

Demand Simulation for Dynamic Traffic Assignment

Constantinos Antoniou

Diploma in Civil Engineering
National Technical University of Athens, Greece (1995)

Submitted to the Department of Civil and Environmental Engineering
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

January 1997

[February 1997]

© Massachusetts Institute of Technology, 1997. All Rights Reserved.

Author
Civil and Environmental Engineering Department
January 24, 1997

Certified by
Moshe Ben-Akiva
Professor of Civil and Environmental Engineering
Thesis Supervisor

Certified by
Michel Bierlaire
Research Associate, Center for Transportation Studies
Thesis Supervisor

Certified by
Rabi Mishalani
Research Associate, Center for Transportation Studies
Thesis Supervisor

Accepted by
Joseph M. Sussman
Chairman, Departmental Committee on Graduate Studies

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

JAN 29 1997



Demand Simulation for Dynamic Traffic Assignment

by

Constantinos Antoniou

Submitted to the Department of Civil and Environmental Engineering on January 24,
1997, in partial fulfillment of the requirements for the degree of
Master of Science

Abstract

In this thesis a pre-trip demand simulator with predictive capabilities and explicit simulation of the response of the travelers to real-time pre-trip information is proposed and implemented. The demand simulator is an extension of conventional OD estimation models aimed at overcoming their inability to explicitly capture the effect of information on demand. This is achieved by explicitly simulating driver behavior in response to both descriptive and prescriptive information at the disaggregate level.

The true demand can be constructed from the historical with the addition of two systematic deviations –the effect of information in the demand and the daily demand fluctuations– and a random error. Conventional OD estimation models do not take explicitly into account the first systematic deviation and, hence, their estimation accuracy may be limited. In the proposed demand simulator, the systematic component of the deviation that is due to the available information is captured by a set of disaggregate behavioral models which update the historical demand. The OD estimation uses the updated OD matrix, instead of the historical OD matrix itself, as a starting point to compute an estimated OD matrix consistent with the observed link counts.

A series of case studies are performed to illustrate some of the capabilities and assess some of the properties of the demand simulator, as well as investigate some of its potential shortcomings. Furthermore, a framework that can be used for a more comprehensive evaluation of the demand simulator is presented.

Thesis supervisor: Moshe Ben-Akiva

Title: Professor, Department of Civil and Environmental Engineering

Thesis supervisor: Michel Bierlaire

Title: Research Associate, Center for Transportation Studies

Thesis supervisor: Rabi Mishalani

Title: Research Associate, Center for Transportation Studies

Acknowledgments

I would like to thank my thesis supervisors Professor Moshe Ben-Akiva, Michel Bierlaire, and Rabi Mishalani for the guidance, advise and constructive feedback that they provided at all stages of this research.

I would also like to thank the staff of the MIT ITS Program, who provided a friendly research environment, and valuable information and help every time it was needed.

I am also thankful to Oak Ridge National Laboratory and Federal Highway Administration (FHWA) for financially supporting this research.

I would also like to thank my friends and fellow students that made MIT a unique experience and taught me a lot of things.

Finally, I would like to thank my family.

Table of Contents

1. Introduction	10
1.1 Background	11
1.2 Problem Statement and Thesis Contribution	15
1.3 Application	17
1.4 Thesis Outline	17
2. Literature Review	18
2.1 Dynamic OD Estimation and Prediction Models	18
2.1.1 “Closed” Networks	18
2.1.2 General Networks	19
2.1.3 Conclusion	20
2.2 Behavioral models	20
2.2.1 Departure time choice	21
2.2.2 Mode choice	21
2.2.3 Route choice	22
2.2.4 Joint choice	23
2.2.5 Conclusion	23
3. Design of the Pre-Trip Demand Simulator	25
3.1 Framework	25
3.2 Variable and Parameter Definition	28
3.3 Time Discretization	32
3.4 Choice Set Generation	33
3.5 Off-line Disaggregation	34
3.5.1 Habitual Route Model Structure	34
3.5.2 Habitual Route Model Specification	36
3.5.3 Algorithm	36
3.6 Pre-Trip Behavior Update	39
3.6.1 Descriptive Model Structure	39
3.6.2 Descriptive Model Specification	41
3.6.3 Prescriptive Model Structure	45
3.6.4 Prescriptive Model Specification	47
3.6.5 Algorithm	47
3.7 Aggregation	49
3.7.1 Algorithm	49
3.8 OD Estimation and Prediction Model	50
3.8.1 Model Overview	51
3.8.2 The Transition Equation	51
3.8.3 The Measurement Equation	52
3.8.4 Approximate Formulation	53
3.8.5 Algorithm	54
3.8.6 OD Estimation from Deviations	57

3.8.7 Discussion on Algorithm	57
4. Implementation	59
4.1 Overview	59
4.2 Objectives	61
4.2.1 Object Oriented (OO) Design	61
4.2.2 Computational Efficiency	61
4.2.3 Numerical Robustness	62
4.3 Requirements	62
4.3.1 Design Flexibility	63
4.3.2 Element Access	64
4.3.3 Memory Requirements	65
4.4 Objects	66
4.4.1 List of Objects	66
4.4.2 Object Associations	67
4.5 Implementation Issues	69
4.5.1 Off-line Disaggregation	69
4.5.2 Pre-Trip Behavioral Model	70
4.5.3 Aggregation	70
4.5.4 OD Estimation and Prediction Model	70
4.5.5 On-line Disaggregation	74
5. Evaluation	76
5.1 Scenarios	77
5.1.1 Network (A)	77
5.1.2 Historical demand (B)	79
5.1.3 Population-wide characteristics (C)	81
5.1.4 Pre-trip behavior (D)	82
5.1.5 En route behavior (E)	87
5.1.6 Guidance (F)	88
5.1.7 Habitual travel time data (G)	88
5.1.8 Link Counts (H)	88
5.1.9 Assignment matrices (I)	89
5.1.10 Transition matrices (J)	89
5.1.11 Error covariance matrices (K)	89
5.1.12 Initial state vector covariance (L)	90
5.2 Impact of Behavioral Update	90
5.2.1 Description of the process	90
5.2.2 Analysis of the Results	91
5.2.3 Conclusions	93
5.3 Stochasticity Analysis	93
5.3.1 Description of the Process	93
5.3.2 Analysis of the results	94
5.3.3 Conclusions	97
5.4 Sensitivity to Input	97
5.4.1 Description of the Process	97
5.4.2 Analysis of the results	99
5.4.3 Conclusions	104
6. Conclusions	105

6.1 Summary	105
6.2 Further Research	108
6.2.1 Extensions of Performed Tests	108
6.2.2 Assessment of Real-time performance	109
6.2.3 Evaluation of the Demand Simulator	110
7. References	114

List of Figures

Figure 1. Structure of the DynaMIT system	14
Figure 2. Representation of demand estimation	16
Figure 3. Pre-trip Demand Simulator	26
Figure 4. Simple road network with overlapping paths	35
Figure 5. Structure of the C-logit model	35
Figure 6. Habitual path choice generation	38
Figure 7. Pre-trip choice tree in the case of descriptive information	41
Figure 8. Pre-trip choice tree in the case of prescriptive information	46
Figure 9. Travel behavior update process	48
Figure 10. Estimation and prediction structure	55
Figure 11. Association example	60
Figure 12. Multiplicity example	60
Figure 13. Aggregation example	60
Figure 14. Inheritance example	61
Figure 15. Matrix Classes	63
Figure 16. Vector Classes	64
Figure 17. Read-write iterator inheritance	65
Figure 18. Object associations	68
Figure 19. ODMatrix data structure	73
Figure 20. Assignment matrix data structure	74
Figure 21. Map of the network	78
Figure 22. Schematic representation of the network	79
Figure 23. Demand pattern	80
Figure 24. Impact of behavioral update process overview	91
Figure 25. Deviation of updated from historical demand	92
Figure 26. Stochasticity analysis process overview	94
Figure 27. Total updated demand	95
Figure 28. Distribution of total updated demand values	96
Figure 29. Total updated demand standard deviation	96
Figure 30. Sensitivity process overview	98
Figure 31. Deviation of updated demand due to perturbation of the behavioral parameters	101
Figure 32. Deviation of estimated demand due to perturbation of the behavioral parameters	101
Figure 33. Relative error in the updated and estimated demand due to perturbations in the behavioral parameters	102
Figure 34. Deviation in updated demand due to perturbation of the historical demand	102
Figure 35. Deviation of estimated demand due to perturbation of the historical demand	103
Figure 36. Relative error in the updated and estimated demand due to perturbations of historical demand	103
Figure 37. Relative errors in estimated link flows due to perturbation in the historical OD flows	104
Figure 38. Simulation Based Evaluation of Demand Estimation	112

List of Tables

<i>Table 1. List of variables for travel behavior models</i>	29
<i>Table 2. List of superscripts of coefficients for travel behavior models</i>	31
<i>Table 3. List of subscripts of coefficients for travel behavior models</i>	32
<i>Table 4. List of subscripts of structural coefficients for travel behavior models</i>	32
<i>Table 5. OD matrix for 3rd interval (7:30-7:45)</i>	80
<i>Table 6. Generated OD matrices</i>	81
<i>Table 7. Population-wide characteristics distribution</i>	82
<i>Table 8. Travel time profiles</i>	84
<i>Table 9. Estimated coefficients for the behavioral model</i>	86
<i>Table 10. Estimated constants for the behavioral model</i>	87

1. Introduction

Road traffic congestion and all the side-effects it has on the environment, productivity and mental state of the people, is one of the leading problems of cities today. Congestion steadily increases and the conventional approach of building more roads is faced with skepticism for a variety of political, economic, social, and environmental reasons. The realization that the existing terrestrial infrastructure can indeed perform better than it is doing at present, as a result of better utilization, led to the Intelligent Transportation Systems (ITS) initiative, which was introduced in the late 80s as Intelligent Vehicle Highway Systems (IVHS). For an overview of ITS, the reader is referred to Sussman (1992).

ITS is a broad term used to describe a variety of advanced technologies in the areas of advanced communication, computers, information display, road infrastructure, traffic control systems and advanced vehicle control systems. It envisages the development of a Dynamic Traffic Management System that would, in real-time, attempt to improve capacity utilization by:

- Providing both pre-trip and en route information to motorists with respect to optimal paths to their destinations, and
- Using advanced traffic control systems that are adaptive to rapidly changing traffic conditions in real-time.

Congestion management measures can be categorized as either demand-based or supply-based. Supply-based measures are intended to increase the existing capacity of the system in order to improve the traffic flow for all modes. These measures include signal control improvements, incident management and preferential treatment of high occupancy vehicles (HOV) and public transit vehicles. Incident management refers to the necessary actions that need to be taken in real time to mitigate the effect of an incident on the network and alleviate the congestion that it may generate. The area of ITS that deals with these traffic management schemes is called Advanced Traffic Management Systems (ATMS).

Demand-based measures, on the other hand, are designed to reduce car demand on the system by increasing vehicle occupancy and public transit share, distributing the travel demand over time (peak spreading) and distributing the travel demand more uniformly on the traffic network. Demand-based systems rely heavily on the availability

of reliable, real-time information about the traffic conditions in the network. Collected information, such as current travel times and speeds on links, can be used as is or as the basis for the calculation of more complex information, such as the shortest path for a given origin-destination (OD) pair. These systems can improve the information available to the travelers by updating them about current and anticipated conditions on the roadway and transport network in general, and providing route guidance, using a variety of audio and visual means, both in-vehicle and on the roadway and other transport facilities. The disseminated information assists the individual travelers in decisions about their travel patterns.

A particular sort of demand-based measures take advantage of technological advancements, such as Advanced Traveler Information Systems (ATIS), for the faster and more effective dissemination of real-time information to travelers. Information can be provided either in the pre-trip stage or en route. Pre-trip information can be used by the travelers for departure time, route and mode choice. Furthermore, the travelers may decide to change their destination or cancel the trip altogether in response to pre-trip information. On the other hand, the effect of information available en route is usually restricted to subsequent route switching decisions. Nevertheless, it is theoretically possible to have mode choice made en route, e.g. in the case of a driver deciding to use a park-n-ride facility and switch to transit for the remainder of the trip, instead of driving.

A desirable feature of such ATMS and ATIS efforts is the ability to predict future traffic, since without projection of traffic conditions into the future, control or route guidance strategies are likely to be irrelevant and outdated by the time they take effect. In predicting traffic, however, it is essential that the effect of the provided information and guidance on travelers' travel decisions and, consequently, the network's traffic conditions is captured. Otherwise, the information and guidance will not be reliable.

In this thesis, a pre-trip demand simulator which captures the effect of information on estimating and predicting future roadway traffic demand is designed and implemented. The simulator uses disaggregate behavioral models to predict the drivers' travel behavior in response to real-time pre-trip information, and an aggregate OD estimation and prediction model.

1.1 Background

A congestion management tool that has been under investigation over the past years is Dynamic Traffic Assignment (DTA). A DTA system uses historical and real-time information to estimate and predict traffic and, consequently, provide travel information

and guidance. Real-time information about the traffic conditions on the network is essential in the performance of a DTA system, since this is the base of the generated guidance. In addition of being a necessary element of ATIS, DTA is also an essential element of ATMS. Travel guidance will be more effective when guidance information is provided as a component of an integrated approach including advanced traffic management and road pricing.

In this context, MIT Intelligent Transportation Systems (ITS) Program is developing DynaMIT (Dynamic Network Assignment for the Management of Information to Travelers), a DTA system that generates guidance based on predicted traffic conditions. Provided information can be either descriptive or predictive. Descriptive guidance provides tripmakers with information on network conditions so that they can make better decisions about departure time, mode and route choice. Prescriptive guidance consists of actual recommendations about route choice which users decide whether to accept (comply) or reject. The way information is communicated may affect drivers' compliance rates. For example, based on driver simulation experiments, some researchers have reported higher compliance with prescriptive information when descriptive information is also presented (Bonsall, 1992, and Kitamura and Jovanis, 1991).

The assumptions made about user tripmaking behavior, and particularly about user response to travel choice guidance information, are central to any procedure which attempts to generate such information. Hence, different categories of user behavior are represented in the form of user classes in DynaMIT. In this context, a user class is a particular combination of access to information (e.g. through an in-vehicle unit, a roadway sign or a radio broadcast) and user behavior parameters (e.g. utility function specification and coefficients) so that the reaction of different tripmakers to the provided information is taken into account (e.g. different degrees of compliance to guidance recommendations).

The structure of the DynaMIT system is shown in Figure 1. DynaMIT simulates both demand and supply. The demand is simulated by the demand simulator which estimates and predicts the demand in the network, taking into account the effect of available information. The supply (traffic) simulator then assigns this demand on the network and simulates the vehicles' trips considering en route demand modifications due to response to guidance. Using historical data and data from the surveillance system as input, an iterative process between the demand and the supply simulator is used to obtain a consistent estimate of the current state of the network in terms of OD flows, link flows, queues and densities. This is an important task, since information obtained from the

traffic sensors is generally sparse, i.e. the link counts are known only for a fraction of the links. Therefore, estimation is required to provide a complete picture of traffic conditions. The demand simulator incorporates an OD estimation model, which uses historical OD flows, real-time measurements of actual link flows on the network, and estimates of assignment fractions (the mapping of OD flows to counts) to estimate the true OD flows. The demand simulator is also sensitive to the information provided to the drivers. This information represents the true conditions on the network. By taking this into account, the simulator updates the historical demand in response to information from the network and uses this updated historical demand as a starting point for the estimation. Therefore, behavioral models which update the historical OD flows before they are used by the OD estimation model are incorporated in the demand simulator. As a result, the updated OD matrix already incorporates changes that took place in response to guidance and information.

The supply simulation model simulates the actual traffic conditions on the network. Inputs include OD flows estimated by the OD estimation model, network capacities and traffic dynamic parameters, as well as prior control strategies and guidance broadcast. The supply simulator outputs an estimate of the traffic conditions on the network, including link flows. These link flows will generally be different than the actual link counts in the network provided by the surveillance system. Therefore, an iterative process is required between the supply and the demand simulator until the simulated link counts are sufficiently close to the observed. In each iteration the supply simulator uses the estimated demand as input to provide an assignment matrix, which is used by the demand simulator to estimate a new demand. The demand and supply simulators may have to go through several iterations to obtain a consistent estimate. The output of this process is an estimate of the actual traffic conditions on the network, including link flows, queues and densities.

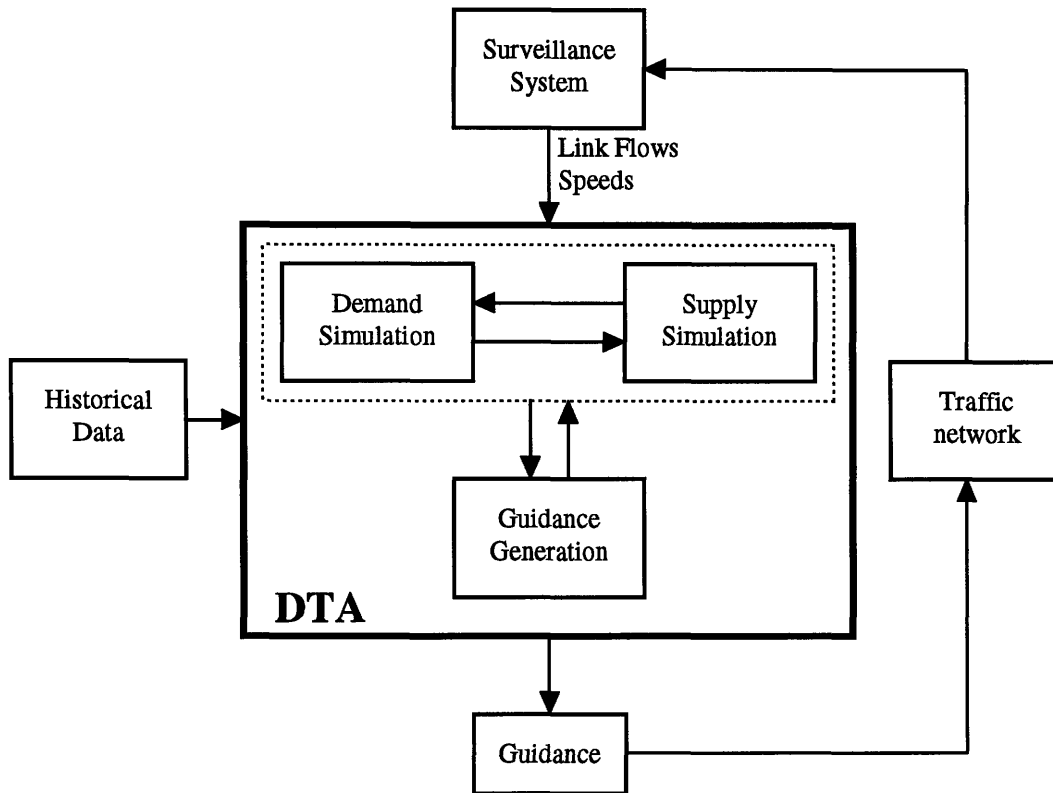


Figure 1. Structure of the DynaMIT system

After the state of the network has been estimated, the demand simulator uses it, in conjunction with the supply simulator, as a starting point to predict future performance of the network. The guidance generation model uses the predicted traffic conditions to generate guidance according to the various ATIS in place. The guidance provided must be such that if the driver follows the guidance, there will be no better path that the driver could have taken instead. This is required, because if the drivers realize that the provided guidance is not the best possible, they will increasingly start ignoring the guidance recommendations resulting in a decrease in the compliance rate. In order to obtain a guidance that satisfies these requirements, an iterative process is required. An iteration consists of the generation of a trial strategy, the state prediction under the strategy, and the evaluation of the predicted state. The guidance that will result in traffic conditions consistent with the information it is based on is the one that is implemented on the network.

