Each iteration of Phase 1, both with and without path generation, generates a new extreme point, which may be used in the Smith direction search of Phase 2. Ideally, the number of extreme points generated would be selected based on the convergence acheived by Phase 1, or number required for efficient search in Phase 2; however, it remains unclear what gap value may be reasonably expected at termination of Phase 1, as well as how the number of extreme points affects the convergence rate of Phase 2. As such, the selection of the number of extreme points to be generated may be driven by computational resources.

For Test Network 1, it was found that only 18 extreme points were generated before the extreme search was no longer able to reduce the demand gap; however, for larger, more realistic networks, time constraints would make it unreasonable to perform so many iterations of Phase 1. Further, the larger number of OD pairs in a large-scale network would likely require more even more than 18 iterations of Phase 1 before the extreme search was no longer able to reduce the demand gap. As such, to address the practical issues that would be encountered using the DTA-IADUE algorithm, the number of extreme points generated for Test Runs D and E was limited to 10, where eight were created with path generation, and 2 were created as all-or-nothing assignments to different combinations of the paths generated. Test Run D included all phases of the algorithm and Test Run E omitted Phase 2. In these tests, the line search parameters were as shown in Table 2-2.

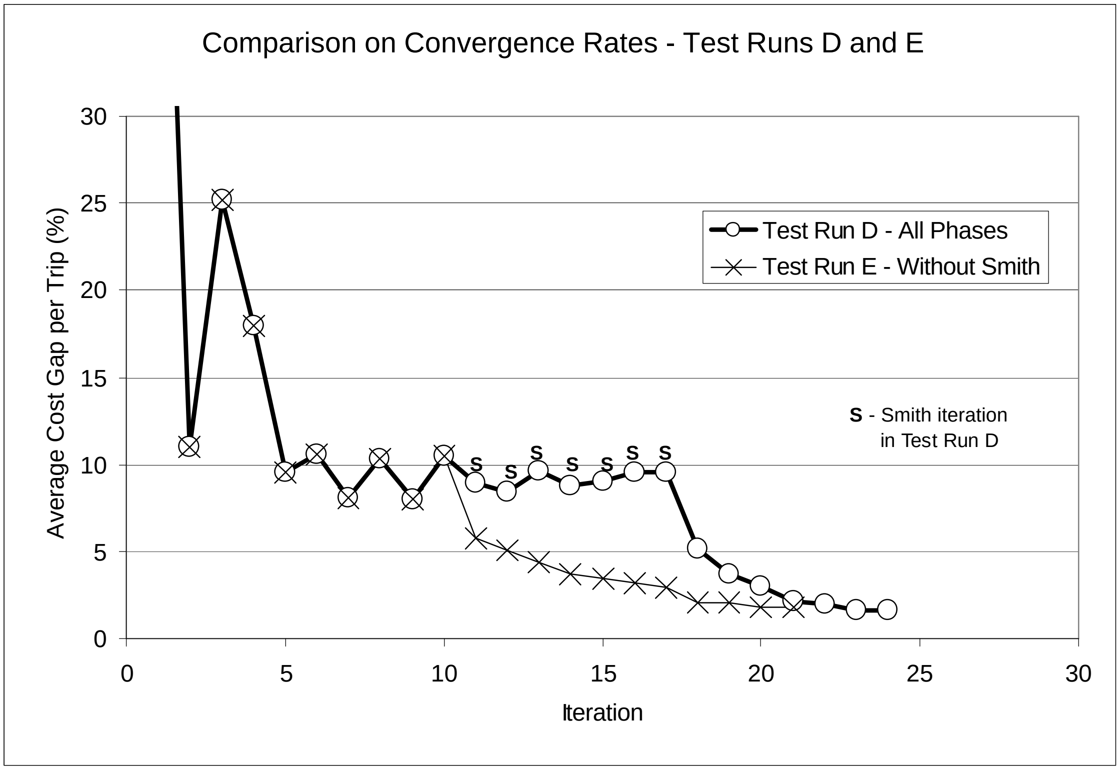


Figure 2-13: Convergence of Test Runs D and E

As with Test Run A, Test Runs D and E were performed with a routing-stop-percentage of 1%, and the path generation phase yielded the same results as shown in Figure 2-10, with 659 paths generated in 12 iterations.

The change in the average cost gap per trip throughout the DTA-IADUE procedure is shown in Figure 2-13 for Test Runs D and E. The plot shows that Test Run E, without the Smith search, approached equilibrium much more quickly than did Test Run D. Test Run E terminated with an average cost gap of 1.8% or about 9.4 seconds per trip when it was found that Phase 3 could no longer reduce the cost gap. Test Run D terminated with a slightly lower average cost gap of 1.7% or about 8.7 seconds per trip when it was found that Phase 3 could no longer reduce the cost gap.

Comparison of Test Runs D and E show that use of the Phase 2 Smith search increased the number of iterations required before termination. Moreover, while the final solution achieved by Test Run D was slightly better than of Test Run E, the Figure 2-13 shows that the cost gap in fact increased during Smith search, despite the fact that the Smith gap decreased from iteration to iteration. This suggests, that the Smith search may not be effective when the number extreme points available is limited. As before, however, these convergence patterns may simply be a result of the search paths chosen and the resulting local minima found by the different variations of the algorithm, and conclusive statements regarding the behavior of the algorithm will require further testing on different networks under different demand conditions.

2.5.1.4 Test Runs F and G

In realistic large-scale networks, the path generation procedure can be extremely time-consuming, so it may be necessary to limit the number of path generation iterations. Test Runs F and G examine the effect of limiting the number of path generation iterations on the convergence behavior of the algorithm. More specifically, whereas the path generation iterations naturally leveled of and terminated after eight iterations for this network, in Test Runs F and G, only four path generation iterations were performed. The growth of the path in those four path generation iterations is shown in Figure 2-14, and is the same as that depicted in the first four iterations shown in Figure 2-11. Figure 2-14 shows that the path generation iterations were stopped with 599 paths between 199 OD pairs. The full set of paths used in Test Runs A-E included 659 paths.

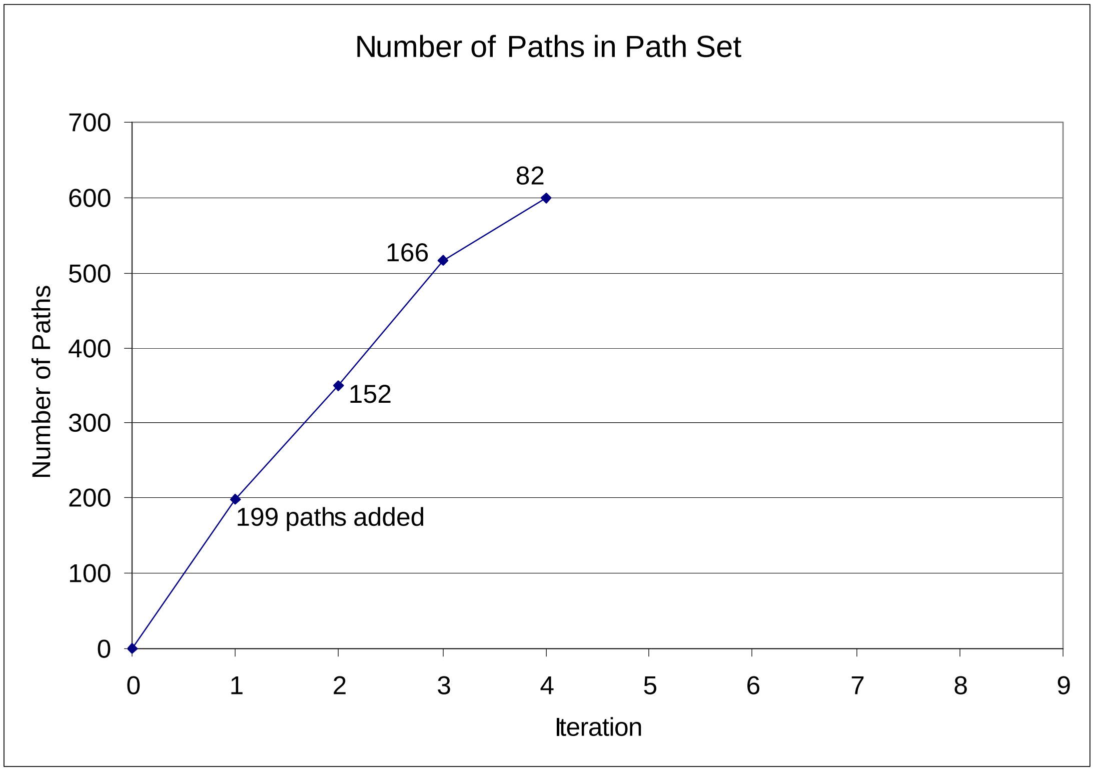


Figure 2-14: Growth of the Path Set (Limited Path Set)

In addition, the number of extreme point generation iterations was also limited. Specifically, Phase 1 was terminated after a total of 11 iterations, including the 4 path generation iterations. This further simulates practical constraints on computational resources.

The change in the average cost gap per trip throughout the DTA-IADUE procedure is shown in Figure 2-15 for Test Runs F and G. The plot shows that Test Run G, without the Smith search, approached equilibrium much more quickly than did Test Run F. Test Run G terminated with an average cost gap of 2.0% or about 10.8 seconds per trip when it was found that Phase 3 could no longer reduce the cost gap. Test Run F terminated with a slightly lower average cost gap of 2.5% or about 13.2 seconds per trip when it was found that Phase 3 could no longer reduce the cost gap.

Comparison of Test Runs F and G show that use of the Phase 2 Smith search increased the number of iterations required before termination. Moreover, the final solution achieved by Test Run G, without the Smith search, was better than of Test Run F, despite the fact that the Smith gap decreased from iteration to iteration. This suggests that the Smith search may not be effective when the number paths and extreme points available is limited. However, these convergence patterns may simply be a result of the search paths chosen and the resulting local minima of the cost gap found by the different variations of the algorithm, and conclusive statements regarding the behavior of the algorithm will require further testing on different networks under different demand conditions.

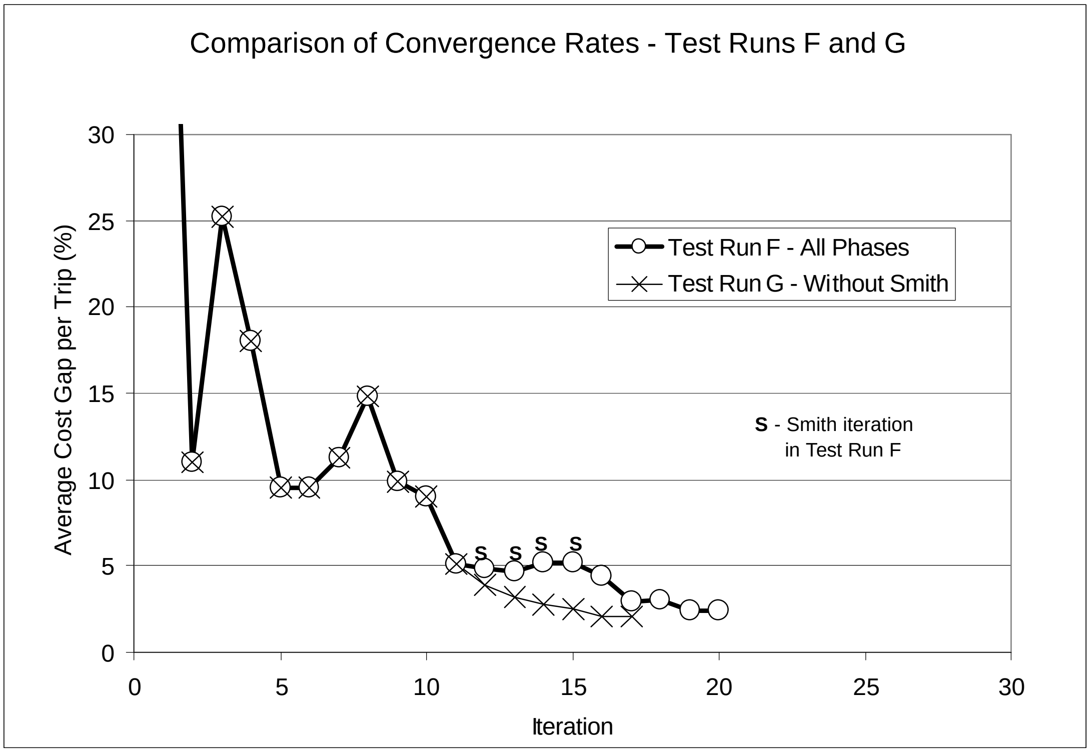


Figure 2-15: Convergence of Test Runs F and G

2.5.1.5 Test Run H

Test Run H was similar to Test Run C in that the Phase 3 search was invoked immediately after the path generation iterations, such that the extreme direction search of Phase 1 following path generation and the Smith search of Phase 2 were omitted. In contrast to Test Run C, however, the number of path generation iterations permitted in Test Run H was limited. Specifically, in Test Run H, only 4 iterations of path generation were permitted, and the path set of 599 paths was thus generated as shown in Figure 2-14.

This test was performed to explore the convergence properties of the algorithm when a limited path set is supplied without and no extreme points are available to be supplied to Phase 2. In this situation, the only option is to begin the search in Phase 3, since this phase does not require extreme points. This situation might occur when the path set is generated using an MSA approach, which is a more efficient means of generating paths. More specifically, whereas the Phase 1 path generation stage of the IADUE algorithm requires 5-8 simulations between each path generation, the MSA algorithm requires only 1. The MSA approach does not generate paths based on the traffic conditions closest to equilibrium, as does IADUE; however, practical constraints on computational time may require the use of MSA for path generation, such that concerns with equilibration are left to Phase 3 of the IADUE algorithm.

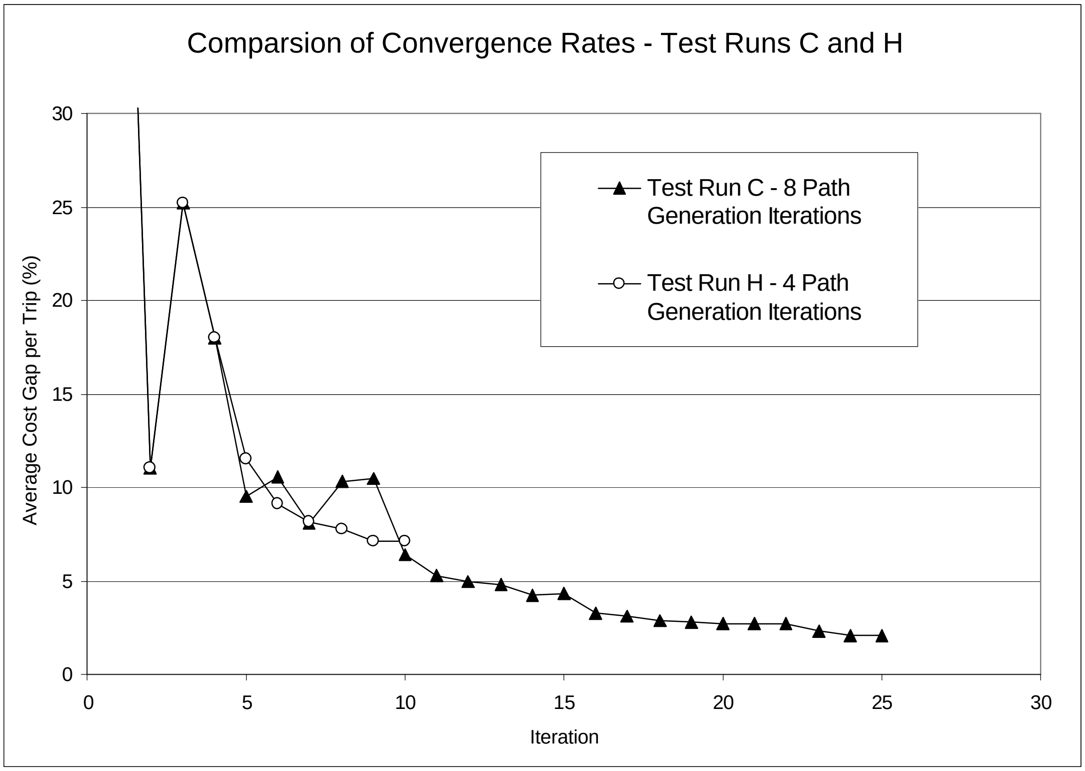


Figure 2-16: Convergence of Test Runs C and H

The convergence of Test Run H is shown in Figure 2-16, with a plot of the convergence of Test Run C for comparison. Test Run H required far fewer iterations before reaching a solution that could not be further improved using the cost gap search. Specifically, only 10 iterations were required compared with 25 for Test Run C; however, the final average cost gap was much higher at 7.1%. This high cost gap is likely a result of a shortage of path alternatives provided to Phase 3. However, these convergence patterns may also simply be a result of the search paths chosen and the resulting local minima of the cost gap found by the different variations of the algorithm, and conclusive statements regarding the behavior of the algorithm will require further testing on different networks under different demand conditions.

2.5.1.6 Test Network 1 - Summary

In most of the tests, the average cost gaps per trip were reduced to values of 1-3%, but never satisfied the 1% termination criterion. Instead, the algorithm stopped when the cost gaps could not be further improved in the descent direction defined for Phase 3. Several tests were performed using variations of the DTA-IADUE algorithm, to approximate the computational constraints that might affect application of the algorithm to a large-scale network. A summary of the tests performed, with the number of iterations performed in each phase, is shown in Table 2-4. Further, the total number of iterations and the average cost gap of the final solution are plotted for each test run in Figure 2-17.

Table 2-4: Summary of Test Runs Performed

Test Run	Phase 1 – path generation iterations	Phase 1 – no path generation iterations	Phase 2 – Smith search iterations	Phase 3 – cost gap search iterations
A	8	10	5	4
B	8	10	--*	7
C	8	--*	--*	17
D	8	2*	7	7
E	8	2*	--*	11
F	4*	7*	4	5
G	4*	7*	--	6
H	4*	--*	--*	6

* A limit was imposed on the number of iterations permitted at the given phase.

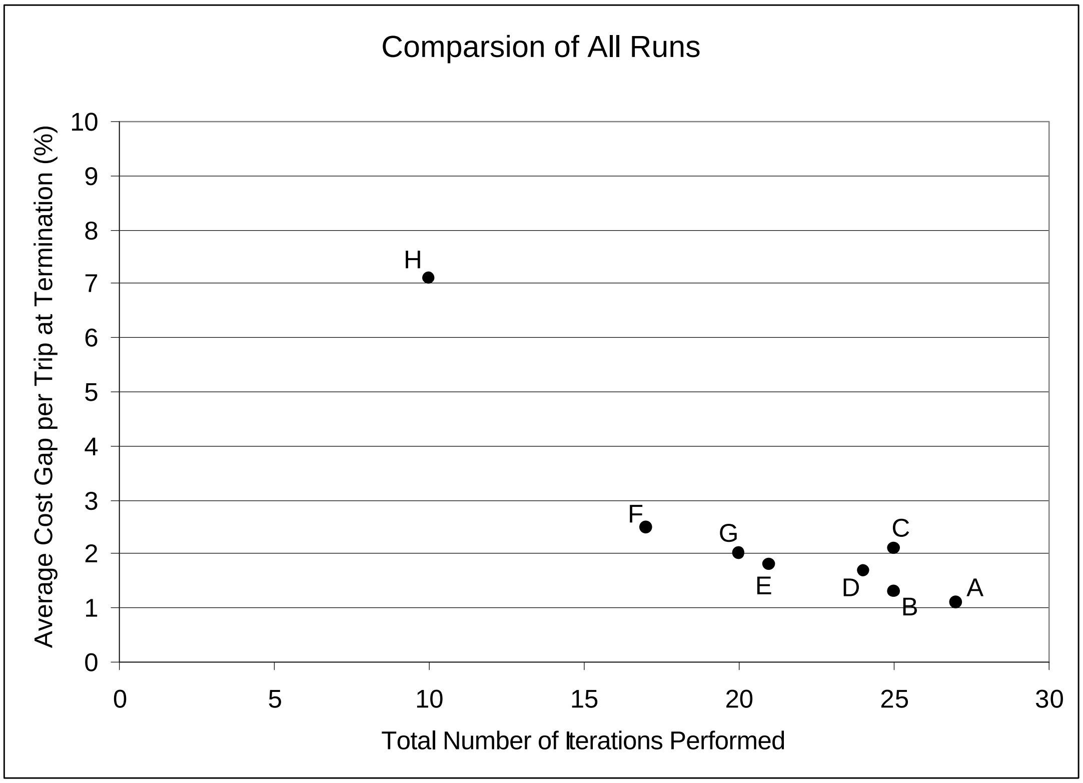


Figure 2-17: Comparison of Iterations Required and Solutions Achieved in each Test Run

It was found that the Phase 2 Smith search does not perform well when an insufficient number of paths and extreme points have been created. With limited paths and extreme points, omission of Phase 2 reduces the number of iterations required, and results in final solutions that are comparable to or better than the solutions generated with the full algorithm. Further, as expected, limiting the number of paths and extreme points both resulted in cost gaps that were higher than for the case in which path and extreme point generation were not limited.

In addition, the highest average cost gap of 7% per trip resulted from limiting the number of path generation iterations, omitting the remainder of the extreme direction search iterations, as well as the Smith search iterations. In other words, a limited path set was supplied directly to the cost gap search. This variation of the algorithm terminated in only 11 iterations, which was much faster than the other variations of the algorithm; however, the final solution was farther from equilibrium.

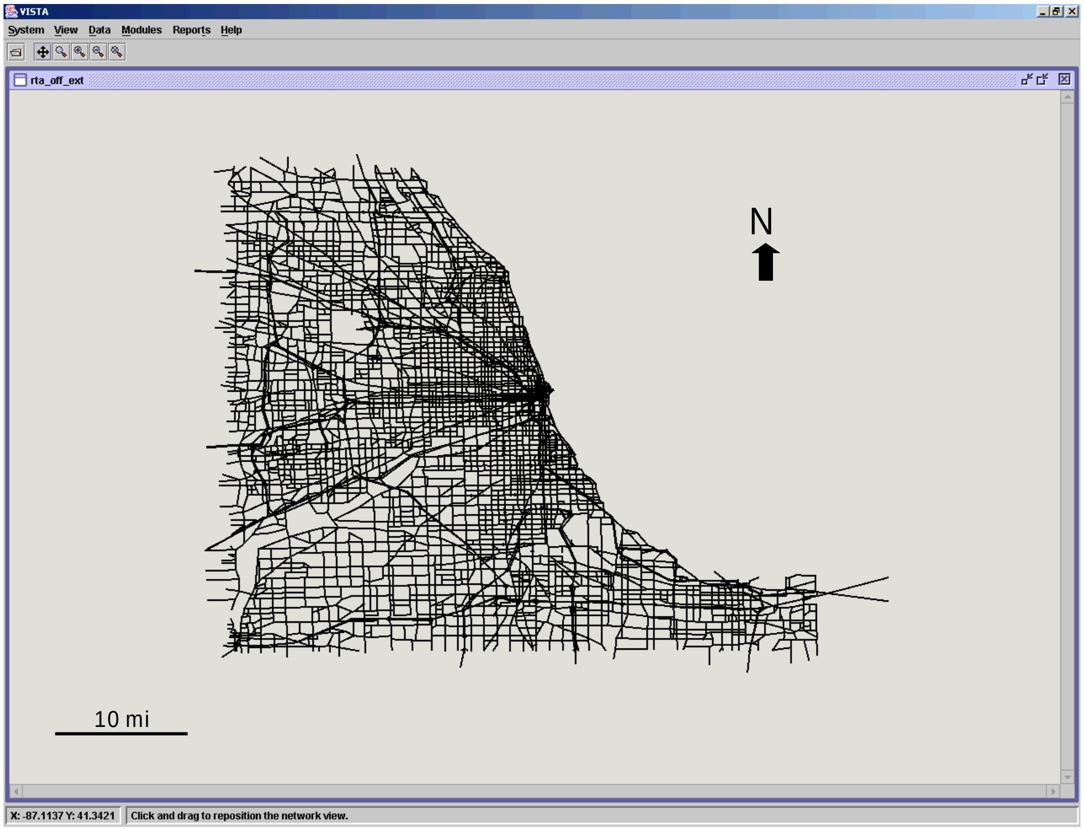
2.5.2 Test Network 2 – Chicago Regional Network

The automobile assignment-based multimodal DTA model described in this section is being used to evaluate the regional impacts of TSP in Chicago. The multimodal model captures stopping and dwelling behavior of buses, as well as the effect of that behavior on the general traffic stream. Further, the DTA-IADUE algorithm will capture travelers’ route switching behavior resulting from even subtle TSP-induced travel time changes. This section presents computational requirements of the DTA-IADUE algorithm for application to the large-scale, regional network. Next, validation results for the AM peak and off-peak base cases are discussed, and preliminary findings regarding the regional impacts of TSP are presented. A full report on the analysis and evaluation of TSP impacts will be available in the TSP study’s Final Report to be submitted to the RTA.

2.5.2.1 Computational Time and Convergence

The Chicago regional network is shown in Figure 2-18. The network, obtained from the Chicago Area Transportation Study (CATS), includes 10,081 nodes and 28,363 links, and stretches 45 miles along the southern border and 48 miles along the westsern border. Further, the region is divided into 1077 zones, with an origin-destination demand matrix of average daily automobile trips for the year 2002 obtained from CATS. The daily averages were factored down to an AM peak demand of 1,037,976 automobile trips and off-peak demand of 830,825, loaded in the first 2 hours of the 4-hour simulation periods. In addition, 5,739 buses were loaded on 125 routes operated by the Chicago Transit Authority (CTA) and 161 Pace routes in the AM peak’s 2-hour vehicle loading period. For the off-peak, 5,207 buses were loaded in the 2-hour loading period. The network includes 11,716 bus stops on the CTA routes and 4,539 bus stops on the Pace routes. Data sets describing the bus network were obtained from CATS and CTA. Further, the network includes 4,764 signals where signal locations were determined based on a signal inventory compiled by CATS and signal timing plans were optimized based on simulation results. A detailed description of the data sets used to create the model is included in the document, “Interim Report: Data for Model Input, Calibration and Validation” (7).

The large size of the network resulted in long computational times, with dual 2GHz Athlon servers with 4GB RAM requiring an average 30 minutes for calculation of test assignments, 55 minutes for each simulation, and 82 minutes for each the path generation. As such, it was estimated that each iteration of IADUE’s Phase 1 procedure could take up to 12 hours, since typically each test point in the iteration’s line search procedure requires an assingment caluculation and a simluation. To reduce this computational time, paths were generated using MSA instead of IADUE, so that only one simulation was performed for each path generation iteration. Using the off-peak demand set, ten iterations of MSA were completed in 35.4 hours. Next, the solution and path set obtained from MSA were used in the IADUE algorithm continuing from where on path generation is typically terminated.



X: -87.1137 Y: 41.3421

Click and drag to reposition the network view.

Figure 2-18: Test Network 2 – Chicago Regional Network

Beginning with the assignment obtained from MSA, Phase 1 with no path generation was performed. For this assignment, the algorithm found that the demand gap was 0.00; however, the cost gap remained quite high at an average of 18.2% per trip. The demand gap value of 0.00 suggests that an equilibrium assignment has been found, but the cost gap indicates that the solution may still be improved. The apparent inconsistency occurs because the demand gap is calculated as a system cost, so an inefficiency in one traveler's aassignment may be compensated by an improvement in another traveler's assignment. In contrast, the cost gap considers travelers individually, without allowing for one traveler's inefficiency to be compensated by an improvement in another traveler's route choice. Since both Phases 1 and 2 of the IADUE algorithm attempt only to mnimize the demand gap, the stopping criteria for both phases were satisfied by the demand gap of 0.00, the algorithm continued with Phase 3 to minimize the cost gap. In ten iterations of Phase 3, the cost gap was reduced from an average of 18.2% to 4.9% per trip for the off-peak demand set, where the progress of the algorithm is shown in Figure 2-19. The ten iterations of Phase 3 included 51 simulations and required a total of 81.6 hours.

Similarly, for estimation of the AM peak assignment the pattern assignment obtained from MSA was used as a starting point for the IADUE algorithm. Since the MSA solution was developed using the off-peak demand, the path assignments for each origin-destination-departure time were factored up to match the AM peak demand. In twelve iterations of Phase 3, the cost gap was reduced from an average of 24.2% to 6.4 % per trip, with the progress of the algorithm is shown in Figure 2-19. The twelve iterations of Phase 3 included 66 simulations and required a total of 338.3 hours. In this case, computational speed was slowed severely due to multiple tests being run on the same server.

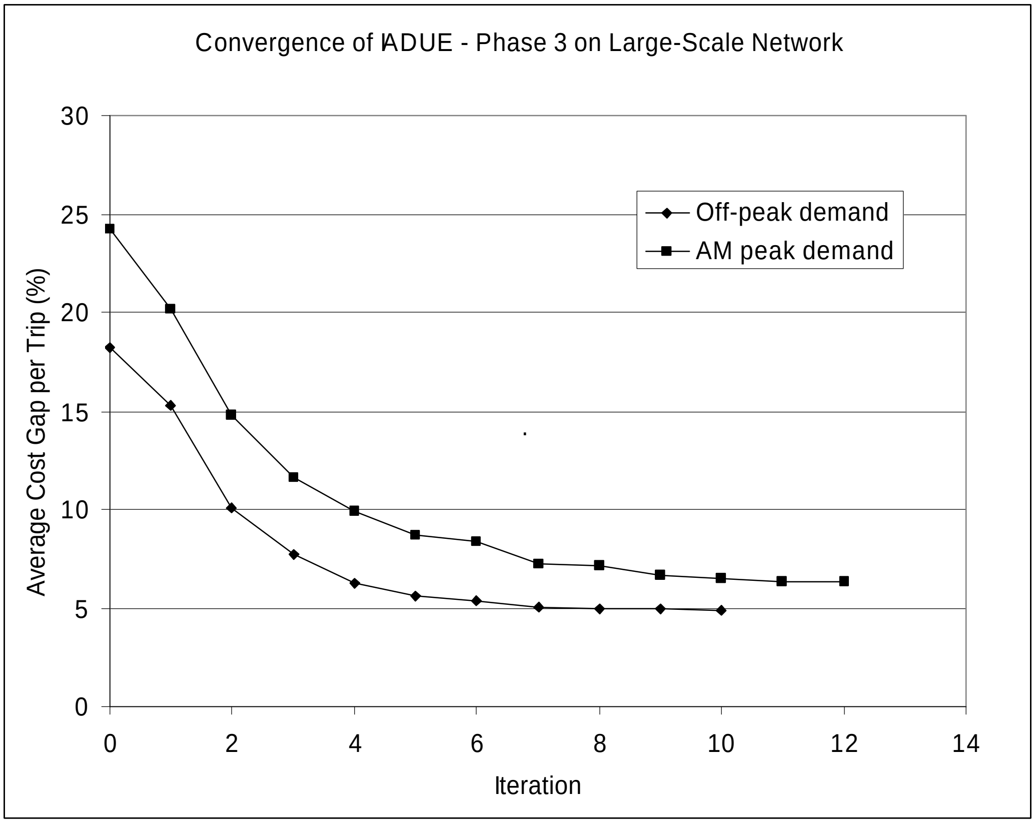


Figure 2-19: Convergence of IADUE - Phase 3 on Large-Scale Network of Chicago with Off-peak and AM peak Base Scenarios

In short, assignment patterns for the large-scale network were estimated such that average cost gaps of 6.4% and 4.9% per trip were achieved for the AM and off-peak base cases and under 10% for the TSP test scenarios. Lengthy computational time requirements resulted in practical modifications of the IADUE algorithm, such as use of MSA in place of the IADUE Phase 1 procedure to generate paths. Further, in practice it was impossible to dedicate the server and database solely to TSP test scenarios, and it was found that computational time varied greatly as a result.