From the presentation of the DynaMIT framework, and general research experience in the area, the importance of the following can be highlighted:

- Basing the generation of guidance on predicted traffic conditions, and

- Incorporating realistic models of driver behavior in general, and of drivers' response to information in particular to achieve accurate predictions.

The demand simulator provides the future OD flows that are necessary for the prediction of traffic conditions, while it also incorporates the driver behavior models that are necessary to update the historical OD flows in response to available real-time information.

A key component of the demand simulator process is dynamic OD estimation and prediction. In a real-time application, estimation refers to the computation of the OD matrix for departure time intervals up to and including the current interval of interest. Prediction refers to the computation of the OD matrix for future departure time intervals. The model uses aggregate historical OD matrices, representing demand at the OD level, as input and computes an estimated OD matrix. Making assumptions about time evolution of demand, the model is also capable of predicting future OD matrices. Since historical demand is known at the OD level, an aggregate OD estimation model is used.

Besides the need for predicted traffic conditions, the validity and relevance of the demand simulator's output is dependent on its ability to capture drivers' travel decisions at the individual level. In order to capture these decisions, the demand simulator incorporates disaggregate travel behavior models that predict trip making decisions such as departure time, mode and route choice, based on real-time information and the characteristics of the individuals. The use of disaggregate models allows the demand simulator to model each driver individually and thus potentially capture the travelers' behavior more accurately. Thus, the variations in demand due to the effect of information in the individual travelers' decision are captured.

The combined use of an aggregate (macroscopic) and a disaggregate (microscopic) model results in a *mesoscopic* demand simulator. This combination provides a good tradeoff between computational performance and estimation accuracy.

1.2 Problem Statement and Thesis Contribution

The focus of this thesis is on the pre-trip component of the demand simulator of DynaMIT discussed above. Pre-trip demand is the demand that enters the network and thus reflects the effect of pre-trip information on drivers' decisions. The overall representation of pre-trip demand estimation is presented in Figure 2. The true demand can be constructed from the historical with the addition of two systematic and one

random deviations. The two systematic terms are the effect of information in the demand and the daily demand fluctuations. The third deviation is a random error term.

$$\text{True demand} = \left\{ \begin{array}{l} \text{Historical demand} \\ + \\ \text{Effect of information} \\ + \\ \text{Daily demand fluctuations} \\ + \\ \text{Error } \varepsilon \end{array} \right. \begin{array}{l} \text{Systematic} \\ \text{deviations} \\ \\ \text{Random} \end{array}$$

Figure 2. Representation of demand estimation

Conventional OD estimation models update the historical demand to reflect actual network demand, thus capturing drivers' travel patterns and demand deviations at the OD level. The models start from a historical OD matrix and observed link counts and estimate an OD matrix that is consistent with these link counts. Therefore, these models do not explicitly take into account the first systematic deviation and, hence, their estimation accuracy may be limited since they fail to directly capture the deviation from the historical demand resulting from the pre-trip information that has become available to the drivers about the network traffic conditions.

In this thesis a *pre-trip demand simulator* with predictive capabilities and explicit simulation of the response of the travelers to real-time pre-trip information is proposed and implemented. The demand simulator is an extension of conventional OD estimation models aimed at overcoming their inability to explicitly capture the effect of information on demand. This is achieved by explicitly simulating driver behavior in response to both descriptive and prescriptive information at the disaggregate level. The systematic component of the deviation that is due to the available information is captured by a disaggregate behavioral model which updates the historical demand. The OD estimation uses the updated OD matrix, instead of the historical OD matrix itself, as a starting point to compute an estimated OD matrix consistent with the observed link counts. Thus, the OD estimation procedure starts with the pre-trip guidance information already taken into account. Thus, the final estimated demand explicitly includes both systematic components of the deviation.

The prediction makes an assumption about the time evolution of the demand fluctuations to predict future OD matrices. The demand simulator is designed and implemented taking into account computational and other feasibility concerns. Furthermore, the demand simulator is validated through an application using a simulated

real network in order to assess its potential methodological and operational capabilities in a DTA context.

This methodology, characterized by the explicit representation of pre-trip travel decisions in response to information, differs from existing approaches. To the best of the author's knowledge and based on published literature, existing DTA models either do not capture pre-trip travel decisions (e.g. Mahmassani et al. (1993)), or when they do, assume complete knowledge of future traffic conditions (e.g. Ran et al.(1992) and Friesz et al. (1993)).

1.3 Application

The proposed demand simulator is a tool with a wide variety of potential uses. Although this thesis focuses on its application in a DTA environment, the demand simulator is applicable to all kinds of networks, including rural and regional networks. Furthermore, although the simulator is described in a real-time context (short term), it can be used in a wide variety of applications, including evaluation (medium term) and planning (long term). An important distinction between the use of the demand simulation in a real-time context and in planning problems is the time scale of the interactions between demand and supply. In the case of planning applications, these interactions develop over possibly long periods of time. Therefore, the day to day learning and adjustment mechanisms of individual drivers should be explicitly represented and captured by the behavioral models. Depending on the type of application the effects can be really long term and may include revisions of residential location or car ownership. In medium term cases, such as the evaluation of the effectiveness of an ATMS project, the effects may include changes in the mode or path that travelers customarily take to work.

1.4 Thesis Outline

This thesis is organized as follows. The literature review is presented in Section 2. The design of the pre-trip demand simulator is described in Section 3. The software implementation of the system is presented in Section 4. A preliminary evaluation of the simulator is presented in Section 5. Finally, the summary, conclusions, and further research are discussed in Section 6.

2. Literature Review

The objective of this literature review is to provide the necessary background for the conceptual and analytical design of the demand simulator. The demand simulator proposed in this thesis explicitly incorporates the effect of information on the users' travel behavior, and estimates and predicts demand. To achieve this task, the demand simulator uses:

- Dynamic OD estimation and prediction models (i.e. aggregate or macroscopic level), and
- Behavioral models (i.e. disaggregate or microscopic level).

A literature review on these two topics is presented in the next two sections.

2.1 Dynamic OD Estimation and Prediction Models

Extensive experience with static OD estimation is available (see Ben-Akiva (1987) and Cascetta and Nguyen (1989) for a review and bibliography), but less with dynamic (or time-dependent) OD estimation. Furthermore, few of the available dynamic OD estimation models can be extended to predict demand for future time intervals which, as was discussed above, is very important for a traffic assignment application.

A comprehensive review of dynamic OD estimation models can be found in Ashok (1996), from which the following review is partially derived. Literature on dynamic OD estimation can be classified into models applicable for two types of networks:

- “Closed” networks, where complete information is available on the entry and exit counts of the network at all points of time, and
- General networks, where available information is limited to the instrumented entries and exits of the network.

2.1.1 “Closed” Networks

A variety of methods have been proposed for dynamic OD estimation for closed networks. Cremer and Keller (1987) proposed several estimators for the simple case of a single intersection, where the turning movements can be considered to be OD flows. Most of these are least squares based methods, with constrained versions of the problem also proposed in order to ensure non-negativity of turning fractions. In addition, recursive algorithms, such as Kalman filtering, have also been proposed (Nihan and Davis, 1987).

Nihan and Davis (1987) also report some experience with estimating freeway OD matrices where an OD flow is taken as the flow between an origin ramp and a destination ramp. Chang and Wu (1994) utilize on- and off-ramp counts as well as main-line counts to estimate the OD matrix. Main-line counts at different points on the freeway are used to obtain approximate estimates of travel times, which are then used to define the dynamic mapping between OD flows and link flows.

2.1.2 General Networks

The approaches mentioned in the previous paragraph all suffer from the major limitation that all entry and exit counts need to be known, which is unlikely to be the case in a realistic network. The methods proposed for general networks are extensions of the static OD matrix estimation problem. Cascetta et al. (1993) obtain estimates of dynamic OD flows by optimizing a two part objective function. The first part seeks to minimize the difference between the estimated OD matrix for an interval and an a priori estimate of the OD matrix for that interval, while the second part seeks to minimize the difference between link volumes predicted by the model when the estimated OD flows are assigned to the network and the link volumes actually measured. Their model, though, cannot be used in a real time application with the presence of information, because it does not offer predictive capabilities.

An alternative model with predictive capability was developed as part of the DRIVE-II DYNA project (Inaudi et al., 1994). Nevertheless, estimation and prediction are dealt with separately. Estimation is based on the work of Cascetta et al. (1993) mentioned above. The estimated values are then used to generate predictions by a separate *filtering* technique; they combine historical and estimated OD information using the concept of “deviations” proposed by Ashok and Ben-Akiva (1993), presented below. This approach has the disadvantage that the prediction component is exogenous to the estimation, resulting in a statistically inefficient estimator.

An approach based on Kalman Filtering has been suggested by Okutani (1987). The set of decision variables or *state vector* is defined as the vector of unknown OD flows. The model includes an autoregressive formulation, in which the state vector for period h is related by correlation factors to state vectors for previous periods. Okutani uses standard linear Kalman Filter theory to get optimal estimates of the state vector for each time interval. Although this model has predictive and updating capabilities and could be used in real-time applications, there are problems with the autoregressive

specification, which fails to capture any structural information on trip making patterns and is limited in capturing temporal interdependencies between OD flows.

Ashok and Ben-Akiva (1993) introduced the notion of *deviations* of OD flows from historical estimates, in order to overcome the inadequacy of Okutani's autoregressive specification for OD flows. While their measurement equation is the same as Okutani's, the state vector is defined in terms of OD deviations that conform to an autoregressive process. Since the historical OD matrices incorporate all the information about structural relationships that drive travel demand, the estimation and prediction process takes into account all the experience gained over many prior estimations and is hence richer in structural content. This model has been implemented on a linear network with very encouraging results. Nevertheless, the model does not only estimate OD flows for the current interval but also updates OD flow estimates for several past intervals, and is therefore computationally very intensive. Later work by Ashok and Ben-Akiva (1994) showed that by appropriate approximations in the measurement equation, significant computational savings can be achieved at almost no accuracy loss. Ashok (1996) developed and tested an approximate OD estimation and prediction formulation, based on estimating each OD flow only once –the first time it is measured. This is based on the observation that much of the information about an OD flow is likely to be provided the first time it is counted. The algorithm gives good results, provides the capability for prediction of future OD matrices and has the potential for use in a real-time application.

2.1.3 Conclusion

The pre-trip demand simulator requires a dynamic OD estimation model with prediction capabilities, applicable to networks with incomplete entry and exit link count information. Furthermore, the model needs to be suited for real-time applications, i.e. its computational cost should not be prohibitive. From the above review, it is concluded that the only available model that concentrates all these features is the approximate model by Ashok (1996). Furthermore, the model has been tested and given promising results. Therefore, as is evident in chapter 3, this model is selected and implemented in the demand simulator.

2.2 Behavioral models

A literature review of existing behavioral models is attempted in this section. Behavioral models attempt to capture the travel decisions of individual drivers. The reviewed models are classified into:

- Departure time choice,
- Mode choice,
- Route choice, and
- Joint choice models.

2.2.1 Departure time choice

Departure time choice has been related empirically to the cost of early or late arrival relative to some preferred arrival time. Initial models assumed a deterministic departure time choice (see Small (1995) for a review). Probabilistic demand models have been developed by various researchers such as de Palma et al. (1983) and Ben-Akiva et al. (1984) who presented departure time choice as a general continuous logit model, where the set of alternatives is assumed to be continuous. Mahmassani and Chang (1987) applied the bounded rational user response concept to departure time choice introducing the notion of an indifference band of tolerable arrival delay. In the scientific literature, the concept of *rational behavior* is used to describe a consistent and calculated decision process in which the individual follows his or her own objectives (Ben-Akiva and Lerman, 1985). Simon (1957) developed the distinction between *perfect* and *bounded* rationality. Unlike perfect rationality, bounded rationality recognizes the constraints on the decision process that arise from limitations of human beings as problem solvers with limited information-processing capabilities.

More recently Noland and Small (1995) developed a model that incorporates the effects of uncertainty of travel times on travelers' departure time choice. The model includes penalties for late as well as early arrival and captures temporal variations in congestion. Jou and Mahmassani (1994) calibrated a Poisson regression model on the daily departure time switching frequency. Factors affecting switching decisions included tolerance in being late at the destination, travel time fluctuations, and travelers' socioeconomic characteristics.

2.2.2 Mode choice

The most common mode choice models are random utility models (RUM) assessing the choice between car and transit alternatives. Examples of estimated models are those in Ben-Akiva and Lerman (1985), Bradley et al. (1991), and Badoe and Miller (1995). Factors affecting mode choice include purpose of trips, in-vehicle travel time,

out-of-vehicle travel time, travel cost, car availability, destination and travelers' socioeconomic characteristics.

2.2.3 Route choice

A comprehensive literature review of route choice models is presented by Bovy and Stern (1990). The factors that in general affect route choice are travel time, travel distance, number of traffic lights along the route, congestion, length of the route on highways, road quality, and presence of commercial areas. Route choice models belong to one or more of the following major categories:

- Random utility models, and
- Production rule models.

Random utility models assume that travelers maximize their travel utility. If the error term in the utility function is independent and identically Gumbel distributed, then the model becomes a logit model. An example is presented by Ben-Akiva et al. (1984) who estimated a route choice model that captures the effects of travel time, distance, number of signals, and highway distance. If the error term of the utility function is normally distributed then the model becomes a probit model. Yai and Iwakura (1994) estimated a transit route choice probit model with the independent variables being cost, access time, egress time, walking, congestion rate, and transfer time. Cascetta et al. (1995) proposed a modified specification of the logit model, named C-logit, which overcomes the main shortcoming of multinomial logit (MNL), i.e. unrealistic path probabilities for paths sharing a number of links. Furthermore, the proposed model has an easily computable closed form, thus overcoming the drawback of the probit model which has no closed analytical form. The C-logit model requires explicit path enumeration. The basic idea is to deal with similarities among overlapping paths through an additional "cost" attribute, named commonality factor, in the utility function of a logit model rather than through covariances of the random residuals of perceived path utilities as assumed by probit models.

Production rule systems are based on the assumption that decisional behavior in a certain context (e.g. route choice) can be described as a system of IF-THEN rules. Lotan and Koutsopoulos (1993) have explored route choice processes and driver perceptions in the presence of information using concepts from fuzzy set theory, approximate reasoning and fuzzy control. Variables affecting drivers' behavior are observations on current travel time perceptions, congestion levels, and accidents.

2.2.4 Joint choice

Mahmassani and Herman (1984) and Ben-Akiva et al. (1991) developed joint departure time and route choice models, in which drivers are assumed to minimize the generalized cost incurred. This cost is a function of travel time, early arrival, late arrival, and travel cost. Cascetta and Biggiero (1992) calibrated a joint departure time and path choice model as a function of travel time, safety and comfort. Jou and Mahmassani (1993) presented a bounded rational departure time and route switching model as a function of travel time fluctuations, early and late arrival, and socioeconomic characteristics. Early or late arrival is defined as the difference between the arrival time given by a choice alternative and a desirable arrival time, e.g. official work start time.

In the context of real-time applications, though, a joint model that will be able to capture departure time, mode and route choice, as well as response of drivers to information is required. Drivers that do not receive information rely on historical perceptions and experiences and, therefore, follow their habitual travel pattern. Travelers that are provided with guidance, may decide not to travel, or adjust their destination, departure time, mode, or route choice accordingly.

Khattak et al. (1996) model pre-trip travel response to ATIS. They investigate the influence of unexpected and expected congestion, various types and quality of information received about congestion, and travelers' experience with congestion and related information on pre-trip travel decisions (trip cancellation, departure time, mode and route choice). They examine the effects of various factors, such as source(s) of congestion, information, trip characteristics, and route attributes on traveler response to unexpected congestion. The model is formulated as a multinomial logit. They also model the relationship between traveler response to qualitative, quantitative, predictive and prescriptive information in a hypothetical ATIS context in combination with actual behavior. However, the developed model is very specific to the context from which the data was collected.

2.2.5 Conclusion

From the above review, it appears that none of the available models independently provide all the functionality that is required for the demand simulator. The only model that captures departure time, mode and route choice is the last one presented above, by Khattak et al. (1996). However, this model requires the knowledge of information that is very context specific and, consequently, is not general enough for use in the demand simulator. Therefore, elements from this and other models will be combined into new

models that will be specially designed to meet the requirements of the demand simulator in terms of both comprehensiveness and generality.

3. Design of the Pre-Trip Demand Simulator

In this section, the design of the demand simulator is presented and motivated. The overall framework is presented in the first section. The variable definition, the time discretization, as well as the choice set generation procedure are presented in the following sections. Finally, the various components of the system are presented and their corresponding models and algorithms are described.

3.1 Framework

The input to the demand simulator includes:

- Aggregate historical OD matrices,
- Real-time information and guidance, and
- Link counts.

The historical OD matrices give the average number of trips for each OD pair in some past period, for which the OD matrices have been constructed. These OD matrices are provided from a historical database. This database can be constructed from results of estimations conducted in previous days, and can be stratified by day-of-week, type of weather, special events, etc. Real time information and guidance made accessible to the travelers via a number of ATIS media, including radio, telephone and the Internet can be provided from the guidance generation module of the DTA system used by the ATIS in place. Finally, link counts are provided by the surveillance system as direct measurements from traffic sensors.

The pre-trip demand simulator incorporates explicitly the effect of pre-trip information and guidance provision to update the historical OD matrices prior to OD estimation, in order to capture the drivers' response to real-time information available at the pre-trip stage. Although the OD estimation model is applied on aggregate OD matrices, the individual choice of drivers is captured by disaggregate behavioral models. Thus, variations of travel behavior can be captured at the individual driver level. This is important because it allows the simulator to use individual driver characteristics to capture travel behavior in a potentially more accurate fashion, rather than being limited in capturing behavior at the OD level. In order to be able to use disaggregate models, though, the demand simulator needs to disaggregate the historical OD matrices into a population of drivers, which will be updated and subsequently aggregated to produce the updated OD matrices that will be used as input to the OD estimation model.

The overall structure of the proposed pre-trip demand simulator is presented in Figure 3.

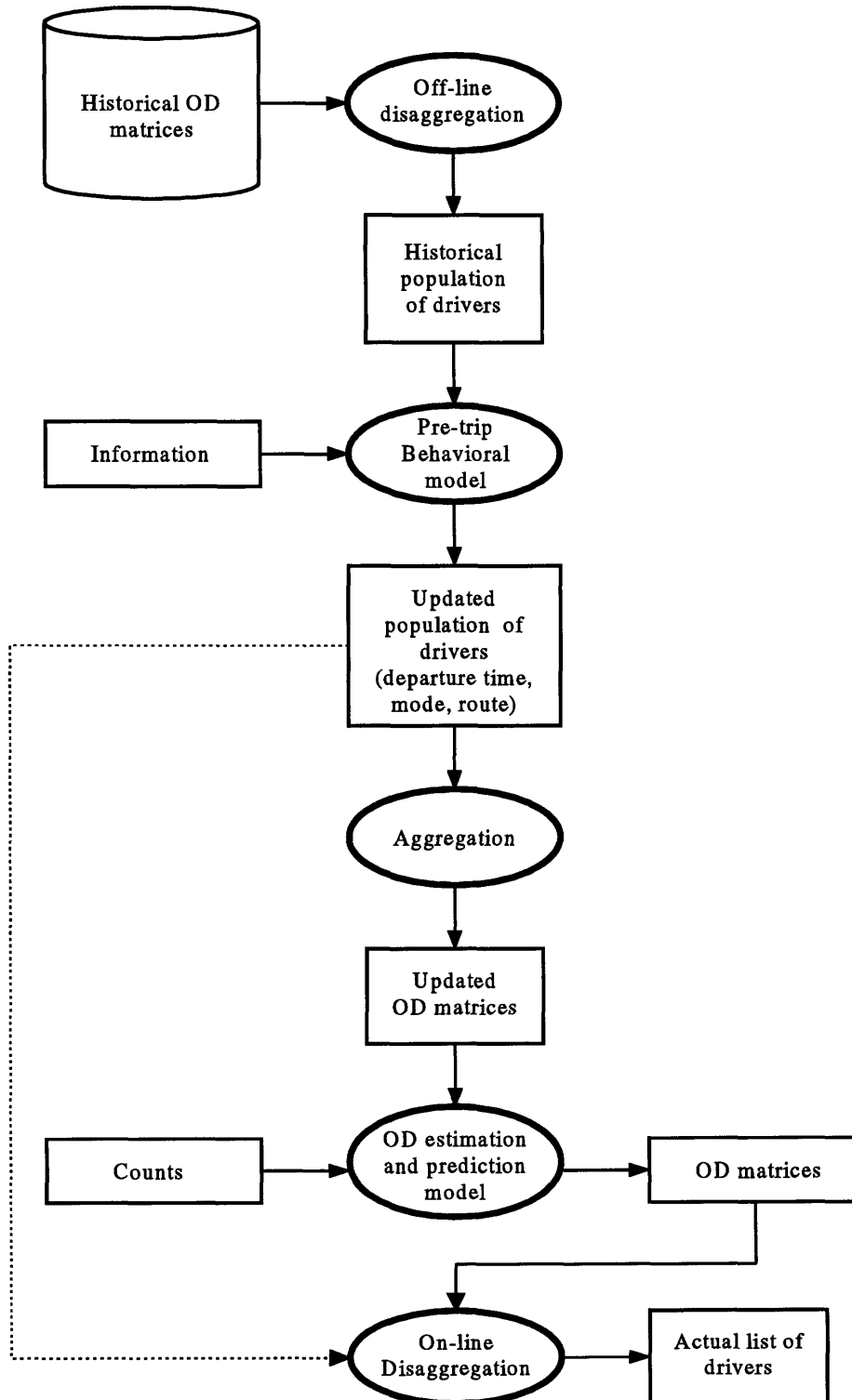


Figure 3. Pre-trip Demand Simulator

The functionality of the pre-trip demand simulator can be separated in two main functions:

- Travel behavior update in response to information, and
- Dynamic OD estimation and prediction.

The travel behavior model is disaggregate and is applied on individual drivers. On the other hand, historical demand is available as aggregate OD matrices. In order to be able to use the disaggregate behavioral model, the demand simulator transforms the aggregate OD matrices into a disaggregate population of drivers, upon which the behavioral model is applied. The off-line disaggregation is responsible for this. The behavioral model then updates the travel choices of each traveler, using the available real time information to determine if they will stick to their initial travel pattern or they will change departure time, mode or route, including canceling their trip altogether. After the travel behavior of each driver has been updated to reflect the available information and guidance, the aggregate OD estimation model is applied on the updated population of drivers. Therefore, the population of drivers is aggregated to a set of updated OD matrices (one OD matrix is generated for each departure time interval).

At this stage, the historical OD matrices have been updated to reflect the response of the drivers to available real-time information. If this step had been omitted, then the simulator would ignore significant information that could potentially affect its success in estimating current OD matrices. After the behavioral update has been completed, the OD estimation model accepts the updated OD matrices as input and uses traffic counts from the surveillance system to estimate aggregate demand for the current interval in the form of an estimated OD matrix. The OD prediction model makes an assumption about time evolution of demand to predict OD matrices for a given number of future intervals.

The pre-trip demand simulator is designed with the capability to be used in a wide range of applications, as discussed in Section 1.3. Depending on the nature and the requirements of each application, the output of the module can be either:

- Aggregate demand, or
- Disaggregate demand.

If aggregate demand is required as output, then no further operation is performed and the estimated and predicted OD matrices are the desired output. On the other hand, if disaggregate demand is required, then the estimated and predicted OD matrices are disaggregated to a list of drivers by an additional on-line disaggregation component. This

procedure uses the previously generated updated population of drivers as basis, and adds or removes drivers to reflect the OD estimation results. For example, if for a given OD pair the final estimated flow is 100 drivers, but in the updated population of drivers there are only 95, then five additional drivers will be generated in the same fashion that was used in the off-line disaggregation. Similarly, if there are 105 drivers in the updated population for that OD pair and time interval, then five of them will be selected, using Monte Carlo simulation, and removed from the population of drivers.

In the remainder of the chapter, the variables and parameters that are used in the simulator are defined and the time discretization and the choice set generation procedure are presented. Furthermore, the major components of the demand simulator are described in more detail, including the models and the algorithms that are used.

3.2 Variable and Parameter Definition

The pre-trip demand simulator uses a number of variables and parameters in the specifications of the disaggregate travel behavior models. These are presented in this section. The variables are summarized in Table 1.

In the following definitions superscript H refers to historical information, whereas superscript I refers to information provided by the information system. Subscript h refers to departure time interval h and subscript p refers to path p . Finally, prime ($'$) is used to denote habitual, e.g. h' is the habitual departure time interval and p' is the habitual path.

Abbreviation	Variable
tt_{hp}^H	historical travel time for departure time interval h and path p
tt_m^H	historical travel time for <i>other</i> (than car) mode
tt_{hp}^I	travel time provided by the information system for departure time h and path p
$tt_{hp'}^I$	travel time provided by the information system for departure time h and habitual path p'
$tt_{h'p}^I$	travel time provided by the information system for habitual departure time h' and path p
$tt_{h'p'}^I$	travel time provided by the information system for habitual departure time h' and habitual path p'
dt_h	departure time of traveler departing in interval h

at_{hp}^I	arrival time of traveler departing in interval h on path p
$at_{h'p}^I$	arrival time of traveler departing in habitual interval h' on path p
$at_{hp}^{I'}$	arrival time of traveler departing in interval h on habitual path p'
$at_{h'p}^{I'}$	arrival time of traveler departing in habitual interval h' on habitual path p'
$at_{h'p}^H$	habitual arrival time for traveler with habitual interval h' and path p'
T	time that the information is sent to the traveler
VOT_{lo}	low value of time indicator
VOT_{med}	medium value of time indicator
VOT_{hi}	high value of time indicator
P_w	work as trip purpose (0-1 dummy variable)
P_l	leisure as trip purpose (0-1 dummy variable)
CF_p	commonality factor for path p
l_p	length of path p
w_p	length of path p in highway
c_p	out-of-pocket monetary cost for path p
s_p	number of signalized intersections in path p
f_p	number of left turns on current path p

Table 1. List of variables for travel behavior models

The historical travel time tt_{hp}^H is the average travel time observed for departure time interval h and path p over prior observations. The variable tt_m^H is the minimum from the historical travel times for all alternative non-driving modes. In other words, this is assumed to be the historical travel time of the alternative mode that a driver is considering. The travel time tt_{hp}^I is the travel time that the information system gives to drivers departing during interval h on route p . Variables tt_{hp}^I , $tt_{h'p}^I$, and $tt_{h'p}^{I'}$ can be interpreted similarly.

The departure time dt_h is the actual departure time of a traveler departing in interval h . It is calculated by assuming uniform departure rate for the departure time

interval and using Monte Carlo distribution to draw the exact departure time within that interval. The arrival time of a driver departing in departure time interval h on path p is given by $at_{hp}^I = dt_h + tt_{hp}^I$. When the habitual departure time h' or habitual path p' are considered, the arrival time variable becomes $at_{h'p'}^I$, $at_{hp'}^I$, and $at_{h'p}^I$, accordingly. Similarly, the habitual arrival time for a driver departing in departure time interval h on path p is given by $at_{h'p'}^H = dt_{h'} + tt_{h'p'}^H$.

The time that the traveler receives the information is denoted by T . There are also three indicator variables VOT_{lo} , VOT_{med} , VOT_{hi} to reflect the value of time of the driver (low, medium, or high). Only one of these variables is equal to one for each individual, while the other two are zero. In addition, there are two 0-1 dummy variables to capture the trip purpose of the driver (work or leisure). The variable that corresponds to other trips purposes is used as reference and, therefore, is not included in the utility functions. The selection of the referent alternative has no effect of the outcome of the model. Changes in the referent only shift the values of the estimated constants, preserving their difference.

The commonality factor CF_p (Cascetta et al., 1996) for path p is an additional “cost” attribute in the utility of a logit model, which deals with the similarities among overlapping paths (this will be described in Section 3.5.1). The commonality factor does not affect travel behavior. Its role in the model is to overcome the limitation of the IIA property. The length of path p , l_p , the length of path p that is on highways, w_p , the monetary cost of path p , c_p , the number of signalized intersections in path p , s_p , and the number of left turns in path p , f_p , are attributes of the paths. The same variables with subscript p' are used to refer to the habitual path.

Functions of the aforementioned variables appear in the models, as well. The variable $dt_{h'} - T$ is used to describe how early the traveler receives the information (how much earlier than the habitual departure time). For example, information received two hours before the habitual departure time may not be as relevant as information received 15 minutes before. In the specification of the utilities this variable appears as $\max(dt_{h'} - T, 0)$. This is to ensure that the variable takes a value of zero if the information comes after the departure (since then it is not relevant for pre-trip decisions).

The variables $at_{h'p'}^H - at_{hp}^I$ and $at_{hp'}^I - at_{h'p}^H$, express the deviation in the arrival time of the traveler departing at departure time interval h on path p , relative to the habitual arrival time for the habitual arrival time and path. This, in association with the possible early or late arrival penalty, captures the effect of arrival time variations in the

travelers' decisions. The variables are introduced in the specification of the utilities as the following terms:

- $\max(at_{h'p'}^H - at_{hp}^I, 0)$, and
- $\max(at_{hp}^I - at_{h'p'}^H, 0)$.

The first term captures the early arrival. It is zero if the arrival time is later than the habitual arrival time (i.e. $at_{h'p'}^H < at_{hp}^I$ - the user is expected to be late). Otherwise, it is equal to the amount of time that the driver is expected to be early. Similarly, the second term captures the late arrival. It is zero if the expected arrival time is earlier than the habitual arrival time (i.e. $at_{h'p'}^H > at_{hp}^I$ - the user is expected to be early). Otherwise, it is equal to the amount of time that the driver is expected to be late. The use of two such variables allows for designing different penalties for late and early arrivals. Terms $\max(at_{h'p'}^H - at_{hp}^I, 0)$, $\max(at_{h'p'}^H - at_{hp}^I, 0)$, $\max(at_{hp}^I - at_{h'p'}^H, 0)$, and $\max(at_{hp}^I - at_{h'p'}^H, 0)$ are used similarly in the utility functions.

Tables 2, 3, and 4 present the coefficients of the models. The coefficients β and the structural coefficients θ appear in the utility and probability equations as well. There are superscripts and subscripts to these parameters that need to be defined. Superscripts specify the model to which the coefficients refer to. These superscripts are defined in Table 1. Some β coefficients are specific to the corresponding alternative, denoted by the appropriate subscript. These subscripts are defined in Table 3. Furthermore, all of these variables are associated with a numerical identifier, ranging from 0 to 14. Finally, the structural coefficients, θ , have a subscript that defines the alternative they are associated with. These subscripts are presented in Table 4.

Coefficient with superscript	model indicated by superscript
β^h	habitual behavior
β^d	behavior based on descriptive information
β^p	behavior based on prescriptive information

Table 2. List of superscripts of coefficients for travel behavior models

coefficient with subscript	alternative indicated by subscript to β
β_p	path p (in the utility of the paths for the habitual behavioral model)
β_{DNC}	do not change
β_M	change mode
β_{DT}	change departure time
β_{DT_h}	change to departure time interval h
β_{CP}	change path
β_{CP_p}	change to path p
β_{CDTP}	change departure time and path
$\beta_{CDT_hP_p}$	change to departure time h and path p

Table 3. List of subscripts of coefficients for travel behavior models

coefficient with subscript	alternative indicated by subscripts to θ
θ_M	change mode
θ_{DT}	change departure time
θ_P	change route
θ_{DTP}	change departure time interval and route
θ_{CT}	cancel trip

Table 4. List of subscripts of structural coefficients for travel behavior models

3.3 Time Discretization

Time is a very important dimension in simulation environments. The demand simulator is a *discrete step* simulator, i.e. the state of the system is updated in a series of steps, each corresponding to some fixed time interval (Lerman, 1993). The demand simulator uses two such time discretizations intervals:

- Departure time interval, and
- Estimation interval.

The departure time choice of individual drivers is modeled as a discrete choice. Therefore, a time discretization is necessary in order to define the choice set. This time interval needs to be small enough in order to realistically represent the drivers' change in departure time in response to information. Nevertheless, very small departure time intervals would increase the computational strain and the data requirements of the simulator. Therefore, in order to make a decision for the departure time interval length, this tradeoff should be examined and the length of the departure time interval should be selected based on the needs of the application.

Similarly, the OD estimation is performed for discrete intervals of time. Therefore, an interval that refers to the time period for which the demand is estimated is also necessary. The selection of the length of the estimation time interval depends on the application. As it will be explained in Sections 3.5 and 3.7, the two time intervals do not need to coincide. In such a case, a procedure that maps one time discretization into the other is applied (see sections 3.5 and 3.7).

3.4 Choice Set Generation

The behavioral models that are incorporated in the demand simulator are discrete choice models and require that the individual drivers' choice sets are explicitly specified. The travel decisions that the demand simulator considers are:

- Mode choice,
- Departure time choice,
- Route choice, and
- Trip cancellation.

Regarding mode, only the decision whether to drive or switch to public transit is considered and, therefore, the alternatives are clear. Since the demand simulator focuses on private transportation, only travelers switching from the automobile to transit are captured. Similarly, for the trip cancellation choice, the driver may choose whether to make the trip or not. Regarding departure time choice, the feasibility of the time intervals is bounded by the decision time, since an individual can not decide to depart earlier than the decision time. Since the habitual departure time interval is captured by the *do not change* alternative, the departure times that are considered for the *change departure time* alternatives do not include it. In the case of path choice, the choice set of an individual is considered to comprise from all paths connecting the origin and destination of interest. Again, the *change path* alternatives do not include the habitual path, since it is covered by

the *do not change* alternative. In the case of both departure time and path change, the choice set is comprised from all possible combinations of the time intervals in the departure time choice set with the paths in the path choice set. The only combinations that are excluded from the choice set are those that include the habitual departure time, the habitual path, or both.

In practical applications, it may be desirable to keep the number of alternatives to a reasonable size. In this case, the choice set can be filtered to exclude paths and departure time intervals that are very unlikely to be selected. Paths with very high travel times can be removed from the choice set, since they are unlikely to be chosen. Similarly, departure time intervals that are a lot later than the habitual departure time may also be safely removed from the choice set, based on the assumption that the travelers need to complete their trip within some time frame. Nevertheless, one needs to be cautious in such an elimination procedure, in order not to exclude alternatives that may become attractive under certain conditions, such as an incident on a generally preferable route.

3.5 Off-line Disaggregation

The role of the *off-line disaggregation* component is to generate a population of drivers from the historical OD matrices. A number of socioeconomic characteristics, such as value of time, and trip characteristics, such as trip purpose, are generated and assigned to each driver. Origin, destination, departure time interval and mode are assigned to the drivers using information from the OD matrices. The origin and the destination come from the particular OD pair for which the driver is generated, the departure time interval is the interval to which the OD matrix corresponds and the mode is by default the car (since it is assumed that the OD matrices contain only car trips). Furthermore, a habitual behavior model is applied to each driver in order to generate habitual travel behavior based on historical information. This is presented in sections 3.5.1 and 3.5.2. Section 3.5.3 presents the entire off-line disaggregation algorithm.

3.5.1 Habitual Route Model Structure

The behavioral model that is used to provide choice probabilities for each path is the C-logit model, a modified multinomial logit (MNL) model, proposed by Cascetta (1993). The model specification of the C-logit is that of a MNL with a modified utility to account for overlapping paths. The model overcomes the main shortcoming of MNL where unrealistic choice probabilities result for paths sharing a number of links. For example, consider the simple network represented in Figure 4. This network is composed

from three nodes, A, B and C, and four links. There are three alternative routes connecting A and B. Assuming equal utility for the three alternative routes, a logit model would result in equal choice probabilities, 0.333, for each alternative route. This is not realistic, since paths 1 and 2 are practically identical. In the C-logit model this is captured by the introduction of an additional “cost” parameter in the utility specification.

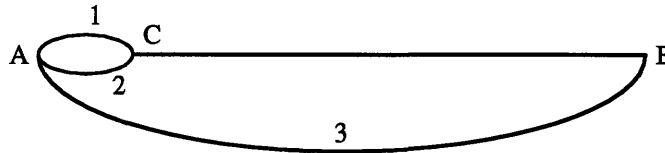


Figure 4. Simple road network with overlapping paths

The model requires explicit path enumeration. The structure of the model is presented in Figure 5.