2.5.2.2 Validation of the AM and Off-peak Base Cases

The AM and off peak base case models were validated by comparing the simulation results with expressway counts and observed travel times on selected corridors. More specifically, Figures 2-20 and 2-21 compare the simulation link counts to the observed IDOT expressway counts for the AM and off-peak periods, respectively. The plots map observed counts on the x-axis, and simulation counts on the y-axis, such that a perfect match would lie on the diagonal reference line. The figure shows that link counts are in fact distributed both above and below the reference line, with slightly more points below the reference line. A better match might be obtained by calibrating the OD demand table; however, no efficient means of performing this type of calibration was available. As such, we strive only to have link counts relatively evenly distributed around the reference line, and not heavily above or below the reference line. Points that lie along the y-axis represent counts of zero vehicles, and suggest that the counters were likely off or malfunctioning, since it is highly unlikely that any segment of the expressway was empty for a full hour during either the peak or off-peak periods.

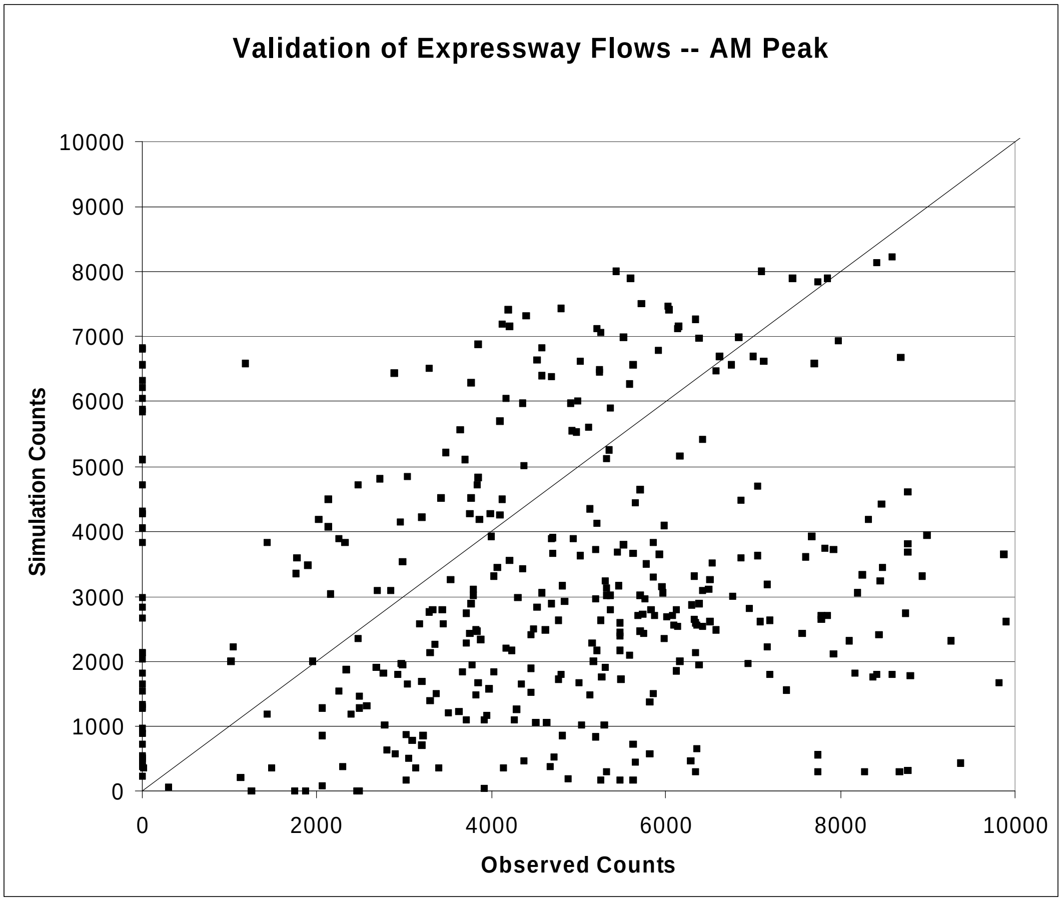


Figure 2-20: Comparison of Simulation Expressway Flows with Observed Counts for the AM Peak Period

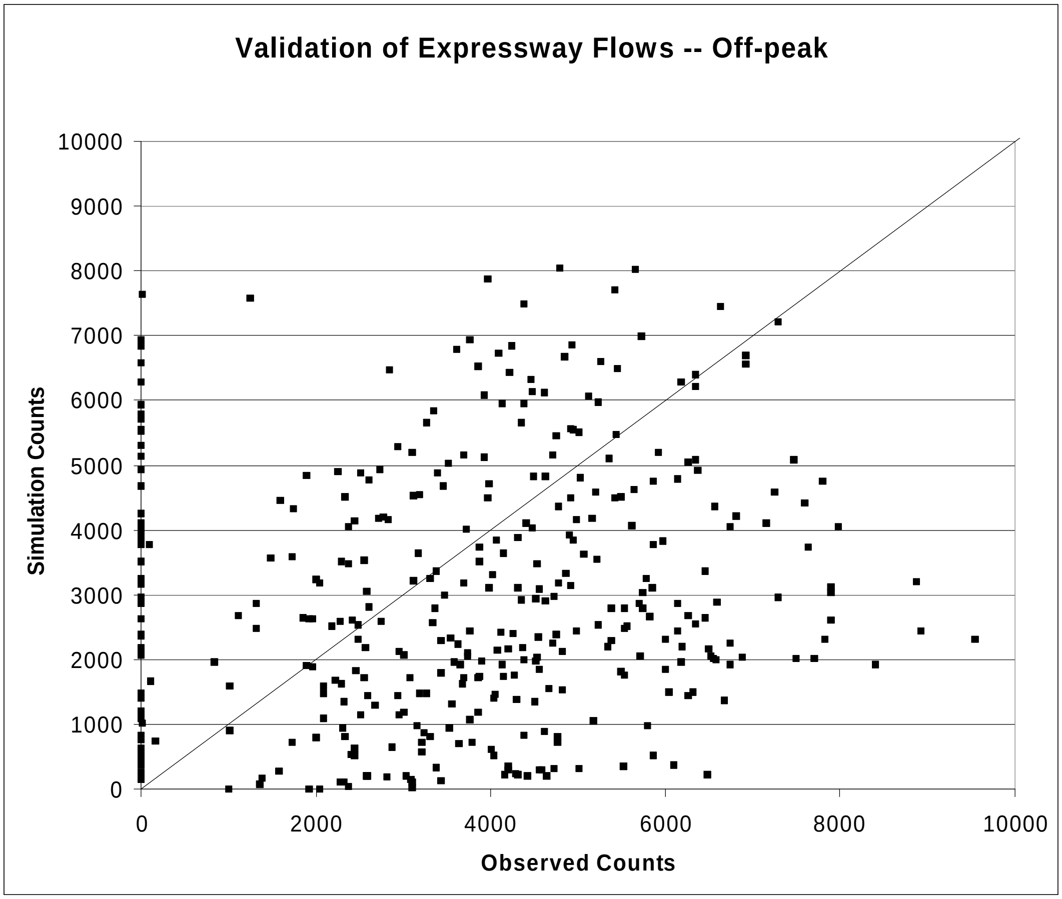


Figure 2-21: Comparison of Simulation Expressway Flows with Observed Counts for the Off-Peak Period

For the AM peak base case, the simulation results were also compared with observed travel times on selected arterial corridors in the network (corridors in the RTA/ITC/C micro-simulation study of TSP). Figure 2-22 shows that the travel times are distributed both above and below the reference line, with a few outliers.

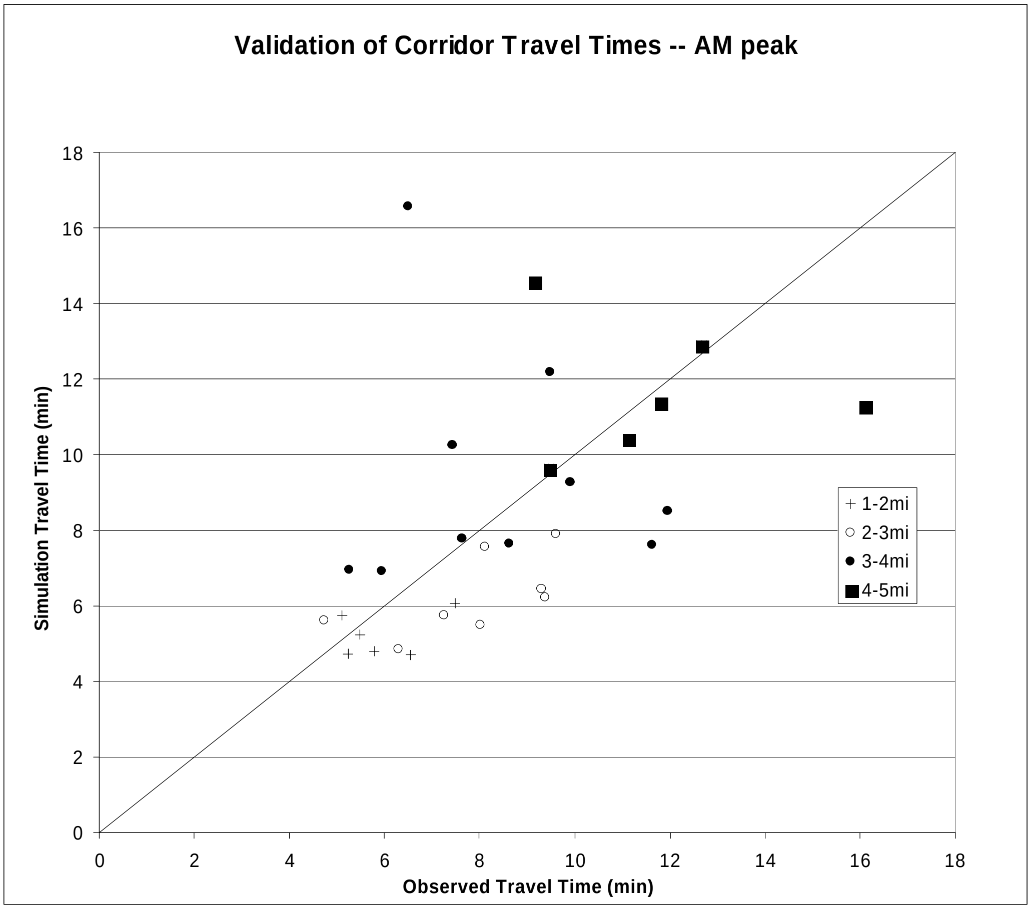


Figure 2-22: Comparison of Simulation Travel Times with Observed Travel Times for the AM Peak Period

The RTA/ITC/C did not collect travel time data for the off-peak period, so the travel time comparison was not performed for the off-peak period.

In short, the validation plots of expressway flows and observed travel times show that, in general, the simulation estimates were distributed both above and below the observed values. In other words, while the estimates did not match the observed values exactly, they were neither systematically low nor systematically high. Ideally, the demand would be calibrated to obtain a better match in flows and travel times; however, no efficient means of performing this type of calibration was available. Further, to the best of the authors' knowledge, there is exists no published experience with large-scale regional DTA analysis, and no standards established in terms of how closely a DTA model's estimates may reasonably be expected to match observed data. As such, the peak and off-peak models are thus taken as a reasonable representation of the Chicago network for the purposes of the TSP study, and the validation results in this section are presented as an initial contribution in the area of validation of large-scale DTA models for real-world analyses.

2.5.2.3 Regional Impacts of TSP

This section presents analyses of impacts of TSP on a single corridor, Western Ave., using both strategies 1 and 2. Next, impacts of multiple TSP corridors are examined. In addition, a test of TSP without OD path changes is analysed to determine the importance of capturing route choice in the evaluation of TSP. A more complete analysis and evaluation of TSP impacts will be available in the TSP study's Final Report to be submitted to the RTA.

2.5.2.3.1 Impacts of Transit Signal Priority Strategy 1 on a Single Corridor

This section discusses the impacts of TSP Strategy 1 implemented on a 27.5-mile length of Western Ave. from Birchwood to Dixie-Sibley in both the AM peak and off peak conditions. Impacts on priority buses, general traffic and non-priority buses are analysed.

It was found that Strategy 1, with a detector location 600ft upstream of the intersection, resulted in clear improvements in the bus travel times of routes that used the priority corridor. Tables 2-5 and 2-6 list the differences in average route travel times between the base case (no TSP) and the case with TSP along Western Avenue for routes that travel along a significant portion of the prioritized segment of Western Ave. Specifically, bus routes that travel along less than 1 mile of the TSP corridor were omitted.

Table 2-5: Bus Route Travel Times (min) with TSP on Western Ave. – AM peak

Route		Route Length (mi)	Nveh	Base No TSP		TSP on Western Strategy 1			
				AvgTT	StdDev	AvgTT	StdDev	ΔTT	%ΔTT
49 Western Ave	SB	34.4	23	59.2	*1.8*	55.5	*2.6*	-3.7	-6.6
49 Western Ave	NB	34.7	20	60.1	*3.5*	55.3	*1.8*	-4.8	-8.6
49A S Western Ave	SB	26.4	6	32.4	*0.5*	26.2	*0.4*	-6.2	-23.8
49A S Western Ave	NB	26.8	6	33.0	*0.0*	28.8	*0.8*	-4.1	-14.3
49B N Western Ave	SB	8.9	19	12.8	*0.8*	10.6	*0.4*	-2.2	-20.4
49B N Western Ave	NB	7.9	19	10.7	*0.7*	9.3	*0.4*	-1.4	-15.3
X49 Western Exp	SB	18.0	12	53.7	*2.5*	49.2	*4.1*	-4.6	-9.3
X49 Western Exp	NB	18.4	12	55.3	*3.1*	51.7	*2.2*	-3.7	-7.1
349 S Western	SB	11.4	7	29.1	*1.5*	24.0	*0.4*	-5.1	-21.3
349 S Western	NB	11.5	7	30.3	*0.5*	26.7	*1.1*	-3.6	-13.5

Table 2-6: Bus Route Travel Times (min) with TSP on Western Ave. – off-peak

Route		Route Length (mi)	Nveh	Base No TSP		TSP on Western Strategy 1			
				AvgTT	StdDev	AvgTT	StdDev	ΔTT	%ΔTT
49 Western Ave	SB	34.4	23	59.0	*1.7*	45.4	*1.3*	-13.6	-29.9
49 Western Ave	NB	34.7	20	58.9	*2.4*	44.9	*1.1*	-13.9	-31.0
49A S Western Ave	SB	26.4	6	32.0	*1.3*	25.6	*0.4*	-6.4	-25.0
49A S Western Ave	NB	26.8	6	32.7	*0.7*	29.4	*1.0*	-3.3	-11.2
49B N Western Ave	SB	8.9	19	13.0	*1.0*	10.4	*0.2*	-2.6	-25.5
49B N Western Ave	NB	7.9	19	11.2	*1.3*	9.1	*0.2*	-2.2	-23.8
X49 Western Exp	SB	18.0	12	54.0	*2.7*	39.7	*1.6*	-14.3	-36.1
X49 Western Exp	NB	18.4	12	53.3	*3.0*	39.5	*1.9*	-13.8	-35.0
349 S Western	SB	11.4	7	29.2	*1.2*	23.7	*0.1*	-5.5	-23.3
349 S Western	NB	11.5	7	30.1	*0.6*	26.6	*0.7*	-3.5	-13.2

From the tables, it is clear that travel time improvements were achieved with TSP Strategy 1, and further that the improvements were generally much greater during the off-peak than the am peak. This suggests that higher levels of congestion and queuing reduce the effectiveness of TSP. One possible reason for this is that higher levels of congestion result in lower bus travel speeds, and thus longer time required for the bus to travel from the detector location to the intersection. In some cases, if the bus is excessively delayed, the maximum green time will elapse (a maximum green time of 100 seconds was used in these tests) before the bus is able to traverse the intersection, and the priority opportunity will have been wasted. Otherwise, even if the bus succeeds in traversing within the maximum green limit, the slower bus travel speed results in a longer holding of the green extension, and may result in a longer recovery periods if minimum green times for non-priority phases are reched. Since new TSP requests are not served during the recovery period, this results in a lower number of TSP requests being served.

The tables also show that the standard deviations of route travel times both increased and decreased in the AM peak test, but generally decreased in the off-peak test. This suggests that TSP may reduce travel time variability in conditions of low congestion; however, during the peak period bus travel times are affected not only by signal delays but also by congestion, and as such, TSP alone may not reduce the travel time variabiity.

While Strategy 1 clearly improves bus travel times along the TSP corridor, its effect on the average travel speed along the corridor is relatively low. Table 5 shows that in the AM peak, travel speeds improved by 0.21 mph in the SB direction and 0.41 mph in the NB direction. Table 6 shows that in the off-peak period, travel speeds improved slightly more by 0.88 mph in the SB direction and 0.63 mph in the NB direction.

In addition, Tables 2-7 and 2-8 show the average travel speeds along major corridors parallel to the priority corridor, Western Avenue. The corridors are listed in order from west to east across the city, and the corridor segments considered are defined in Table 2-9. Table 2-7 and 2-8 show that changes in average speed along parallel corridors are both positive and negative, between –1.96 mph and +1.02 mph in the AM peak, and between –0.18 mph and 1.04 mph in the off-peak period. The table reveals little correlation between the distance of the corridor from the TSP corridor and the resulting change in average corridor speed.

Table 2-7: Average Speeds along TSP Corridor and Parallel Corridors – AM peak

Corridor		Base Case Average Speed (mph)	TSP Strategy 1 on Western Ave Average Speed (mph)	Difference in Speed (mph)
Pulaski	SB	26.44	26.3	-0.10
Pulaski	NB	25.60	25.8	0.17
Kedzie	SB	15.65	16.7	1.02
Kedzie	NB	20.02	18.9	-1.09
California	SB	24.77	24.8	0.06
California	NB	24.34	24.1	-0.21
Western	SB	26.10	26.3	0.21
Western	NB	25.22	25.6	0.41
Damen	SB	23.67	23.6	-0.03
Damen	NB	6.98	7.4	0.45
Ashland	SB	23.01	21.1	-1.96
Ashland	NB	25.27	25.4	0.14
Racine	SB	25.94	25.9	0.00
Racine	NB	25.75	25.7	-0.01
Halsted	SB	25.95	26.0	0.06
Halsted	NB	25.83	25.9	0.06
King	SB	30.53	30.4	-0.12
King	NB	30.98	30.9	-0.05
Cottage Grove	SB	30.28	30.3	-0.02
Cottage Grove	NB	29.58	29.6	0.00

Table 2-8: Average Speeds along TSP Corridor and Parallel Corridors – Off peak

Corridor		Base Case Average Speed (mph)	TSP Strategy 1 on Western Ave Average Speed (mph)	Difference in Speed (mph)
Pulaski	SB	27.05	26.9	-0.17
Pulaski	NB	25.93	26.0	0.04
Kedzie	SB	25.79	25.8	0.01
Kedzie	NB	24.91	25.1	0.18
California	SB	24.64	24.7	0.05
California	NB	24.82	24.9	0.10
Western	SB	26.46	27.3	0.88
Western	NB	25.78	26.4	0.63
Damen	SB	28.19	28.2	0.02
Damen	NB	13.25	14.3	1.04
Ashland	SB	26.80	26.8	0.01
Ashland	NB	26.27	26.4	0.08
Racine	SB	25.89	25.9	-0.01
Racine	NB	25.73	25.8	0.03
Halsted	SB	25.95	26.0	0.04
Halsted	NB	25.84	25.9	0.05
King	SB	30.52	30.5	-0.07
King	NB	31.33	31.3	-0.03
Cottage Grove	SB	29.74	29.8	0.05
Cottage Grove	NB	30.26	30.2	-0.07

Table 2-9: Major N-S Corridors

Time of Day	From	To	Length (mi)
Pulaski	Howard	147th	26.7
Kedzie	Lincoln	115th	27.1
California*	Howard	71st	17.6
Western	Birchwood	Dixie-Sibley	27.5
Damen	Bryn Mawr	87th	17.1
Ashland	Irving Park	95th	16.1
Racine	31st	87th	7.0
Halsted	Waveland	Dixie-Sibley	22.4
MLK	Cermak	95th	9.0
Cottage Grove	35th	95th	7.6

* Includes break from Grand to Fulton. Total length measures only actual roadway length.

It was expected that corridors perpendicular to the TSP corridor would experience slower travel speeds as a result of TSP; however, tests did not show any consistent decrease in travel speed along these corridors. Tables 2-10 and 2-11 show the average travel speeds along major corridors perpendicular to the priority corridor during the AM peak and off-peak periods, respectively. The corridors are listed in order from north to south, and the corridor segments considered are defined in Table 2-12. Tables 2-10 and 2-11 show that changes in average speed along perpendicular corridors are both positive and negative between –2.25 mph and +1.24 mph in the AM peak and between –2.68 mph and +1.59 mph in the off-peak. In other words, just as the tests showed no obvious pattern in the effects of TSP on average corridor speed or parallel corridors, there is similarly little pattern in its effect on the average speeds along perpendicular corridors.

Table 2-10: Average Speeds along TSP Corridor and Parallel Corridors – AM peak

Corridor		Base Case Average Speed (mph)	TSP Strategy 1 on Western Ave Average Speed (mph)	Difference in Speed (mph)
Lawrence	WB	23.89	23.7	-0.19
Lawrence	EB	5.06	5.7	0.61
Montrose	WB	23.09	23.1	0.04
Montrose	EB	23.28	23.2	-0.07
Irving Park	WB	23.25	23.3	0.09
Irving Park	EB	23.23	23.1	-0.10
Addison	WB	24.98	24.9	-0.07
Addison	EB	2.76	2.4	-0.39
Belmont	WB	24.43	24.4	-0.01
Belmont	EB	24.48	24.2	-0.25
Diversey	WB	23.91	24.1	0.15
Diversey	EB	22.30	22.2	-0.14
Fullerton	WB	22.71	22.3	-0.42
Fullerton	EB	11.01	8.8	-2.25
North	WB	22.68	23.0	0.30
North	EB	12.13	12.2	0.05
Division	WB	23.57	23.5	-0.07
Division	EB	22.95	22.8	-0.13
Chicago	WB	22.09	22.1	0.01
Chicago	EB	22.63	22.6	0.02
Archer	WB	25.65	25.7	0.02
Archer	EB	27.05	26.6	-0.49
79th	WB	15.58	15.2	-0.42
79th	EB	27.12	27.1	-0.03
87th	WB	20.49	21.0	0.55
87th	EB	12.61	13.9	1.24
95th	WB	23.79	22.6	-1.18
95th	EB	25.90	25.4	-0.46
103rd	WB	25.33	25.3	-0.01
103rd	EB	25.60	25.6	0.03

Table 2-11: Average Speeds along TSP Corridor and Parallel Corridors – Off peak

Corridor		Base Case Average Speed (mph)	TSP Strategy 1 on Western Ave Average Speed (mph)	Difference in Speed (mph)
Lawrence	WB	23.87	23.7	-0.18
Lawrence	EB	23.08	22.6	-0.48
Montrose	WB	23.27	23.4	0.08
Montrose	EB	23.38	23.3	-0.06
Irving Park	WB	23.88	23.8	-0.11
Irving Park	EB	23.42	23.4	0.02
Addison	WB	25.22	25.2	-0.02
Addison	EB	23.57	23.3	-0.23
Belmont	WB	24.83	24.7	-0.09
Belmont	EB	24.48	24.4	-0.05
Diversey	WB	23.86	24.0	0.10
Diversey	EB	22.82	22.8	0.02
Fullerton	WB	22.84	22.9	0.03
Fullerton	EB	18.80	16.1	-2.68
North	WB	23.17	23.2	0.04
North	EB	18.86	18.7	-0.13
Division	WB	23.72	23.7	0.01
Division	EB	23.06	23.2	0.13
Chicago	WB	22.43	22.5	0.03
Chicago	EB	22.36	22.5	0.16
Archer	WB	26.88	26.9	0.05
Archer	EB	26.01	26.1	0.04
79th	WB	19.37	18.7	-0.68
79th	EB	27.51	27.4	-0.11
87th	WB	25.15	24.4	-0.79
87th	EB	19.94	21.5	1.59
95th	WB	26.27	26.0	-0.22
95th	EB	24.38	23.7	-0.66
103rd	WB	25.34	25.5	0.11
103rd	EB	25.84	25.7	-0.11

Table 2-12: Major E-W Corridors

Time of Day	From	To	Length (mi)
Lawrence	Austin	Lakeshore	6.5
Montrose	Narragansett	Lakeshore	7.5
Irving Park	Cumberland	Lake Shore Drive	9.9
Addison	Cumberland	Lake Shore Drive	9.9
Belmont	Cumberland	Lake Shore Drive	10.1
Diversey	Harlem	Lake Shore Drive	8.8
Fullerton	Narragansett	Lake Shore Drive	7.7
North	Austin	Lake Shore Drive	7.5
Division	Austin	Lake Shore Drive	7.5
Chicago	Austin	Fairbanks	8.0
Archer	1st Ave	Clark	11.9
79th	LaGrange	Lake Shore Drive	15.2
87th	I-294	ake Shore Drive	16.4
95th	LaGrange	Dan Ryan Expressway	11.8
103rd	Central	Michigan	7.2

Further, just as general traffic conditions were not greatly affected by TSP, bus routes on non-priority corridors also experienced conditions that were very similar to base case conditions. For example, Table 2-13 shows that bus travel times on non-priority corridors, with a few exceptions, are generally the same for the TSP case as for the base case.

Table 2-13: Bus Routes on Parallel Corridors – Off peak

Route		Route Length (mi)	Nveh	Base No TSP		TSP on Western Strategy 1		
				AvgTT	StdDev	AvgTT	StdDev	ΔTT
53 Pulaski	SB	23.6	21	40.9	0.9	40.9	0.9	0.0
53 Pulaski	NB	23.7	19	43.5	1.3	43.5	1.3	0.0
53A South Pulaski	SB	30.8	14	35.3	1.3	35.3	1.3	0.0
53A South Pulaski	NB	30.0	14	33.5	1.2	33.5	1.2	0.0
385 87th-111th	SB	26.3	3	69.8	7.4	67.2	5.7	-2.6
385 87th-111th	NB	26.8	3	65.4	1.2	66.1	1.6	0.7
52 Kedzie-California	SB	22.0	17	42.8	1.3	42.8	1.3	0.0
52 Kedzie-California	NB	22.7	15	39.8	1.6	39.8	1.6	0.0
52A South Kedzie	SB	32.0	20	37.2	1.5	37.2	1.5	0.0
94 South California	SB	29.4	16	40.3	2.0	40.1	2.1	-0.2
94 South California	NB	31.1	20	41.2	1.3	41.6	1.2	0.4
Western Ave								
48 South Damen	SB	16.4	10	15.8	1.0	15.8	1.0	0.0
50 Damen	SB	26.1	14	41.2	3.1	41.1	2.8	-0.2
50 Damen	NB	25.8	14	65.0	12.9	62.5	11.1	-2.5
9 Ashland Ave	SB	39.4	18	65.4	2.1	65.4	2.1	0.0
9 Ashland Ave	NB	35.1	18	62.4	2.5	62.4	2.5	0.0
44 Wallace-Racine	SB	29.4	20	32.1	0.7	32.1	0.7	0.0
44 Wallace-Racine	NB	28.7	20	32.2	2.2	32.2	2.2	0.0
8 Halsted St	SB	32.4	21	52.8	1.9	52.8	1.9	0.0
8 Halsted St	NB	33.8	21	53.2	1.9	53.2	1.9	0.0
8A S Halsted St	SB	24.6	14	25.2	1.3	25.2	1.3	0.0
8A S Halsted St	NB	25.0	15	22.5	0.8	22.5	0.8	0.0
108 Halsted/95th	SB	16.7	24	18.7	0.9	18.7	0.9	0.0
108 Halsted/95th	NB	16.7	21	16.1	0.8	16.1	0.8	0.0
352 S Hlstd-Chic Hts	SB	17.2	9	47.5	1.5	47.5	1.5	0.0
352 S Hlstd-Chic Hts	NB	17.2	9	46.6	2.6	46.6	2.6	0.0
359 Robbins-S Kedzie	SB	12.9	6	32.4	1.2	32.4	1.2	0.0
359 Robbins-S Kedzie	NB	12.9	6	32.9	1.3	32.5	1.2	-0.3
3 King Drive	SB	33.7	18	49.5	2.0	49.5	2.0	0.0
3 King Drive	NB	22.1	18	42.3	1.7	42.3	1.7	0.0
4 Cottage Grove	SB	28.6	27	46.4	2.3	46.4	2.3	0.0
4 Cottage Grove	NB	28.4	20	46.7	2.5	46.7	2.5	0.0

In short, tests of Strategy 1 on Western Ave showed that travel times on priority routes improved with TSP, and further that improvements were greater during the off-peak than the AM peak. In addition, bus travel time variability improved during the off-peak, but did not consistently improve for all routes in the AM peak. Moreover, the speed of general traffic along the TSP corridor was higher with TSP, but only slightly. On the other hand, the effects of TSP on parallel and intersecting non-priority corridors were neither consistent nor significant.