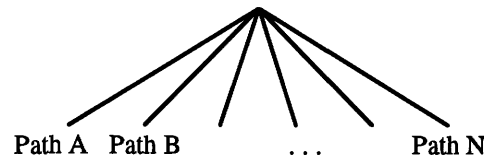


Figure 5. Structure of the C-logit model

The choice alternatives are the paths that connect the origin and the destination of each driver and are included in the choice set.

In the C-logit model, the effect of overlapping paths is taken into account by the introduction of a term called “commonality factor” in the utility function. The commonality factor can be interpreted as the degree of overlapping of a path with the other paths in the choice set of the individual. The function of the commonality factor is to decrease the utility of heavily overlapping paths by introducing an additional “cost” in the utility of the paths, which is higher for overlapping paths. The reader is referred to Cascetta et al. (1996) for a more complete presentation of the commonality factor. In this reference, one can find alternative specifications, as well as a behavioral interpretation of the commonality factor. For the purposes of this application, the following specification has been selected:

$$CF_p = \ln \sum_{j \in p} \omega_{jp} N_j \quad (1)$$

where:

CF_p is the commonality factor for path p ,

ω_{jp} is the proportional weight of link j in path p , defined as the fraction of the total length of path p which can be attributed to link j (the link weights sum up to one for all links of a given path), and

N_j is the number of paths connecting the same OD pair, which share link j .

3.5.2 Habitual Route Model Specification

Using the notation defined in Section 3.2, the systematic utility (from now on, the term *utility* will be used instead of *systematic utility* for the sake of brevity) specification for route p is given by the following formula:

$$V(p) = \beta_{p,0}^h + \beta_1^h VOT_{lo} tt_p^H + \beta_2^h VOT_{med} tt_p^H + \beta_3^h VOT_{lo} tt_p^H + \beta_4^h CF_p + \beta_5^h l_p + \beta_6^h \omega_p + \beta_7^h c_p + \beta_8^h s_p + \beta_9^h f_p + \beta_{10}^h P_w + \beta_{11}^h P_l \quad (2)$$

(From now on, the term *utility* will be used instead of *systematic utility* for the sake of brevity)

The alternative specific constant $\beta_{p,0}^h$, and the alternative specific dummy variables P_w , and P_l can only appear in $N-1$ utilities. Without loss of generality, they are omitted from the utility of the N th alternative path, which acts as a referent. The selection of the referent alternative has no effect on the outcome of the model. Changes in the referent only shift the values of the estimated constants, preserving their difference.

Finally, the probability that an individual will choose path p from the choice set P is:

$$P(p) = \frac{e^{V(p)}}{\sum_{p' \in P} e^{V(p')}} \quad (3)$$

3.5.3 Algorithm

The input to the disaggregation component includes:

- Historical OD matrices, and
- Information about the socioeconomic and trip characteristics of the drivers, such as the value of time and the trip purpose.

As mentioned earlier, historical OD matrices are available from a historical database. The information about the socioeconomic characteristics of the drivers can be available in different forms and varying levels of detail. Population-wide distributions are the minimum requirement, however origin, destination, or even OD pair-specific distributions are preferred, if available, since they allow for capturing the spatial variation in the behavior of the travelers. Population-wide distributions give the proportion of the population that is characterized by each value of an attribute. Similarly, distributions of a subset of the population, provide these proportions for that subset, e.g. the travelers from origin A to destination B. The output of the disaggregation component is a population of drivers, each characterized by a list of socioeconomic and trip characteristics, and a habitual travel behavior.

In the disaggregation the historical OD demand is disaggregated into a historical population of drivers, corresponding to the OD flows in the matrices. It is assumed that the OD matrix represents the number of trips (i.e. vehicles or drivers) and not total passengers traveling from an origin to a destination. Therefore, for each OD cell of the matrix being disaggregated, a number of drivers equal to the OD flow is generated. The origin and the destination of each driver are determined from the OD pair that they were derived from. Furthermore, the departure time interval corresponding to the disaggregated OD matrix is assigned as the habitual departure time interval of all the drivers that are generated from the disaggregation of the OD matrix. Also, it is assumed that the historical OD matrices represent only those trips that are made by private automobile. Therefore, the habitual mode for all generated drivers is by default the automobile.

Besides origin, destination, habitual departure time, and mode, the individual drivers need to have socioeconomic and trip characteristics as well as a habitual travel behavior assigned to them. The socioeconomic and trip characteristics are necessary for the determination of the utility of each alternative in the behavioral models. Furthermore, the habitual travel behavior is used as the base alternative for the drivers, which they decide if they want to change or not, based on the available information. Monte Carlo simulation (also known as *sampling from probability distributions* or *random variate generation*; refer to Winston (1994) for more details) is used to assign characteristics to the drivers, based on the available distributions. As mentioned above, the more area-specific the distributions are, the more representative the generated characteristic will be. The characteristics that are assigned to each generated driver are:

- Value of time, which is categorically represented by low, medium or high (in this thesis), and

- Trip purpose, which is classified into work or leisure (in this thesis).

These characteristics have been selected from a larger set of possible characteristics (including e.g. age and gender), because they appear to be more suitable to the application. This set of characteristics is more directly relevant to the drivers' choice and sensitive to information. Furthermore, it provides a good tradeoff between complexity (number of characteristics) and ability to capture the drivers' behavior.

After the assignment of the characteristics to the drivers, the only attribute that is missing is the habitual path (since socioeconomic and trip characteristics, origin, destination, habitual mode and departure time have already been assigned to them). The disaggregation component applies the habitual behavior model presented earlier in this section to generate habitual paths for the drivers. The procedure is presented in Figure 6.

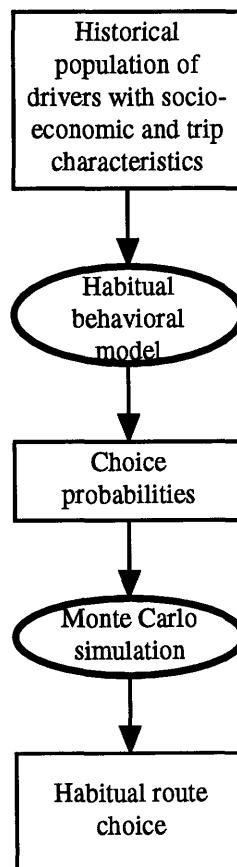


Figure 6. Habitual path choice generation

A set of enumerated paths is assumed for each OD pair. This choice set can be generated off-line once (and perhaps be re-computed occasionally, to reflect new information collected from the network). The habitual behavioral model uses the characteristics of each individual driver, along with historical information on the path

attributes (e.g. travel time, length, monetary cost) to generate the probabilities that each path will be chosen. Once the probabilities are known, Monte Carlo simulation is used to randomly select one. The selection is assigned to the driver as the habitual route choice.

3.6 Pre-Trip Behavior Update

At this stage of the pre-trip demand simulation process, a population of individual drivers with corresponding socioeconomic and trip characteristics, as well as habitual choices on departure time, mode and route, is available. These choices reflect a priori decisions which can be updated based on real-time information and guidance. The pre-trip behavior update applies a behavioral model to each individual driver in the updated historical population to capture their travel behavior in response to available information. The drivers may decide to change departure time, path, mode, a combination of these, or even cancel their trip. It is assumed, that the drivers' destination is fixed and it cannot be changed in response to available information.

There are two types of information:

- Descriptive, and
- Prescriptive

Descriptive information reflects traffic conditions on the network and leaves the responsibility of using this information in deciding on a travel choice to the driver. On the other hand, prescriptive information provides specific recommendations to the user about travel decisions (e.g. leave one interval earlier, change to route x , change to route x and leave one interval later, or switch to public transit) and the user decides whether to comply or not. Different models have been formulated depending on the available information. The descriptive model is presented in sections 3.6.1 and 3.6.2, whereas the prescriptive is described in sections 3.6.3 and 3.6.4.

3.6.1 Descriptive Model Structure

The descriptive behavioral model is formulated as a nested logit model. The nested logit model was first derived as a generalization of the joint logit model by Ben-Akiva (1973) and subsequently formalized in different ways based on utility maximization by several researchers. The nested logit model structure is hierarchical in nature. This allows for grouping subsets of alternatives at a particular level of the hierarchy reflecting their similarity in comparison to alternatives outside the group. The utility of the alternatives of a level is represented by a combination of the utility of the

alternatives of the lower level. Thus, the choice at each level can be expressed in term of the alternatives of that level only. The utility of each alternative is expressed with a term that attempts to capture the utility of the best alternative of the lower level. This term is called expected maximum utility or systematic component of the maximum utility of a subset of alternatives and is equivalent with the terms *inclusive value* (McFadden, 1978) and *accessibility* (Ben-Akiva and Lerman, 1979). The random components or random parts of the utility are called disturbances. The relation of the disturbance of the lower level alternatives with the disturbance of the upper level alternative are represented by a *structural coefficient* θ , which takes values between 0 and 1. If the structural coefficient is equal to one, then it means that the disturbance is the same for the two levels and, therefore, the model simplifies to a MNL. For a more rigorous explanation of the structural coefficient, the reader is referred to Ben-Akiva and Lerman (1985).

The choice tree for the descriptive model is presented in Figure 7. At the first level, the traveler decides whether to change habitual behavior or not. If the user decides to change, then the next level choice is whether to change mode, departure time, path, both departure time and path or cancel the trip altogether. The mode change and trip cancellation alternatives do not require any further decisions. In the case of departure time change, the lower level alternatives are the departure time intervals in the choice set of the individual. Similarly, in the case of path change, the lower level alternatives are the paths in the choice set of the individual. In the case of both departure time and path choice, though, more than one tree structures can be used. For example, the individual may first make a departure time choice and then, having decided on departure time interval, make the path choice, or first make the path choice and then choose a departure time interval. Neither of these approaches are adopted in this model. It is assumed that the individual driver makes both choices simultaneously. Therefore, only one subsequent level is required and the choice set is composed by the combination of all departure time intervals and paths in the choice set of the individual (see Section 3.4 for more details). This approach is selected mainly due to the lack of data and relative ease of estimation. This tree structure is an arbitrary choice and may be revisited. A comparison of alternative tree structures would be an interesting topic for further research, but it is out of the scope of this thesis.

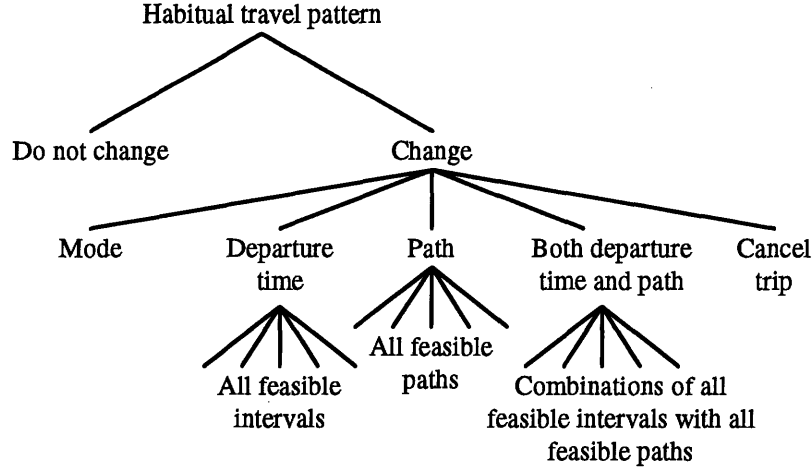


Figure 7. Pre-trip choice tree in the case of descriptive information

3.6.2 Descriptive Model Specification

The utility of the *do not change* (DNC) alternative is:

$$\begin{aligned}
 V(DNC) = & \beta_1^d VOT_{lo} tt_{h'p'}^I + \beta_2^d VOT_{med} tt_{h'p'}^I + \beta_3^d VOT_{hi} tt_{h'p'}^I + \\
 & \beta_4^d CF_{p'} + \beta_5^d l_{p'} + \beta_6^d \omega_{p'} + \beta_7^d c_{p'} + \beta_8^d s_{p'} + \beta_9^d f_{p'} + \\
 & \beta_{10}^d \max(dt_k - T, 0)
 \end{aligned} \quad (4)$$

The utility for the *change* (C) alternative is the *expected maximum utility* of the set of lower level alternatives:

- Change mode (CM),
- Change departure time (CDT),
- Change path (CP),
- Change departure time and path (CDTP), and
- Cancel trip (CT).

The utility of the *change mode* (CM) alternative is:

$$\begin{aligned}
 V(CM) = & \beta_{M,0}^d + \beta_1^d VOT_{lo} tt_m^H + \beta_2^d VOT_{med} tt_m^H + \\
 & \beta_3^d VOT_{hi} tt_m^H + \beta_{CM,13}^d P_\omega + \beta_{CM,14}^d P_l
 \end{aligned} \quad (5)$$

The utility of the *change departure time* (CDT) is the *expected maximum utility* of the lower level alternatives, i.e. the intervals in the choice set of the individual. The utility

of a lower level alternative, i.e. change to departure time interval h of the departure time interval choice set H , for the habitual route choice p' is:

$$\begin{aligned}
V(CDT_h) = & \beta_{CDT_h,0}^d + \beta_1^d VOT_{lo} tt_{hp'}^I + \beta_2^d VOT_{med} tt_{hp'}^I + \beta_3^d VOT_{hi} tt_{hp'}^I + \\
& \beta_{11}^d \max(at_{h'p'}^H - at_{hp'}^I, 0) + \beta_{12}^d \max(at_{hp'}^I - at_{h'p'}^H, 0) + \\
& \beta_{CDT_h,13}^d P_\omega + \beta_{CDT_h,14}^d P_l
\end{aligned} \tag{6}$$

If there is a total of N departure time intervals in the choice set, the alternative specific constant $\beta_{CDT_h,0}^d$ can only appear in $N-1$ utilities. Without loss in generality, it is omitted from the N th utility.

The utility of the *change departure time* alternative is:

$$V(CDT) = \theta_{CDT} \log \sum_{h \in H} e^{V(h)/\theta_{CDT}} \tag{7}$$

where H is the departure time interval choice set for the individual and θ_{CDT} is the structural coefficient associated with the change departure time alternative.

The utility of the *change path (CP)* alternative is the *expected maximum utility* of the lower level alternatives, i.e. change to path p in path choice set P . The utility of a lower level alternative, i.e. change to a feasible path p in the choice set P is:

$$\begin{aligned}
V(CP_p) = & \beta_{CP_p,0}^d + \beta_1^d VOT_{lo} tt_{h'p}^I + \beta_2^d VOT_{med} tt_{h'p}^I + \beta_3^d VOT_{hi} tt_{h'p}^I + \\
& \beta_4^d CF_p + \beta_5^d l_p + \beta_6^d \omega_p + \beta_7^d c_p + \beta_8^d s_p + \beta_9^d f_p + \\
& \beta_{11}^d \max(at_{h'p'}^H - at_{h'p}^I, 0) + \beta_{12}^d \max(at_{h'p}^I - at_{h'p'}^H, 0) + \\
& \beta_{CP_p,13}^d P_\omega + \beta_{CP_p,14}^d P_l
\end{aligned} \tag{8}$$

If there is a total of N feasible paths in the choice set P , the alternative specific constant $\beta_{CP_p,0}^d$ can only appear in $N-1$ utilities. Without loss in generality, the alternative specific constant is omitted from the N th utility.

The utility of the *change path* alternative is:

$$V(CP) = \theta_{CP} \log \sum_{p \in P} e^{V(p)/\theta_{CP}} \tag{9}$$

where P is the departure time interval choice set for the individual and θ_{CP} is the structural coefficient associated with the change path alternative.

The utility of the *change departure time and path (CDTP)* alternative is the *expected maximum utility* of the lower level alternatives, i.e. change to combination hp of path p in path choice set P with departure time interval h in departure time interval choice set H . The utility of a lower level alternative, i.e. a combination hp of route p and departure time interval h , is:

$$\begin{aligned}
V(CDT_{hP_p}) = & \beta_{CDT_{hP_p},0}^d + \beta_1^d VOT_{lo} tt_{hp}^I + \beta_2^d VOT_{med} tt_{hp}^I + \beta_3^d VOT_{hi} tt_{hp}^I + \\
& \beta_4^d CF_p + \beta_5^d l_p + \beta_6^d \omega_p + \beta_7^d c_p + \beta_8^d s_p + \beta_9^d f_p + \\
& \beta_{11}^d \max(at_{h'p'}^H - at_{hp}^I, 0) + \beta_{12}^d \max(at_{hp}^I - at_{h'p'}^H, 0) + \\
& \beta_{CDT_{hP_p},13}^d P_\omega + \beta_{CDT_{hP_p},14}^d P_l
\end{aligned} \tag{10}$$

If there is a total of N combinations of route and departure time choices, the alternative specific constant $\beta_{CDT_{hP_p},0}^d$ can only appear in $N-1$ utilities. Without loss of generality, the constant is omitted from the N th utility specification.

The utility of the *change departure time and path* alternative is:

$$V(CDTP) = \theta_{CDTP} \log \sum_{hp \in PxH} e^{V(hp)/\theta_{CDTP}} \tag{11}$$

where PxH is the set of all combinations of the departure time intervals in the choice set H with the feasible paths in the choice set P , and θ_{CDTP} is the structural coefficient associated with this alternative.

The utility of the *cancel trip (CT)* alternative is:

$$V(CT) = \beta_{CT,0}^d + \beta_{CT,13}^d P_\omega + \beta_{CT,14}^d P_l \tag{12}$$

With all the lower level utilities already described, the utility for the *change (C)* alternative is:

$$V(C) = \theta_C \log \left(e^{V(M)/\theta_C} + e^{V(DT)/\theta_C} + e^{V(P)/\theta_C} + e^{V(DT+P)/\theta_C} + e^{V(CT)/\theta_C} \right) \tag{13}$$

where θ_C is the structural coefficient associated with the *change* alternative.

Once all the utilities have been specified, the probabilities that each choice will be made can be calculated based on the following formulas. The probability that alternative A will be selected from a choice set S , comprising from all the alternatives in the same level, is:

$$P(A) = \frac{e^{V(A)}}{\sum_{B \in S} e^{V(B)}} \quad (14)$$

whereas, the probability that a lower level alternative a will be chosen from choice set S , provided that the higher level alternative A has been chosen, is:

$$P(a) = P(A) \frac{e^{V(a)}}{\sum_{b \in S} e^{V(b)}} \quad (15)$$

Using the above two formulas, the probability that a driver will not change travel behavior is:

$$P(DNC) = \frac{e^{V(DNC)}}{e^{V(DNC)} + e^{V(C)}} \quad (16)$$

and the probability that a driver will change travel behavior is:

$$P(C) = \frac{e^{V(C)}}{e^{V(DNC)} + e^{V(C)}} \quad (17)$$

The probability that a driver will change mode, provided that the decision to change travel behavior has been made, is:

$$P(CM) = P(C) \frac{e^{V(CM)}}{e^{V(CM)} + e^{V(CDT)} + e^{V(CP)} + e^{V(CDTP)} + e^{V(CT)}} \quad (18)$$

The probability that a driver will change departure time, provided that the decision to change travel behavior has been made, is:

$$P(CDT) = P(C) \frac{e^{V(CDT)}}{e^{V(CM)} + e^{V(CDT)} + e^{V(CP)} + e^{V(CDTP)} + e^{V(CT)}} \quad (19)$$

The probability that a driver will change path, provided that the decision to change travel behavior has been made, is:

$$P(CP) = P(C) \frac{e^{V(CP)}}{e^{V(CM)} + e^{V(CDT)} + e^{V(CP)} + e^{V(CDTP)} + e^{V(CT)}} \quad (20)$$

The probability that a driver will change departure time and path, provided that the decision to change travel behavior has been made, is:

$$P(CDTP) = P(C) \frac{e^{V(CDTP)}}{e^{V(CM)} + e^{V(CDT)} + e^{V(CP)} + e^{V(CDTP)} + e^{V(CT)}} \quad (21)$$

The probability that a driver will cancel his/her trip, provided that the decision to change travel behavior has been made, is:

$$P(CT) = P(C) \frac{e^{V(CT)}}{e^{V(CM)} + e^{V(CDT)} + e^{V(CP)} + e^{V(CDTP)} + e^{V(CT)}} \quad (22)$$

The probability that a driver will switch to departure time interval h , provided that the decision to change departure time has been made, is:

$$P(CDT_h) = P(CDT) \frac{e^{V(CDT_h)}}{\sum_{j \in H} e^{V(CDT_j)}} \quad (23)$$

where H is the departure time choice set of the individual.

The probability that a driver will switch to path p , provided that the decision to change path has been made, is:

$$P(CP_p) = P(CP) \frac{e^{V(CP_p)}}{\sum_{r \in P} e^{V(CP_r)}} \quad (24)$$

where P is the path choice set of the individual.

The probability that a driver will switch to departure time interval h and path p , provided that the decision to change path has been made, is:

$$P(CDT_h P_p) = P(CDTP) \frac{e^{V(CDT_h P_p)}}{\sum_{j \in P \times H} e^{V(CDT_j P_r)}} \quad (25)$$

where $P \times H$ is the set of all combinations of the departure time intervals in the choice set H with the feasible paths in the choice set P of the individual.

3.6.3 Prescriptive Model Structure

In the case of prescriptive information, the behavioral model simplifies to a compliance model (Figure 8). The travelers are assumed to decide whether they will change their habitual travel behavior or not, and if they decide to change, they are

assumed to follow the suggested alternative. The model is assumed to apply only to route choice. This is due to the fact that –bearing current technological constraints in mind– no prescriptive guidance for departure time or mode choice can be generated based only on information derived from the prevailing network conditions. Although the system can evaluate the alternative paths and provide route choice recommendations for each OD pair, in order to perform a similar operation for departure time or mode choice, additional information about the individual needs to be known. Departure time and mode choice depend heavily on information about the individual traveler which is not available unless an information technology with feedback capabilities (from the user to the information provider) is in use.

An interactive guidance system, which would have the potential to acquire information from the drivers and compute and send customized guidance could allow for the extension of this framework to model departure time and mode choice under prescriptive guidance. An alternative approach would be the use of a distributed guidance system that transmits necessary traffic data to separate processing units, e.g. personal computers or on-board units, that combine this information with information provided to the unit by the driver to generate customized prescriptive guidance on departure time and mode choice. Although this approach is potentially limited with technological constraints associated with the capacity of transmission between the traffic control center (TCC) and the distributed processing units, two such systems have already been designed and implemented for en route applications. ADVANCE broadcasts current link travel times and incident reports to the vehicles, which use an on-board computer to derive guidance information (Lappin et al., 1994). EURO-SCOUT broadcasts a look-up table to all vehicles with specific recommendations about the direction that the user should follow in every crossroad (i.e. turn left or right, or continue on the same road) (Sodeikat, 1995). If a system with similar capabilities was available for pre-trip application, it would be possible to provide pre-trip prescriptive information regarding departure time choice and mode.

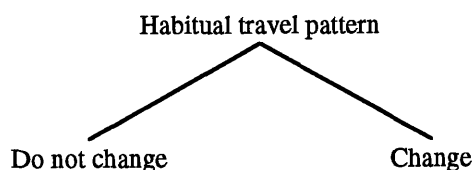


Figure 8. Pre-trip choice tree in the case of prescriptive information

3.6.4 Prescriptive Model Specification

The utility of the *do not change* alternative is:

$$\begin{aligned}
 V(DNC) = & \beta_1^p VOT_{lo} tt_{h'p}^I + \beta_2^p VOT_{med} tt_{h'p}^I + \beta_3^p VOT_{hi} tt_{h'p}^I + \\
 & \beta_4^p CF_{p'} + \beta_5^p l_{p'} + \beta_6^p \omega_{p'} + \beta_7^p c_{p'} + \beta_8^p s_{p'} + \beta_9^p f_{p'} + \\
 & \beta_{10}^p \max(dt_{h'} - T, 0)
 \end{aligned} \tag{26}$$

The utility of the *change* (to the proposed path p) alternative is:

$$\begin{aligned}
 V(C) = & \beta_{C,0}^p + \beta_1^p VOT_{lo} tt_{h'p}^I + \beta_2^p VOT_{med} tt_{h'p}^I + \beta_3^p VOT_{hi} tt_{h'p}^I + \\
 & \beta_4^p CF_p + \beta_5^p l_p + \beta_6^p \omega_p + \beta_7^p c_p + \beta_8^p s_p + \beta_9^p f_p + \\
 & \beta_{11}^p \max(at_{h'p}^H - at_{h'p}^I, 0) + \beta_{12}^p \max(at_{h'p}^I - at_{h'p}^H, 0) + \beta_{13}^p P_\omega + \\
 & \beta_{14}^p P_l
 \end{aligned} \tag{27}$$

The probabilities that each alternative will be selected are respectively:

$$P(DNC) = \frac{e^{V(DNC)}}{e^{V(DNC)} + e^{V(C)}} \tag{28}$$

and

$$P(C) = \frac{e^{V(C)}}{e^{V(DNC)} + e^{V(C)}} \tag{29}$$

3.6.5 Algorithm

The input to the behavioral update procedure includes:

- A historical population of drivers, and
- Information and guidance, expressed as travel times for all alternatives.

The output is a population of drivers, with updated travel decisions, that reflect the available information.

The procedure that is followed in the generation of the updated travel behavior of each driver is presented in Figure 9.

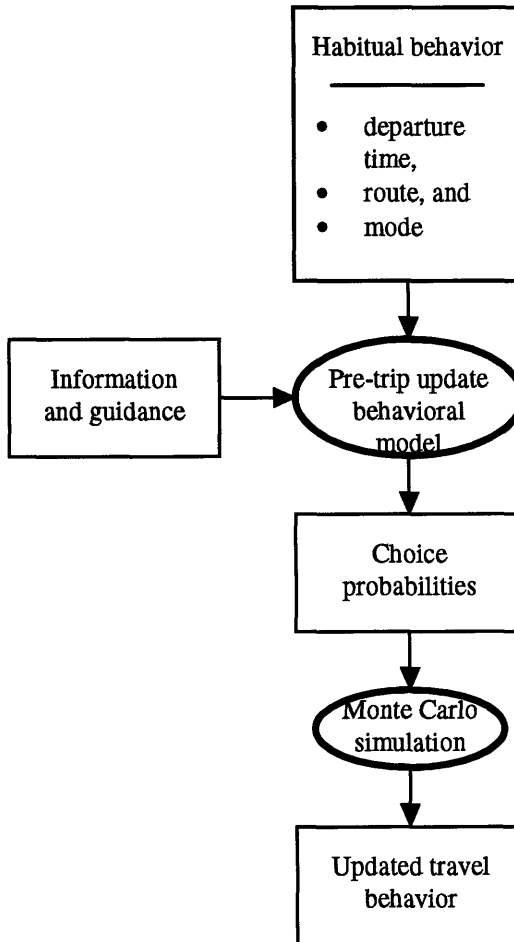


Figure 9. Travel behavior update process

The first step in the pre-trip behavior update is the application of the appropriate behavioral model, in order to get choice probabilities for all alternative options. The component deals with both descriptive and prescriptive information. The traveler behavior is captured by the descriptive or prescriptive models, which –selected based on the type of available information– provide choice probabilities for each of the alternatives. If the user receives both prescriptive and descriptive information, then it is assumed that the prescriptive overrides the descriptive and the user behaves as if receiving only prescriptive information. Furthermore, it is assumed that the system provides full information, that is travel times (and any other required information) for all combinations of paths and departure time intervals.

Once all the choice probabilities are computed, Monte Carlo simulation is used to select an alternative. This concludes the computation of the updated traveler behavior. The updated departure time interval, mode and path are then assigned to the driver, by overwriting the default values generated for the driver at the off-line disaggregation stage.

At this point the effect of pre-trip information in the drivers' travel behavior has been captured, and a disaggregate updated population is available. The OD estimation needs aggregate demand as input. Therefore, an aggregation procedure is required. This is presented in the next section.

3.7 Aggregation

The role of the aggregation component is to generate updated OD matrices by aggregating the updated population of drivers. The updated OD matrices are used as input for the OD estimation model. The aggregation is based on the departure time interval, the origin and the destination of the drivers.

3.7.1 Algorithm

The input to the aggregation component is the updated population of drivers. The output is a set of updated OD matrices, with one OD matrix corresponding to each estimation time interval.

The aggregation component generates OD matrices by iterating over the drivers that have decided to depart in a given departure time interval and have selected the car as their mode and summing them, based on their origin and destination. The number of drivers that depart in a given departure time interval from a given origin with a given destination corresponds to the cell of the OD matrix for that departure time interval and OD pair.

The effect of the behavioral update becomes apparent at this stage. Drivers are now organized in OD matrices based on their updated departure time interval rather than their habitual intervals that the historical OD matrices assume. Drivers that have decided to cancel their trip or switch to public transit are not included in the updated OD matrices. It must be noted that although the model explicitly captures mode switch from automobile to transit and removes the drivers that decide to switch to public transit, it does not capture explicitly travelers switching from transit to car. It is assumed that all potential drivers (i.e. those with access to a car that possess a driver's license) were included in the historical OD matrix.

At this point, the aggregate demand has been updated to reflect the travelers' response to information. The next step is the OD estimation, which uses data from the network and attempts to achieve consistency between the link counts and the demand in order to estimate the actual demand.

3.8 OD Estimation and Prediction Model

The OD estimation and prediction model is presented in this section. The general model is described first. An approximation of the model, which is more suitable for the context of real-time applications, is also presented and motivated. Finally, the implemented algorithm that solves the approximate model is presented and discussed.

Time-dependent OD matrices are key inputs to a DTA system. The ability of the DTA to accurately model existing traffic conditions depends on the quality of the OD matrices, since the OD matrices are translated into actual traffic that is loaded and simulated in the network. To be useful, these OD matrices need to be updated in real-time to reflect the changing traffic conditions in the network. The basic problem of dynamic OD estimation is to compute, in real-time, an estimate of OD flows for a given time interval from link traffic counts. The formulation of the real-time dynamic OD matrix estimation problem based on a Kalman Filtering framework is described by Ashok (1996). The basic idea of this approach is to use all the information contained in updated OD data in conjunction with data on traffic counts to generate OD estimates in real-time. The OD matrices that are used in this procedure have already been updated so that they reflect the response to information available to the travelers at the pre-trip stage.

Unlike other approaches in the literature, the adopted model is based on deviations from historical values as proposed by Ashok and Ben-Akiva (1993) rather than the values themselves. This approach seeks to incorporate structural relationships in the estimation process by including all estimations of prior days. This is important, since the already estimated OD matrices subsume a wealth of information about the latent relationships that affect travel demand and their variations over time. Thus the estimation process indirectly takes into account all the experience gained over many prior estimations and is richer in structural content.

The basic problem of OD prediction is to compute, in real-time, estimates of future OD flows from the current OD estimates. The autoregressive process used by the Kalman filtering approach provides a prediction tool, with real-time capabilities, that is consistent with the estimation process and models the temporal relationship among deviations in OD flows. Unobserved factors that are correlated over time (like weather conditions, unusual events, etc.) give rise to correlation of deviations over time which are modeled by the autoregressive process. The autoregressive process is characterized by a set of coefficients describing the effect of the deviations during one time interval on the deviation during another time interval. These coefficients are computed off-line, using a

linear regression model for each OD pair and for each time interval. Predicted deviations are, therefore, obtained by applying this autoregressive model to the deviations estimated for the current time interval.

Estimated and predicted deviations are finally added to a historical OD matrix to get estimated and predicted OD matrices.

3.8.1 Model Overview

The OD estimation and prediction model is a model for real-time OD estimation and prediction, proposed by Ashok (1996). The model is a state-space formulation that uses deviations of OD flows from historical values as unknown variables. A state-space model is formulated as a set of:

- Transition equations, and
- Measurement equations.

The formulation of the model is presented in the next two subsections. In Section 3.8.4, an approximate formulation is presented, which has many computational advantages at virtually no accuracy cost. This makes it attractive for use in a real-time application, such as the proposed pre-trip demand simulator. The Kalman filter algorithm that is used to solve the approximate system of equations is presented in the next subsection. The procedures that are used to get estimated and predicted OD matrices from deviations are presented next. Finally, the overall algorithm of the OD estimation and prediction process is discussed.

3.8.2 The Transition Equation

Assuming the following notation:

- x_h is the vector representing the number of vehicles between each OD pair departing their origins during time interval h ,
- x_h^H is the corresponding updated historical estimate, and
- $X_h = x_h - x_h^H$ is the deviation of x_h from x_h^H ,

the transition equation can be expressed in matrix form as:

$$\hat{X}_{h+1} = \sum_{p=h-q}^h f_h^p X_p + w_h \quad (30)$$

where:

- f_h^P is an $n_{OD} * n_{OD}$ matrix of effects of X_p on \hat{X}_{h+1} ,
- w_h is an $n_{OD} * 1$ vector of gaussian errors,
- q is the degree of the autoregressive process, and
- \hat{X}_{h+1} is an estimate of X_{h+1} .

The following assumptions are made about the error vectors:

1. $E[w_h] = 0$
2. $E[w_h w_l'] = Q_h \delta_{hl}$

where

- $\delta_{hl} = 1$ if $h = l$ and 0 otherwise $\forall h, l$ and
- Q_h is an $n_{OD} * n_{OD}$ transition error covariance matrix.

The second assumption implies that there is no serial correlation and is justified because the unobserved factors in the transition equation that could be correlated over time are captured by the updated historical matrix x_{h+1}^H .

The model in the above form is highly general and assumes dependence of deviations corresponding to one OD pair on deviations corresponding to other OD pairs in prior periods. In practical application this is unnecessarily general and relationships between deviations across different OD pairs may be safely ignored. This simplification is adopted in our implementation. In other words, we are assuming a diagonal structure for the matrices f_h^P . If a non-diagonal matrix is provided as input to some estimation, then it is diagonalized prior to the application of the algorithm. Also, computation of the matrices f_h^P involves estimating linear regression models for each OD pair and for each interval. If one makes the additional assumption that the structure of the autoregressive process remains constant with respect to h , the values of the matrix f_h^P would only depend on the difference $(h-p)$ and not on the individual values of h and p . Given the data limitations, this is a reasonable assumption.

3.8.3 The Measurement Equation

Assuming the following notation:

- y_h are the link flows obtained by assigning the updated historical OD flows,

- a_h^p is an $n_l * n_{OD}$ assignment matrix of contributions of x_p to y_h ,
- $A_h = a_h^h$ is the assignment matrix mapping the drivers that departed in interval h to the link counts observed in interval h ,
- p' is the maximum number of time intervals taken to travel between any OD pair of the network, and
- $Y_h = y_h - \sum_{p=h-p'}^{h-1} a_h^p \hat{x}_p$ is the deviation in the link counts,

the measurement equation, which relates unknown OD flows to the observed link counts, can be stated in matrix form as follows:

$$Y_h = A_h x_p + v_h \quad (31)$$

where:

v_h is the vector of measurement errors.

The following assumptions are made about the error vectors:

1. $E[v_h] = 0$
2. $E[w_h v_l'] = 0 \forall h, l$ i.e. transition and measurement errors are uncorrelated
3. $E[v_h v_l'] = R_h \delta_{hl}$

where

- $\delta_{hl} = 1$ if $h = l$ and 0 otherwise $\forall h, l$ and
- R_h is the $n_l * n_l$ measurement error covariance matrix.

3.8.4 Approximate Formulation

The model, as presented above, filters a very large number of flows during each interval. This imposes an enormous computational strain especially for large and congested networks potentially making a real-time application of the model infeasible. An approximation is made, based on the intuition that most of the information about an OD flow is likely to be provided the first time it is counted. Ashok (1996) found that the approximation has only a slight impact on quality of estimated OD flows relative to the base model. Furthermore, the computational savings that this approximation gives make it the preferred method for real-time implementation. Finally, he found that the model is fairly robust with respect to measurement error in link counts. The measurement and transition equations would then contain constant terms and could be expressed as follows:

$$Y_h - y_h^H = A_h X_h + v_h \quad (32)$$

$$\hat{X}_{h+1} = \Phi_h X_h + c_h + w_h \quad (33)$$

where

- $y_h^H = \sum_{p=h-p'}^h a_h^p x_p^H$,
- $c_h = \sum_{p=h-q}^{h-1} f_h^p \hat{X}_p$, and
- $\Phi_h = f_h^h$.

3.8.5 Algorithm

The solution of the presented formulation is given by a *square root* Kalman Filtering algorithm (Chui and Chen, 1987). The selection of the particular algorithm has been based on several criteria –presented in Section 3.8.7– and the proposed algorithm fits well in the need of the system.

The input to the OD estimation and prediction model for time interval h consists primarily of the following matrices, where n_l is the number of links and n_{OD} is the number of OD pairs in the network:

- One updated OD matrix x_h of drivers departing in that interval with dimension $n_{OD} \times 1$.
- $p'+1$ assignment matrices a_h^p . Each assignment matrix is $n_l \times n_{OD}$ and provides information about the contribution of the OD flows of a time interval (for each of the p' past intervals plus the current one) to the link flows for the current time interval.
- One observed link flow matrix y_h with dimension $n_l \times 1$. This matrix contains information about the traffic conditions on the links of the network which has become available from the surveillance system.

The output is estimated and predicted OD matrices.