2.5.2.3.2 Impacts of Transit Signal Priority Strategy 2 on a Single Corridor

This section discusses the impacts of TSP Strategy 2 implemented on a 27.5-mile length of Western Ave. from Birchwood to Dixie-Sibley in off peak conditions. Impacts on priority buses, general traffic and non-priority buses are analysed.

It was found that Strategy 2, with a detector location 600ft upstream of the intersection, resulted in improvements in the bus travel times of routes that used the priority corridor; however these improvements were generally lower than those obtained using Strategy 1. Table 2-14 lists the average route travel times for the base case (no TSP) and the cases with TSP Strategies 1 and 2 along Western Ave., for routes that travel along at least 1 mile of the prioritized segment of Western Ave. The table shows that travel times obtained with Strategy 2 were generally several minutes slower than those obtained using Strategy 1. The reason for this is likely that priority buses receive a lower amount of total green time using Strategy 2, since Strategy 2 recovers from priority treatment by removing green time from the next occurrence of the priority phase. By contrast, Strategy 1 recovers from priority treatment by removing green time from the cycle's remaining phases, which are non-priority phases. Further, since Strategy 2 requires at least a full cycle to recover, the recovery period lasts longer, resulting in a lower number of priority requests served.

Table 2-14 also shows that standard deviations of bus travel time are generally higher using Strategy 2 than using Strategy 1. Again, this is likely a result of fewer priority requests being served.

Table 2-14: Comparison of Bus Route Travel Times (min) on Western Ave. – off-peak

Route		Route Length (mi)	Nveh	Base No TSP		TSP on Western Strategy 1		TSP on Western Strategy 2	
				AvgTT	StdDev	AvgTT	StdDev	AvgTT	StdDev
49 Western Ave	SB	34.4	23	59.0	*1.7*	45.4	*1.3*	50.9	*2.2*
49 Western Ave	NB	34.7	20	58.9	*2.4*	44.9	*1.1*	49.8	*3.3*
49A S Western Ave	SB	26.4	6	32.0	*1.3*	25.6	*0.4*	26.5	*0.5*
49A S Western Ave	NB	26.8	6	32.7	*0.7*	29.4	*1.0*	29.0	*0.5*
49B N Western Ave	SB	8.9	19	13.0	*1.0*	10.4	*0.2*	11.4	*0.7*
49B N Western Ave	NB	7.9	19	11.2	*1.3*	9.1	*0.2*	10.1	*0.6*
X49 Western Exp	SB	18.0	12	54.0	*2.7*	39.7	*1.6*	43.9	*1.8*
X49 Western Exp	NB	18.4	12	53.3	*3.0*	39.5	*1.9*	44.3	*2.3*
349 S Western	SB	11.4	7	29.2	*1.2*	23.7	*0.1*	24.0	*0.7*
349 S Western	NB	11.5	7	30.1	*0.6*	26.6	*0.7*	26.5	*0.5*

While Strategy 2 improves bus travel times along the TSP corridor, it has no appreciable effect on the average travel speed along the corridor. More specifically, as shown in Table 2-15 use of Strategy 2 resulted in an average increase in corridor speed of 0.1 mph along the priority corridor in the southbound direction, and virtually no change in the northbound direction.

Further, Table 2-15 compares the travel speeds along N-S non-priority corridors for the base case, and TSP cases with Strategies 1 and 2 on Western Ave. The corridors are listed in order from west to east across the city, and the corridor segments considered are defined in Table 7.

The table shows that all of the average corridor speeds are similar, generally within ±0.7 mph of both the base case, except for southbound Damen. However, the table reveals little correlation between the difference in average corridor speed and the distance of the corridor from the TSP corridor.

Table 2-15: Average Speeds along TSP Corridor and Parallel Corridors – Off peak

Corridor		Base Case Average Speed (mph)	TSP Strategy 1 on Western Ave Average Speed (mph)	TSP Strategy 2 on Western Ave Average Speed (mph)	Difference Between TSP Strategy 2 Test and Base Case
Pulaski	SB	27.1	26.9	26.9	-0.2
Pulaski	NB	25.9	26.0	26.0	0.1
Kedzie	SB	25.8	25.8	25.8	0.0
Kedzie	NB	24.9	25.1	25.6	0.7
California	SB	24.6	24.7	24.9	0.2
California	NB	24.8	24.9	24.6	-0.2
Western	SB	26.5	27.3	26.5	0.1
Western	NB	25.8	26.4	25.8	0.0
Damen	SB	28.2	28.2	28.2	0.0
Damen	NB	13.3	14.3	16.5	3.3
Ashland	SB	26.8	26.8	26.8	0.0
Ashland	NB	26.3	26.4	26.5	0.2
Racine	SB	25.9	25.9	25.9	0.0
Racine	NB	25.7	25.8	25.8	0.0
Halsted	SB	26.0	26.0	26.0	0.1
Halsted	NB	25.8	25.9	25.8	0.0
King	SB	30.5	30.5	30.5	0.0
King	NB	31.3	31.3	31.3	0.0
Cottage Grove	SB	29.7	29.8	29.8	0.1
Cottage Grove	NB	30.3	30.2	30.2	0.0

Just as the results showed no obvious pattern in the effects of TSP on average corridor speed or parallel corridors, there is similarly little pattern in its effect on the average speeds along perpendicular corridors. Table 2-16 shows the average travel speeds along major corridors perpendicular to the priority corridor, Western Avenue. The corridors are listed in order from north to south, and the corridor segments considered are defined in Table 2-12. Table 2-16 shows that changes in average speed along perpendicular corridors are both positive and negative between –0.5 mph and +2.7 mph in the off-peak.

It was expected that the use of Strategy 2 would result in higher speeds along E-W corridors than using Strategy 1, since non-priority movements on these corridors are compensated using Strategy 2. Table 2-16, shows that several corridors did indeed experience higher travel speeds than using Strategy 1; however, many corridors also experienced no difference, and a few even experienced a negative difference.

Table 2-16: Average Speeds along Corridors the Intersect the TSP Corridor – Off-peak

Corridor		Base Case Average Speed (mph)	TSP Strategy 1 on Western Ave Average Speed (mph)	TSP Strategy 2 on Western Ave Average Speed (mph)	Difference Between TSP Strategy 2 Test and Base Case (mph)	Difference Between TSP Strategy 2 Test and TSP Strategy 1 Test (mph)
Lawrence	WB	23.9	23.7	23.8	-0.1	0.1
Lawrence	EB	23.1	22.6	22.8	-0.2	0.2
Montrose	WB	23.3	23.4	23.3	0.0	0.0
Montrose	EB	23.4	23.3	23.4	0.0	0.1
Irving Park	WB	23.9	23.8	24.0	0.1	0.2
Irving Park	EB	23.4	23.4	23.5	0.1	0.1
Addison	WB	25.2	25.2	25.2	0.0	0.0
Addison	EB	23.6	23.3	23.6	0.0	0.3
Belmont	WB	24.8	24.7	24.7	-0.1	0.0
Belmont	EB	24.5	24.4	24.4	-0.1	-0.1
Diversey	WB	23.9	24.0	23.9	0.1	0.0
Diversey	EB	22.8	22.8	23.1	0.3	0.3
Fullerton	WB	22.8	22.9	22.8	0.0	-0.1
Fullerton	EB	18.8	16.1	19.4	0.6	3.3
North	WB	23.2	23.2	23.3	0.1	0.1
North	EB	18.9	18.7	18.3	-0.5	-0.4
Division	WB	23.7	23.7	23.7	0.0	0.0
Division	EB	23.1	23.2	23.2	0.2	0.0
Chicago	WB	22.4	22.5	22.5	0.0	0.0
Chicago	EB	22.4	22.5	22.6	0.2	0.1
Archer	WB	26.9	26.9	26.9	0.0	0.0
Archer	EB	26.0	26.1	26.0	0.0	-0.1
79th	WB	19.4	18.7	20.3	1.0	1.6
79th	EB	27.5	27.4	27.4	-0.1	0.0
87th	WB	25.2	24.4	25.7	0.5	1.3
87th	EB	19.9	21.5	22.6	2.7	1.1
95th	WB	26.3	26.0	26.0	-0.2	0.0
95th	EB	24.4	23.7	24.0	-0.4	0.3
103rd	WB	25.3	25.5	25.6	0.3	0.2
103rd	EB	25.8	25.7	25.8	-0.1	0.0

Just as general traffic conditions are minimally affected by TSP, bus routes on non-priority corridors also experience conditions that are very similar to base case conditions. For example, Table 2-17 and 2-18 show that bus travel times on non-priority corridors are virtually the same for the test case with TSP Strategy 2 as for the base case. Further, these results are very similar to the results obtained with TSP Strategy 1.

Table 2-17: Bus Routes on N-S Corridors – Off peak

Route		Route Length (mi)	Nveh	Base No TSP		TSP on Western Strategy 1		TSP on Western Strategy 2	
				AvgTT	StdDev	AvgTT	StdDev	AvgTT	StdDev
53 Pulaski	SB	23.6	21	40.9	0.9	40.9	0.9	40.9	0.9
53 Pulaski	NB	23.7	19	43.5	1.3	43.5	1.3	43.5	1.3
53A South Pulaski	SB	30.8	14	35.3	1.3	35.3	1.3	35.3	1.3
53A South Pulaski	NB	30.0	14	33.5	1.2	33.5	1.2	33.5	1.2
385 87th-111th	SB	26.3	3	69.8	7.4	67.2	5.7	66.1	3.9
385 87th-111th	NB	26.8	3	65.4	1.2	67.6	3.0	66.8	4.3
52 Kedzie-California	SB	22.0	17	42.8	1.3	42.8	1.3	42.8	1.3
52 Kedzie-California	NB	22.7	15	39.8	1.6	39.8	1.6	39.8	1.6
52A South Kedzie	SB	32.0	20	37.2	1.5	37.2	1.5	37.2	1.5
94 South California	SB	29.4	16	40.3	2.0	40.0	2.0	39.9	1.8
94 South California	NB	31.1	20	41.2	1.3	41.3	1.3	41.2	1.3
Western Ave									
48 South Damen	SB	16.4	10	15.8	1.0	15.8	1.0	15.8	1.0
50 Damen	SB	26.1	14	41.2	3.1	41.2	3.1	40.9	2.6
50 Damen	NB	25.8	14	65.0	12.9	62.7	11.2	55.2	8.5
9 Ashland Ave	SB	39.4	18	65.4	2.1	65.4	2.1	65.4	2.1
9 Ashland Ave	NB	35.1	18	62.4	2.5	62.4	2.5	62.4	2.5
44 Wallace-Racine	SB	29.4	20	32.1	0.7	32.1	0.7	32.1	0.7
44 Wallace-Racine	NB	28.7	20	32.2	2.2	32.2	2.2	32.2	2.2
8 Halsted St	SB	32.4	21	52.8	1.9	52.8	1.9	52.8	1.9
8 Halsted St	NB	33.8	21	53.2	1.9	53.2	1.9	53.2	1.9
8A S Halsted St	SB	24.6	14	25.2	1.3	25.2	1.3	25.2	1.3
8A S Halsted St	NB	25.0	15	22.5	0.8	22.5	0.8	22.5	0.8
108 Halsted/95th	SB	16.7	24	18.7	0.9	18.7	0.9	18.7	0.9
108 Halsted/95th	NB	16.7	21	16.1	0.8	16.2	0.8	16.2	0.8
352 S Hlstd-Chic Hts	SB	17.2	9	47.5	1.5	47.5	1.5	47.5	1.5
352 S Hlstd-Chic Hts	NB	17.2	9	46.6	2.6	46.6	2.6	49.0	2.0
359 Robbins-S Kedzie	SB	12.9	6	32.4	1.2	32.4	1.2	32.4	1.2
359 Robbins-S Kedzie	NB	12.9	6	32.9	1.3	32.9	1.3	32.9	1.3
3 King Drive	SB	33.7	18	49.5	2.0	49.5	2.0	49.5	2.0
3 King Drive	NB	22.1	18	42.3	1.7	42.3	1.7	42.3	1.7
4 Cottage Grove	SB	28.6	27	46.4	2.3	46.4	2.3	46.4	2.3
4 Cottage Grove	NB	28.4	20	46.7	2.5	46.7	2.5	46.7	2.5

Table 2-18: Bus Routes on E-W Corridors – Off peak

Route		Route Length (mi)	Nveh	Base No TSP		TSP on Western Strategy 1		TSP on Western Strategy 2	
				AvgTT	StdDev	AvgTT	StdDev	AvgTT	StdDev
78 Montrose	EB	18.4	16	33.0	1.1	33.1	1.3	33.0	1.1
78 Montrose	WB	18.4	13	31.6	1.5	31.6	1.5	31.6	1.5
80 Irving Park	EB	29.1	14	34.5	1.5	35.0	1.8	34.5	1.5
80 Irving Park	WB	27.1	14	34.5	1.6	34.4	1.6	34.3	1.8
152 Addison	EB	28.1	18	36.9	0.5	37.0	0.6	36.9	0.5
152 Addison	WB	29.7	18	35.5	1.1	35.2	1.0	35.5	1.1
77 Belmont	EB	27.1	28	38.2	1.4	38.4	1.2	38.1	1.5
77 Belmont	WB	28.0	16	35.4	1.4	35.2	1.5	35.9	1.4
76 Diversey	EB	20.3	28	35.5	1.4	35.9	1.6	35.5	1.5
76 Diversey	WB	19.4	28	33.4	1.9	33.5	1.3	33.5	2.0
74 Fullerton	EB	16.5	8	35.5	2.7	37.9	3.3	34.6	1.7
74 Fullerton	WB	15.9	8	31.4	1.4	31.1	1.1	31.7	1.4
70 Division	EB	15.6	20	27.2	1.2	27.3	1.1	27.3	1.1
70 Division	WB	14.9	17	26.6	0.7	26.9	0.9	26.6	0.7
66 Chicago	EB	18.3	34	35.8	1.0	35.9	1.1	35.8	1.1
66 Chicago	WB	18.7	28	36.8	1.2	36.8	1.2	36.8	1.3
62 Archer	EB	25.2	25	46.7	2.1	46.8	2.3	46.7	2.1
62 Archer	WB	27.3	21	47.5	2.1	47.1	1.7	47.5	2.1
62H Archer/Harlem	EB	13.6	12	17.7	1.0	17.7	1.0	17.7	1.0
62H Archer/Harlem	WB	12.7	12	15.1	0.8	15.1	0.8	15.2	0.8
831 Joliet-Midway	EB	32.0	2	57.1	0.0	57.1	0.0	57.1	0.0
79 79th	EB	37.6	21	36.5	1.2	36.3	1.0	36.4	1.2
79 79th	WB	35.8	26	43.1	4.4	43.5	4.4	42.1	3.9
379 W 79th	SB	11.6	6	26.6	0.3	26.6	0.3	26.6	0.3
379 W 79th	NB	11.3	6	25.4	0.5	25.4	0.5	25.4	0.5
87 87th	EB	27.1	13	30.7	1.5	29.7	1.8	30.4	1.7
87 87th	WB	30.3	14	33.1	1.4	32.9	1.3	33.2	1.5
385 87th-111th	SB	26.3	3	69.8	7.4	67.2	5.7	66.1	3.9
385 87th-111th	NB	26.8	3	65.4	1.2	67.6	3.0	66.8	4.3
95W 95th	EB	8.5	22	11.0	0.7	10.9	0.5	10.9	0.6
95W 95th	WB	7.3	18	10.5	0.7	10.5	0.4	10.3	0.5
381 95th Street	EB	13.0	6	34.7	0.6	34.6	0.4	35.4	1.1
381 95th Street	WB	13.0	6	35.2	0.9	35.9	1.4	35.4	1.2
103 West 103rd	EB	16.8	18	20.9	0.9	21.1	1.0	21.1	1.0
103 West 103rd	WB	17.9	17	21.7	1.0	21.7	1.1	21.7	1.0
382 Central S	SB	8.5	6	22.0	1.1	22.0	1.1	22.0	1.1

In short, tests of Strategy 2 on Western Ave showed that travel times on priority routes improved with TSP, but not as much as with Strategy 1. Further, the average speed of general traffic along the TSP corridor was almost unchanged with Strategy 2, which is makes sense since Strategy 2 maintains the same overall phase split by compensating non-priority phases after each priority treatment. Further, Strategy 2 was found to have minimal effects on parallel and intersecting non-priority corridors.

2.5.2.3.3 Spatial Impacts with Increasing Numbers of Corridors

This section explores the effect of implementing multiple TSP corridors with Strategy 1. Results are presented for tests with a single corridor (Western), two parallel corridors (Western and Halsted), and three parallel corridors (Western, Halsted and Ashland). In addition, results for a test with eight corridors (four NS and 4 EW) are presented. The results for these tests will be compared to illustrate the effects of combining multiple parallel and intersecting TSP corridors. The TSP corridors used in these tests are listed in Table 2-19.

Table 2-19: TSP Corridors for Tests with Multiple Corridors

Time of Day	From	To	Length (mi)
Irving Park	Cumberland	Lake Shore Drive	9.9
Chicago	Austin	Fairbanks	8.0
Archer	1st Ave	Clark	11.9
95th	LaGrange	Dan Ryan Expressway	11.8
Western	Birchwood	Dixie-Sibley	27.5
Ashland	Irving Park	95th	16.1
Halsted	Waveland	Dixie-Sibley	22.4
MLK	Cermak	95th	9.0

It was expected that the implementation of TSP would result in shorter bus travel times along the TSP corridors. Further, it was expected that as travelers select new routes in reaction to the implementation of TSP, traffic conditions beyond the TSP corridor would also change. To explore these hypotheses, travel times were examined for buses that traveled along both priority and non-priority corridors. In this section, bus travel times on the selected corridors are compared across scenarios with varying numbers of TSP corridors throughout the region.

Table 2-20 shows the travel times of bus routes on N-S corridors, listed in order from west to east. Grey shading indicates that the given bus route received priority treatment in the given scenario. The table shows that in general, buses that traveled along non-priority corridors experienced minimal, if any, changes in travel time from the base case scenario, regardless of the proximity of the corridor to the priority corridors. One exception is route 50 Damen NB, which experienced a travel time of 65.0 minutes under base case conditions, and under 63 minutes in all of the TSP scenarios. Since Damen is extremely close to the priority corridors, these results suggest that rerouting due to TSP have resulted in less congestion on Damen in the northbound direction. However, results for other non-priority corridors suggest that the travel time changes are typically much lower.

Table 2-20: Travel Times of Bus Routes on N-S Corridors

Route	Direction	Route Length (mi)	Nveh	Base No TSP AvgTT (min)	TSP on Western Strategy 1 AvgTT (min)	TSP on Western, Halsted Strategy 1 AvgTT (min)	TSP on Western, Halsted, Ashland Strategy 1 AvgTT (min)	TSP on 8 corridors Strategy 1 AvgTT (min)
53 Pulaski	SB	23.0	21	40.9	40.9	40.9	40.9	41.1
53 Pulaski	NB	23.0	19	43.5	43.5	43.5	43.5	43.7
53A South Pulaski	SB	30.0	14	35.3	35.3	35.3	35.3	35.6
53A South Pulaski	NB	30.0	14	33.5	33.5	33.5	33.5	33.2
52 Kedzie-California	SB	22.0	17	42.8	42.8	42.8	42.8	42.2
52 Kedzie-California	NB	22.0	15	39.8	39.8	39.8	39.8	40.1
52A South Kedzie	SB	31.0	20	37.2	37.2	37.2	37.2	37.2
94 South California	SB	29.0	16	40.3	40.0	40.3	40.0	40.4
94 South California	NB	31.0	20	41.2	41.3	41.3	41.3	41.4
49 Western Ave	SB	34.0	23	59.0	45.4	45.6	45.4	45.7
49 Western Ave	NB	34.0	20	58.9	44.9	45.1	45.2	44.9
49A S Western Ave	SB	26.0	6	32.0	25.6	25.7	25.8	25.7
49A S Western Ave	NB	26.0	6	32.7	29.4	29.5	29.3	29.0
49B N Western Ave	SB	8.0	19	13.0	10.4	10.4	10.4	10.4
49B N Western Ave	NB	7.0	19	11.2	9.1	9.0	9.2	9.1
X49 Western Exp	SB	18.0	12	54.0	39.7	39.4	39.0	39.5
X49 Western Exp	NB	18.0	12	53.3	39.5	39.3	39.6	39.6
349 S Western	SB	11.0	7	29.2	23.7	23.8	23.8	24.0
349 S Western	NB	11.0	7	30.1	26.6	26.6	26.8	26.5
48 South Damen	SB	16.0	10	15.8	15.8	15.8	15.8	15.8
50 Damen	SB	26.0	14	41.2	41.2	41.2	41.1	41.3
50 Damen	NB	25.0	14	65.0	62.7	62.9	62.2	62.4
9 Ashland Ave	SB	39.0	18	65.4	65.4	65.4	49.7	49.8
9 Ashland Ave	NB	35.0	18	62.4	62.4	62.4	48.4	47.7
44 Wallace-Racine	SB	29.0	20	32.1	32.1	32.1	32.1	32.2
44 Wallace-Racine	NB	28.0	20	32.2	32.2	32.0	32.0	32.3
8 Halsted St	SB	32.0	21	52.8	52.8	42.2	42.4	42.5
8 Halsted St	NB	33.0	21	53.2	53.2	42.7	43.0	43.7
8A S Halsted St	SB	24.0	14	25.2	25.2	21.4	21.4	21.6
8A S Halsted St	NB	25.0	15	22.5	22.5	20.6	20.6	20.8
108 Halsted/95th	SB	16.0	24	18.7	18.7	15.9	15.9	16.0
108 Halsted/95th	NB	16.0	21	16.1	16.2	15.3	15.3	14.7
352 S Hlstd-Chic Hts	SB	17.0	9	47.5	47.5	43.4	43.4	42.8
352 S Hlstd-Chic Hts	NB	17.0	9	46.6	46.6	44.4	44.4	44.4
359 Robbins-S Kedzie	SB	12.0	6	32.4	32.4	32.1	32.1	32.1
359 Robbins-S Kedzie	NB	12.0	6	32.9	32.9	32.6	32.6	32.6
3 King Drive	SB	33.0	18	49.5	49.5	49.5	49.5	44.0
3 King Drive	NB	22.0	18	42.3	42.3	42.3	42.3	36.8
4 Cottage Grove	SB	28.0	27	46.4	46.4	46.4	46.4	45.8
4 Cottage Grove	NB	28.0	20	46.7	46.7	46.7	46.7	45.9

Table 2-20 also shows that bus routes on priority corridors experience significant travel time improvements over the base case, saving more than 25% of the travel time in many cases. At the same time, the table shows that the addition of TSP corridors parallel to Western Ave did not result in consistently increasing or decreasing travel times for routes on Western Ave. For example, route 49 Western Ave NB experienced an increase in travel time with the addition of TSP on Halsted, and a further increase with the addition of TSP on Ashland. On the other hand, route X49 Western Express SB experienced a decrease in travel time with the addition of TSP on Halsted, and a further decrease with the addition of TSP on Ashland. Moreover, other routes on Western Ave experienced both increases and decreases in travel time with the addition of parallel TSP corridors. Regardless, the differences between route travel times were within one minute for all of the scenarios with parallel N-S TSP corridors. In other words, the results suggest that the addition of parallel TSP corridors does not result in any consistent pattern of increase nor decrease in travel times of priority buses, and further, the changes in travel time are minimal.

In addition, a test with TSP on eight corridors was performed, and the last column of Table 2-20 presents these results to show the impacts of implementing multiple parallel and intersecting TSP corridors. In this scenario, the N-S corridors listed in Table 2-20 were intersected by four E-W TSP corridors. As such, it was expected that bus routes on non-priority N-S corridors would experience travel times slower than the base case, since non-priority buses had to cross four priority intersections. At the same time, it was expected that bus routes on priority N-S corridors would experience travel times faster than the base case, but slower than the scenarios with TSP corridors only in the N-S direction. In fact, the table shows that there is no consistent relative increase, nor decrease in the bus route travel times, either for the priority or the non-priority bus routes. Again, the increases and decreases in bus travel times observed across scenarios is minimal at less than half a minute in most cases.

A similar analysis was performed for bus routes on E-W corridors. Specifically, Table 2-21 shows the travel times of bus routes on E-W corridors, listed in order from north to south. Grey shading indicates that the given bus route received priority treatment in the given scenario, where E-W routes were provided priority only in the scenario with eight TSP corridors. All of the routes listed in Table 2-21 had to traverse the N-S TSP corridors, and it was thus expected that buses traveling in the E-W direction would experience longer travel times as N-S priority corridors were added.

The table shows that in general, buses that traveled along non-priority corridors experienced minimal, if any, changes in travel time from the base case scenario, regardless of the proximity of the corridor to the priority corridors. Further, the table shows that non-priority routes experienced both increases and decreases in travel time; however, those changes were minimal at less than a minute in most cases.

On the other hand, it was expected that priority corridors would experience improvements in travel time, and this is indeed the case, as shown in the shaded cells in last column of Table 2-21.

Table 2-21: Travel Times of Bus Routes on E-W Corridors

Route	Direction	Route Length (mi)	Nveh	Base No TSP AvgTT (min)	TSP on Western Strategy 1 AvgTT (min)	TSP on Western, Halsted Strategy 1 AvgTT (min)	TSP on Western, Halsted, Ashland Strategy 1 AvgTT (min)	TSP on 8 corridors Strategy 1 AvgTT (min)
78 Montrose	EB	18.0	16	33.0	33.1	33.0	33.1	33.2
78 Montrose	WB	18.0	13	31.6	31.6	31.6	31.8	31.7
80 Irving Park	EB	29.0	14	34.5	35.0	34.9	35.0	29.8
80 Irving Park	WB	27.0	14	34.5	34.4	34.2	34.5	29.2
152 Addison	EB	28.0	18	36.9	37.0	37.0	37.1	37.2
152 Addison	WB	29.0	18	35.5	35.2	35.2	35.2	35.2
77 Belmont	EB	27.0	28	38.2	38.4	38.5	38.1	38.2
77 Belmont	WB	28.0	16	35.4	35.2	35.3	35.7	35.8
76 Diversey	EB	20.0	28	35.5	35.9	35.3	35.8	35.4
76 Diversey	WB	19.0	28	33.4	33.5	33.4	33.4	33.4
74 Fullerton	EB	16.0	8	35.5	37.9	37.3	37.0	36.9
74 Fullerton	WB	15.0	8	31.4	31.1	31.2	31.4	31.5
70 Division	EB	15.0	20	27.2	27.3	27.7	27.4	27.3
70 Division	WB	14.0	17	26.6	26.9	26.9	27.4	27.6
66 Chicago	EB	18.0	34	35.8	35.9	36.2	36.4	31.2
66 Chicago	WB	18.0	28	36.8	36.8	37.0	37.3	30.7
62 Archer	EB	25.0	25	46.7	46.8	46.7	46.7	41.2
62 Archer	WB	27.0	21	47.5	47.1	47.1	47.3	39.3
62H Archer/Harlem	EB	13.0	12	17.7	17.7	17.7	17.7	16.6
62H Archer/Harlem	WB	12.0	12	15.1	15.1	15.1	15.1	14.5
831 Joliet-Midway	EB	32.0	2	57.1	57.1	57.1	57.1	57.1
79 79th	EB	37.0	21	36.5	36.3	36.5	36.5	36.2
79 79th	WB	35.0	26	43.1	43.5	44.3	44.5	44.4
379 W 79th	SB	11.0	6	26.6	26.6	26.6	26.6	26.8
379 W 79th	NB	11.0	6	25.4	25.4	25.4	25.4	25.4
87 87th	EB	27.0	13	30.7	29.7	29.7	29.5	30.0
87 87th	WB	30.0	14	33.1	32.9	33.0	34.9	35.2
385 87th-111th	SB	26.0	3	69.8	67.2	67.2	68.2	68.2
385 87th-111th	NB	26.0	3	65.4	67.6	67.6	67.6	68.5
95W 95th	EB	8.0	22	11.0	10.9	10.9	10.9	9.8
95W 95th	WB	7.0	18	10.5	10.5	10.5	10.6	9.7
381 95th Street	EB	13.0	6	34.7	34.6	34.6	34.6	30.8
381 95th Street	WB	13.0	6	35.2	35.9	36.5	36.2	31.5
103 West 103rd	EB	16.0	18	20.9	21.1	20.7	20.6	20.8
103 West 103rd	WB	17.0	17	21.7	21.7	22.0	22.1	21.9
382 Central S	SB	8.0	6	22.0	22.0	22.0	22.0	22.2

In summary, the results shown in Tables 2-20 and 2-21 indicate that TSP Strategy 1 results in improvements in travel time for priority bus routes, while at the same time having minimal impact on travel times of both parallel and intersecting non-priority routes. Further, the addition of parallel and intersecting TSP corridors did not result in consistent nor significant changes in travel times of priority buses. These findings suggest that impacts on TSP corridors are relatively independent of other parallel and intersecting TSP corridors for the configurations tested. It is expected, however, that corridor configurations that are more dense with more intersecting corridors may result in more significant impacts due to conflicting TSP requests.

The average speeds of general traffic along major corridors throughout the region were examined, where the average speed includes both automobile and bus speeds. Table 2-22 shows the average speeds on N-S corridors, listed in order from west to east. Grey shading indicates that the given corridor received priority treatment in the given scenario. The table shows that in general, vehicles that traveled along non-priority corridors experienced minimal, if any, changes in average speed from the base case scenario, regardless of the proximity to the priority corridors. One exception was northbound Damen, which experienced an improvement in travel speed of 1 mph. This exception was also seen in the changes in bus route travel times for route 50 Damen NB. Again, since Damen is extremely close to the priority corridors, these results suggest that rerouting due to TSP have resulted in less congestion on Damen in the northbound direction; however, results for other non-priority corridors suggest that changes in speed are typically much lower.