The assignment matrix is an important input to the model and can be provided from a traffic simulator, if such a capability exists, or computed from flow and speed data available either from the surveillance system of a real network or from a traffic simulator. A detailed description of methods for computing the assignment matrix can be found in Ashok (1996).

The structure of the module is presented in Figure 10.

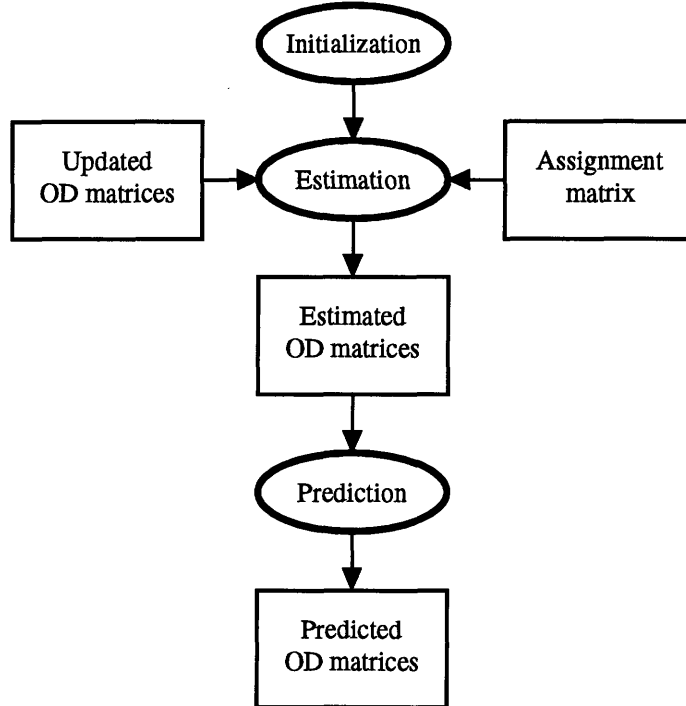


Figure 10. Estimation and prediction structure

The OD estimation and prediction has three main components. The first one is called initialization and takes place only once, prior to the estimation of the first interval. The use of the initialization step is to produce necessary information for the first estimation interval. The initialization is equivalent to a *preprocessing* step. The *square root* of the variance of the state vector, which is required by the specific *square-root* Kalman Filtering algorithm that is used for the solution of the model, is computed at the initialization step.

Besides the OD, assignment and link flow matrices, the model also requires:

- q transition matrices f_h , with dimension $n_{OD} \times n_{OD}$. These matrices do not depend on the actual intervals, but only on their difference q ,
- transition and measurement error covariances of the system Q_h and R_h , for each time interval h . The dimension of Q_h is $n_{OD} \times n_{OD}$, while the dimension of R_h is $n_l \times n_l$,
- the initial state vector x_0 , i.e. the set of decision variables, and the variance of its deviation from a historical value x_0^H , $\text{Var}(x_0 - x_0^H)$, with dimensions $n_{OD} \times 1$ and $n_{OD} \times n_{OD}$ respectively.

In the formulation of the algorithm the following Kalman Filter terminology is used:

- $\hat{X}_{h|h-1}$ represents a *one-step* prediction of the state X_h , i.e. it represents the best knowledge of the deviation X_h prior to obtaining the link counts for interval h .
- $\Sigma_{h|h-1}$ and $\Sigma_{h|h}$ represent the variances of $\hat{X}_{h|h-1}$ and $\hat{X}_{h|h}$ respectively.

The algorithm proceeds as follows:

(i) Initialization:

The square root of the variance of the state vector is computed:

$$\Sigma_{0|0} = \left(\text{Var}(X_0) \right)^c \quad (34)$$

where

c denotes the Cholesky factor, which is the equivalent matrix operation to the square root of a number.

Furthermore, one-step prediction is performed on historical data, to give an initial estimate of the state vector for the first estimation interval:

$$\hat{X}_{h+1|h} = \Phi_h X_{h|h} + \sum_{p=h-q}^{h-1} f_h^p \left(\hat{X}_p - x_p^H \right) \quad (35)$$

The following computations are then performed for each estimation interval $h=0,1,2,3, \dots$

(ii) Estimation of deviations:

(a) Variance propagation of the filter

$$\text{Compute} \quad \Sigma_{h|h-1} = \left[\Phi_{h-1} \Sigma_{h-1|h-1} \quad Q_{h-1}^c \right]_{n \times (n+p)} \quad (36)$$

$$\text{and} \quad H_h = (A_h \Sigma_{h|h-1} \Sigma_{h|h-1}^T A_h^T + R_h^c)^c \quad (37)$$

and use them to compute

$$\Sigma_{h|h} = \Sigma_{h|h-1} \left[I - \Sigma_{h|h-1}^T A_h^T (H_h^T)^{-1} (H_h + R_h^c)^{-1} A_h \Sigma_{h|h-1} \right] \quad (38)$$

(b) Calculation of the Kalman gain

$$G_h = \Sigma_{h|h-1} \Sigma_{h|h-1}^T A_h^T (H_h^T)^{-1} H_h^{-1} \quad (39)$$

(c) *Measurement update*

$$\hat{X}_{hlh} = \hat{X}_{hlh-1} + G_h \left[Y_h - A_h (\hat{X}_{hlh-1} + x_h^H) - A_{h-1} (\hat{X}_{h-1lh-1} + x_{h-1}^H) - A_{h-2} (\hat{X}_{h-2lh-1} + x_{h-2}^H) \right] \quad (40)$$

(iii) **Prediction of deviations:**

After the estimation of the deviation has been performed, i.e. \hat{X}_{hlh} is known, k -step prediction is performed, in order to get \hat{X}_{h+1lh} , \hat{X}_{h+2lh} , \hat{X}_{h+3lh} , ..., \hat{X}_{h+klh} . The k -step prediction algorithm follows:

$$\hat{X}_{h+1lh} = \Phi_h \hat{X}_{hlh} + \sum_{p=h-q}^{h-1} f_h^p \hat{X}_{plh} + x_{h+1}^H \quad (41)$$

$$\hat{X}_{h+2lh} = \Phi_{h+1} \hat{X}_{h+1lh} + \sum_{p=h+1-q}^h f_{h+1}^p \hat{X}_{plh} + x_{h+2}^H \quad (42)$$

.

.

.

$$\hat{X}_{h+klh} = \Phi_{h+k-1} \hat{X}_{h+k-1lh} + \sum_{p=h+k-1-q}^{h+k-2} f_{h+k-1}^p \hat{X}_{plh} + x_{h+k}^H \quad (43)$$

3.8.6 OD Estimation from Deviations

The application of the Kalman Filter that has been described above outputs the deviations \hat{X}_{hlh} of the OD flows from the historical values for the estimation interval h . In order to get the final OD flows x_h one needs to add these deviations to the historical OD flows x_h^H :

$$x_h = \hat{X}_{hlh} + x_h^H \quad (44)$$

3.8.7 Discussion on Algorithm

The square root algorithm was selected for its numerical robustness and efficiency. Indeed, in going to the square root, small numbers become larger and large numbers become smaller, improving accuracy. The major time-consuming operation in the Kalman filtering process is the computation of the Kalman gain matrices. This is due to the need to perform a particularly computationally expensive inversion. The square-root algorithm takes the Cholesky factor of this matrix, thus reducing the matrix to a lower triangular. The square-root algorithm is thus more efficient since the inverse of a lower triangular matrix can be computed efficiently (time-wise). The fact that some

matrices are lower triangular is beneficial also from a memory-requirements point of view, since the upper triangle -which contains only non-zero elements- does not need to be stored. This can significantly limit the size of required computer memory, which is important, since the dimensions of the matrices describing a realistic urban network -and, hence, the memory requirements- can be very large.

The algorithm contains two functions that require from the matrices upon which they operate to be non-singular. These functions are the Cholesky factorization and the inversion. The matrices that need to be non-singular are: $\text{Var}(X_0)$ and H_h . If either happen to be singular, then backup algorithms are required, that do not depend on the non-singularity of these matrices. If $\text{Var}(X_0)$ is singular, this means that some OD flows are known precisely. Therefore they do not need to be estimated and will be removed from the OD matrix, thus eliminating the singularity. Since this operation happens in the beginning of the algorithm, it does not affect it. A way to detect in which measurements the singularity occurs is to perform standard eigenvalue analysis. Furthermore, H_h is unlikely to be singular, since its singularity depends on the variance-covariance of the measurement error; if R_h is strictly positive definite, then H_h is non singular.

A modified Cholesky factorization algorithm that does not depend on the non-singularity of the matrix (Schnabel and Eskow, 1988) is used to overcome potential singularity problems. The algorithm is based on a modified Cholesky factorization introduced by Gill and Murray (1974), which is commonly used in optimization algorithms. Given a symmetric, but not necessarily positive definite matrix A , the modified Cholesky factorization computes a Cholesky factorization of $A+E$, where $E=0$ if A is positive definite, and E is a diagonal matrix chosen to make $A+E$ positive definite otherwise. The algorithm succeeds in generating a small E matrix and a well conditioned $A+E$, in practice.

4. Implementation

The pre-trip demand simulator is a complex system. The implementation of such a system is not trivial and should be made in an organized manner. The steps that were followed in the implementation phase of the system are presented in this section. The first subsection presents an overview of the implementation environment. The second discusses a set of objectives, whereas, the guidelines that were set for the implementation of the required objects are presented in the third subsection. Finally, object design and specific issues associated with the implementation of each individual component are presented in the last two subsections.

4.1 Overview

The system is implemented using the Object Oriented (OO) paradigm (Rumbaugh et al., 1991). The programming language of choice is C++ with the Standard Template Library (STL). The system is implemented as a client/server distributed application, using Orbix, a Common Object Request Broker Architecture (CORBA) implemented by Iona Technologies. Finally, the Object Modeling Technique (OMT) (Rumbaugh et al., 1991) has been selected as the object design methodology.

C++ is an object oriented extension of the widely used programming language C, and was developed by Bjarne Stroustrup at the AT&T Bell Laboratories (Stroustrup, 1991). C++ allows the user to define data types that behave in nearly the same way as built-in types. Nevertheless, unlike earlier interpreted object oriented language, such as *Smalltalk* and *Lisp* based languages, C++ is a compiled language that does not sacrifice run time efficiency. Some of the advantages of C++ -besides object oriented features- include portability into virtually any platform, compatibility with C, efficiency, and performance.

The Standard Template Library is a template-based library of generic C++ data structures and algorithms that work together in an efficient and flexible fashion (Nelson, 1995). STL was proposed by Alexander Stepanov, developed by Hewlett Packard, and accepted by the ANSI/ISO C++ Standards Committee in July 1994. STL offers a set of ready-to-use components that -from an efficiency point of view- are very close to their hand-coded equivalents. Therefore, the need for custom-made vectors, lists and other -more complicated- structures has disappeared, allowing the developers to concentrate on more substantial aspects of the development procedure.

CORBA is the Object Management Group (OMG) consortium’s standard for distributed application communications. It is based on object-oriented design principles and client/server technology. Orbix is a CORBA implementation, which allows software interfaces to be defined in a standard language and then accessed from anywhere in a distributed system. Servers are launched and managed automatically from the Orbix runtime system. Orbix’s performance has been shown to be comparable to other communication protocols such as BSD sockets and PVM (Fatoohi, 1996).

The Object Modeling Technique uses the object model to describe the static data structure of objects, classes and their relationships to one another. Nevertheless, no behavior, dynamic evolution or data flow are captured by this model. The main relationships used in this document and their representation on the diagrams are presented in the next figures. Two objects can be connected with a one-to-one relationship, called association. An association between two objects is represented by a link (Figure 11): the PreTripBehavioralModel is associated with one MonteCarlo object. In the case of a one-to-many relationship, a multiplicity relationship can also be specified (Figure 12): the PreTripBehavioralModel is associated with several (zero or more) PreTripInfo objects. An aggregation is an association that represents the “is a part of” relationship (Figure 13): a driver is a part of the ListOfDrivers. Finally, inheritance, also known as generalization, represents the “is a special case of” relationship (Figure 14): the full and sparse vectors are special cases of Vector.



Figure 11. Association example

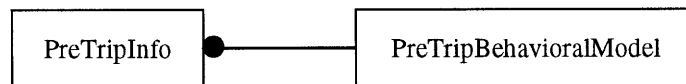


Figure 12. Multiplicity example

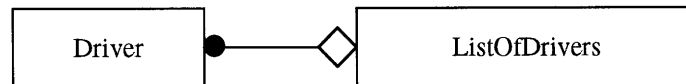


Figure 13. Aggregation example

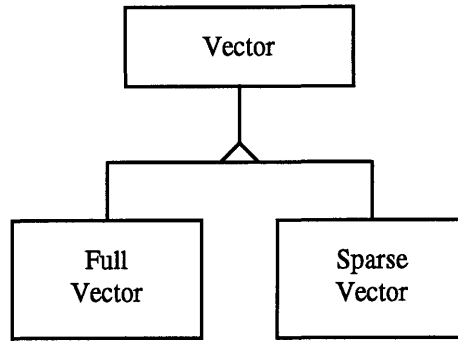


Figure 14. Inheritance example

4.2 Objectives

The following general objectives were set and followed during the implementation stage:

- Object oriented (OO) design,
- Computational efficiency, and
- Numerical robustness.

Each of these objectives is discussed in the following subsections.

4.2.1 Object Oriented (OO) Design

The term “object-oriented” means that software is organized as a collection of discrete objects that incorporate both data structure and behavior, in contrast to conventional programming in which data structure and behavior are loosely connected (Rumbaugh et al, 1991). Object oriented design supports several concepts, such as abstraction, encapsulation and sharing, and offers substantial advantages over conventional programming practices. The benefits from using the object-oriented approach include:

- Flexibility of the design,
- Better communication of the objects and easy exchange of information, and
- Easier debugging, maintainability and reusability of the code.

4.2.2 Computational Efficiency

The computational efficiency of the application can be broken down into three major sub-tasks:

- Efficiency of the algorithms,
- Efficiency of the individual functions called by the algorithm, and
- Efficiency of the data structures.

The benefits from computational efficiency are two-fold:

- Higher execution speed, which is critical for a real-time application, and
- Large size of network, that can be processed, under current technological constraints.

4.2.3 Numerical Robustness

The term numerical robustness refers to the elimination of numerical errors. The numerical robustness of the demand simulator is ensured by:

- i. Selection of the square root Kalman filtering algorithm (Chui and Chen, 1987) for the solution of the OD estimation and prediction equation system, and
- ii. Elimination of potential singularity problems.

As mentioned in Section 3.8.7, the square root algorithm uses the Cholesky factor of some matrices, instead of the actual matrices. This improves the numerical robustness, since the square root of very small numbers is larger and the square root of very large numbers is smaller. In this way, extreme values, that are potential sources of numerical instability, are avoided and accuracy is improved. Besides that, the algorithm also offers computational benefits.

Furthermore, some of the functions that are used in the algorithm, namely the Cholesky factorization and the inversion, require from the matrices upon which they operate to be not close to singularity and positive definite. Although these matrices are very likely to be non-singular, the necessary steps have been incorporated in the algorithm to ensure that such an event will not influence the execution of the application. For example, a modified Cholesky factorization is used that can be applied even in the case of matrices close to singularity and non-positive definite matrices (Schnabel and Eskow, 1988).

4.3 Requirements

The objects that comprise the demand simulator need to be designed in a way that they provide:

- Flexibility in the design,
- Efficient access of their elements, and
- Efficient storage, that meets the memory requirements.

4.3.1 Design Flexibility

The flexibility is achieved by the OO design of the system, which makes the substitution of a component with another one, that complies with the specified interface, possible. This substitution should be transparent to the rest of the system and no modification of other objects should be necessary to complement it.

In order to satisfy this requirement, the demand simulator exploits the advantages of a full OO development. Where appropriate, abstract classes are defined and the concrete classes are derived from these. For example, in the OD estimation and prediction component, a series of matrices and vectors are required, such as dense, sparse, symmetric, and lower triangular matrices, as well as dense and sparse vectors. Abstract *matrix* and *vector* objects are defined, and the concrete classes are derived from them. In this way, a consistent interface is defined and the object elements are accessed through that. Figure 15 and Figure 16 show the abstract and concrete classes for *matrix* and *vector*.

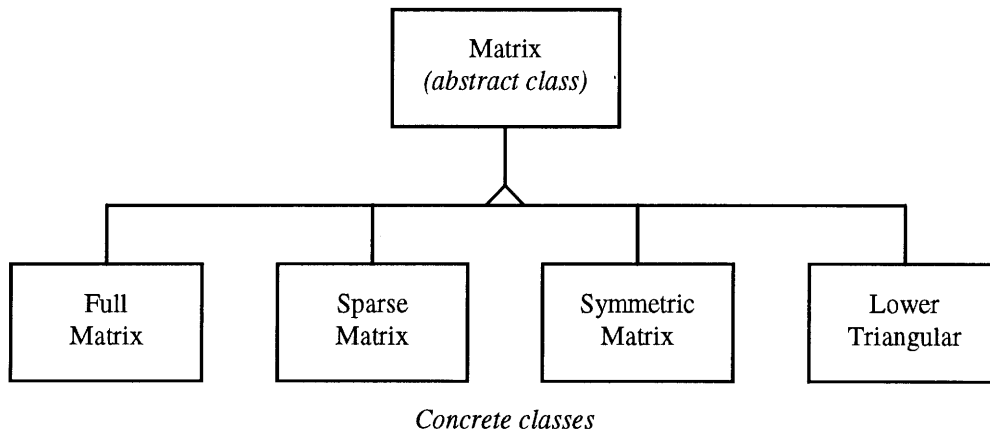


Figure 15. Matrix Classes

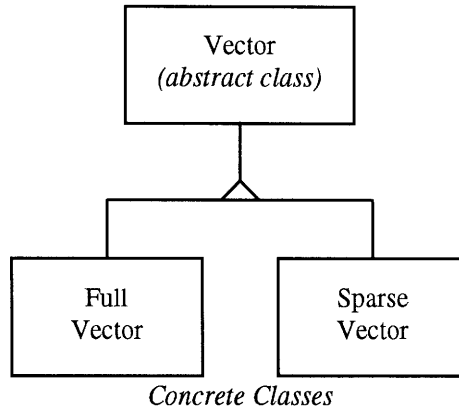


Figure 16. Vector Classes

The data of an object is not accessed directly from other objects or components, but all interaction takes place through abstract interfaces. In most cases these interfaces are defined as iterators¹ and are independent of the internal representation and structure of the object. *Iterators* provide a way to access the elements of an aggregate object sequentially, without exposing its underlying representation (Gamma et al., 1995). If the user wishes to change the internal representation of the object, then the code modifications are limited in changing the iterators to reflect the new structure. Once the modification is complete, the other objects should be able to access the modified object in exactly the same way as they used to. In other words, any change in the internal representation of an object is transparent to other objects.