Table 2-22: Average Speed of General Traffic on N-S Corridors

Corridor		Base No TSP AvgSp (mph)	TSP on Western Strategy 1 AvgSp (mph)	TSP on Western, Halsted Strategy 1 AvgSp (mph)	TSP on Western, Halsted, Ashland Strategy 1 AvgSp (mph)	TSP on 8 corridors Strategy 1 AvgSp (mph)
Pulaski	SB	27.1	26.9	26.9	26.9	27.0
Pulaski	NB	25.9	26.0	26.0	26.0	25.9
Kedzie	SB	25.8	25.8	25.8	25.8	25.7
Kedzie	NB	24.9	25.1	25.3	25.4	25.3
California	SB	24.6	24.7	24.7	24.7	24.7
California	NB	24.8	24.9	24.9	24.9	24.9
Western	SB	26.5	27.3	27.3	27.4	27.2
Western	NB	25.8	26.4	26.5	26.4	26.4
Damen	SB	28.2	28.2	28.2	28.1	28.1
Damen	NB	13.3	14.3	14.3	14.4	14.3
Ashland	SB	26.8	26.8	26.8	27.0	26.8
Ashland	NB	26.3	26.4	26.4	27.2	27.3
Racine	SB	25.9	25.9	25.9	25.8	25.8
Racine	NB	25.7	25.8	25.8	25.8	25.8
Halsted	SB	26.0	26.0	26.9	26.9	26.8
Halsted	NB	25.8	25.9	26.3	26.4	26.3
King	SB	30.5	30.5	30.4	30.5	30.9
King	NB	31.3	31.3	31.3	31.3	32.0
Cottage Grove	SB	29.7	29.8	29.8	29.7	29.6
Cottage Grove	NB	30.3	30.2	30.2	30.2	30.1

Table 2-22 also shows that general traffic on priority corridors experienced slight, but consistent, increases in average speed over the base case. More specifically, average speeds along priority corridors improved by 0.5 to 1.0 mph. At the same time, the table shows that the addition of TSP corridors parallel to Western Ave did not result in consistent increases or decreases average speeds along Western Ave. For example, southbound Western Ave experienced no change in speed with the addition of TSP on Halsted, but the average speed increased with the addition of TSP on Ashland. On the other hand, northbound Western Ave experienced an increase in speed with the addition of TSP on Halsted, and then a decrease with the addition of TSP on Ashland. Regardless, the differences between speeds were within 0.1 mph, indicating that the addition of parallel TSP corridors does not result in any significant, nor consistent pattern of increase nor decrease in average corridor speeds.

In addition, the test with TSP on eight corridors showed that changes in corridor speeds were neither significant nor consistent for both priority and non-priority corridors. In other words, non-priority N-S corridors were intersected by four priority E-W corridors, and yet did not in general experience significantly slower travel speeds than the base case, nor than the other TSP scenarios. Similarly, priority corridors, which also had to cross four non-priority E-W corridors did not experience significantly slower travel times than they did in scenarios with no intersecting E-W corridors.

A similar analysis was performed for average speeds along E-W corridors. Table 2-23 lists the E-W corridors in order from north to south. Further, cells that represent corridors with TSP are shaded in grey. All of the routes listed in Table 2-23 had to traverse the N-S TSP corridors, and it was thus expected that E-W corridors would experience slower average speeds as N-S priority corridors were added. In fact, the table shows that in general non-priority corridors experienced minimal, if any, changes in average speed from the base case scenario, regardless of the proximity of the corridor to the priority corridors. Further, the table shows that non-priority corridors experienced both increases and decreases in speed; however, those changes were minimal at less than 0.2 mph in most cases.

In addition, it was expected that priority corridors would experience increased average speed, and as shown in the shaded cells in last column of Table 2-23, changes ranged from none to increases of 0.9 mph.

Table 2-23: Average Speed of General Traffic on E-W Corridors

		Base	TSP on Western	TSP on Western, Halsted	TSP on Western, Halsted, Ashland	TSP on 8 corridors
Corridor		No TSP	Strategy 1	Strategy 1	Strategy 1	Strategy 1
		AvgSp (mph)	AvgSp (mph)	AvgSp (mph)	AvgSp (mph)	AvgSp (mph)
Lawrence	WB	23.9	23.7	23.7	23.7	23.8
Lawrence	EB	23.1	22.6	22.5	22.5	22.7
Montrose	WB	23.3	23.4	23.4	23.3	23.3
Montrose	EB	23.4	23.3	23.3	23.3	23.3
Irving Park	WB	23.9	23.8	23.8	23.8	23.9
Irving Park	EB	23.4	23.4	23.5	23.6	23.6
Addison	WB	25.2	25.2	25.2	25.1	25.1
Addison	EB	23.6	23.3	23.3	23.3	23.5
Belmont	WB	24.8	24.7	24.8	24.8	24.8
Belmont	EB	24.5	24.4	24.4	24.4	24.4
Diversey	WB	23.9	24.0	24.0	23.9	23.9
Diversey	EB	22.8	22.8	22.8	22.7	22.7
Fullerton	WB	22.8	22.9	22.8	22.9	22.8
Fullerton	EB	18.8	16.1	16.0	16.6	16.6
Division	WB	23.7	23.7	23.5	23.5	23.6
Division	EB	23.1	23.2	23.1	23.1	23.0
Chicago	WB	22.4	22.5	22.4	22.4	23.0
Chicago	EB	22.4	22.5	22.4	22.3	22.8
Archer	WB	26.9	26.9	27.0	26.7	27.2
Archer	EB	26.0	26.1	26.0	25.9	26.3
79th	WB	19.4	18.7	18.6	19.0	19.3
79th	EB	27.5	27.4	27.4	27.5	27.4
87th	WB	25.2	24.4	24.1	23.9	22.6
87th	EB	19.9	21.5	21.5	21.3	21.3
95th	WB	26.3	26.0	26.0	25.9	26.2
95th	EB	24.4	23.7	23.8	23.6	24.5
103rd	WB	25.3	25.5	25.5	25.5	25.5
103rd	EB	25.8	25.7	25.8	25.9	25.8

In summary, the results shown in Tables 2-22 and 2-23 indicate that TSP Strategy 1 results in minor improvements in average speed for priority corridors, while at the same time having minimal impact on speeds along both parallel and intersecting non-priority routes. Further, the addition of parallel and intersecting TSP corridors did not result in consistent nor significant changes in speeds for priority corridors. These findings suggest that impacts on TSP corridors are relatively independent of other parallel and intersecting TSP corridors for the configurations tested. It is expected, however, that corridor configurations that are more dense may result in more significant impacts.

Since TSP shifts the signal timing advantage to priority approaches at TSP intersections, it is expected that the implementation of TSP will result changes in travelers' travel times and route choices. As such, system-wide travel times and trip distances are examined to determine whether the overall impacts of TSP are increases or decreases in travel times and distances. Table 2-24 shows the system-wide miles traveled, travel time and average speed for automobiles in the base case and the TSP tests. The travel statistics include all 830,825 vehicles loaded into the network, including those that did not complete their trips within the simulation period. The table also lists the number of trips completed, since the number of trips completed must be considered in comparing the total vehicle miles traveled and travel times.

Table 2-24: System-wide Statistics

	Base	TSP on Western	TSP on Western, Halsted	TSP on Western, Ashland, Halsted	TSP on 8 corridors
	None	Strategy 1	Strategy 1	Strategy 1	Strategy 1
Vehicles Miles Traveled	10,036,611	10,022,073	10,021,136	10,028,253	10,028,634
Travel Time (hours)	412,623	419,055	418,998	415,119	415,208
Average Speed (mph)	24.3	23.9	23.9	24.2	24.2
Number of Completed Trips	828,753	828,344	828,420	828,766	828,802

The table shows that all of the TSP tests experienced lower system-wide travel distances than the base case; however, in the case of the TSP tests with one TSP corridor and two TSP corridors, the savings in VMT is partly a result of fewer vehicles completing their trips. In these two tests, the lower VMT is accompanied by higher system travel times and thus lower average system travel speeds. On the other hand, the TSP tests with three and eight TSP corridors experienced higher system-wide travel distances than the base case along with a higher number of vehicles completing their trips. In these two tests, the higher VMT is accompanied by slightly higher system travel times and lower average system travel speeds.

The differences in the system-wide travel distances and times suggest that TSP does indeed result in overall changes in network conditions. Further, a comparison of the TSP test scenarios indicates that implementation of TSP on only one or two corridors results in an overall decrease in system efficiency; however, the implementation of TSP on three or more corridors results in improved system travel conditions over the base case.

While the system-wide travel statistics indicate the overall impact of TSP on the network as a whole, it is also important to consider the effect of TSP on individual travelers. More specifically, it is important to examine how many travelers experience longer and shorter travel times, and to what degree they experience those changes. As such, the distribution of changes in trip travel times as a result of TSP implementation is examined for the four TSP tests (see Figure 2-23). In all of the TSP tests, almost half of all automobiles experience no change in travel time, with fewer vehicles experiencing "no change" as the number of TSP corridors is increased. Further, in all TSP tests, the majority of vehicle travel time changes are increases or decreases of less than five minutes; however, some vehicles do experience travel time changes of more than one hour. These extreme changes are rare outliers and are likely results of imperfect equilibrium assignment.

The travel time changes shown in the distribution include both direct and indirect impacts of TSP. More specifically, in each TSP test, over 400,000 vehicles experience travel time changes, and while some of those changes result from the changes in the signal timing at TSP intersections, many of those changes also result from changes in general network conditions due to rerouting of travelers. To illustrate, Table 2-25 compares the number of automobiles that experienced longer and shorter trip travel times, with the number of automobiles that traveled either along or across the TSP corridors. The table shows that in all of the TSP tests, the numbers of trips that experienced decreased travel times are much higher than the numbers of travelers that traveled along the TSP corridors. Similarly, the numbers of trips that experienced increased travel times are much higher than the numbers of travelers that traveled across the TSP corridors. These results show that the indirect effects of TSP extend beyond the TSP corridors

Table 2-25: Travel Time Changes and Use of TSP Intersections

Travel Time Change	TSP on Western Strategy 1	TSP on Western, Halsted Strategy 1	TSP on Western, Ashland, Halsted Strategy 1	TSP on 8 corridors Strategy 1
No change	356,154	347,305	341,107	332,588
Decrease	231,219	234,603	243,416	250,390
Use priority approaches	15,986	28,070	36,650	87,877
Increase	237,497	242,962	240,347	241,892
Use non-priority approaches	59,055	85,211	99,453	147,941

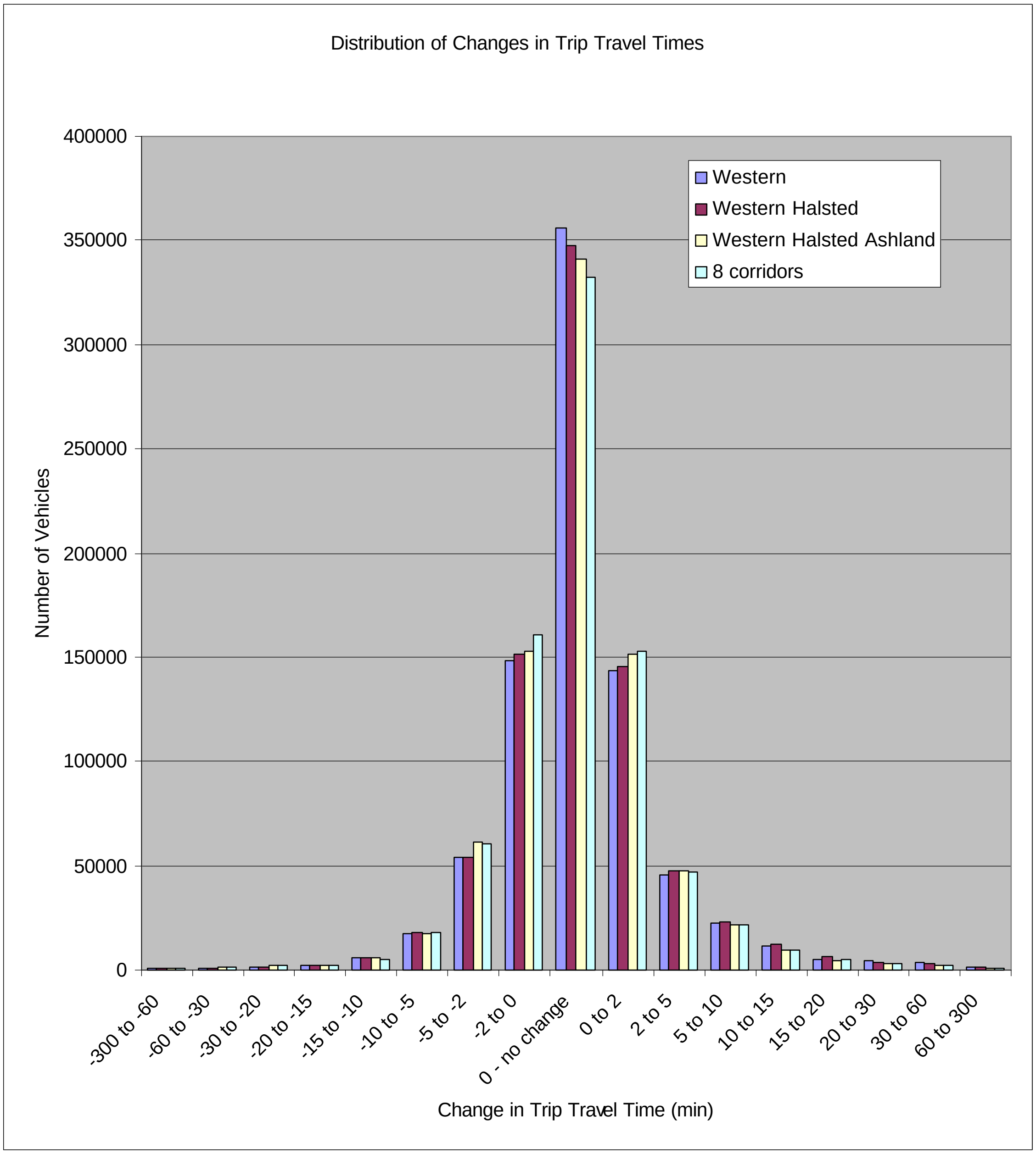


Figure 2-23: Distribution of Changes in Trip Travel Times

The distribution of changes in trip distances as a result of TSP implementation is also examined for the four TSP tests (see Figure 2-24). In all of the TSP tests, the majority of automobiles experience no change in trip distance. Further, in all TSP tests, the majority of vehicle travel distance changes are increases or decreases of less than three miles; however, some vehicles do experience changes of more than 40 miles. As with extreme changes in travel times, these extreme changes in trip distance are rare outliers and are likely results of imperfect equilibrium assignment.

Table 2-26 shows that in all of the scenarios, slightly more than 100,000 vehicles experienced changes in trip distance. Changes in trip distance occur as a result of travelers selecting routes that minimize their travel times with the implementation of TSP. As such, changes in trip distances serve to indicate that changes in route choice do indeed take place as a result of TSP, and will cause changes in network conditions.

Table 2-26: Number of Vehicles Experiencing Changes in Trip Distance

Travel Time Change	TSP on Western Strategy 1	TSP on Western, Halsted Strategy 1	TSP on Western, Ashland, Halsted Strategy 1	TSP on 8 corridors Strategy 1
No change	716,907	717,298	714,074	714,011
Decrease	54,812	54,894	57,323	57,459
Increase	53,151	52,678	53,473	53,400

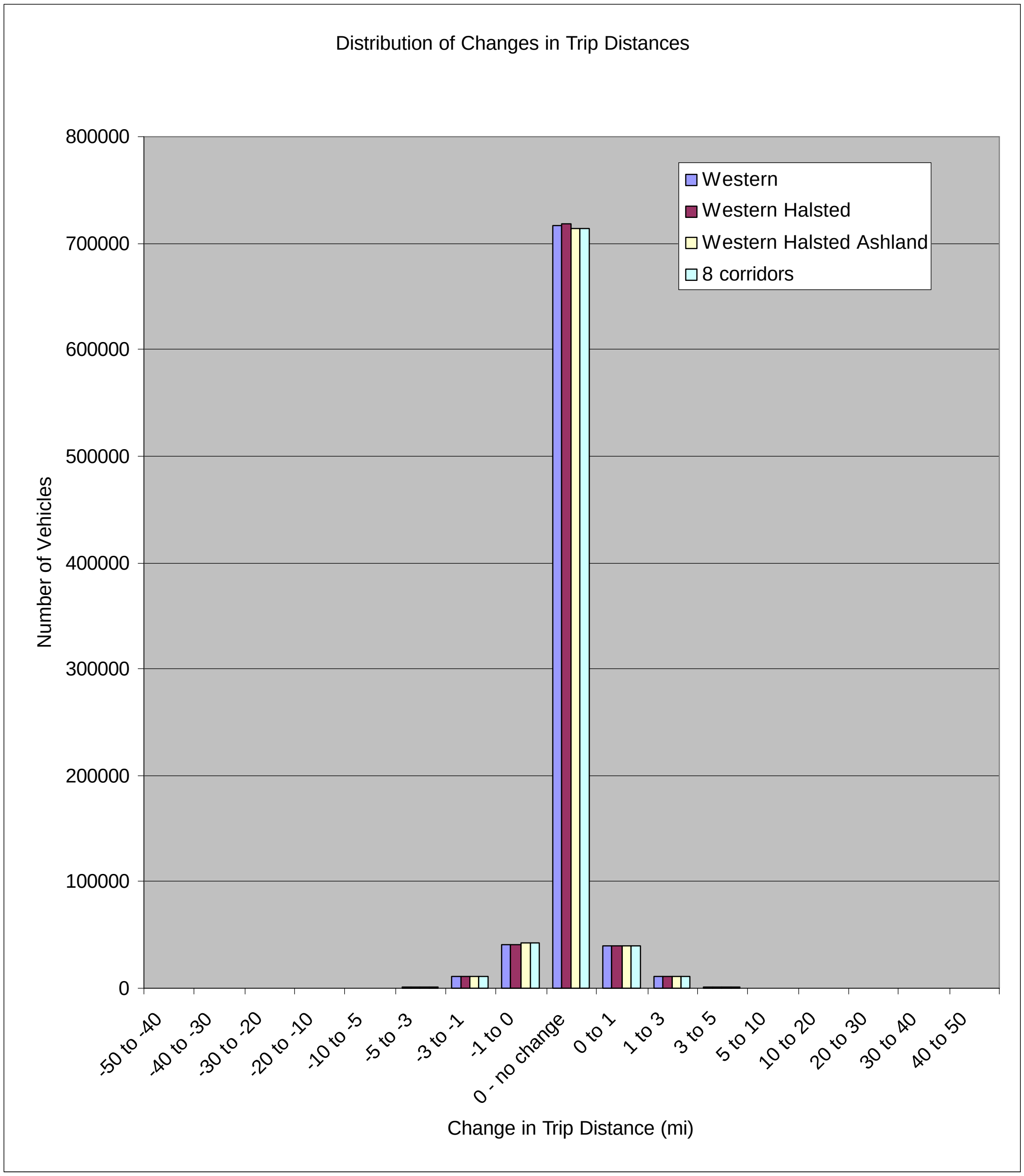


Figure 2-24: Distribution of Changes in Trip Distances

It is expected direct impact of TSP on route choice behavior may be observed in the use priority and non-priority approaches at TSP intersections. Specifically, it is expected that implementation of TSP would attract more vehicles to travel along the priority corridor, and at the same time would encourage vehicles to avoid non-priority approaches. To explore this hypothesis, Table 2-27 lists for each TSP scenario the vehicle miles traveled along TSP corridors compared with the use of those corridors in the base case, as well as the average speeds along those corridors. In addition, Table 2-28 lists for each TSP scenario the number of non-priority movements made compared with the number of movements made on those same links in the base case, along with the travel times averaged over all non-priority approach links.

Table 2-27: Changes in Travel Along Priority Corridor

	Base Case Distance Traveled (mi)	TSP Test Distance Traveled (mi)	Change in Distance Traveled (mi)	Base Case Average Speed (mph)	TSP Test Average Speed (mph)	Change in Average Speed (mph)
TSP Strategy 1 on Western	16,658	16,915	257	22.2	22.7	0.5
TSP Strategy 1 on Western, Halsted	23,864	24,219	354	21.8	22.5	0.7
TSP Strategy 1 on Western, Ashland, Halsted	31,868	31,937	69	21.7	22.3	0.6
TSP Strategy 1 on 8 corridors	116,052	118,285	2,233	23.4	23.5	0.1

Table 2-27 shows that as expected, values of corridor VMT increase with the implementation of TSP. At the same time, travel speeds improve with the implementation of TSP.

In addition, Table 2-28 shows that, except in the first TSP test scenario (TSP Strategy 1 on Western), the number of non-priority movements made decreases with implementation of TSP. The exception is likely a result of overall changes in network conditions negating the effects of the increased travel time across the corridor. Since the second and third scenarios include multiple parallel TSP corridors, travelers would likely have to make multiple non-priority movements in their trips, so it they would stand to benefit more from a change in route, than would the travelers in the first scenario.

Moreover, changes in the numbers of non-priority movements are, as expected, accompanied by increases in overall travel times to traverse non-priority approach links.

Table 2-28: Changes in Travel on Non-priority Approaches

	Base Case Number of Crossings	TSP Test Number of Crossings	Change in Number of Crossings	Base Case Average Approach Time (min)	TSP Test Average Approach Time (min)	Change in Average Approach Time (min)
TSP Strategy 1 on Western	58,006	58,047	41	2.3	2.5	0.2
TSP Strategy 1 on Western, Halsted	102,078	101,953	-125	1.8	2.0	0.2
TSP Strategy 1 on Western, Ashland, Halsted	139,117	138,815	-302	1.5	1.7	0.2
TSP Strategy 1 on 8 corridors	216,222	215,807	-415	1.2	1.4	0.2

In short, Tables 2-27 and 2-28 show that changes in signal timing and travel benefits resulting from implementation of TSP do indeed result in changes in travelers' route choices¸ specifically by attracting more users to travel along the priority corridor and discouraging use of the non-priority approaches.

2.5.2.3.4 Examination of TSP with No Route Changes

This section examines the importance of considering route choice behavior in the evaluation of TSP impacts. Specifically, since TSP shifts the signal timing advantage to priority approaches at TSP intersections, it is expected that the implementation of TSP will result changes in travelers' travel times and route choices. As such, the regional model has been used to capture the effect of changes in traffic conditions along and across the TSP corridors as drivers take advantage of the priority approaches and avoid the non-priority approaches. Further, the regional model has been used to capture changes in traffic conditions in beyond the TSP corridor, as a result of changes in route choice stemming from TSP-related travel time changes.

This section compares the impacts estimated based on fixed route choices with impacts esimated based on route choices that optimize travel time. First, system-wide impacts are examined, then vehicle usage of the priority corridor and the bus travel times along the priority corridor are analysed. Since previous analyses found that non-priority corridors (including those that are both intersecting with and parallel to the TSP corridor) experience little change with the implementation of TSP, the effects of ignoring route changes will not be examined for non-priority corridors.

System-wide travel times and trip distances are shown in Table 2-29 for the base case, the TSP test case with route choice and the TSP test case without route choice. The travel statistics include all 830,825 automobiles loaded into the network, including those that did not complete their trips within the simulation period. The table also lists the number of trips completed, since the number of trips completed must be considered in comparing the total vehicle miles traveled and travel times.

The table shows that the TSP test case without route changes yielded results that were much more similar to the base case than the TSP test case with route changes. More specifically, the total vehicle miles traveled and system travel time in the TSP test case without route changes were only slightly higher than the in the base case, and the resulting average network speed was the same at 24.3 mph. The results for the TSP test case with route changes differed from the other two tests with much lower VMT and average speed, and much higher system travel time. These results suggest that at the system-wide level, the route-changes do indeed have a significant effect on the estimated impacts of TSP.

Table 2-29: System-wide Statistics

	Base None	TSP on Western Strategy 1	TSP on Western - No Route Change Strategy 1
Vehicles Miles Traveled	10,036,611	10,022,073	10,036,657
Travel Time (hours)	412,623	419,055	413,154
Average Speed (mph)	24.3	23.9	24.3
Number of Completed Trips	828,753	828,344	828,746

The direct impacts of TSP on route choice behavior may be observed in the use priority approaches at TSP intersections. Specifically, it is expected that implementation of TSP would attract more vehicles to travel along the priority corridor. As such, for the tests with and without route choice, Table 2-30 lists the vehicle miles traveled along the TSP corridor compared with the use of those corridors in the base case, as well as the average speeds along those corridors.

Table 2-30: Changes in Travel Along Priority Corridor

	Base Case Distance Traveled (mi)	TSP Test Distance Traveled (mi)	Change in Distance Traveled (mi)	Base Case Average Speed (mph)	TSP Test Average Speed (mph)	Change in Average Speed (mph)
TSP Strategy 1 on Western	16,658	16,915	257	22.2	22.7	0.5
TSP Strategy 1 on Western - No Route Change	16,658	16,658	0	22.2	22.8	0.6

Table 2-30 shows that as expected, with route choice permitted, the vehicle miles traveled along the Western Ave. corridor increased with the implementation of TSP. Further, as required, the TSP test with no route choice resulted in no change in vehicle miles traveled along the Western Ave. corridor. As a result of increased demand, the TSP test with route choice estimated a slightly lower average travel speed along the corridor, than did the test without route choice. This suggests that route choice behavior does indeed affect traffic conditions along the TSP corridor.

Since changes in route choice affect general traffic conditions, estimated travel times for bus routes may be affected by assumptions regarding whether or not travelers change routes. Table 2-31 shows the estimated bus route travel times for the base case and the TSP test cases both with and without route choice. The table shows that the TSP test case without route choice yields bus travel times that are quite similar to those estimated with route choice. More specifically, the bus route travel time estimates are all within 0.6 min of the esimates produced using the model with route choice, and in fact, many of the estimates are the same.

Table 2-31: Bus Route Travel Times for Priority Buses

Route		Route Length (mi)	Nveh	Base	TSP on Western	TSP on Western, No Route Changing		
				No TSP	Strategy 1	Strategy 1		
				AvgTT	AvgTT	AvgTT	Diff Base (min)	Diff Route Change (min)
49 Western Ave	SB	34.4	23	59.0	45.4	45.9	-13.0	0.5
49 Western Ave	NB	34.7	20	58.9	44.9	45.0	-13.9	0.1
49A S Western Ave	SB	26.4	6	32.0	25.6	25.8	-6.2	0.2
49A S Western Ave	NB	26.8	6	32.7	29.4	29.4	-3.3	0.0
49B N Western Ave	SB	8.9	19	13.0	10.4	10.3	-2.7	0.0
49B N Western Ave	NB	7.9	19	11.2	9.1	9.1	-2.2	0.0
X49 Western Exp	SB	18.0	12	54.0	39.7	39.1	-14.9	-0.6
X49 Western Exp	NB	18.4	12	53.3	39.5	39.0	-14.3	-0.5
349 S Western	SB	11.4	7	29.2	23.7	23.7	-5.5	0.0
349 S Western	NB	11.5	7	30.1	26.6	26.5	-3.6	0.0

These results suggest that for the off-peak implementation of TSP on Western Ave., the effect of route choice does not have a significant impact on travel times for priority bus routes; however, system-wide impacts and corridor-level impacts on general traffic are affected by the route changes. Further, the effect of route choice on estimated TSP impacts may be more significant under more congested conditions, or with increased numbers of TSP corridors.

2.5.2.3.5 Summary of Regional TSP Impacts

This section presented an analysis of regional impacts of transit signal priority. It was found that TSP Strategy 1 was more effective at reducing travel time and travel time variability of priority buses during the off-peak period than during the AM peak. Further, it was found that Strategy 1 was more effective at reducing travel times of priority vehicles than Strategy 2. In addition, Strategy 1 was found to improve travel speed of general traffic along the TSP corridor, while Strategy 2 did not have a significant effect on general traffic along the TSP corridor. At the same time, neither strategy had a consistent, nor significant effect on non-priority corridors.

In addition, tests of TSP with multiple corridors showed that the addition of parallel and intersecting TSP corridors did not result in consistent nor significant changes in travel times of priority buses, nor of parallel and intersecting non-priority routes. Further, minor improvements were seen in average speed for priority corridors, while at the same time speeds along both parallel and intersecting non-priority routes experienced minimal change with addition of TSP corridors. Further, the addition of parallel and intersecting TSP corridors did not result in consistent nor significant changes in speeds for priority corridors. These findings suggest that impacts on TSP corridors are relatively independent of other parallel and intersecting TSP corridors for the configurations tested.

In addition, it was found that for the off-peak implementation of TSP on Western Ave., the effect of route choice does not have a significant impact on travel times for priority bus routes; however, system-wide impacts and corridor-level impacts on general traffic are affected by the route changes.

Further analysis of the regional impacts of TSP will be presented in the TSP study's Final Report to be submitted to the RTA.

2.6 Summary

This section described the automobile assignment-based multimodal DTA model implemented in the framework of Northwestern University's VISTA software. The software enhancements and algorithms described in this section will support a study of the regional impacts of TSP in Chicago. The multimodal model described in this section captures stopping and dwelling behavior of buses, and the effect of that behavior on the general traffic stream. Further, the DTA-IADUE algorithm will capture travelers' route switching behavior resulting from even subtle TSP-induced travel time changes.

As with other DTA models, the automobile assignment-based multimodal VISTA model iterates between traffic simulation, shortest path calculations, and network assignment. The RouteSim simulator propagates vehicles according to cell transmission model logic. Further, turning movements, signalized intersection and transit signal priority logic are included in the simulator. The simulator also captures the added length of bus vehicles, as well as their frequent stopping behavior; however, dwell times are defined exogenously, and do not reflect the number of boardings and alightings.

The shortest path algorithm calculates only automobile paths, so intermodal trips are not modeled. In addition, a IADUE assignment algorithm is used to find the equilibrium path assignment. This approach replaces the MSA algorithm, which assigns vehicle paths probabilistically, and thus does not guarantee equilibrium. The IADUE algorithm uses a non-linear optimization search approach to find the assignment that minimizes an equilibrium gap function. The IADUE algorithm is heuristic, but preliminary tests are promising. This algorithm will capture travelers' route switching behavior resulting from subtle TSP-induced travel time changes, which might not be captured in the random assignment produced by MSA. The DTA-IADUE assignment algorithm will also be applied to the intermodal model presented in Section 3.