4.3.2 Element Access

Many objects need to access the elements of other objects. These operations are performed through the interface of the objects and the calling objects do not have direct access to the elements, which are private, but access them via the public interface. The public interface consists from simple functions that return the requested data, and iterators on the elements of the objects.

The *iterator* pattern, as defined by Gamma et al. (1995), is used as the basis for these iterators. Iterators are intended to provide a way to access the elements of an aggregate objects sequentially without exposing its underlying representation. Furthermore, iterators support multiple traversals of aggregate objects by the use of multiple iterators (e.g. iterators on the rows and the columns of a matrix). Finally, iterators provide a uniform interface for traversing different aggregate structures. An iterator has at least the following operations:

¹ Iterators are discussed in the next section.

- First(): initializes the current element to the first element of the aggregate object,
- Next(): advances the current element to the next element of the object,
- IsDone(): tests whether the iteration over the elements of the aggregate object is complete, i.e. it has advanced beyond the last element, and
- CurrentItem(): returns the current element.

Besides this general read-only iterator, one can enhance the concept by adding functionality. Such an addition, that was used in the current project, was the SetCurrentItem() operation that sets the value of the current element to the desired one. The resulting iterator is a read-write iterator, that inherits from the read-only iterator (Figure 17) and provides a uniform interface in the update of the values of aggregate objects.

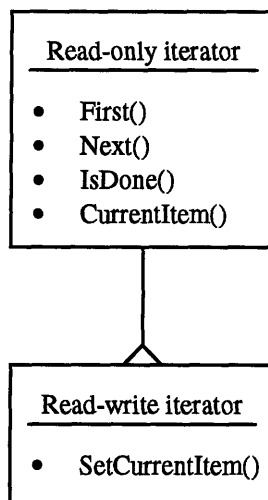


Figure 17. Read-write iterator inheritance

4.3.3 Memory Requirements

The demand simulator involves a number of objects, many of which have to be loaded on the memory at the same time, while others need to be read from and written to the storage devices in real-time. It is therefore important that these objects are created in a way that they consume as little storage capacity as possible. Furthermore, since the size of the network has closely relation with the size of some of these objects, the more efficient the structure of the objects, the larger the network that can be processed. Therefore, it is important that the representation of these objects is optimized in a way that their storage needs are reasonable.

The storage efficiency is achieved by exploiting the special form of several of the demand simulator objects. For example, some of the matrices are very likely to be highly sparse, whereas others are symmetric or lower triangular. In the case of sparse matrices, only non-zero elements are stored to decrease the memory requirements². Similarly, in the case of symmetric or lower triangular matrices only the lower triangle needs to be stored.

4.4 Objects

In this section, the objects that comprise the pre-trip demand simulator are presented. In the first subsection, the objects are listed and a selective list of iterators are presented. In the second, the object associations are described.

4.4.1 List of Objects

The following objects are required in the pre-trip demand simulator:

- **OffLineDisaggregation**: disaggregates the historical OD matrices into a **ListOfDrivers**,
- **PreTripBehavioralModel**: updates the travel behavior of the drivers to reflect their response to information made available at the pre-trip stage,
- **Aggregation**: aggregates the **ListOfDrivers** into an **ODMatrixList**,
- **ODEstimationAndPrediction**: estimates OD flows for the current time interval and predicts OD flows for future intervals,
- **OnLineDisaggregation**: in the case that disaggregate demand is required as final output of the demand simulator, disaggregates the aggregate demand output, by removing drivers from the **ListOfDrivers** or generating and adding new drivers to it,
- **ODMatrix**: a historical, estimated, or predicted OD matrix for a time interval,
- **ODMatrixList**: a list of **ODMatrices**,
- **LinkCounts**: the link counts, observed by the surveillance system.
- **AssignmentMatrix**: an assignment matrix (see Section 3.8),
- **TransitionMatrix**: a transition matrix (see Section 3.8),

² Nevertheless, additional information needs to be stored in this case -namely, the row and column index of each non-zero element- and therefore this tradeoff should be examined before using this representation for a matrix.

- **ErrorCovarianceMatrices:** the error covariance matrices Q and R (see Section 3.8),
- **Driver:** one driver,
- **ListOfDrivers:** a list of drivers,
- **PreTripInfo:** pre-trip information provided by the guidance generator in the form of travel times for all alternatives,
- **Distributions:** distributions of characteristics, based on which the characteristics of the individual drivers are generated,
- **Coefficients:** coefficients of the behavioral models, and
- **MonteCarlo:** an object that given a distribution returns a randomly chosen alternative.

Also, besides these objects, the matrices and vectors presented in Section 4.3.1 are used for intermediate computations and the representation of additional objects that do not require further functionality.

The last category of demand simulator objects are the *iterators*. As mentioned earlier, iterators provide a way to access the elements of an aggregate object sequentially without exposing its underlying representation. The iterators have been implemented efficiently and exploit the data structure of the objects on which they iterate. An example of iterators is given in Section 4.5.4.

4.4.2 Object Associations

The objects of the simulator and their associations are presented in Figure 18. Each association has been numbered to facilitate easier reference at the description that follows. For the sake of controlling the figure's complexity, the iterators have not been presented, although they are included in the subsequent discussion. This does not affect the clarity of the description, because the associations of the iterators are straightforward.

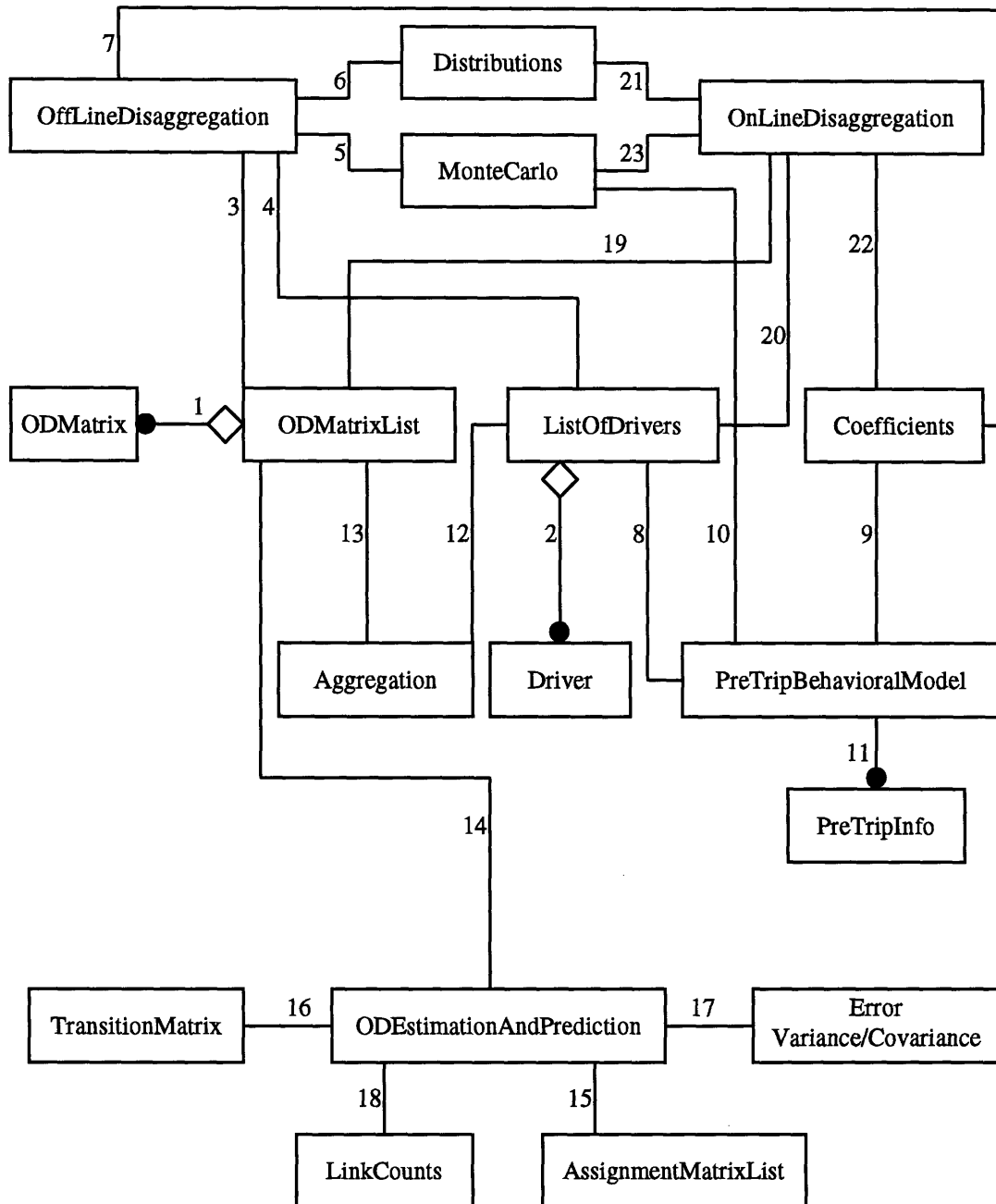


Figure 18. Object associations

The ODMatrixList is composed of ODMatrices (1). Similarly, the ListOfDrivers is composed of Drivers (2).

The OffLineDisaggregation uses the ODMatrixListROiterator to read the aggregate historical demand from the ODMatrixList (3). It subsequently disaggregates the aggregate demand using the Coefficients (7), the Distributions (6), and the MonteCarlo (5). Finally, the OffLineDisaggregation uses the ListOfDriversRWiterator to fill the ListOfDrivers (4) with the disaggregate demand.

The `PreTripBehavioralModel` uses a `ListOfDriversRWIterator` to iterate on the `ListOfDrivers` (8), and `PreTripInfo` (11), `Coefficients` (9) and `MonteCarlo` (10) to calculate and update the current behavior of each driver. The updated behavior of the drivers is written in the `ListOfDrivers` through the `ListOfDriversRWIterator`.

The `Aggregation` uses a `ListOfDriversROIterator` to read the characteristics of the individual drivers from the `ListOfDrivers` (12). As the drivers are read, an `ODMatrixListRWIterator` is used to update the entry in each cell of the `ODMatrixList` (13), by incrementing the value of the appropriate existing cell or creating a new one, if it does not exist.

The `ODEstimationAndPrediction` uses the `ODMatrixList` (14), the `AssignmentMatrixList` (15), the `TransitionMatrix` (16), the `ErrorCovariance` matrices (17) and the `LinkCounts` (18) as input to estimate the aggregate demand. Read-only iterators are used for all these objects in order to get data, required in the estimation. Furthermore, an `ODMatrixListRWIterator` is used to write the estimated aggregate demand in the `ODMatrixList`.

The `OnLineDisaggregation` uses an `ODMatrixListROIterator` to read the estimated aggregate demand from the `ODMatrixList` (19), and a `ListOfDriversRWIterator` to read the disaggregate demand, as it was prior to the estimation, from the `ListOfDrivers` (20). The `OnLineDisaggregation` checks the consistency of the two representations of demand and updates the `ListOfDrivers` to reflect the results of the estimation. If further drivers need to be generated, then the `Distributions` (21), `Coefficients` (22) and the `MonteCarlo` (23) are used to generate the additional drivers.

4.5 Implementation Issues

Implementation issues concerning the five components of the system are outlined in this section. These summarize the areas of the implementation where assumptions have been made that may need to be revisited in the future in order to improve the performance of the demand simulator.

4.5.1 Off-line Disaggregation

This procedure is performed off-line, since it is not dependent on real-time information and takes place at the beginning of the simulation. Therefore, its efficiency is not critical. Nevertheless, one should note that the procedure is computationally intensive, since a large number of randomly generated numbers will be needed for the assignment of characteristics and initial route to the drivers. The `ListOfPackets` needs to keep the drivers

sorted, in terms of departure time intervals. The STL set has been used in the implementation of the `ListOfPackets` which keeps the list sorted at all times, with little computational cost (Nelson, 1995).

4.5.2 Pre-Trip Behavioral Model

The pre-trip behavioral model is applied on every driver in the population of drivers, which is represented by the `ListOfDrivers` object. Therefore, it is important that the implemented algorithm is efficient. Several attributes of each driver have to be passed to the component in order to compute the drivers' behavior and therefore the existence of efficient iterators on their characteristics is important. Iterators are defined on the `ListOfDrivers` object, that return their static characteristics and their dynamic travel behavior.

Finally, a large number of random numbers needs to be generated in order to determine the drivers' travel behavior from the choice probabilities of the alternatives. This is a computationally intensive process.

4.5.3 Aggregation

The focus in the aggregation procedure is in the implementation of iterators on the `ListOfDrivers` object, that will enable efficient iteration on each drivers' departure time interval, origin and destination, which are necessary for their aggregation into OD matrices.

4.5.4 OD Estimation and Prediction Model

The *square root* Kalman filtering algorithm that has been selected to solve the system does not require many computationally intensive functions, e.g. inversions or Cholesky factorizations. Nevertheless, the algorithm requires that the matrices to be decomposed are non-singular and definite positive. Therefore, as discussed earlier, a modified Cholesky decomposition algorithm (Schnabel and Eskow, 1988) that overcomes this deficiency is used.

The functions that are called by the algorithm have also been designed efficiently. Inversion, a particularly computationally intensive function, need only be performed on lower triangular matrices. A special inversion algorithm has been implemented, which exploits the special structure of the lower triangular matrices (Golub and Van Loan, 1983). Also, other functions have been designed to take advantage of the special structure of symmetric, lower triangular or sparse matrices, whenever that was possible. Finally,

some operations required the identity matrix (\mathbf{I}). Instead of generating and storing a potentially very big identity matrix, the necessary steps have been incorporated in the functions to “simulate” the identity matrix, i.e. return a value equal to *one* if the element is on the diagonal, and *zero* otherwise. Besides being more efficient in terms of storage requirements, this is a computationally inexpensive procedure.

The third efficiency aspect is on the design of the data structures. The OD estimation and prediction model is associated with several objects, most of which contain a large amount of information. Therefore, they are very intensive from a storage point of view. Several of these objects, though, can be optimized in terms of required storage capacity. This has dual benefits:

- Size of network that can be processed given a certain size of random access memory (RAM), and
- Speed with which the objects associated with a network can be loaded from the database.

Some of these matrices, namely the OD matrix and the assignment matrix, are going to be highly sparse. Data structures that exploit this sparseness could result in significant benefits. A small example follows, to show the comparative requirements of a *sparse* representation of the assignment matrix over its conventional *dense* or *full* counterpart.

Consider the Boston, Ma., metropolitan area inside Route 128, which is a reasonably sized urban network. There are some 4800 nodes and 11000 links in the network, as well as 240 centroids, leading to 57360 OD pairs³. Assuming that 25% of the links have sensors, the size of each OD matrix will be (57360*1) and the size of each assignment matrix will be (2750*57360), or around 158 million cells. Nevertheless, both matrices will be very sparse.

To show the benefit that can be obtained from a data structure, that takes sparseness into account, the following example is presented. Considering on average:

- k paths per OD pair p ,
- l links per path, and
- L links in the network,

³ $N_{OD} = N_{centroids} * (N_{centroids} - 1)$, where N_{OD} and $N_{centroids}$ are the numbers of OD pairs and centroids in the network, respectively.

a measure called *degree of sparseness* is defined, which gives the percentage of cells in the matrix that have zero value:

$$\text{Degree of sparseness} = \left(1 - \frac{kl}{L}\right)\% \quad (45)$$

In this example, assuming reasonable values ($k = 10$, $l = 30$, and $L = 11000$), the degree of sparseness is 97.3%, which translates to a total number of 4.3 million non-zero cells. Therefore, it is clear that significant benefits can arise from the use of efficient data structures for the assignment matrix. Similarly, one can show that benefits can arise from the use of a sparse data structure for the OD matrix.

The data structure that is used for the representation of the OD matrix needs to have the following characteristics:

- Efficient iteration over destinations, for a given origin, and
- Management of the sparsity.

The data structure of the object is implemented as a map of maps (Figure 19). A map is an STL structure that can store and retrieve data based on a key. Data is stored using a tree organization, and there is no need for sorting. For each origin, a map contains the non-zero flows to all destinations. Therefore, iteration over the destinations -for a given origin-, which is required often, is very efficient. Iteration over origins -for a given destination- is also possible, though somewhat less efficient. There is some overhead associated with the use of the map (Nelson, 1995). Nevertheless, the functionality that it provides through the ability to keep the elements sorted at all times, as well as the easy retrieval and setting of elements outweigh this overhead.

- Efficient iteration over time intervals and OD pairs, and
- Management of the sparsity.

The data structure of the object is implemented as a vector of vectors of maps (Figure 20). For each time interval, there is a vector of links. For each link there is a vector of (current and prior) time intervals, and for each time interval there is a map of cells. Each cell is stored using the OD pair it is referring to as key and contains the corresponding assignment fraction for that time interval. For each link and prior interval, the corresponding cell represents the fraction of the OD flow for that OD pair and interval that contributes to the current link flow of the link. This data structure allows efficient iteration over time intervals and OD pairs and stores only non-zero cells. Like the OD matrix data structure, there is an overhead in this implementation, but the benefits are greater.