Computational test results were presented for a small test network, as well as the large-scale regional Chicago network, which will be used in the TSP study. The tests on the small network showed that the model is capable of finding the equilibrium solution within an average travel cost deviation of 1-3% per vehicle. Several variations of the algorithm were tested to explore the effects of limiting and suppressing certain search phases. It is expected that similar limits will be encountered due to computational resource constraints when running the algorithm on large-scale networks. The tests suggested that use of the Smith search phase did not approach the equilibrium solution when the path set and extreme point set were limited. In addition, it was found that the algorithm terminated in the fewest number of iterations when the cost gap search was invoked directly following path generation, where path generation was limited; however, the resulting cost gap of 7% was higher than for other variations of the algorithm. In order to save computational time, tests on the regional Chicago network were performed using MSA to generate a limited path set, and then continuing with Phase 3 of the IADUE. For the large-scale network, an average cost deviation of 6.4% and 4.9% per trip were achieved for the AM and off-peak base cases and under 10% for the TSP test scenarios. Validation of base case results with with observed counts and travel times showed that estimates were reasonably evenly distributed both above and below the observed values. Further, a preliminary analysis of the regional impacts of TSP was presented, with a complete analysis of the regional impacts of TSP to be submitted to the RTA in the form of the TSP study's Final Report.

In short, this section presented an automobile assignment-based multimodal simulation model, which captures interactions between cars and buses in the simulator, such that bus travel times and travel time variability can be observed in the simulator output. Further, an IADUE assignment algorithm was developed to estimate the equilibrium assignment pattern accurately enough to account for TSP-induced changes in path travel time. The measures produced by the multimodal model can be used to draw conclusions about operating efficiency and asset management, as well as service and ridership; however, since automobile demand is exogenously determined, the model does not capture nor directly address the impacts of transit policies on mode choice. Person assignment-based approaches, which simultaneously capture mode and route choice behavior are presented in Sections 3 and 4.

3.0 PERSON ASSIGNMENT-BASED INTERMODAL MODEL

The previous section presented a simulation-based multimodal DTA model of cars and buses incorporated into the simulator; however, vehicle demand remained exogenously determined, so mode split and transit ridership could not be directly observed in the model results. The approach presented in this section assigns person demand, rather than vehicle demand. Mode choice is captured by assigning person-trips, rather than automobile trips, to a multimodal network using an algorithm that calculates intermodal least cost paths. The model can be considered a combined mode split and assignment model.

3.1 Implementation in DTA and VISTA

A person assignment-based intermodal model has been implemented in VISTA. As with the automobile assignment-based model presented in Section 2, the person assignment-based approach iterates between traffic simulation, shortest path calculations, and network assignment; however, this approach assigns person trips to intermodal paths instead of assigning automobile trips to single-mode roadway paths.

The dynamic combined model has been implemented in VISTA, according to the iterative procedure shown in Figure 3.1. The ovals represent data, and the rectangles represent processes. Further, the ovals in bold represent data that is entered into the model, whereas the regular ovals represent data that is derived from calculations or processes within the iterative procedure.

As with the vehicle assignment-based DTA model, the person assignment-based model requires network, OD demand, bus routes and schedules and signal data as inputs; however in the combined model OD demand is inputted in terms of person trips, rather than vehicle trips. In other words, in the combined model, the OD demand is entered as person demand, and each person trip is assigned to a multimodal network, so a car is generated only for the trips or portions of trips that take place in a car. In addition, the intermodal model also accounts for parking, and train and pedestrian trip segments. Further, disutility or cost parameters must be entered to reflect traveler preferences.

Preparation modules are run to convert the node and link network into a cell network, the static person demand into time-dependent person demand and the bus routes and bus schedules into bus runs. In addition, parking lots must be converted to automobile source and sink cells in the roadway network.

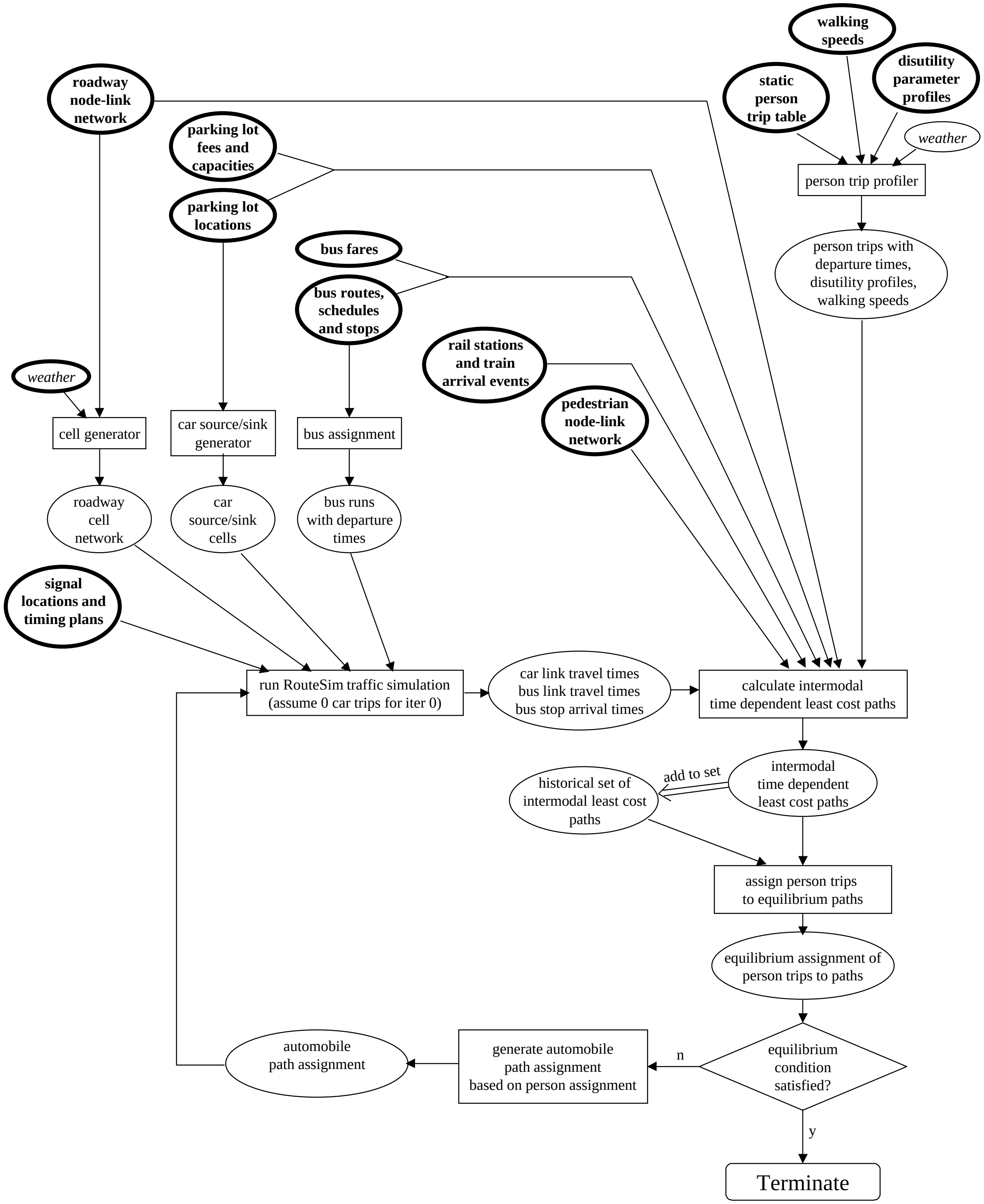

walking speeds
static person trip table
disutility parameter profiles
weather
person trip profiler
person trips with departure times, disutility profiles, walking speeds
roadway node-link network
parking lot fees and capacities
parking lot locations
bus fares
bus routes, schedules and stops
rail stations and train arrival events
pedestrian node-link network
weather
cell generator
car source/sink generator
bus assignment
roadway cell network
car source/sink cells
bus runs with departure times
signal locations and timing plans
run RouteSim traffic simulation (assume 0 car trips for iter 0)
car link travel times
bus link travel times
bus stop arrival times
calculate intermodal time dependent least cost paths
intermodal time dependent least cost paths
add to set
historical set of intermodal least cost paths
assign person trips to equilibrium paths
equilibrium assignment of person trips to paths
equilibrium condition satisfied?
n
y
generate automobile path assignment based on person assignment
automobile path assignment
Terminate

Figure 3-1: VISTA Implementation of the Person Assignment-based Intermodal Model

The iterative procedure begins with RouteSim simulation, which simulates the movement of vehicles through the roadway cell network according to the cell transmission model (see Section 2.2.1). By modeling the complex interactions of vehicles and traffic conditions, the simulator determines the travel time for each roadway link and the arrival times of buses at each bus stop. Based on these roadway link travel times and bus arrival times, time dependent intermodal least cost paths are calculated for each origin, destination, departure time and traveler behavior type. These intermodal least cost paths are then added to the set of all previously calculated paths, and person trips are assigned to the equilibrium paths from the path set. Next, if the equilibrium condition is satisfied, then the procedure terminates; otherwise, automobile trip segments are generated based on the most recent trip assignment and the automobile trip segments are simulated in RouteSim to obtain updated link travel times and bus arrival times. The iterative process continues until equilibrium is reached.

The follow sections provide an overview of the procedure’s main components, namely, the RouteSim simulator, the intermodal time dependent least cost path algorithm, and the equilibrium assignment algorithm, as they are applied to the intermodal equilibrium trip assignment problem.

3.2 RouteSim Simulator

In the VISTA implementation, RouteSim is limited to simulating roadway traffic, including both automobile and bus movements. As such, RouteSim operates exactly as described in Section 2.2. Since the simulator will only model roadway traffic, rail movements will be based on fixed train schedules, and pedestrian movements will be based on fixed walking speeds. In other words, train delays and pedestrian congestion will not be considered.

Ideally the simulator would capture person movements, such that dwell times could be modeled as a function of boardings and alightings; however, since person movements are not explicitly simulated, it is impossible to assign a passenger and passenger boarding time to a specific bus run. As such, dwell times will remain exogenously defined.

3.3 Intermodal Time Dependent Shortest Path Algorithm

Whereas the vehicle assignment-based model included a shortest path algorithm to calculate single-mode (automobile) paths, the person assignment-based model will use an intermodal least cost path algorithm to calculate intermodal paths. When assigning demand to a single mode, typically only the travel time is minimized; however, in a multimodal case, many other costs, such as fares and transfer time, must be considered in order to compare different intermodal alternatives. These costs can be combined into a generalized cost function, which forms the basis for selecting user optimal paths. Yang (8) proposed a time dependent intermodal minimum cost path (TDIMCP) algorithm to calculate least cost paths for freight. His algorithm will form the basis for the intermodal least-cost passenger routing algorithm implemented in DTA in this dissertation. The algorithm is described in this section, but with variations made to accommodate passenger travel rather than freight movement. Further, proof of the algorithm’s correctness and a discussion of its cycling behavior, not provided in Yang’s paper, are presented.

3.3.1 Problem Definition

The TDIMCP algorithm finds the minimum cost path on a multimodal network, given time dependent origin and destination locations, travel times and transportation costs. The network is represented by a directed graph, G=(N,A,T,M), where N is the set of nodes, A is the set of arcs, T is the discretized time period of interest, and M is the set of modes. Each arc (i,j) is assigned a set of non-negative travel times $\tau_{ij}^{m}(t)$ associated with mode m at when departing node i at time t. In addition, arcs can be assigned a fixed travel cost, $\mu_{ij}^{m}(t)$, to represent tolls that must be paid to travel on arc (i,j) using mode m, when departing from node i at time t. The arc cost function (Equation 3.1) is a combination of the time-related and fixed travel costs, where the time-related cost is assumed to be a linear function of the travel time. The parameter, $\alpha^{m}(t)$, reflects the level of dislike that traveler's tend to have for a certain mode. For example, buses, which are less comfortable than private cars, would have a higher value of $\alpha^{m}(t)$, resulting in a higher travel cost for the same amount of travel time.

$$f_{ij}^{m}(t)=m_{ij}^{m}(t)+a^{m}(t)\cdot t_{ij}^{m}(t) \qquad (3.1)$$

Mode-switching arcs are also assigned travel times and costs. Switching delays between modes are represented by $\xi_{ijk}^{m1,m2}(t)$, to describe the time required to switch from mode m1 to mode m2 when traveling from node i through node j to node k, departing from node j at time t. Fixed transfer costs, such as bus fares and parking fees, are represented by $\upsilon_{ijk}^{m1,m2}(t)$, to describe the fixed cost associated with switching from mode m1 to mode m2 when traveling from node i through node j to node k, departing from node j at time t. The mode switching cost function (Equation 3.2) is a combination of the time-related and fixed transfer costs, where the time-related cost is assumed to be a linear function of the transfer time. The parameter, $\beta^{m1,m2}(t)$, reflects the level of dislike that traveler's tend to have for a certain transfer. For example, a waiting at a bus shelter might be less onerous than waiting at an open bus stop, especially in the rain, so a transfer at a node with a bus shelter would have a lower value of $\beta^{m1,m2}(t)$, and would thus incur a transfer lower cost.

$$k_{ijk}^{m1,m2}(t)=n_{ijk}^{m1,m2}(t)+b^{m1,m2}(t)\cdot x_{ijk}^{m1,m2}(t) \qquad (3.2)$$

At origin nodes, parallel entry nodes, i', are introduced to capture possible transfer delays, $\xi_{i'ij}^{m1,m2}(t)$, and costs, $\upsilon_{i'ij}^{m1,m2}(t)$ and $\kappa_{i'ij}^{m1,m2}(t)$, caused when entering a node. For example, waiting for a bus and paying bus fare at the beginning of a trip would be captured by a transfer from the entry node to the origin node. Similarly, an exit node D" is added to the destination node D to capture trip end costs such as parking delays and fees.

3.3.2 Design of the Algorithm

The TDIMCP algorithm maintains cost and travel time labels for each node. The label, $\theta_{ij}^{m}(t)$, denotes the optimal path cost from node j to the destination node when arriving at node j from node i at time interval t. The label, $\gamma_{ij}^{m}(t)$, denotes the total travel time on this optimum path from node j to the destination node at time interval t. The necessary and sufficient condition for a cost label, $\theta_{ij}^{m}(t)$, to be optimal is shown in Equation 3.3, with terminal conditions shown in Equation 3.4. These conditions are a straightforward extension of Bellman's principle of optimality (Bellman 1957).

$$\mathrm{q}_{ij}^{m1}(t)=\min\left\{\mathrm{k}_{ijk}^{m1,m2}(t)+\mathrm{f}_{jk}^{m2}\left(t+\mathrm{x}_{ijk}^{m1,m2}(t)\right)+\mathrm{q}_{jk}^{m2}\left(t+\mathrm{x}_{ijk}^{m1,m2}(t)+\mathrm{t}_{jk}^{m2}\left(t+\mathrm{x}_{ijk}^{m1,m2}(t)\right)\right)\right\} \tag{3.3}$$

$$\forall k\in\Gamma(j),\forall i\in\left\{\Gamma^{-1}(j),j'\right\},\forall m1\in M,\forall m2\in M,\forall t\in T,\forall j\in N\setminus D$$

$$\mathrm{q}_{kD}^{m}(t)=\mathrm{k}_{kDD''}^{m,mf}(t)\quad\forall k\in\left\{\Gamma^{-1}(D),D'\right\},\forall m\in M,\forall t\in T \tag{3.4}$$

where Γ(i) is the set of successor nodes to node i, Γ^{-1}(i) is the set of predecessor nodes to node i, D is the destination node, m_f is the exit mode at the destination node, and N\D is the set of all nodes except node D.

The TDIMCP starts from the destination node and iteratively scans all nodes for optimality. The scanning process consists of examining all predecessor nodes to check whether extending the path from the current node to a predecessor node provides a lower cost path from those nodes to the destination for all modes and time intervals. If such an extension does provide a better path for a predecessor node j for at least one mode, time interval and arc combination, node j is considered to have the potential to improve the paths to its predecessor nodes, its cost and time path labels are updated and it is marked as eligible to be scanned. The steps of the TDIMCP algorithm are shown in Figure 3-2.

```
Step 1
Initialize the labels L_i and F_i as follows:
$g_{ij}^{m}(t)$ = ¥, "j Î N½D " i Î {G$^{-1}$(j),j'}, t Î T
$q_{ij}^{m}(t)$ = ¥, "j Î N½D " i Î {G$^{-1}$(j),j'}, t Î T
$g_{k,D}^{m}(t) = x_{k,D,D²}^{m,mf}(t)$, " t Î T, " m½m_f Î M , " k Î {G$^{-1}$(D),D'}
$q_{k,D}^{m}(t) = k_{k,D,D²}^{m,mf}(t)$, " t Î T, " m½m_f Î M, " k Î {G$^{-1}$(D),D'}
$g_{k,D}^{mf}(t) = 0$, " t Î T, " k Î {G$^{-1}$(D),D'}
$q_{k,D}^{mf}(t) = 0$, " t Î T, " k Î {G$^{-1}$(D),D'}
Insert the destination node D into the "Scan Eligible" (SE) list.

Step 2
If the SE list is empty, go to Step 4; otherwise, delete the first node k from the SE list and do the following:
For every node j Î G$^{-1}$(k) do the following:
  For every mode m1 Î M do the following:
    For every mode m2 Î M do the following:
      For every node i Î {G$^{-1}$(j), j'} do the following:
        For all time intervals t Î T do the following:
          If $\mathrm{q}_{i,j}^{m2}(t)>\left\{\mathrm{k}_{i,j,k}^{m2,m1}(t)+\mathrm{f}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)\right)+\mathrm{q}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)+\mathrm{t}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)\right)\right)\right\}$
            then set
$\mathrm{q}_{i,j}^{m2}(t)=\left\{\mathrm{k}_{i,j,k}^{m2,m1}(t)+\mathrm{f}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)\right)+\mathrm{q}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)+\mathrm{t}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)\right)\right)\right\}$
$\mathrm{g}_{i,j}^{m2}(t)=\left\{\mathrm{x}_{i,j,k}^{m2,m1}(t)+\mathrm{t}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)\right)+\mathrm{g}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)+\mathrm{t}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)\right)\right)\right\}$
            and mark node j as eligible to be scanned;
          otherwise,
            do nothing with this time interval and modal combination.
```

If node *j* has been marked as eligible to be scanned and is not currently in the SE list, insert it into the SE list.

Step 3
Go to Step 2.

Step 4
Terminate the algorithm.

Figure 3-2: Time Dependent Intermodal Minimum Cost Path Algorithm

3.3.3 Algorithm Correctness and Cycling Behavior

Upon termination of the TDIMCP algorithm, every cost label $\mathrm{g}_{ij}{}^{m}(t)$ is either an infinite number, meaning that no path exists from node j to the destination node for this mode, predecessor node and time interval combination, or a finite number that represents the least-time path from this node, mode, predecessor node and time interval combination, to the destination node D and exit mode m_F. Proof of the algorithm's correctness is provided first with proof of the following three lemmas:

Lemma 1: At every stage of the computation, the cost labels of a node are either infinite of finite numbers. An infinite label means that no path exists from that node to the destination node for that time step. A finite label is an upper bound on the least-cost path from that node to the destination node at that time step.

Proof:

An infinite label of a node j at a given time step t, for predecessor node i and mode m1, at any stage of computation means that there is no node k with mode m2 that has updated the label of node j for predecessor node i and mode m1 at time interval t up to this stage. Therefore, there exists no path from node j at time interval t to any other node k on mode m2, and consequently to the destination node D.

On the other hand, if the label is finite, then node j at time t has a successor node j that updated the label of the node i at an earlier stage of the computation, according to Equation 3.5.

$$\mathrm{q}_{i,j}^{m2}(t)=\left\{\mathrm{k}_{i,j,k}^{m2,m1}(t)+\mathrm{f}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)\right)+\mathrm{q}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)+\mathrm{t}_{j,k}^{m1}\left(t+\mathrm{x}_{i,j,k}^{m2,m1}(t)\right)\right)\right\} \tag{3.5}$$

At any later step, only one of the following events can take place:

(a) no change in the cost label of node j for predecessor node i and mode m1, and no change in the label of node k for predecessor node j and mode m2

(b) the label of node j for predecessor node i and mode m1 is reduced via a successor node k (the label for the predecessor node i and mode m1 of which would have changed in the interim)

(c) the cost label of node j for predecessor node i and mode m1 is reduced via some successor node other than k, which then becomes the current successor node to node j

(d) the cost label of node k for predecessor node j and mode m2 is reduced, but node k has not been scanned so as to improve the label of node j

Under cases (a), (b) and (c) the equality continues to hold between j and its current successor, while under case (d) the equality becomes an inequality. The relation between a label of any node i≠D and the labels of its successor node k is thus as shown in Equation 3.6.

$$\mathrm{q}_{i,j}^{m2}(t) \geq \left\{\mathrm{k}_{i,j,k}^{m2,m1}(t) + \mathrm{f}_{j,k}^{m1}\left(t + \mathrm{x}_{i,j,k}^{m2,m1}(t)\right) + \mathrm{q}_{j,k}^{m1}\left(t + \mathrm{x}_{i,j,k}^{m2,m1}(t) + \mathrm{t}_{j,k}^{m1}\left(t + \mathrm{x}_{i,j,k}^{m2,m1}(t)\right)\right)\right\} \qquad (3.6)$$

The above reasoning applies to successor node k as well, and that in turn applies to its successor node. Going this way forward, the destination node D will eventually be reached.

Consequently, there exists a path leading from every node j for predecessor node i and mode m1 to the destination node D for time interval t with a finite travel time equal to the label $\mathrm{q}_{ij}^{m}(t)$. This label is an upper bound to the minimum cost path from node j for predecessor node i and mode m to the destination node D at time interval t. ð

Lemma 2: The algorithm converges in a finite number of iterations.

Proof:

Recall that the algorithm deals only with a finite set of discrete time steps, $T=\{t_0,\ t_0+\Delta t,\ t_0+2\Delta t, \ldots,\ t_0+(|T|-1)\Delta t\}$. Since there are a finite number of values of t, there must also be a finite number of values of $\mu_{ij}^{m}(t)$, $\tau_{ij}^{m}(t)$, and $\alpha^{m}(t)$. As such, there is also a finite number of values of travel costs $\phi_{ij}^{m}(t)$. Similarly, there are a finite number of values of $\upsilon_{ijk}^{m1,m2}(t)$, $\xi_{ijk}^{m1,m2}(t)$, $\beta^{m1,m2}(t)$, and transfer costs $\kappa_{ijk}^{m1,m2}(t)$.

Further, $\tau_{ij}^{m}(t)>0$ and $\xi_{ijk}^{m1,m2}(t)\geq 0$, where both must be no more detailed than the level of detail of the time discretization. In addition, there is a maximum of one mode transfer per node visit. As such, traversal of an arc requires at least 1 time step, so each path can include a maximum of |T| arcs and transfers. Therefore, there must be a finite number of combinations of arcs and times that can create a path $\pi_{i'i}^{m}(t)$ containing at most |T| arcs and transfers from node i to node D.

According to Lemma 1, every node j for predecessor node i and mode m with a finite cost label $\mathrm{q}_{ij}^{m}(t)$ at any stage has a corresponding path that leads to the destination node D for all time steps t. Let

$\pi_{j'j}^{x}(t) = \{(n_1=i,\ x_1=x,\ t_1^{0}=t),\ (n_1,\ x_2,\ t_1^{1}),\ (n_2,\ x_2,\ t_2^{0}),\ \ldots,\ (n_k=D,\ x_2=x_f,\ t_k^{1})\}$

such that the cost of the path $\pi_{j'j}^{x}(t)$ is

$\mathrm{q}_{ij}^{x}(t)=\Sigma_{\text{arcs,times in path ijx-D}}\ \phi_{arc}^{m}(t) + \Sigma_{\text{transfers,times in path ijx-D}}\ \kappa_{transfers}(t)$.

If $\pi_{j'j}^{x}(t)$ is not the least cost path, then there exists another path

$\pi_{j'j}^{x*}(t) = \{(n_1^{*}=j,\ x_1^{*}=x,\ t_1^{0*}=t),\ (n_1^{*},\ x_2^{*},\ t_1^{1*}),\ (n_2^{*},\ x_2^{*},\ t_2^{0*}),\ \ldots,\ (n_k^{*}=D,\ x_2^{*}=x_f,\ t_k^{1*})\}$

with $\mathrm{q}_{ij}^{x*}(t)=\Sigma_{\text{arcs,times in path* ijx-D}}\ \phi_{arc}^{m}(t) + \Sigma_{\text{transfers,times in path* ijx-D}}\ \kappa_{transfers}(t)$.

such that $\mathrm{q}_{ij}^{x*}(t) < \mathrm{q}_{ij}^{x}(t)$. $\mathrm{q}_{ij}^{x}(t)$ will then be updated as $\mathrm{q}_{ij}^{x}(t) = \mathrm{q}_{ij}^{x*}(t)$. Let the improvement in $\mathrm{q}_{ij}^{x}(t)$ be represented by $r_{ij}^{x}{}_{\text{path1 path2}} = \mathrm{q}_{ij}^{x*}(t) - \mathrm{q}_{ij}^{x}(t)$.

Since there are a finite number of combinations of paths $\pi_{j'j}^{x}(t)$, there are a finite number of possible values $\mathrm{q}_{ij}^{x}(t)$. Thus there are a finite number of possible reductions $r_{ij}^{x}{}_{\text{path1 path2}}$, where $r_{ij}^{x}{}_{\text{path1 path2}} > 0$ (otherwise, no update would occur).

Let r_{min} = min { $r_{ij}^{x}{}_{path1\ path2} > 0$ ∀ i ∀ path1 j to D ∀ path2 j to D }, such that r_{min} represents the smallest possible change in q $_{ij}^{x}(t)$ that can be made during a time step. Suppose that the SE list is not empty after a finite number of iterations. This means that at least one node is repeatedly inserted, which in turn implies that at least one label is infinitely improved by at least r_{min} at each time step. This eventually leads to negative cost labels q $_{ij}^{x}(t)$, which contradicts that all cost labels are non-negative. ð

Lemma 3: Upon termination, the following relation holds for every label q $_{i}(t)$*:* $q_{i,j}^{m2}(t) \le \left\{k_{i,j,k}^{m2,m1}(t)+ f_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)\right)+q_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)+t_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)\right)\right)\right\}$ " k Î {G*(j)*}, t Î T

Proof:

The proof of this lemma derives directly from Step 2 of the algorithm. Suppose there exists a node k∈{Γ(j)} such that

$$q_{i,j}^{m2}(t) > \left\{k_{i,j,k}^{m2,m1}(t)+ f_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)\right)+q_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)+t_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)\right)\right)\right\} \text{ " } k \text{ Î } \{\text{G}(j)\},\ t \text{ Î } T$$

then the algorithm does not terminate, since node j has not been scanned. Thus node j is still in the SE list. This contradicts the statement that the algorithm has terminated. ð

Next, proof of the algorithm's correctness is proven based on the proofs of the three lemmas.

Theorem 1: Upon termination of the algorithm, every cost label q $_{ij}^{m}(t)$ *is either an infinite number, meaning that no path exists from this node to the destination node at this time step, or a finite number that represents the cost of the least-cost path from this node for this mode predecessor node and time interval combination to the destination node D and exit mode* m_f*.*

Proof:

According to Lemma 1, every node j with a finite cost label q $_{ij}^{m}(t)$ at any stage has a corresponding path that leads to the destination node D for all time steps t. Let

$\pi_{jj'}^{x}(t) = \{(n_1=j, x_1=x, t_1^0=t), (n_1, x_2, t_1^1), (n_2, x_2, t_2^0), \ldots, (n_k=D, x_2=x_f, t_k^1)\}$

which is the least-cost path that corresponds to the label q $_{ij}^{x}(t)$ upon termination of the algorithm. We prove by contradiction that $\pi_{jj'}^{x}(t)$ is the least-cost path from j to D at starting at time t on mode x at i.

Assume that $\pi_{jj'}^{x}(t)$ is not the least-cost path, then there exists another path

$\pi_{jj'}^{x*}(t) = \{(n_1^*=j, x_1^*=x, t_1^{0*}=t), (n_1^*, x_2^*, t_1^{1*}), (n_2^*, x_2^*, t_2^{0*}), \ldots, (n_k^*=D, x_2^*=x_f, t_k^{1*})\}$

which is the least-cost path. This would mean that

$$q_{i,j}^{m2}(t) > \left\{k_{i,j,n2*}^{m2,m1}(t)+ f_{j,n2*}^{m1}\left(t+x_{i,j,n2*}^{m2,m1}(t)\right)+q_{j,n2*}^{m1}\left(t+x_{i,j,n2*}^{m2,m1}(t)+t_{j,n2*}^{m1}\left(t+x_{i,j,n2*}^{m2,m1}(t)\right)\right)\right\} \text{ where } n_2^* \in \{\Gamma(i)\}.$$

The above relationship contradicts Lemma 3, which states upon termination there is no node j∈{Γ(i)} with a label such that

$$q_{i,j}^{m2}(t) > \left\{k_{i,j,k}^{m2,m1}(t)+ f_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)\right)+q_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)+t_{j,k}^{m1}\left(t+x_{i,j,k}^{m2,m1}(t)\right)\right)\right\}$$

Thus $\pi_{jj'}^{x}(t)$ is the least-cost path from j to D at time t starting on mode x at i with total travel time $\mathrm{q}_{ij}^{m}(t)$. ð

It is also important to note that the algorithm will not fall into an infinite cycle. This is proven as follows:

Theorem 2: Assuming that $\mathrm{t}_{ij}^{m}(t)>0$ *and* $\mathrm{x}_{ijk}^{m1,m2}(t)³0$*, it is impossible for infinite cycles to develop.*

Proof:

A cycle occurs when a path returns to a node that has already been included, such that the node sequence would appear as

$\ldots$-(i,x,t_1)-$\ldots$-(i,x,t_2)-$\ldots$

where $t_2 = t_1+\Sigma_{ij\in \text{arcs in cycle}}\tau_{ij}(t) + \Sigma_{\text{transfers in cycle}}\ \xi_{ijk}^{m1,m2}(t)$. Since $\Sigma_{ij\in \text{arcs in cycle}}\tau_{ij}(t) + \Sigma_{\text{transfers in cycle}}\ \xi_{ijk}^{m1,m2}(t)$. time elapses with each cycle, and since |T| is finite, the algorithm eventually runs out of time steps and the cycle ends.

In order for an infinite cycle to exist in a finite time period, a sequence such as $\ldots$-(i,x,t_1)-$\ldots$-(i,x,t_1)-$\ldots$must exist. This implies that $t_1=t_2$; however, $t_1=t_2$ contradicts $t_2 = t_1+\Sigma_{ij\in \text{arcs in cycle}}\tau_{ij}(t) + \Sigma_{\text{transfers in cycle}}\ \xi_{ijk}^{m1,m2}(t)$ where $\tau_{ij}(t)>0$, $\xi_{ijk}^{m1,m2}(t)\geq 0$. As such, assuming that $\tau_{ij}(t)>0$, $\xi_{ijk}^{m1,m2}(t)\geq 0$ it is impossible to have an infinite cycle. ð

3.3.4 Implementation of the TDIMCP in VISTA

The TDIMCP is implemented in the VISTA to extend the DTA procedure to solve the combined mode split and traffic assignment problem. Link travel and mode transfer times are obtained from the simulation and used in the path cost calculations. In addition, several sets of cost parameters can be defined to represent different behavior types and travel preferences. The TDIMCP is then repeated for each distinct cost function. Further description of the use of the intermodal routing algorithm in a combined dynamic mode split and traffic assignment algorithm is provided in Section 3.4.

3.4 Equilibrium Assignment Algorithm

The equilibrium assignment algorithm used in the intermodal time dependent trip assignment algorithm is fundamentally the same as that described in Section 2.4 for the vehicle assignment-based approach; however, since intermodal trips are incur several different types of costs, the trip cost used in the intermodal version is based on a generalized cost function instead of just travel time. Further, different different behavior types are captured by defining different cost functions, and the algorithm notation must be slightly modified to account for those different functions.

3.4.1 Notation

On a network G(V,A,T) where V is the set of nodes, A the set of arcs and [0,T] an assignment period, let d^{t}_{rsb} be the number of people of behavior type b generated at node r and destined to node s $(r,s\in V)$ at time $t\in[0,T]$. Let P be the set of all spatiotemporal paths from all origins to all destinations, i.e. $P=\{p^1, p^2, \ldots p^{\pi}\}$. Each path p^k, $1 \leq k \leq \pi$ belongs to a set P(r,s,t,b), which contains all paths departing at time t in [0,T] from node r to node $s \in V$.

We denote with

x^{p^k} the number of vehicles choosing to follow path p^k -- Ξ in vector notation

$\mathrm{y}^{p^k}(\Xi)$ the travel cost on path p^k -- Ψ(Ξ) in vector notation

The demand relationships $\sum_{p^k \in P(r,s,t,b)} \mathrm{x}^{p^k} = d_{rsb}^{t}$ form a closed, bounded, convex space D⊂R$^{\pi}$. As such, any assignment Ξ in D is feasible, given that the traffic flow propagation and person tranfer laws adopted allow all person trips to be completed within time T. Thus, given a Ξ in D, the travel costs, Ψ(Ξ), can be computed using a simulator to move vehicles through the network, along with additional mode transfer rules.

The space D is compact and Ψ(Ξ) further is assumed to be continuous; therefore, there is at least one solution to the above VI problem. Uniqueness is more difficult to show since it requires that the cost function Ψ(Ξ) is monotonic. In fact, network realities, such as signals and transfer delays prevent the cost function from being truly continuous. Further, due to temporal interactions of the vehicles assigned earlier with those assigned later, monotonicity may not hold. Lack of monotonicity precludes proof of solution uniqueness, as well as algorithm convergence. The effect of deviations from these assumptions on the algorithm's convergence will be explored from a practical standpoint through presentation of computational results.

3.4.2 Equilibrium Conditions

As for the vehicle assignment-base implementation described in Section 2.4, the intermodal implementation attempts to find the dynamic user equilibrium assignment, which can be defined as either (3.7) or (3.8). Derivations and interpretations of these relationships are provided in Section 2.4.2.

$$\Psi\left(\Xi^*\right)^T\left(\Xi - \Xi^*\right) \geq 0 \quad \forall \Xi \in D \tag{3.7}$$

$$\Xi^{*T}\left(\Psi\left(\Xi^*\right) - \Psi_{\min}\left(\Xi^*\right)\right) = 0 \tag{3.8}$$

3.4.3 Minimization of Gap Functions

As for the vehicle assignment-base implementation described in Section 2.4, the intermodal implmentation relies on conversion of the equilibrium conditions into gap functions, such that the optimum point of the gap functions corresponds to the equilibrium solution. Three different gap functions are defined for three different search phases. Further, for each gap function a descent direction is defined. The different phases take advantage of different convergence properties of each gap function and descent direction, and will thus be used in different stages of the search for the equilibrium solution. The gap functions and descent directions defined for each phase of the algorithm are detailed in Section 2.4.3.

3.4.4 Algorithm

The intermodal implementation of the IADUE assignment algorithm is fundamentally the same as that for the vehicle assignment-based version; however, for the intermodal version, least cost paths must be calculated for each behavior type, as differentiated by generalized cost functions.

Further, following each simulation, bus arrival times must be calculated along with link travel times, in order determine bus passenger travel times and transfer delays. The intermodal version of the IADUE assignment algorithm is shown in Figure 3-3. The detailed description of the algorithm procedure, as well as the golden section search, provided in Section 2.4.4 also apply to this intermodal version of the IADUE assignment algorithm.

```
Phase 0 – Initialize
Set n=0.
Set link travel time to free flow.
Calculate least cost paths for each rstb.
Set $\Xi_0$ to all-or-nothing assignment of $d^t_{rsb}$ to least cost path for rstb.

Phase 1 – Search in Extreme Point Direction
Simulate traffic conditions with assignment $\Xi_n$.
Update link travel times and bus arrival times.
If (n=0) or ((|path set|$_n$ - |path set|$_{n-1}$) > routing-stop-precentage*|path set|$_{n-1}$)
   Calculate least cost paths for each rst.
   Add new paths to path set.
Set descent direction (a-o-n assignment to least cost paths, extreme point $P_n$).
Select step length $\lambda_n = \text{argmin}_\lambda (V_{demand}((1-\lambda)\Xi_n + \lambda P_n))$ using golden section search.
Assign demand to $\Xi_{n+1} = (1-\lambda_n)\Xi_n + \lambda_n P_n$ .
If n <assignment-max-extremes
   Repeat Phase 1.
Else
   Go to Phase 2.
Set n=n+1.

Phase 2 – Search in Smith Direction
Simulate traffic conditions with assignment $\Xi_n$.
Update link travel times and bus arrival times.
Choose descent direction $\Delta_n$ .
Select step length $\lambda_n = \text{argmin}_\lambda (V_{smith}(\Xi_n + \lambda\Delta_n))$ using golden section search.
Assign demand to $\Xi_{n+1} = \Xi_n + \lambda_n\Delta_n$ .
If $V_{smith}(\Xi_{n+1}) / V_{smith}(\Xi_n) >$ assignment-switch-ratio
   Repeat Phase 2
Else
   Go to Phase 3
Set n=n+1

Phase 3 – Search in Non-extreme Direction
Simulate traffic conditions with assignment $\Xi_n$.
Update link travel times and bus arrival times.
Choose descent direction $X_n$.
Select step length $\lambda_n = \text{argmin}_\lambda (V_{cost}((1-\lambda)\Xi_n + \lambda P_n))$ using golden section search.
Assign demand to $\Xi_{n+1} = \Xi_n + \lambda X_n$ .
If ( $V_{cost}(\Xi_{n+1}) / \Sigma_{rstp}\psi_{rstp}(\Xi_{n+1}) >$ cost-gap-percentage)
```

```
   Repeat Phase 3.
   Set n=n+1.
Else
   Terminate.
```

Note:
The routing-stop-precentage, assignment-switch-threshold, assignment-switch-ratio and cost-gap-percentage are convergence parameters that can be set within the VISTA implementation of DTA-IADUE.

Figure 3-3: The DTA-IADUE algorithm for Intermodal Problems

3.4.5 Algorithm Convergence Issues

As with the vehicle assignment-based implementation described in Section 2.4, the intermodal version of the IADUE assignment algorithm approximates the cost function as continuous and monotonic, whereas real-world problems may not be consistent with these assumptions. For example, traffic signals and mode transfers lead to discontinuities in travel time costs, which suggest that an equilibrium solution may not exist; however, by shifting demand from higher cost paths to lower cost paths, the algorithm is expected find a solution that better approximates equilibrium than the random assignment of the traditionally used MSA approach.

Further, while the assumption that a path's cost increases with the number of vehicles loaded on that path may be rendered inaccurate by the effect of path interactions, in practice, preliminary results show reasonably close approximations to equilibrium. Further, it should be noted that while the rate of convergence is not guaranteed, it is certain, due to the nature of the line search, that the algorithm will not diverge.

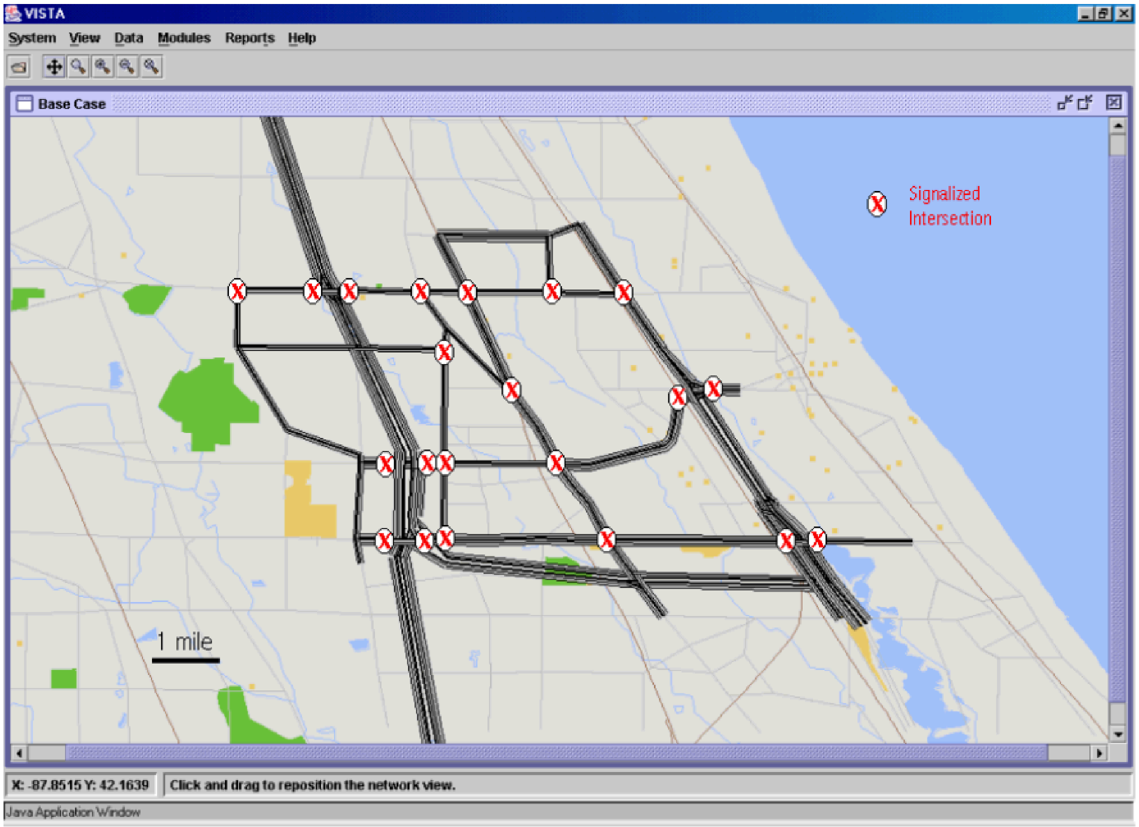
Moreover, as for the vehicle assignment-base implementation described in Section 2.4, the question of when to switch from phase to phase demands further attention. In other words, it remains unclear what values of **assignment-switch-threshold** and **assignment-switch-ratio** would provide the most efficient convergence, and how those values should be selected. Similarly, it remains unclear how the convergence of the line search (**assignment-stop-difference**, **assignment-previous-gap-ratio**, **assignment-worst-gap-ratio**) affects the progress of the following iterations.

3.5 Computational Results

The person assignment-based intermodal DTA model has been implemented in VISTA, and test results on a small test network of cars and buses are presented in this section. No tests on real-world networks are currently planned, since real-world tests will require detailed person trip data and disutility parameters, which may be difficult to obtain, calibrate and validate.

3.5.1 Description of the Test Network, Demand and Costs

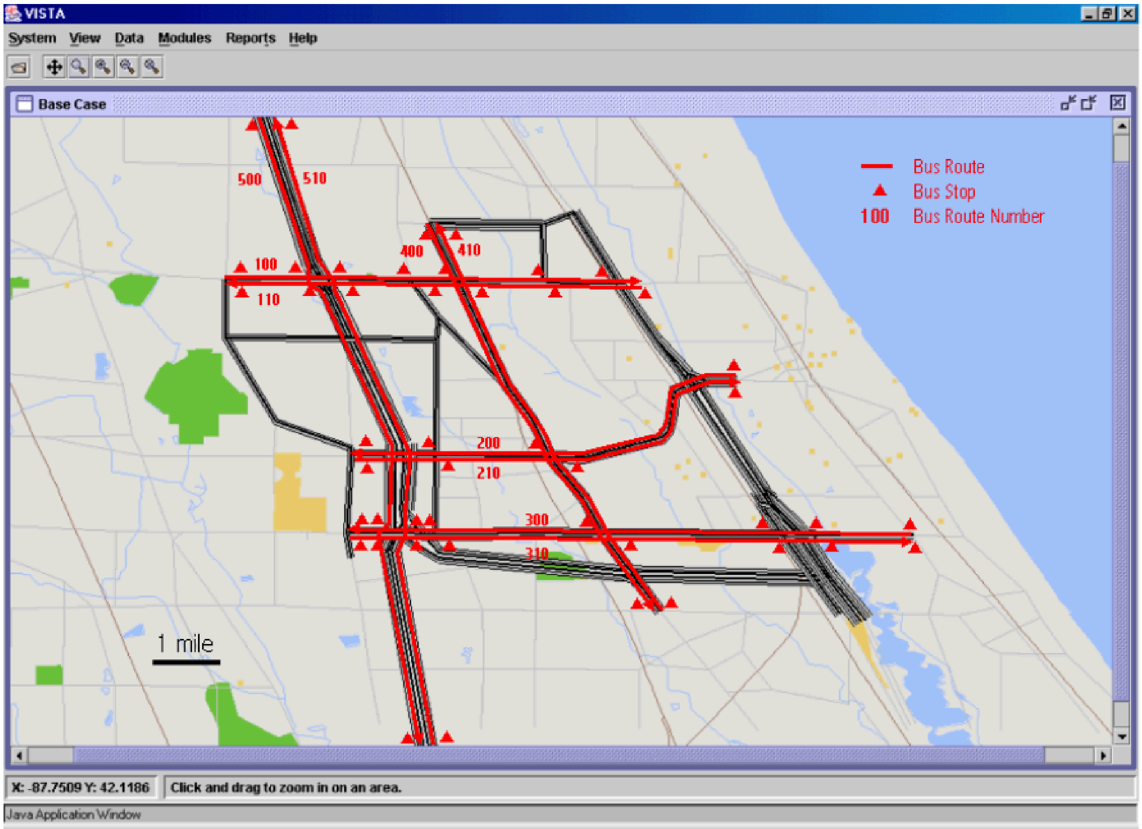
The intermodal test network, shown in Figure 3.4, includes 82 roadway intersection nodes and 169 roadway links. The network includes 21 signalized arterial and ramp intersections, as shown in Figure 3.4. In addition, the network is traversed by five bus routes, each traveling in two directions, and bus stops are connected the nearest roadway nodes by pedestrian links to allow for intermodal transfers. Trips are generated at 26 nodes in 19 zones, and parking areas are located at all trip ends, as well as at bus stop transfer locations.



X: -87.8515 Y: 42.1639

Click and drag to reposition the network view.

Figure 3-4: Signalized Intersections in the Intermodal Test Network



X: -87.7509 Y: 42.1186

Click and drag to zoom in on an area.

Figure 3-5: Intermodal Test Network

A total demand of 17,996 person trips are loaded into the network in the first 2 hours of a 3-hour simulation period. Traveler costs are calculated according to Equation (3.9), where the β's represent cost parameters for automobile travel time, bus travel time, bus out-of-vehicle travel time (ovtt), and pedestrian travel time. The x's represent travel times including automobile travel time, bus travel time, bus ovtt, which are determined from the simulation results, as well as pedestrian travel time, which is based on a fixed walking speed of 3 miles per hour. The y values represent fixed costs, including parking fees and bus fares. For the tests, parking fees have been set to $5 at all locations, and bus fares have been set to $1.5 with no discounts for transfers.

$$C = \beta_{\text{auto tt}} x_{\text{auto tt}} + \beta_{\text{bus tt}} x_{\text{bus tt}} + \beta_{\text{bus ovtt}} x_{\text{bus ovtt}} + \beta_{\text{ped tt}} x_{\text{ped tt}} + y_{\text{parking}} + y_{\text{bus fare}} \qquad (3.9)$$

For the test case, cost parameters were defined for two behavior types, where within each behavior group the population is assumed to be homogeneous in their travel preferences. The cost parameters for each group are listed in Table 4.1. According to the parameters, travelers of behavior type 1 prefer automobile travel over bus travel, and find bus waiting time more onerous and bus travel time, and walking time more onerous than bus waiting time. Travelers of behavior type 1 will choose the automobile mode over bus travel, except where the parking fee makes bus travel less costly. By contrast, travelers of behavior type 2 find automobile travel time, bus travel time and bus waiting time equally onerous, and further find walking time slightly more onerous than the other modes. These travelers are more likely to choose the bus mode than travelers of behavior type 1. Behavior type 1 included 11,242 person trips between 209 origin-destination pairs, and behavior type 2 included 6,754 person trips between 139 origin-destination pairs.

Table 4-1: Travel Cost Parameters for Intermodal Tests

Parameter	Behavior Type 1	Behavior Type 2
$\beta_{\text{auto tt}}$ ($/hr)	$5.04	$5.40
$\beta_{\text{bus tt}}$ ($/hr)	$5.40	$5.40
$\beta_{\text{bus ovtt}}$ ($/hr)	$5.76	$5.40
$\beta_{\text{ped tt}}$ ($/hr)	$7.20	$5.76

The computational time and convergence of the algorithm are discussed for testing on the base case network and demand described in this section. Further, results of a test with signal priority are discussed to show how the intermodal model can be used to evaluate the impacts transit signal priority.

3.5.2 Computational Time and Convergence of the Algorithm

The total computational time required for the base case was 01:30:47. Table 3-1 shows a breakdown of the number of iterations, the number of simulations required, and the computational time required for each phase. The table shows that the most time consuming portion of the algorithm was the path generation portion of Phase 1. Phase 2, was not performed because the algorithm found that the demand gap at the end of Phase 1 was 0, thus producing a 0 value in the denominator of calculation of the Smith direction, Equation (2.43).

As with the vehicle assignment-based implementation of IADUE, in each phase, each iteration includes a line search during which several step lengths are tested. Further, for each step length test, a simulation run is performed. With the line search parameters set to the values shown in Table 3-2, each line search required 3-8 simulations, where each simulation required an average of 15 seconds.

Table 3-1: Computational Time for Intermodal Test Run

Phase	Number of Iterations	Number of Simulations	Computational Time
1 – path generation	7	31	00:53:03
1 – no path generation	1	4	00:04:51
2	--	--	--
3	3	26	00:21:53
Total	11	61	01:20:47

Table 3-2: Line Search Parameters

Parameter	Value
Assignment-search-low	0.38
Assignment-search-high	0.62
Assignment-stop-difference	0.05
Assignment-previous-gap-ratio	0.99
Assignment-worst-gap-ratio	0.05

Outside of simulation, the remainder of the computational time is devoted to updating of link and path travel times following each simulation, calculation of assignment patterns and calculation of time dependent shortest paths. Calculation of time dependent shortest paths was also a relatively time-consuming task, requiring about 2.5 minutes for each path generation.

In Phase 1, the routing-stop-percentage was set to 1%, such that when the number of new paths generated was less than 1% of the total number of paths in the path set, the time dependent shortest path calculation was no longer invoked for subsequent iterations. Figure 3-6 shows the growth of the path set through the seven iterations of Phase 1 during which paths were generated. As with the vehicle assignment-based implementation of the IADUE algorithm, many paths were added in the early iterations, but the growth of the path set leveled off in later iterations. The final size of the path set was 526 for 212 origin-destination pairs.

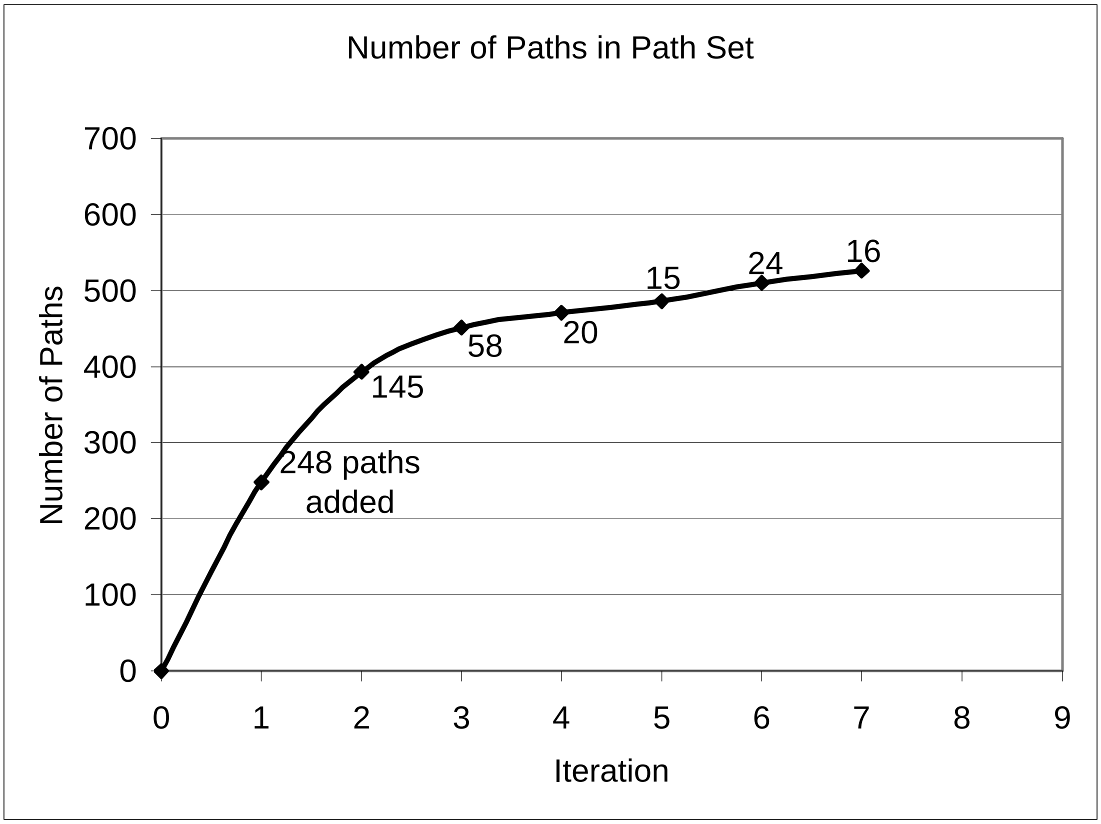


Figure 3-6: Growth of Path Set

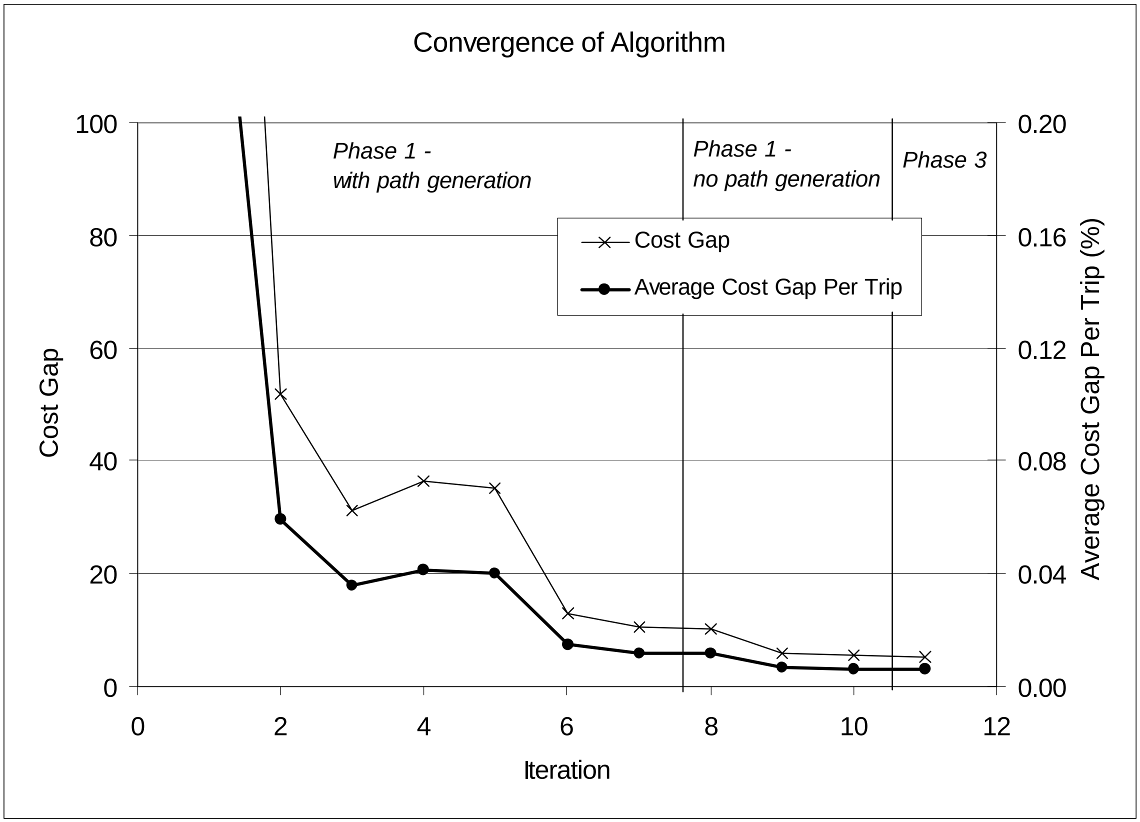


Figure 3-7: Convergence of the IADUE algorithm on the Intermodal Test Netork

The convergence of the algorithm is shown in Figure 3-7 in terms of the total cost gap for the network, as well as the average cost gap per trip. The plot shows that the IADUE algorithm found a path assignment that reduced the cost gap to $5.25 for the system, or 0.0062% per trip.

As previously mentioned, Phase 2 was not performed because the algorithm found that the demand gap at the end of Phase 1 was 0, thus producing a 0 value in the denominator of calculation of the Smith direction, Equation (2.43). While the demand gap value was calculated to be 0, the cost gap at the end of Phase 1 was in fact $5.42 for the system. In other words, the demand gap of 0 suggested that an equilibrium solution had been found; however, the cost gap indicated that some travelers were still assigned to non-equilibrium paths. This discrepancy occurs because the demand gap is derived from an equilibrium formulation that, as shown in Equation (2.20), allows some travelers to choose more costly routes, as long as the total system route cost is reduced. By contrast, the cost gap includes any costs beyond the minimum possible path cost for an origin, destination, departure time and behavior type.

3.5.4 Base Case Results

The IADUE algorithm estimated the mode split shown in Table 3-3. The table shows that of travelers of behavior type 1, about ten times more chose the automobile mode than the bus mode.

Of travelers of behavior type 2, about four times more chose the automobile mode than the bus mode. No intermodal automobile-bus trips, nor purely pedestrian trips were selected.

Table 3-3: Base Case Mode Split

	Behavior Type 1	Behavior Type 2	All
Automobile Trips	10,176	4,422	15,598
Bus Trips	1,066	1,332	2,398

The travel pattern resulted in a total system cost of $87,962. Travel costs and travel times for the system are shown in Table 3-4. The average trip cost for travelers of behavior type 1 was $4.96. This was slightly higher than the average trip cost of $4.76 for travelers of behavior type 2. However, the average person trip travel time was much lower for behavior type 1, at 3.6 minutes, than the average person trip travel time of 5.1 minutes for behavior type 2.

In general, automobile trips are associated with higher costs, since each automobile trip ends with a $5 parking fee, and since the network small, the travel time costs are relatively small compared with that fee. Also, bus trips are in general longer in travel time than automobile trips, since bus trips include a walk portion to reach the bus stop, as well as an out-of-vehicle waiting delay. As such, it is not surprising that travelers of behavior type 2 experience lower trip costs and higher travel times than travelers of behavior type 1, since type 2 travelers are more likely to select the bus mode than type 1 travelers.

The total vehicle travel time for the network, including bus and automobile travel times, was found to be 7,349.4 hours, and the total vehicle miles traveled for the system was 118,776 miles. These values may be used to provide an indication of air quality impacts of traffic. Further, the average vehicle speed for the network was 16.2 mph.

Table 3-4: Base Case Travel Costs and Travel Times

	Behavior Type 1	Behavior Type 2	All
Average Cost ($)	4.96	4.76	4.89
Average Person Travel Time (min)	3.6	5.1	4.1
Vehicle Travel Time (hours, bus + automobile)	--	--	7,349.4
Vehicle Miles Traveled (bus + automobile)	--	--	118,776
Average Vehicle Speed (mph, bus + automobile)	--	--	16.2

3.5.5 Evaluation of Transit Signal Priority

In addition to these system-level cost and travel time statistics, the person assignment-based intermodal DTA model also provides more detailed insights into travel patterns and impacts by user classes. Results from a signal priority test case are presented and compared with the base case results to show some ways in which the model may be used to examine the impacts of transit policies, such as TSP.

For the test, signal priority was applied to the intersection signals along bus routes 100/110 and 400/410. The estimated mode split for the TSP case is shown in Table 3-5. The table shows that overall, the number of bus trips increases by almost 600. A more detailed look at the ridership for each bus route, shown in Table 3.6, indicates that most of the gain is on the priority routes. This suggests that the priority treatment did indeed have an impact on ridership.

Table 3-5: TSP Case Mode Split

	Behavior Type 1	Behavior Type 2	All
Automobile Trips	9,974	5,048	15,022
Bus Trips	1,268	1,706	2,974

Table 3-6: Comparison of Bus Route Ridership

Route	Base Case	TSP Case

100*	139	243
110*	84	135
200	221	218
210	49	49
300	73	76
310	16	15
400*	638	763
410*	192	240
500	673	677
510	556	558

* Route with signal priority

The travel pattern resulted in a total system cost of $87,737. Travel costs and travel times for the system are shown in Table 3-7. Since more bus trips were made in the TSP case than in the base case, in general costs were lower and travel times were higher for both behavior types in the TSP case. Further, the total vehicle travel time and miles traveled for the network, including bus and automobile movements, was found to be less than in the base case, which is consistent with the shift of trips from automobile to bus. The reduction in automobile movements resulted in an increase in the average speed on the network to 19.1 mph.

Table 3-7: TSP Case Travel Costs and Travel Times

	Behavior Type 1	Behavior Type 2	All
Average Cost ($)	4.94	4.70	4.86
Average Person Travel Time (min)	4.0	5.4	4.5
Vehicle Travel Time (hours, bus + automobile)	--	--	6,221.4
Vehicle Miles Traveled (bus + automobile)	--	--	118,560
Average Vehicle Speed (mph, bus + automobile)	--	--	19.1

3.6 Summary

A person assignment-based multimodal approach DTA model has been proposed for implemention within the framework of Northwestern University's VISTA DTA software. This approach assigns person trips to intermodal equilibrium least cost paths. As with the automobile assignment-based multimodal VISTA model, the person assignment-based approach iterates between traffic simulation, path calculations, and network assignment; however, the path calculation finds the least cost intermodal path, where costs are defined to reflect traveler preferences regarding travel time, fares, parking costs and other factors. The intermodal least cost path algorithm used for this problem is based on Yang's (7) time dependent intermodal minimum cost path (TDIMCP) algorithm. Proofs of convergence and correctness, not previously presented by Yang, were developed.

The interactions of cars and buses in the shared roadway network are captured by the RouteSim simulator, which propagates vehicles according to cell transmission model logic. Further, turning movements, signalized intersections and transit signal priority logic are included in the simulator. The simulator also captures the added length of bus vehicles, as well as their frequent stopping behavior. Since person movements are not captured by RouteSim, dwell times are defined exogenously, and do not reflect the number of boardings and alightings.

The IADUE assignment algorithm proposed to find the equilibrium path assignment for the automobile assignment-based problem is also applied to the person assignment-based intermodal problem. The algorithm assumes continuous and monotonic cost functions; however, in reality, traffic signals and mode transfers may result in discontinuous travel time costs, and path interactions may cause path costs to be non-monotonic. Despite the problems with these assumptions, the algorithm is expected find a solution that better approximates equilibrium than the random assignment of the traditionally used MSA approach. Further work is study is required to determine how the algorithm’s convergence paramters affect the overall computational time required.

Computational results on a test network were shown to provide an indication of the computational requirements and convergence of the algorithm. In addition, tests were performed to with a TSP implementation to explore how the model may be used for analysis of transit policies. No tests on real-world networks are currently planned, since real-world tests will require detailed person trip data and disutility parameters, which may be difficult to obtain, calibrate and validate.

In short, this approach captures interactions between cars and buses in the simulator, as well as intermodal route choices, such that bus travel times and travel time variability can be observed in the simulator output. These measures can be used to draw conclusions about operating efficiency and asset management. Further, since mode choice is determined within the model, the impacts of transit policies on ridership are captured by the model and can be directly observed in the model output.

4.0 Analytical Formulation of the Dynamic Combined Model

The approaches described in Sections 2 and 3 can be considered simulation-based DTA approaches since they rely on a traffic simulation module to capture vehicle movements. In this section, we present an analytical formulation of the dynamic mode split and traffic assignment problem.

4.1 Network Definition

The multi-modal network is represented by separate sub-networks, as shown in Figure 4-1. One subnetwork contains the source cells where trips are generated, one contains the sink cells where trips are completed, and one contains the automobile mode's roadway cells. Further, each bus run is assigned a separate sub-network. As a result, there will be at most one bus on each bus subnetwork, and a bus route may be associated with several subnetworks. This type of definition allows the model to track the number of people on each bus vehicle (or run) by tracking the number of people on that subnetwork.

Each subnetwork includes cells that can be accessed by the given mode or bus run. A specific cell is defined by its subnetwork and cell location. A cell location represents a roadway segment that can be traveled by several modes, and that cell location will thus be included in the appropriate sub-networks. Transfer cells join subnetworks at specific cell locations. The last cell of each bus run is an imaginary cell that holds all out-of-service buses.

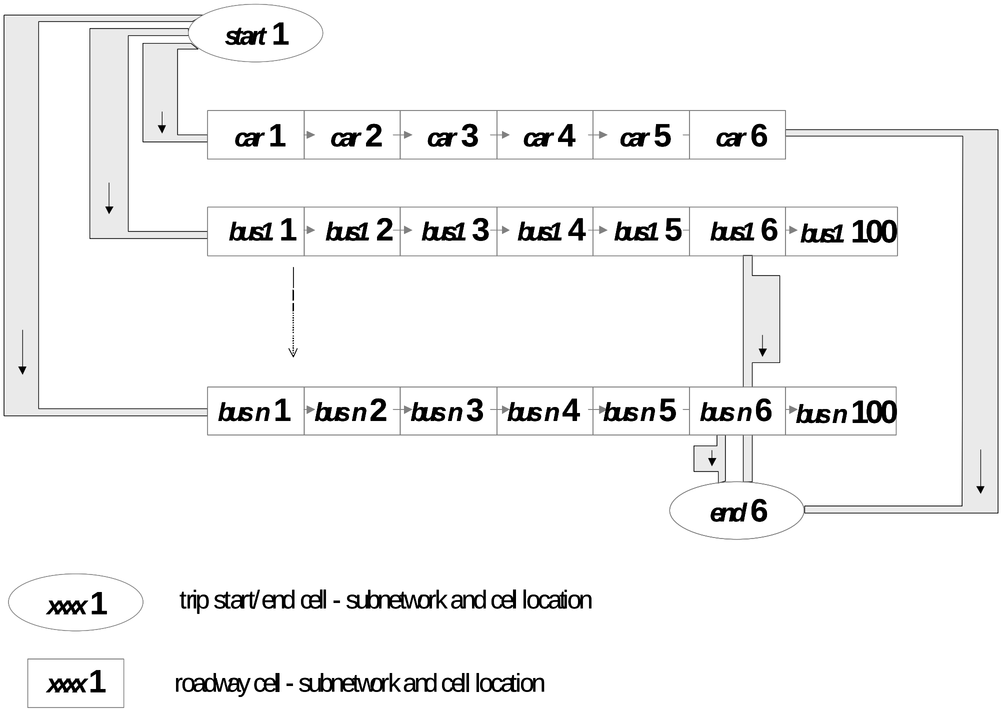


Figure 4-1: Subnetworks in the analytical approach

Demand is generated at the source cells and people enter the network through the network entry cells. There is no movement allowed among cells in the network entry subnetwork, so the only option is to transfer to another subnetwork. From the network entry subnetwork, people may enter the car or bus stop subnetworks. From the bus stop subnetwork, the people may board any bus that stops at that cell. A limit of 1 person per car is assumed.

As with the simulation-based approach, the model developed in this section deals specifically with cars and buses; however, the model can be extended include other modes. For example, commuter rail can be added by adding a rail subnetwork. Because this mode does not share the roadway nor interact with car and bus movements, the cell locations would not coincide with roadway cell locations, and rail vehicles would not be governed by the congestion rules of cell transmission. Instead, propagation of rail vehicles can simply be based on the rail schedule.

4.2 System Optimal Mathematical Programming Formulation

The model presented in this section calculates the intermodal flow pattern that minimizes the total system cost, given network and cost definitions, and demand from a set of origins to a single destination. The network is defined as shown in Section 4.1, and the sets are defined in Table 4-1.

The model is formulated using the cell transmission approach *(14)* to govern vehicle movements, including both buses and cars, and may thus be considered an intermodal version of the cell transmission-based DTA model presented by Ziliaskopoulos *(15)*. The cell transmission relationships regulate vehicle movements according to roadway flow and density limits, and thus capture congestion and queuing effects. Person transfer movements are also handled by cell occupancy and flow equations. Under this representation, transfer delay time is accrued as a traveler is left in a transfer cell awaiting the arrival of a bus. Since the model formulated considers only automobile and bus modes, and does not account for pedestrian walking movements, no pedestrian propagation rules were incorporated.

The travel pattern is calculated in terms of both vehicle and people movements. In this model, there is no pedestrian subnetwork, so it is assumed that all movement through cells occurs due to vehicle movement. As such, the movement of people through the car subnetwork is tracked by the number and flow of cars in each cell, $x^t_{m\ i}$ and $y^t_{m\ ij}$, where each car is assumed to contain only one traveler. The movement of people through a bus subnetwork is also represented by the variables, $x^t_{m\ i}$ and $y^t_{m\ ij}$, however, in this case, the number of people on each bus is represented by n^t_m, where only one bus travels on each bus subnetwork. The number of people at each entry and exit cell is represented by $n^t_{m\ i}$. Transfer cells that precede bus boardings may also be occupied by people, where the occupancy is represented by $n^t_{m1m2\ i}$. Time spent in a transfer cell represents transfer delay time, resulting from waiting for a bus. It is assumed that there is no delay in transferring to a car nor to the exit subnetwork. Further, flow into the transfer cell is represented by $f^t_{m1m2\ \ i}$ and flow out of the transfer cell by $g^t_{m1m2\ \ i}$. The person and vehicle occupancy and flow variables are summarized in Table 4-2.

Table 4-1: List of Sets

Set	Description
L	set of possible cell locations: source (L_R), sink (L_S), transfer (L_T), bus stop boarding ($L_X \subset L_T$), bus stop on route m ($L_{Xm} \subset L_X$)
M	set of subnetworks: network entry (M_R), network exit (M_S), car (M_C), bus runs ($M_B = \{M_{B1}..\ M_{Bn}\}$ for n bus runs)
E	set of vehicle cell connectors: connectors in car subnetwork (E_C), connectors in bus run subnetworks ($E_B = \{E_{B1}..\ E_{Bn}\}$ for n bus runs)
J	transfer subnetwork ($J_T = \{(M_R,M_C), (M_R,M_B), (M_B,M_B), (M_B,M_C), (M_C,M_B), (M_C,M_S), (M_B,M_S)\}$), bus stop subnetwork ($J_X \subset J_T$, $J_X = \{(M_R,M_B), (M_B,M_B), (M_C,M_B)\}$)
$\Gamma(i)$	set of locations of successor cells to cell at location I
$\Gamma^{-1}(i)$	set at location of predecessor cells to cell at location I
T	set of discrete time intervals

Several binary variables are used to indicate bus stoppage status. A positive value of μ^t_{mi} indicates that bus m is moving. Conversely, a positive value of ν^t_{mi} indicates that bus m is stopped, so people are permitted to board and alight. A positive value ψ^t_i indicates that at least one bus is stopped in cell i at time t, and thus that there is a lane blocked. These variables are summarized in Table 4-2.

The parameters used in the model are described in Table 4-3. The model takes as parameters limits on flows and occupancies in roadway and transfer cells to govern propagation of vehicles and people through the network. In addition, cost coefficients can be defined to reflect traveler choice behavior. Also, a demand parameter is included to indicate the number of trips generated in the network.

Table 4-2: List of Variables

Variable	Description
$n^t_{m\,i}$	number of people in cell on subnetwork m at location i at time interval t (entry, exit subnetwork)
n^t_m	number of people on subnetwork m at time interval t (bus subnetwork)
$n^t_{m1m2\,i}$	number of people in the transfer cell from subnetwork m1 to subnetwork m2 at location i at time interval t (transfers)
$f^t_{m1m2\ \ i}$	flow of people from subnetwork m1 to the transfer cell between subnetworks m1 and m2 at location i at time interval t
$g^t_{m1m2\ \ i}$	flow of people from the transfer cell between subnetworks m1 and m2 to subnetwork m2 at location i at time interval t
$x^t_{m\,i}$	number of vehicles moving into cell on subnetwork m at location i at time interval t (n people)
$y^t_{m\,ij}$	number of vehicles moving from cell at location i to cell at location j on subnetwork m at time interval t
μ^t_{mi}	0/1 variable indicating whether or not bus m is moving from cell i at time t (0 if people are boarding/alighting)
ν^t_{mi}	0/1 variable indicating whether or not bus m is stopped in cell i at time t (1 if bus is stopped and people are permitted to board/alight)
ψ^t_i	0/1 variable indicating whether or not a bus is stopped at location i at time t (1 if a bus is stopped and blocking a lane)

Table 4-3: List of Parameters

Parameter	Description
N^P_m	maximum number of people per vehicle in subnetwork m
Q^p_m	maximum flow of people to and from subnetwork m at transfer cells
N^V_i	maximum number of vehicles in roadway cell at location i
$N^V_{i\ lane}$	number of vehicles that can be held in one lane of roadway cell at i
Q^V_i	maximum flow of vehicles into or out of roadway cell at location i
$Q^V_{i\ lane}$	flow capacity of lane in which buses stop
q	vehicle link flow
q_{max}	vehicle maximum flow
k	vehicle density
k_j	vehicle jam density
v	vehicle link free flow speed
w	vehicle backward propagation speed
δ^t_i	ratio v/w for cell at location i at time interval t
τ	discretization time interval
ρ_m	car equivalent coefficient for vehicles on subnetwork m
α_m	travel time cost coefficient for travel on subnetwork m
β_{m1m2}	transfer time cost coefficient for transfers from subnetwork m1 to subnetwork m2
γ_{m1m2}	fixed transfer cost for transfers from subnetwork m1 to subnetwork m2 at cell location i
d^t_i	number of people trips generated at location i at time interval t
η_m	cost of bus travel time (forces a bus to move, higher value prevents holding for passengers, but a value that is too high will prevent the bus from stopping at all)

The system cost includes travel time costs, transfer delay costs and fixed transfer costs. The objective function including these costs is shown in (4.1). The objective function also includes a bus vehicle travel time cost, η_m, which penalizes the holding a bus to wait for passengers; however, if η_m is set very high, then the bus will not stop at all, even if people are waiting at the bus stop. The cost parameters must be balanced to appropriately reflect the expected behavior.

The balance constraints of people at source and sink cells are shown in constraints (4.2) and (4.3). The vehicle balances for cars not at transfer locations and cars at transfer locations are shown in constraints (4.4) and (4.5), respectively. At transfer locations cars are automatically generated and removed as people transfer to and from the car subnetwork. The bus vehicle balance constraint is shown in constraint (4.6), where the bus is generated by constraint (4.7). Constraint (4.8) indicates that no buses can be present on their respective subnetworks before the start times of their runs, and constraint (4.9) states that after the start time of a bus run a that bus run’s subnetwork can only contain a single bus.

Constraints (4.10) to (4.13) describe the bus stopping behavior. Constraint (4.10) indicates that the bus must be considered either moving or stopped, and it cannot be both nor neither. Constraint (4.11) indicates that if the bus is not stopped then it must cross a cell boundary, otherwise, it will be considered a blockage by constraints (4.15) to (4.17). Constraint (4.11) does not consider the possibility that if the bus is stopped due to queuing then it may not cross a cell boundary; however, in reality a queued bus should not be considered a lane blockage. On the other hand, in cases where the model finds that the optimal solution may include holding a bus in a cell, the bus should indeed be considered a blockage. As such, the model assumes a reduced cell capacity whenever a bus does not cross a cell boundary. Constraint (4.12) indicates that a bus can only stop at bus stops to allow boarding and alighting. Constraint (4.13) indicates that buses must be stopped when people are boarding or alighting.

Cars and buses both travel on the same physical road network, so cell transmission relationships apply to the combined movements of buses and vehicles. Constraints (4.14) to (4.17) are derived from cell transmission relationships. Constraint (4.14) indicates that the flow out of a cell is limited by the number of vehicles originally present in that cell. Constraints (4.15) and (4.16) indicate that the flow out of and into a cell are limited by the maximum flow across the boundaries of that cell, where the maximum flow is reduced when a bus is stopped and blocking a lane. Further, the car equivalence factor is included to account for the added length of buses. Constraint (4.17) indicates that the flow into a cell is limited by the space available in the downstream cell. The space available is reduced when a bus is stopped and blocking a lane. Again, the car equivalence factor is included to account for the added length of buses.

Constraints (4.18) and (4.20) define the balance of people at buses and at various transfers. Constraints (4.19) and (4.21) indicate that transfers to cars and exit cells are performed without delay. In other words, the transfer cell occupancy is 0, and thus the flow into the transfer cell is immediately transferred to the flow out of the transfer cell.

Constraint (4.22) limits the bus occupancy to a predefined maximum vehicle capacity. Constraint (4.23) limits the flow out of a transfer cell to the number of people originally in that cell. Constraint (4.24) limits the flow from a bus to a bus stop to the number of people originally on the bus. Constraint (4.25) limits the flow from the car subnetwork to a transfer cell to the number of cars originally in the cell at the transfer location. Constraint (4.26) limits the flow from the entry transfer cells to the number of people originally in the entry cell. Constraint (4.27) limits the speed of boarding and alighting to a predefined maximum person flow, where boarding and alighting can only occur if the bus is stopped at that location and time. Constraint (4.28) indicates that if a bus is stopped for boarding and alighting, then the corresponding lane is blocked. Constraints (4.29) and (4.30) limit the person flow for transfers other than boarding and alighting movements.

Constraints (4.31) to (4.36) limit the flow and occupancy variables to be non-negative integers, thus ensuring that people and vehicles move as whole entities. Without the integrality conditions, a person may be split into fractions, with some fractions taking one mode and some choosing another. Similarly, the integrality condition ensures that cars and buses move as complete vehicles, rather than for example, leaving half a bus at the beginning of its run while the other half travels through its route. The integrality of the variables makes the problem significantly more complex to solve than regular linear programs, but is required to ensure that people and vehicles move as whole entities. It is convenient that if the flow variables ($f^t_{m1\,m2\,i}$, $g^t_{m1\,m2\,i}$, $y^t_{m\,ij}$) are constrained to integer values, then the occupancy variables (n^t_m, $n^t_{m\,i}$, $n^t_{m1\,m2\,i}$, $x^t_{m\,i}$) will naturally take integer values, and vice versa. This fact results from the mass balance relationships between flow and occupancy, and can be used in implementation to reduce the number of integer variables in the integer program.

Constraints (4.37) to (4.39) limit the bus stop behavior variables to binary values to reflect the fact that buses must be considered either moving or stopped (for loading and unloading), and cannot take on any status in between. These binary variables add to the number of integer variables in the problem, and thus increase its complexity; however, that constraint (4.10) forces μ^t_{mi} and ν^t_{mi} to be inverses of each other, so as long as one is binary, then the other will also take on binary values (the second variable must be limited to a maximum value of 1). Further, as long as ν^t_{mi} is binary, then by constraint (4.28) ψ^t_i will also take on a binary value. As such, for implementation purposes, the complexity of the problem can be reduced by constraining only one of μ^t_{mi} or ν^t_{mi} to be binary, and the remainder of the bus stop behavior variables will also take on binary values.

Minimize

$$\begin{aligned}
&\Sigma_{t\in \mathsf{T}}\ \Sigma_{m\in \mathsf{M}_R}\ \Sigma_{i\in \mathsf{L}}\ (\alpha_m\, \tau\, n^t_{m\,i}) \\
&+ \Sigma_{t\in \mathsf{T}}\ \Sigma_{m\in \mathsf{M}_C}\ \Sigma_{i\in \mathsf{L}}\ (\alpha_m\, \tau\, x^t_{m\,i}) \\
&+ \Sigma_{t\in \mathsf{T}}\ \Sigma_{m\in \mathsf{M}_B}\ (\alpha_m\, \tau\, n^t_{m}) \\
&+ \Sigma_{t\in \mathsf{T}}\ \Sigma_{m1,m2\in \mathsf{J}_T}\ \Sigma_{i\in \mathsf{L}_T}\ (\gamma_{m1m2\,i}\ f^t_{m1m2\,i}) \\
&+ \Sigma_{t\in \mathsf{T}}\ \Sigma_{m1,m2\in \mathsf{J}_X}\ \Sigma_{i\in \mathsf{L}_X}\ (\beta_{m1\,m2}\, \tau\, n^t_{m1\,m2\,i}) \\
&+ \Sigma_{t\in \mathsf{T}}\ \Sigma_{m\in \mathsf{M}_B}\ \Sigma_{i\in \mathsf{L}\backslash \mathsf{L}_{mlast}}\ (\eta_m\, \tau\, x^t_{m\,i})
\end{aligned} \tag{4.1}$$

subject to

$$n^t_{m1\,i} - n^{t-1}_{m1\,i} + \Sigma_{m_2\in \{\mathsf{M}\backslash \mathsf{M}_R\}}\ f^{\,t-1}_{m1m2\,i} = d^{\,t-1}_{i} \qquad \forall m1\in \mathsf{M}_R\ \forall i\in \mathsf{L}_R\ \forall t\in \mathsf{T} \tag{4.2}$$

$$n^t_{m2\,i} - n^{t-1}_{m2\,i} - \Sigma_{m_1\in \{\mathsf{M}\backslash \mathsf{M}_S\}}\ g^{\,t-1}_{m1m2\,i} = 0 \qquad \forall m2\in \mathsf{M}_S\ \forall i\in \mathsf{L}_S\ \forall t\in \mathsf{T} \tag{4.3}$$

$$x^t_{m1\,i} - x^{t-1}_{m1\,i} - \Sigma_{j\in \Gamma^{-1}(i)}\ y^{t-1}_{m1\,ji} + \Sigma_{j\in \Gamma(i)}\ y^{t-1}_{m1\,ij} = 0 \qquad \forall m1\in \mathsf{M}_C\ \forall i\in \mathsf{L}\backslash \mathsf{L}_T\ \forall t\in \mathsf{T} \tag{4.4}$$

$$\begin{aligned}
&x^t_{m1\,i} - x^{t-1}_{m1\,i} - \Sigma_{j\in \Gamma^{-1}(i)}\ y^{t-1}_{m1\,ji} + \Sigma_{j\in \Gamma(i)}\ y^{t-1}_{m1\,ij} \qquad \forall m1\in \mathsf{M}_C\ \forall i\in \mathsf{L}_T\ \forall t\in \mathsf{T} \\
&\qquad + \Sigma_{m2\in \mathsf{M}\ \backslash\ \{\mathsf{M}_C\cup \mathsf{M}_S\}}\ g^{\,t-1}_{m2\,m1\,i} \\
&\qquad + \Sigma_{m2\in \mathsf{M}\ \backslash\ \{\mathsf{M}_C\cup \mathsf{M}_R\}}\ f^{\,t-1}_{m1\,m2\,i} = 0
\end{aligned} \tag{4.5}$$

$$x^t_{mi} - x^{t-1}_{mi} - y^{t-1}_{m\,j'i} + y^{t-1}_{m\,ij} = 0 \qquad \begin{aligned}&\forall m\in \mathsf{M}_B\ \forall i\in \mathsf{L}\ \ \forall j'i\in \mathsf{E}_B \\ &\forall ij\in \mathsf{E}_B\ \forall t\in \mathsf{T}, \\ &(i,t) \neq \text{bus start}\end{aligned} \tag{4.6}$$

$$x^t_{m\,i} = 1 \qquad \begin{aligned}&\forall m\in \mathsf{M}_B, \\ &(i,t) = \text{scheduled departure} \\ &\text{location and time of run m}\end{aligned} \tag{4.7}$$

$$\Sigma_{i\in \mathsf{L}}\ x^t_{m\,i} = 0 \qquad \begin{aligned}&\forall m\in \mathsf{M}_B \\ &\forall t\in \{\mathsf{T}\ \ : t < \text{runstart}\}\end{aligned} \tag{4.8}$$

$$\Sigma_{i\in \mathsf{L}}\ x^t_{m\,i} = 1 \qquad \begin{aligned}&\forall m\in \mathsf{M}_B \\ &\forall t\in \{\mathsf{T}\ \ : t \geq \text{runstart}\}\end{aligned} \tag{4.9}$$

$$\mu^t_{mi} + \nu^t_{mi} = 1 \qquad \forall m\in \mathsf{M}_B\ \forall i\in \mathsf{L}_B\ \forall t\in \mathsf{T} \tag{4.10}$$

$$\Sigma_{i\in L_B} \nu^t_{mi} + \Sigma_{(i,j)\in E_M} y^t_{m\,ij} = \Sigma_{i\in L}\, x^t_{m\,i} \qquad \forall m\in M_B\ \forall t\in T \tag{4.11}$$

$$\nu^t_{mi} = 0 \qquad \forall i\in L_B \setminus L_T\ \forall m\in M_B\ \forall t\in T \tag{4.12}$$

$$\Sigma_{ij\in E_M}\, y^t_{m\,ij} - \mu^t_m \leq 0 \qquad \forall m\in M_B\ \forall t\in T \tag{4.13}$$

$$\Sigma_{j\in \Gamma(i)}\, y^t_{m\,ij} - x^t_{mi} \leq 0 \qquad \begin{array}{l}\forall m\in \{M_C \cup M_B\}\\ \forall i\in L \setminus C_S\ \forall t\in T\end{array} \tag{4.14}$$

$$\Sigma_{m\in \{M_C\cup M_B\}}\, \Sigma_{j\in \Gamma(i)}\, \rho_m\, y^t_{m\,ij} \leq Q^V_i - (\psi^t_i\, Q^V_{i\,lane}) \qquad \forall i\in L \setminus L_S\ \forall t\in T \tag{4.15}$$

$$\Sigma_{m\in \{M_C\cup M_B\}}\, \Sigma_{i\in \Gamma^{-1}(j)}\, \rho_m\, y^t_{m\,ij} \leq Q^V_j - (\psi^t_j\, Q^V_{j\,lane}) \qquad \forall j\in L \setminus L_R\ \forall t\in T \tag{4.16}$$

$$\begin{array}{l}\Sigma_{m\in \{M_C\cup M_B\}}\, [\rho_m\, (\delta^t_j\, x^t_{mj} + \Sigma_{i\in \Gamma^{-1}(j)}\ y^t_{m\,ij})]\\ \qquad\qquad \leq \delta^t_j\, (N^V_j - \psi^t_j\, N^V_{j\,lane})\end{array} \qquad \forall j\in L \setminus L_R\ \forall t\in T \tag{4.17}$$

$$\begin{array}{l}n^t_{m1} - n^{t-1}_{m1} - \Sigma_{m2\in M}\, \Sigma_{i\in L_T}\, g^{\,t-1}_{m1\,m2\,i}\\ \qquad\qquad + \Sigma_{m2\in M}\, \Sigma_{i\in L_T}\, f^{\,t-1}_{m1\,m2\,i} = 0\end{array} \qquad \forall m1\in M_B\ \forall t\in T \tag{4.18}$$

$$f^{\,t}_{m1\,m2\,i} - g^t_{m1\,m2\,i} = 0 \qquad \begin{array}{l}\forall m1\in M\ \forall m2\in M_C\\ \forall i\in L_{Tm2}\ \forall t\in T\end{array} \tag{4.19}$$

$$n^t_{m1\,m2\,i} - n^{t-1}_{m1\,m2\,i} - f^{\,t-1}_{m1\,m2\,i} + g^{\,t-1}_{m1\,m2\,i} = 0 \qquad \begin{array}{l}\forall m1\in M\ \forall m2\in M_B\\ \forall i\in L_{Xm2}\ \forall t\in T\end{array} \tag{4.20}$$

$$f^{\,t}_{m1\,m2\,i} - g^{\,t}_{m1\,m2\,i} = 0 \qquad \begin{array}{l}\forall m1\in M\ \forall m2\in M_S\\ \forall i\in L_{Tm2}\ \forall t\in T\end{array} \tag{4.21}$$

$$n^t_m \leq N^P_m \qquad \forall m\in M_B\ \forall t\in T \tag{4.22}$$

$$g^{\,t}_{m1\,m2\,i} - n^t_{m1\,m2\,i} \leq 0 \qquad \begin{array}{l}\forall m1\in M\\ \forall m2\in M\{M_C\cup M_S\}\\ \forall i\in L_{Tm1}\ \forall t\in T\end{array} \tag{4.23}$$

$$\Sigma_{m2\in M} f^{t}_{m1\,m2\,i} - n^{t}_{m1} \leq 0 \qquad \forall m1\in M_{B}\ \forall i\in L_{Xm1}\ \forall t\in T \qquad (4.24)$$

$$\Sigma_{m2\in M} f^{t}_{m1\,m2\,i} - x^{t}_{m1\,i} \leq 0 \qquad \forall m1\in M_{C}\ \forall i\in L_{Tm1}\ \forall t\in T \qquad (4.25)$$

$$\Sigma_{m2\in M} f^{t}_{m1\,m2\,i} - n^{t}_{m1\,i} \leq 0 \qquad \forall m1\in M_{R}\ \forall i\in L_{Tm1}\ \forall t\in T \qquad (4.26)$$

$$f^{t}_{m1\,m2\,i} + g^{t}_{m3\,m1\,i} \leq \nu^{t}_{m1i}Q^{p}_{m1} \qquad \forall m1\in M_{B}\ \forall m2\in M\ \forall m3\in M\ \forall i\in L_{Tm1}\ \forall t\in T \qquad (4.27)$$

$$\psi^{t}_{i} - \max_{m\in M_{B}} \nu^{t}_{m\,i} = 0 \qquad \forall i\in L_{X}\ \forall t\in T \qquad (4.28)$$

$$f^{t}_{m1\,m2\,i} \leq Q^{p}_{m1} \qquad \forall m1\in M\ \forall m2\in M \setminus M_{B}\ \forall i\in L_{Tm1}\setminus L_{X}\ \forall t\in T \qquad (4.29)$$

$$g^{t}_{m1\,m2\,i} \leq Q^{p}_{m2} \qquad \forall m1\in M \setminus M_{B}\ \forall m2\in M\ \forall i\in L_{Tm2}\setminus L_{X}\ \forall t\in T \qquad (4.30)$$

$$n^{t}_{m\,i} \geq 0, \text{ integer} \qquad \forall m\in M_{R}\ \forall i\in L_{R}\ \forall t\in T \qquad (4.31)$$

$$n^{t}_{m} \geq 0, \text{ integer} \qquad \forall m\in \{M_{C}\cup M_{B}\}\ \forall t\in T \qquad (4.32)$$

$$n^{t}_{m1\,m2\,i} \geq 0, \text{ integer} \qquad \forall m1,m2\in J_{T}\ \forall i\in L_{T}\ \forall t\in T \qquad (4.33)$$

$$f^{t}_{m1m2\,i}\,,\ g^{t}_{m1m2\,i} \geq 0, \text{ integer} \qquad \forall m1,m2\in M\ \forall i\in L_{T}\ \forall t\in T \qquad (4.34)$$

$$x^{t}_{m\,i} \geq 0, \text{ integer} \qquad \forall m\in \{M_{C}\cup M_{B}\}\ \forall i\in L\ \forall t\in T \qquad (4.35)$$

$$y^{t}_{m\,ij} \geq 0, \text{ integer} \qquad \forall m\in \{M_{C}\cup M_{B}\}\ \forall i,j\in E\ \forall t\in T \qquad (4.36)$$

$$\mu^{t}_{mi} = \{0, 1\} \qquad \forall m\in M_{B}\ \forall i\in L_{Xm}\ \forall t\in T \qquad (4.37)$$

$$\nu^{t}_{mi} = \{0, 1\} \qquad \forall m \in M_B\ \forall i \in L_{Xm}\ \forall t \in T \qquad (4.38)$$

$$\psi^{t}_{i} = \{0, 1\} \qquad \forall i \in L_X\ \forall t \in T \qquad (4.39)$$

In brief, the model formulated assigns person trips to a single destination in an intermodal network of cars and buses. The cell transmission logic embedded in the formulation realistically captures time-varying congestion and queuing phenomena. Further, bus stopping and dwelling behavior are captured with dwell times determined by boardings and alightings and roadway capacity reduced for the duration of the stop. The model calculates the system optimal travel pattern based on a generalized cost function that applies different weights to travel time, transfer delays and fixed costs of each mode. In addition, a bus vehicle travel time cost is included to ensure that even empty buses will incur cost, and will thus have an incentive to progress along their routes to the cost-free bus run termination cell. Further, the formulation includes many integer variables; however, for implementation, the number of integer variables can be reduced by taking advantage of the relationships between variables. Despite this, the formulation is likely too complex for modeling realistic large-scale networks, and is proposed here only to obtain insights into the behavior of an analytical formulation of the intermodal trip assignment problem. The behavior of the model and computational results are discussed in the next section.

4.3 Computational Results

The single destination system optimal dynamic combined mode split and traffic assignment model formulated in Section 4.2 was programmed in AMPL and tests were performed to observe the behavior of the model with varied cost parameters and congestion conditions. Tests were performed on a simple car and bus roadway of six cells. Demand enters at cell 1 and exits at cell 6, and travelers can transfer between the car and bus subnetworks at cell 4. The bus travels from cell 1 to 6, and completes its route by entering the bus storage cell, labeled cell 100. Demand levels, roadway capacity and cost parameters are varied for each test.

The first test, with results shown in Figure 4.2, demonstrates the propagation of cars through the network. The figure shows the occupancies of the roadway and transfer cells for the test period of 25 time steps. Ten people were entered into the network at time step 2, and the system optimal model assigned all demand was assigned of the demand to the car network. The vehicles move through the network one cell at each time step, and at the end of the roadway the travelers transfer from cell 6 to the network exit cell. In addition, a bus is dispatched at time 4, and it travels empty through the six cells and then completes its route at cell 100.

The travel time costs were set to α_{car}=0.1 cost units/time step and α_{bus}=0.3 cost units/time step. Further, a penalty of α_{entry}=10 cost units/time step was defined to prevent travelers from remaining in the entry cell. A transfer time cost of 0.5 cost units/time step was assigned for bus stop waiting time, and bus fare was set to 0.5 cost units. No parking fee was defined. Also, bus vehicle travel cost was set to 1 cost unit/time step to prevent the bus from being held. Maximum flow and jam density were set to infinity, so that flow is uncongested. The resulting system cost for this test was 112 cost units.

Time Step	Entry Cell	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Exit Cell	Cell 100 (bus only)	Bus Stop Cell 1	Bus Stop Cell 4	Bus Stop Cell 6
0												
1												
2	10 people											
3		10 cars										
4		1 bus (bus empty)	10 cars									
5			1 bus (bus empty)	10 cars								
6				1 bus (bus empty)	10 cars							
7					1 bus (bus empty)	10 cars						
8						1 bus (bus empty)	10 cars					
9							1 bus (bus empty)	10 people				
10								10 people	1 bus			
11								10 people	1 bus			
12								10 people	1 bus			
13								10 people	1 bus			
14								10 people	1 bus			
15								10 people	1 bus			
16								10 people	1 bus			
17								10 people	1 bus			
18								10 people	1 bus			
19								10 people	1 bus			
20								10 people	1 bus			
21								10 people	1 bus			
22								10 people	1 bus			

Figure 4.2: Analytical model test 1

Time Step	Entry Cell	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Exit Cell	Cell 100 (bus only)	Bus Stop Cell 1	Bus Stop Cell 4	Bus Stop Cell 6
0												
1												
2	10 people											
3										10 people		
4		1 bus (bus empty)								10 people		
5		1 bus 2 passengers								8 people		
6		1 bus 4 passengers								6 people		
7		1 bus 6 passengers								4 people		
8		1 bus 8 passengers								2 people		
9		1 bus 10 passengers										
10			1 bus 10 passengers									
11				1 bus 10 passengers								
12					1 bus 10 passengers							
13						1 bus 10 passengers						
14							1 bus 10 passengers					
15							1 bus 8 passengers	2 people				
16							1 bus 6 passengers	4 people				
17							1 bus 4 passengers	6 people				
18							1 bus 2 passengers	8 people				
19							1 bus (bus empty)	10 people				
20								10 people	1 bus			
21								10 people	1 bus			
22								10 people	1 bus			

Figure 4.3: Analytical model test 2

A second test was run to demonstrate the model results when all travelers are assigned to the bus model. The demand, bus start, cost and capacity parameters are identical to those in the first test, except that the car travel time cost was increased to α_{car}=0.1 cost units/time step, and a parking cost of 3 cost units was defined at cell 6. Further, the rate of boarding and alighting was set to 2 people per time step. The resulting cell occupancies are shown in Figure 4.3, and as expected, the figure shows that the bus dwells at the bus stop as passengers board and alight. In addition, the figure shows that travelers are held in the transfer cell until they board the bus. The system cost for this test was 171 cost units.

A third test was run with the same parameters as in the second test, but with the bus vehicle capacity limited to 7 passengers. The results of this test, shown in Figure 4.4, show that 6 travelers are assigned to the bus and the remainder take their cars. The results from the previous test show that without the bus capacity limit, the optimal solution is to assign all travelers to the bus, so it was expected that with the capacity limit, as many travelers as possible would have been assigned to the bus. However, Figure 4.4 shows that the bus is not filled to its capacity, and this occurs because it is not worth the cost of delaying the bus and all of its passengers for another time step to pick up a single passenger. This result illustrates how the model considers the cost of the whole system, and assigns travelers in ways that are not always intuitive. The system cost for this test was 187 cost units.

Time Step	Entry Cell	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Exit Cell	Cell 100 (bus only)	Bus Stop Cell 1	Bus Stop Cell 4	Bus Stop Cell 6
0												
1												
2	10 people											
3		4 cars								6 people		
4		1 bus (bus empty)	4 cars							6 people		
5		1 bus 2 passengers		4 cars						4 people		
6		1 bus 4 passengers			4 cars					2 people		
7		1 bus 6 passengers				4 cars						
8			1 bus 6 passengers				4 cars					
9				1 bus 6 passengers				4 people				
10					1 bus 6 passengers			4 people				
11						1 bus 6 passengers		4 people				
12							1 bus 6 passengers	4 people				
13							1 bus 4 passengers	6 people				
14							1 bus 2 passengers	8 people				
15							1 bus (bus empty)	10 people				
16								10 people	1 bus			
17								10 people	1 bus			
18								10 people	1 bus			
19								10 people	1 bus			
20								10 people	1 bus			
21								10 people	1 bus			
22								10 people	1 bus			

Figure 4.4: Analytical model test 3

The fourth test, with results shown in Figure 4.5, demonstrates a mid-trip transfer from car to bus. In this test, the travel time costs were set to α_{car}=0.2 cost units/time step and α_{bus}=0.3 cost units/time step. As with the previous tests, a penalty of α_{entry}=10 cost units/time step was defined to prevent travelers from remaining in the entry cell. The transfer time cost for bus stop waiting time was set to 0.3 cost units per time step, and bus fare was set to 0.3 cost units. A parking fee of 3 cost units was defined at cell 6, but no parking fee was defined for cell 4. As such, the parking lot at cell 4 can be interpreted as a free park and ride parking lot, and the lot at cell 6 can be considered an expensive downtown parking lot. As with the previous tests, bus vehicle travel cost was set to 1 cost unit/time step to prevent the bus from being held. Bus vehicle capacity, as well as the roadway maximum flow and jam density were set to infinity. The resulting system cost for this test was 146 cost units.

The figure shows that all ten travelers are initially assigned to the car mode when they enter the network, but then four of those travelers transfer to the bus at cell 4. By transferring at cell 4 the parking cost at cell 6 is avoided; however, if more than four travelers transfer at cell 4, then the cost of delaying the bus and the passengers who have already boarded, as well as the added bus travel and transfer costs, will negate the benefit of avoiding the parking cost. The resulting system cost for this test was 146 cost units.

An interesting result is obtained at the transfer cell. Figure 4.5 shows that at time 6, there are 10 cars in the transfer cell 4. At the next time step, 6 cars have continued into cell 5, while 2 cars remain in cell 4 and 2 travelers have transferred to the bus stop cell. At time step 8, the 2 travelers at the bus stop board the bus, and the 2 people from the car move into the bus stop cell before boarding the bus at time step 9. In other words, whereas it would typically be expected that all four people transferring would move into the bus stop cell at time 7, because the travel time cost of remaining in the car subnetwork is lower than the cost of waiting at the bus stop, the model waits as long as possible before moving people to the bus stop. Since only two people can board a bus at a time, the model moves the last pair of people to the bus stop at the latest time step possible before they board the bus. This travel pattern gives the lowest system cost, but may not reflect the behavior that is intended to be modeled.

Time Step	Entry Cell	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Exit Cell	Cell 100 (bus only)	Bus Stop Cell 1	Bus Stop Cell 4	Bus Stop Cell 6
0												
1												
2	10 people											
3		10 cars										
4		1 bus (bus empty)	10 cars									
5			1 bus (bus empty)	10 cars								
6				1 bus (bus empty)	10 cars							
7					***2 cars*** 1 bus (bus empty)	6 cars					2 people	
8					1 bus 2 passengers		6 cars				2 people	
9					1 bus 4 passengers			6 people				
10						1 bus 4 passengers		6 people				
11							1 bus 4 passengers	6 people				
12							1 bus 2 passengers	8 people				
13							1 bus (bus empty)	10 people				
14								10 people	1 bus			
15								10 people	1 bus			
16								10 people	1 bus			
17								10 people	1 bus			
18								10 people	1 bus			
19								10 people	1 bus			
20								10 people	1 bus			
21								10 people	1 bus			
22								10 people	1 bus			

Figure 4.5: Analytical model test 4

Tests were performed to observe the effect of the bus holding penalty, η_{bus}. In these tests, modal costs were set such that the bus mode was far more attractive than the car mode. Specifically, the travel time costs were set to α_{car}=1.5 cost units/time step and α_{bus}=0.3 cost units/time step. Further, the transfer time cost for bus stop waiting time was set to 0.5 cost units per time step, and bus fare was set to 0.5 cost units. A parking fee of 3 cost units was defined at cell 6, and no parking fee was defined for cell 4. With these costs, it is expected that travelers will be assigned entirely to the bus. Also, as with the previous tests, a penalty of α_{entry}=10 cost units/time step was defined to prevent travelers from remaining in the entry cell. Further, bus vehicle capacity, as well as the roadway maximum flow and jam density were set to infinity.

For the tests of the bus holding penalty, the bus was dispatched from cell 1 at time step 2, while the traveler demand did not arrive at the entry cell until time step 5. It was expected that with low values of η_{bus}, the bus would be held at cell 1 until the travelers arrived and boarded, while high values of η_{bus}, would cause the bus to depart without waiting.

Figure 4.6 shows that cell occupancies of the optimal solution with a value of η_{bus}=0. As expected, since there is no penalty for waiting, the bus remains in cell 1 from time step 2 to time step 5 until the travelers appear and begin to board.

In contrast, figure 4.7 shows the results when the value of η_{bus} is set to 5. In this case, the bus is penalized for remaining in the network and thus does not wait for the passengers to arrive. As such, when the travelers arrive at the entry cell, they move into the bus stop cell and remain there for the rest of the test period because they have already missed the bus. The reason they are not assigned to cars instead is that the car mode has been assigned such high costs that it is less costly to have the travelers wait at the bus stop for the whole test period, even if they never reach the destination. This shows that certain combinations of cost parameters may result in travelers getting stranded at bus stops.

Time Step	Entry Cell	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Exit Cell	Cell 100 (bus only)	Bus Stop Cell 1	Bus Stop Cell 4	Bus Stop Cell 6
0												
1												
2		1 bus (bus empty)										
3		1 bus (bus empty)										
4		1 bus (bus empty)										
5	10 people	1 bus (bus empty)										
6		1 bus (bus empty)								10 people		
7		1 bus 2 passengers								8 people		
8		1 bus 4 passengers								6 people		
9		1 bus 6 passengers								4 people		
10		1 bus 8 passengers								2 people		
11		1 bus 10 passengers										
12			1 bus 10 passengers									
13				1 bus 10 passengers								
14					1 bus 10 passengers							
15						1 bus 10 passengers						
16							1 bus 10 passengers					
17							1 bus 8 passengers	2 people				
18							1 bus 6 passengers	4 people				
19							1 bus 4 passengers	6 people				
20							1 bus 2 passengers	8 people				
21							1 bus (bus empty)	10 people				
22								10 people	1 bus			

Figure 4.6: Analytical model test 5

Time Step	Entry Cell	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Exit Cell	Cell 100 (bus only)	Bus Stop Cell 1	Bus Stop Cell 4	Bus Stop Cell 6
0												
1												
2		1 bus (bus empty)										
3			1 bus (bus empty)									
4				1 bus (bus empty)								
5	10 people				1 bus (bus empty)							
6						1 bus (bus empty)				10 people		
7							1 bus (bus empty)			10 people		
8									1 bus	10 people		
9									1 bus	10 people		
10									1 bus	10 people		
11									1 bus	10 people		
12									1 bus	10 people		
13									1 bus	10 people		
14									1 bus	10 people		
15									1 bus	10 people		
16									1 bus	10 people		
17									1 bus	10 people		
18									1 bus	10 people		
19									1 bus	10 people		
20									1 bus	10 people		
21									1 bus	10 people		
22									1 bus	10 people		
23									1 bus	10 people		

Figure 4.7: Analytical model test 6

The tests shown in Figures 4.2-4.7 were run without any flow or density limits on the roadway. Figures 4.8 and 4.9 are included to demonstrate the model's results under congested conditions. In these tests, the travel time costs were set to α_{car}=0.1 cost units/time step and α_{bus}=0.3 cost units/time step. Further, the transfer time cost for bus stop waiting time was set to 0.5 cost units per time step, and bus fare was set to 0.5 cost units. A parking fee of 3 cost units was defined at cell 6, and no parking fee was defined for cell 4. The entry cell penalty was reduced to α_{entry}=0.4 cost units/time step, which is less than the bus stop waiting cost, so that travelers would not be stranded at the bus stop. Bus vehicle capacity was set to 10 and the bus holding cost was set to η_{bus}=1. A traveler demand of 30 people arrived at the entry cell at time 2, and a bus was dispatched from cell 1 at time 4.

The test shown in Figure 4.8, was run with an infinite jam density for each cell, but with a maximum flow rate of 6 cars per time step. The cell occupancies in Figure 4.8 show that all 30 travelers are assigned to cars, and are loaded immediately into cell 1. Since the maximum flow rate is 6 cars per time step, only 6 cars move forward at time step 3, leaving 24 cars to wait in cell 1. A bus appears in time step 4, and while according to FIFO rules, it should be the last to leave cell 1, the model moves it forward immediately. This occurs both because the bus carries a holding penalty, and because the model constraints assume a lane blockage as long as the bus does not cross a cell boundary. Therefore, the model forces the bus to move first in order to avoid a reduction in capacity. As such, the bus moves from cell 1 to cell 2 at time 4 ahead of the cars that were already queued. Since the bus is assigned a car equivalence factor of 2, flows that include the bus movement can include only 4 cars. The remainder of the flows include 6 cars.

The test shown in Figure 4.9, was run with an infinite maximum flow for each cell, but with a jam density of 6 cars per cell. The cell occupancies in Figure 4.9 show that all 30 travelers are assigned to cars, but initially remain in the entry cell, since there is not enough space for them all in cell 1. As in the previous test, the bus is loaded onto cell 1 at time 4, at the back of the queue, but is propagated forward ahead of the cars. Again, this reflects the model's non-FIFO behavior.

The difference between Figures 4.8 and 4.9 is that in Figure 4.8, vehicles are propagated continuously at each time step; however, in Figure 4.9 vehicles are propagated across cell boundaries only at every second time step. This occurs because constraint (4.17), with δ (ratio free flow speed/backward propagation speed) set to 1 for all cells, limits a cell's inflow at time t to the amount of space available at time t, regardless of the amount of outflow at time t. This constraint is derived from the last term of the cell transmission rule as shown in Equation 2.13. If a cell is being rapidly emptied, then constraint (4.17) with δ=1 may appear overly conservative; however, Daganzo (2) defends this formulation by stating that empty slots travel backward at a finite speed, the wave propagation speed, which is unlikely to be greater than the free flow speed. Therefore, the effects of the outflow should only be noticed upstream after some time. In test, the wave propagation speed was assumed to be equal to the free flow speed, so the cars must wait for a time step to pass before an empty slot appears.

Time Step	Entry Cell	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Exit Cell	Cell 100 (bus only)	Bus Stop Cell 1	Bus Stop Cell 4	Bus Stop Cell 6
0												
1												
2	30 people											
3		30 cars										
4		24 cars 1 bus (bus empty)	6 cars									
5		20 cars	4 cars 1 bus (bus empty)	6 cars								
6		14 cars	6 cars	4 cars 1 bus (bus empty)	6 cars							
7		8 cars	6 cars	6 cars	4 cars 1 bus (bus empty)	6 cars						
8		2 cars	6 cars	6 cars	6 cars	4 cars 1 bus (bus empty)	6 cars					
9			2 cars	6 cars	6 cars	6 cars	4 cars 1 bus (bus empty)	6 people				
10				2 cars	6 cars	6 cars	6 cars	10 people	1 bus			
11					2 cars	6 cars	6 cars	16 people	1 bus			
12						2 cars	6 cars	22 people	1 bus			
13							2 cars	28 people	1 bus			
14								30 people	1 bus			
15								30 people	1 bus			
16								30 people	1 bus			
17								30 people	1 bus			
18								30 people	1 bus			
19								30 people	1 bus			
20								30 people	1 bus			
21								30 people	1 bus			
22								30 people	1 bus			

Figure 4.8: Analytical model test 7

Time Step	Entry Cell	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Exit Cell	Cell 100 (bus only)	Bus Stop Cell 1	Bus Stop Cell 4	Bus Stop Cell 6
0												
1												
2	30 people											
3	24 people	6 cars										
4	22 people	2 cars 1 bus (bus empty)	6 cars									
5	20 people	4 cars 1 bus (bus empty)		6 cars								
6	14 people	6 cars	4 cars 1 bus (bus empty)		6 cars							
7	14 people	6 cars		4 cars 1 bus (bus empty)		6 cars						
8	8 people	6 cars	6 cars		4 cars 1 bus (bus empty)		6 cars					
9	8 people	6 cars		6 cars		4 cars 1 bus (bus empty)		6 people				
10	2 people	6 cars	6 cars		6 cars		4 cars 1 bus (bus empty)	6 people				
11	2 people	6 cars		6 cars		6 cars		10 people	1 bus			
12		2 cars	6 cars		6 cars		6 cars	10 people	1 bus			
13		2 cars		6 cars		6 cars		16 people	1 bus			
14			2 cars		6 cars		6 cars	16 people	1 bus			
15				2 cars		6 cars		22 people	1 bus			
16					2 cars		6 cars	22 people	1 bus			
17						2 cars		28 people	1 bus			
18							2 cars	28 people	1 bus			
19								30 people	1 bus			
20								30 people	1 bus			
21								30 people	1 bus			
22								30 people	1 bus			

Figure 4.9: Analytical model test 8

4.4 Summary

An integer linear programming formulation of the system optimal single destination DTA model with buses was presented. Computational tests were run and cell occupancy tables were presented to illustrate the results of the tests. The test results show that vehicle and person flow through the network abide by cell transmission and boarding and alighting rules. Further, it was found that the relative values of the cost parameters may yield counter-intuitive assignment results. For example, if a bus only has capacity for one more passenger, the model may find it less costly to assign that passenger to the car mode, rather than delay the remainder of the bus passengers and incur additional cost of the bus holding penalty. Similarly, if the travel cost parameter for the car is lower than the bus stop delay cost parameter, then the model will leave people in the car subnetwork for as long as possible before transferring them to the bus stop transfer cell, implying that people will sit in their cars on the roadway until just before the bus arrives. In addition, the effect of different bus holding penalty values on bus waiting behavior was demonstrated. Further, it was shown that the model always moves buses before cars, and thus does not observe FIFO principles. Finally, results are included to show that, according to the rules of the cell transmission model, when the jam density is reached, vehicles must wait for the backward propagation of empty slots before moving forward.

It was found that the framework of an analytical model does not easily allow for modeling of complex traffic and travel phenomena. More specifically, the analytical model optimizes the system costs by adjusting both intermodal path choices as well as traffic movements; for example, the computational results showed that this kind of optimization resulted in holding of buses or skipping of bus stops, depending on the definition of the cost function. By contrast, in simulation-based approaches, such as those presented in Sections 2 and 3, traffic movements are determined by the simulator’s traffic flow rules, and system costs are optimized based only on adjustments to path and mode choices. This separation of traffic flow from the optimization prevents counter-intuitive traffic movements exhibited by the analytical model. As such, continuing research efforts will focus on the simulation-based approaches presented in Sections 2 and 3, and no further development of the analytical formulation of the intermodal problem is planned.

5.0 CONCLUSIONS AND FUTURE RESEARCH

In this research project, multimodal and intermodal models were developed for evaluation of transit performance and asset management. The research, supported by the Midwest Regional Urban Transportation Center (MRUTC), was carried out in parallel with a study of the regional impacts of transit signal priority (TSP) in Chicago, funded by Chicago's Regional Transportation Authority (RTA). The algorithms and models proposed in this report were developed to address issues encountered during the TSP study, but are also generally applicable to the evaluation of transit performance and asset utilization as these are affected by network congestion and travel behavior.

The report presented three approaches to modeling multimodal and intermodal car and bus networks. The first approach was an automobile assignment-based approach, while the second and third approaches explored person assignment-based intermodal approaches. The automobile assignment-based approach, which has been implemented in the VISTA software for use in the TSP study, iterates between a cell transmission-based simulator, an automobile-only time-dependent shortest path algorithm, and a path assignment algorithm. For the TSP study, the simulator was enhanced to capture stopping and dwelling bus behavior, as well as interactions between cars and buses on the roadway. Further, an IADUE assignment algorithm was developed to estimate user equilibrium path demands at a level of accuracy that would ensure that path changes resulting from subtle TSP-induced changes in travel time would be captured. Computational results were performed on a small test network to explore the effect of limiting and suppressing different phases of the algorithm. For tests on the regional Chicago network, MSA was used to generate a limited path set, and then Phase 3 of the IADUE was used to improve the estimate. For the large-scale network, an average cost deviation of 6.4% and 4.9% per trip were achieved for the AM and off-peak base cases, and under 10% for the TSP test scenarios. Validation of base case results with with observed counts and travel times showed that estimates were reasonably evenly distributed both above and below the observed values. Further, a preliminary analysis of the regional impacts of TSP was presented, with a complete analysis of the regional impacts of TSP to be submitted to the RTA in the form of the TSP study's Final Report.

The automobile assignment approach captures the effects of congestion and traffic conditions on bus movements, such that bus performance, including travel times and travel time variabilities can be observed. These measures can then be used as a basis for drawing conclusions on transit operating efficiency, transit service and ridership; however, since automobile trips are exogenously defined, the model does not capture mode choice behavior, and ridership impacts cannot be directly observed in the model outputs. As such, methods of solving the combined mode split and assignment problem were also explored. Assignment of person trips to intermodal routes allows mode and route choice to be modeled as a simultaneous decision. This report proposed both simulation-based and analytical approaches to solving the person assignment-based combined mode split and route choice problem.

The simulation-based person assignment approach is an extension of the automobile assignment-based model developed for the Chicago TSP study. As in the automobile assignment-based model, the simulator models car and bus interactions on the roadway network; however, for the person assignment-based approach the single-mode shortest path algorithm is replaced by an intermodal least cost path algorithm to calculate intermodal paths based on a generalized cost function. The intermodal least cost path algorithm is an adaptation of Yang's (8) algorithm for freight problems, and a proof of correctness, not included in Yang's work, has been presented in this report. Further, the path assignment is determined using the IADUE assignment algorithm proposed for the automobile assignment approach. The person assignment-based intermodal model has been implemented in the framework of the VISTA software. As in the automobile assignment-based version, the IADUE algorithm is heuristic; however, preliminary results on a small test network have shown that the algorithm converges to a reasonably close approximation of equilibrium. Tests on real-world networks are currently not planned, since real-world tests will require detailed person trip data and disutility parameters, which may be difficult to obtain, calibrate and validate.

With the person assignment approach, person trips, instead of vehicle trips are assigned to the network, and mode and route choices are modeled as simultaneous decisions. Since the simulator does not model person movements, but only roadway vehicle movements, bus dwell times remain exogenously defined, rather than being determined by boardings and alightings; however, using the intermodal least cost path algorithm ridership and mode share, which are central issues in any transit policy evaluation, can be directly observed in the model.

An integer linear programming formulation of the system optimal combined mode split and assignment model was also presented. Computational results on a test network showed that vehicle and person flow through the network abided by cell transmission and boarding and alighting rules; however, it was also shown that the model does not move vehicles according to FIFO rules, and further that buses may either be held to wait for travelers arriving at a bus stop, or may skip bus stops if delay cost of picking up passengers is greater than cost of leaving the travelers at the bus stop. These results do not reflect realistic traffic movements and occur because the analytical model optimizes the system costs by adjusting both intermodal path choices as well as traffic movements. As such, no further development of the analytical formulation of the intermodal problem is planned. Instead, continuing research efforts will focus on the simulation-based approaches, which separate traffic movement calculations from the optimization of the path assignment, and thus avoid the counter-intuitive traffic movements exhibited by the analytical model.

In summary, this report presented three different approaches to modeling multimodal and intermodal problem for evaluation of transit performance and asset management. The first approach was an automobile assignment-based multimodal approach, which captures bus movements in the simulator, but assigns automobile demand to shortest paths under the assumption that mode split is fixed. A IADUE assignment algorithm for estimating the equilibrium path demands was presented. The model is being used to evaluate the regional impacts of TSP in Chicago. Preliminary results were presented for a small test network, as well as for the regional Chicago network to be used in the TSP study. Also, a person assignment-based intermodal approach was presented such that mode and path choices are modeled as simultaneous decisions. This approach uses the same multimodal car and bus simulator as in the vehicle assignment approach, and also uses the IADUE algorithm to determine the equilibrium path assignment; however, instead of automobile shortest paths, intermodal least cost paths are calculated at each iteration. This approach has been implemented in VISTA, and will be tested on small networks. Third, an analytical formulation of the system optimal intermodal DTA problem was proposed, and computational results were presented; however no further development of this model is planned.

Continuing research efforts will focus on refinement of the IADUE algorithm. Specifically, further understanding of how convergence parameters for each of the IADUE algorithm's three phases affect the algorithm's overall convergence and computational time. Selection of appropriate search and convergence parameters has the potential to save a significant amount of computational time, especially when analyzing realistic large-scale networks, such as the Chicago regional network. In addition, the importance of monotonicity to the algorithm's convergence rate will also be examined. More specifically, since the algorithm relies on the idea that path cost increases with path demand, it is possible that increased path interactions and congestion may affect the convergence of the algorithm. Also, further testing of intermodal problems will be required to determine how closely the IADUE algorithm is able to estimate the equilibrium given increased discontinuities of the cost function in the form of transfer delays.

6.0 REFERENCES

1. Peeta, S. and A.K. Ziliaskopoulos. 2001. Foundations of dynamic traffic assignment: The past, the present and the future. *Networks and Spatial Economics* 1 (3/4): 233-65.
2. Daganzo, C.F. 1994. The cell transmission model : a dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research* 28B (4): 269-287.
3. *Highway Capacity Manual*, Special Report 209. Washington, D.C.: Transportation Research Board. 1985.
4. Ziliaskopoulos, A.K. and H.S. Mahmassani. 1993. Time-dependent, shortest path algorithm for real-time intelligent vehicle highway systems applications. *Transportation Research Record* 1408: 94-100.
5. Cooke, K. L. and E. Halsey. 1966. The shortest route through a network with time-dependent intermodal transit times. *Journal of Mathematical Analysis and Applications* 14 (3): 492-98.
6. Smith, M. J. 1983. The existence and calculation of traffic equilibria. *Transportation research* 17B (4): 299-303.
7. Ziliaskopoulos, A. and E. Chang. 2003. *Interim Report: Data for Model Input, Calibration and Validation*. Madison, WI: Midwest Regional Urban Transportation Center.
8. Yang, K. 1998. Time-Dependent Multimodal Multicommodity Freight Network Assignment Model. Master’s thesis, Northwestern University.