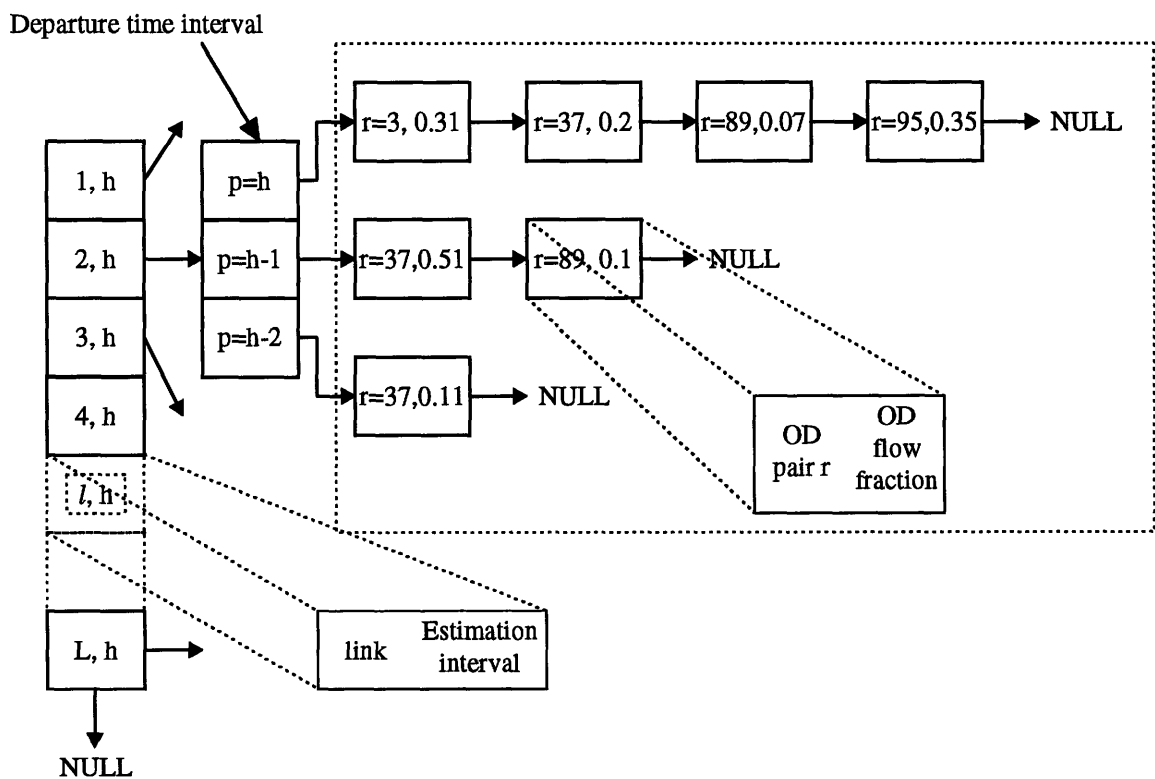


Figure 20. Assignment matrix data structure

4.5.5 On-line Disaggregation

In the case that disaggregate demand is required as output of the demand simulator, the estimated and filtered OD matrices will need to be disaggregated on-line. This is not as computationally demanding as the off-line disaggregation, since the already

generated population of drivers will be used as a basis and drivers will be removed and added to reflect the result of the OD flows filtering. The main computational concerns are the random number generation which is required for the generation of the characteristics of the additional drivers and the *exp* function required for the calculation of the utilities and the choice probabilities of the various alternatives of the habitual behavior model.

5. Evaluation

The demand simulator provides both estimation and prediction capabilities. The purpose of this chapter is to assess some of the properties of this simulator. This assessment focuses in the simulator's estimation capabilities. The estimation process of Ashok (1996) has been extended to include the impact of information in pre-trip decisions regarding departure time, route and mode choice and, therefore, an assessment is warranted.

The pre-trip demand simulator uses a large set of data from different sources and processes them to estimate demand. Depending on the application, this demand can be subsequently used, for example, to generate and provide guidance in a DTA context, generate traffic control strategies, or evaluate such traffic management measures. Therefore, the ability of the simulator to estimate demand as close as possible to the actual demand is important. The ultimate objective of the evaluation of the simulator would be the assessment of its ability to replicate true demand. This, however, is beyond the scope of this thesis. In this chapter a more limited set of assessments are conducted. Nevertheless, a more comprehensive evaluation framework is proposed in Chapter 6 for further research.

The series of case studies performed in this chapter attempt to illustrate some of the capabilities and assess some of the properties of the demand simulator, as well as investigate some of its potential shortcomings. More specifically, the three exercises conducted here include:

- Impact of behavioral update,
- Stochasticity of the output, and
- Sensitivity of the simulator to key inputs.

First, the importance of the pre-trip behavioral update is investigated and presented. The simulator's output should not change drastically as a result of small fluctuations in inputs representing errors. Second, the impact of stochasticity, inherent in some of the models that are incorporated in the demand simulator has to be analyzed. Finally, a sensitivity analysis is performed which aims at capturing the impact of inaccuracies in the inputs on the output of the demand simulator.

The various dimensions of the scenarios that are used in the evaluation exercises are presented in Section 5.1, whereas the evaluation cases are presented in sections 5.2, 5.3, and 5.4. For each case, the process is described, the results are presented and analyzed, and conclusions are drawn.

5.1 Scenarios

Each scenario considered in this evaluation exercise is a combination of up to 12 dimensions. In the following, each dimension is referred to by a capital letter. Some of these dimensions are fixed across all scenarios. The other dimensions can take several values. In this case, the capital letter is qualified by an index referring to each particular value. Not all cases use all the dimensions. Therefore, in the specification of the scenarios for each case, only the required dimensions are included.

5.1.1 Network (A)

The network that is used for the evaluation of the demand simulator is Boston's Central Artery and Third Harbor Tunnel (CA/T) network currently under construction, with an expected 2004 opening date. The network has 185 nodes and 217 links and is represented in Figure 21. A node is either an intersection of several roadways or a source or sink where traffic flows enter or leave the simulated network. Links are directional roadways that connect nodes. The network connects Route 1A, and Logan Airport in East Boston with I-93, Route 1, and the Massachusetts Turnpike (MassPike) through two sets of underwater tunnels, one in the north (Sumner and Callahan tunnels) and one in the south (Ted Williams - Third Harbor tunnel).

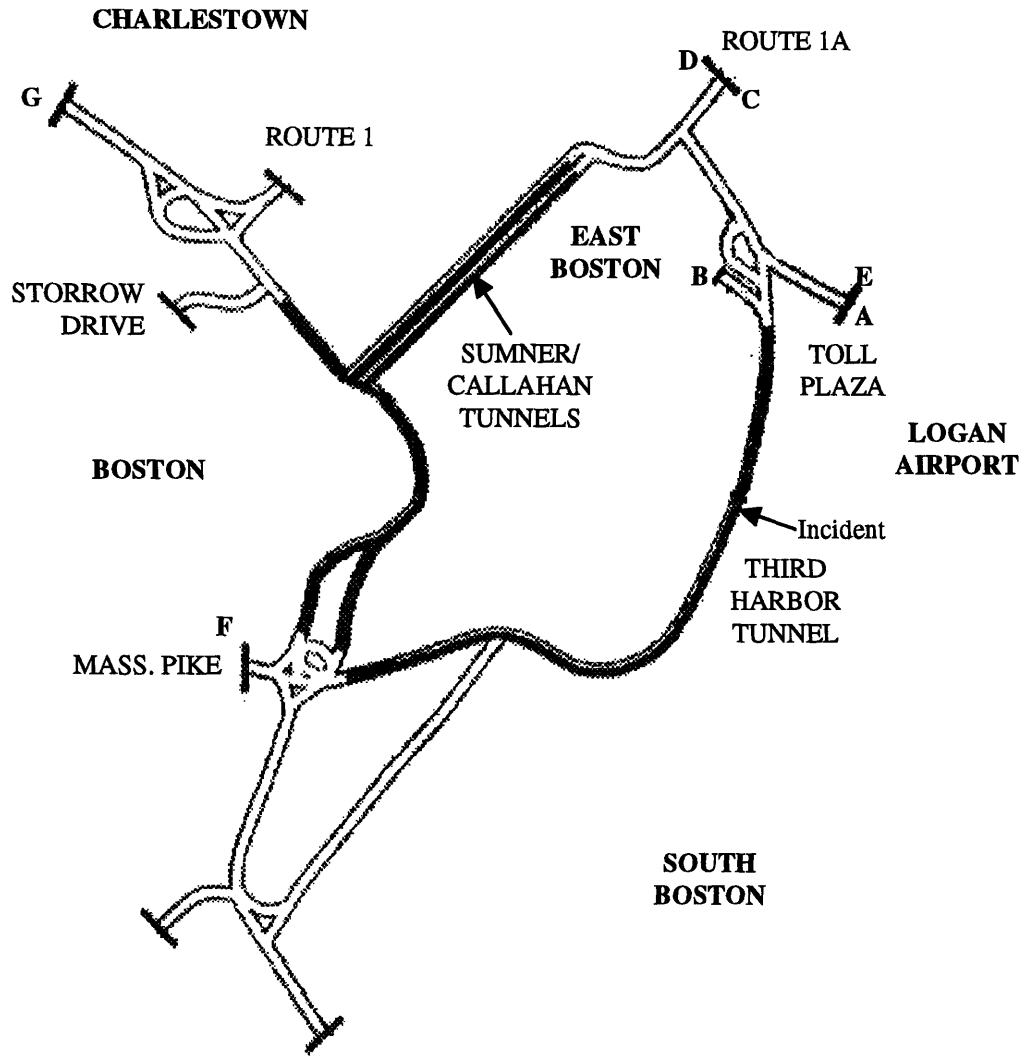


Figure 21. Map of the network

The representation of the network has been slightly extended from that used by other researchers, e.g. Yang (1996), in order to provide a more realistic evaluation environment for the purposes of the exercises presented in this chapter. In particular, drivers moving westbound in the Sumner or Callahan tunnel, cannot turn south towards downtown Boston and further destinations to the south without getting off the freeway and using local streets. The rationale behind this decision is to divert traffic from downtown Boston. As a result, drivers traveling from East Boston or Logan Airport towards South Boston or the MassPike have only one freeway-only route, the Ted Williams Tunnel, and, therefore, have no freeway route choice. In reality, drivers moving westbound in the Sumner or Callahan tunnel might consider bypassing this constraint by using local streets temporarily in Downtown Boston to reach the southbound direction of

I-93. This is made possible in the representation of the network that is used in the current thesis by the addition of a link representing the local street connection.

The existence of an incident on the network is also modeled, resulting in two alternative network representations. In particular:

- No incident (A1), and
- Incident in the Third Harbor (Ted Williams) Tunnel (A2).

The incident was assumed to occur at 7:15 and be cleared out at 7:30. It affected both lanes of the tunnel, blocking the right completely and limiting the speed in the left to 15 mph.

5.1.2 Historical demand (B)

The historical demand is derived from an assumed true demand. This demand was constructed using the following process. A set of 5 origins and 2 destinations are assumed. The origins are placed in the east part of the network (A, B, C, D, and E) and the destinations in the southwest (F) and northwest (G) (see Figure 21). This simplified network structure provides all the desirable conditions, while avoiding unnecessary complexity. Specifically, the network can capture route choice, since there are two paths connecting most OD pairs. Furthermore, the occurrence of an incident in either tunnel does not block the traffic moving between any OD pair.

A schematic representation of the network is represented at Figure 22.

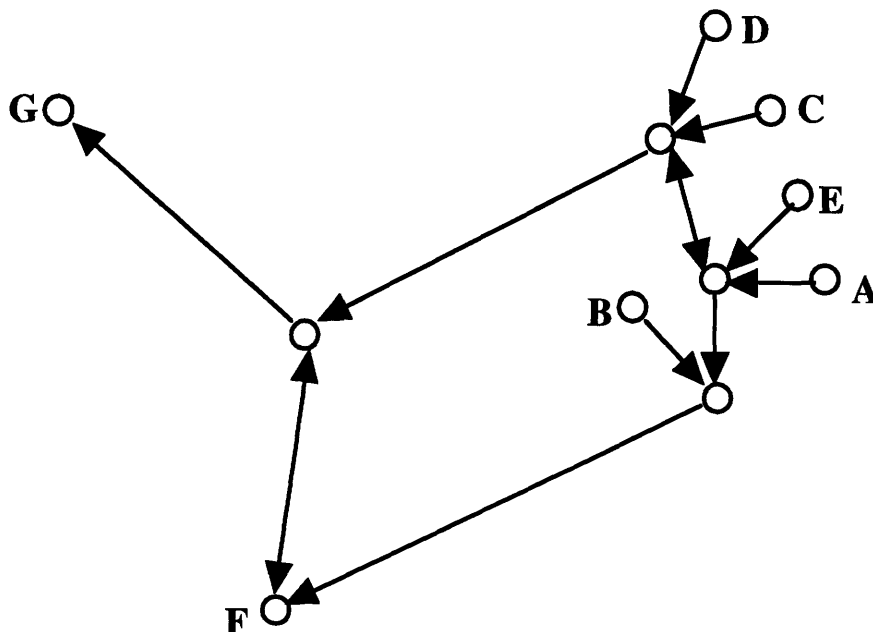


Figure 22. Schematic representation of the network

A sequence of 5 intervals –and their respective OD matrices– are generated. The length of these intervals has been selected to be 15 minutes, resulting in a 75 minute period, from 7:00am to 8:15am. The demand between 7:30 and 7:45 has been defined as follows. The total demand is 7200 vehicles/hour. It has been distributed across origins according to the number of lanes at entry points, and equally distributed across destinations. The resulting OD matrix for the 15 minutes period is given in Table 5.

Origin \ Dest.	F	G
A	300	300
B	150	150
C	225	225
D	75	75
E	150	150

Table 5. OD matrix for 3rd interval (7:30-7:45)

Furthermore, it is assumed, that the first and fifth interval correspond to 80% of the demand of the third interval and the second and fourth correspond to 90% of it. This demand pattern is presented in Figure 23.

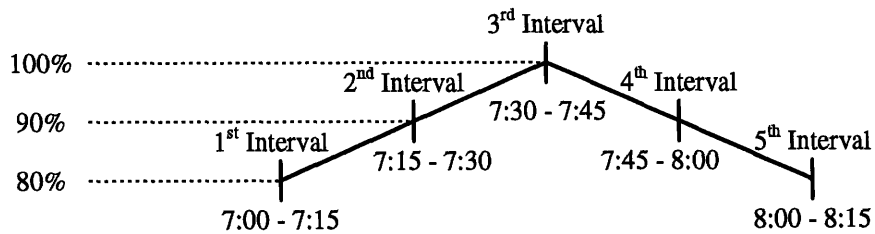


Figure 23. Demand pattern

The OD matrices that result from this process are presented in Table 6.

OD pair		Interval 1	Interval 2	Interval 3	Interval 4	Interval 5
1	A→F	240	270	300	270	240
2	A→G	240	270	300	270	240
3	B→F	120	135	150	135	120
4	B→G	120	135	150	135	120
5	C→F	180	202.5	225	202.5	180
6	C→G	180	202.5	225	202.5	180
7	D→F	60	67.5	75	67.5	60
8	D→G	60	67.5	75	67.5	60
9	E→F	120	135	150	135	120
10	E→G	120	135	150	135	120

Table 6. Generated OD matrices

Based on this demand pattern, three different historical demand scenarios (B1, B2, and B3) are generated:

- B1: 80% of the true demand levels are independently perturbed by a random number uniformly distributed between -10% and +10% of the demand.
- B2: the B1 demand levels are independently perturbed by a random number uniformly distributed between -5% and +5% of the demand.
- B3: the B1 demand levels are independently perturbed by a random number uniformly distributed between -10% and +10% of the demand.

5.1.3 Population-wide characteristics (C)

As described earlier, the pre-trip behavioral model requires a distribution for the value of time of the individuals. The distribution we consider here is arbitrary. It is presented in Table 7. This distribution could represent a time period in which a large proportion of the drivers are going to work and hence have a high value of time. This is consistent with the time that the simulation takes place (7:00am to 8:15am).

<i>Value of time</i>	<i>Proportion</i>
Low	20
Medium	30
High	50

Table 7. Population-wide characteristics distribution

5.1.4 Pre-trip behavior (D)

Pre-trip decisions are captured by a behavioral model (described in Section 3.6), that was estimated for the purposes of this evaluation process using artificial data. The number of parameters in the model were limited to provide an easier to estimate, yet representative, model. The model estimation was performed using the estimation software *Hielow* (Bierlaire, 1995, and Bierlaire and Vandevyvere, 1995) on an artificial data set. The estimated coefficients and constants are presented in Table 9 and Table 10.

The data set generation was based on a simple artificial network, consisting of one origin and one destination connected by two alternative routes.

The steps that were followed in the generation of the data set are described in the remainder of the section.

(i) Choice set generation

Each driver can:

- Maintain the habitual behavior,
- Change mode,
- Change route,
- Leave earlier or later using the same route, and
- Leave earlier or later and change route.

In order to reduce the number of alternatives to a manageable choice set, it was assumed that a driver only considers switching at most to two earlier or two later departure time intervals (five intervals of 15 minutes are considered, which is consistent with the demand described in Section 5.1.2). This assumption was motivated by the fact that drivers are usually constrained to complete their trip within some time frame –which restricts their ability to delay their departure– while at the same time are not willing to

switch their departure time to a lot earlier. The *cancel trip* option is not considered in this exercise.

Therefore, the choice set of the individual drivers consists of the following alternatives:

- Do not change travel behavior,
- Change mode,
- Change route,
- Leave **two** departure time intervals **earlier** using the **same** route,
- Leave **one** departure time interval **earlier** using the **same** route,
- Leave **one** departure time interval **later** using the **same** route,
- Leave **two** departure time intervals **later** using the **same** route,
- Leave **two** departure time intervals **earlier** and **change** route,
- Leave **one** departure time interval **earlier** and **change** route,
- Leave **one** departure time interval **later** and **change** route, and
- Leave **two** departure time intervals **later** and **change** route.

(ii) Identification of variables of interest

The second step in the generation of the data set was the identification of the variables of interest. The following variables were considered:

- Travel times for all choice alternatives,
- Deviation from the expected habitual arrival time,
- Value of time, and
- Information reception time.

(iii) Generation of list of values for each variable

A set of four travel time profiles are hypothesized and shown in Table 8. This table shows the travel times for each path and time profile, as they are supposed to be known to the decision maker. In particular, the third row shows the travel time if the traveler does not change departure time interval. The lines above that correspond to the travel time that the decision maker will experience if the decision is to leave one or two intervals early. Similarly, the lines below that correspond to the travel times that will be

experienced by the decision maker if the departure is delayed by one or two intervals. The first three profiles correspond to the beginning of a peak period and have increasing travel times. Profile 1 describes usual traffic conditions, while profiles 2 and 3 describe the effect of an incident on routes II and I respectively. The fourth profile corresponds to the end of a peak period and has decreasing travel times.

Time profile	1		2		3		4	
Route	I	II	I	II	I	II	I	II
Leave two intervals early	35	40	35	40	35	40	90	100
Leave one interval early	40	45	55	100	80	60	75	85
Do not change	45	50	65	160	130	90	60	70
Leave one interval late	50	55	65	160	130	90	45	55
Leave two intervals late	55	60	55	110	90	65	30	40

Table 8. Travel time profiles

For each route, the travel time of the *do not change* alternative for the first travel time profile was assumed to be the habitual travel time. The travel time for the public transit has been defined in relation to the habitual travel time. Two alternative values were assumed for public transit travel times (constant over travel time profiles):

- 20% higher than the individual's habitual travel time, and
- 20% lower than the individual's habitual travel time.

This implicitly assumes that the travel time for the competing mode is not affected by travel conditions. This is particularly true for modes with exclusive right-of-way such as rapid transit. The use of habitual travel time in setting transit travel times is for convenience only and does not reflect any inherent relationship between the two modes.

The expected arrival time for each choice alternative was computed by adding the travel time associated with that alternative to the respective departure time. Within each time interval the departure time was computed assuming a uniform departure rate. For each driver the departure time was determined randomly within its corresponding interval based on this distribution. The deviation of the individual's expected arrival time from the habitual arrival time, given by the sum of the habitual departure time and habitual travel time, was then computed.

Three values of time were assumed: high, medium and low. Finally, three information reception times were considered:

- 35 minutes prior to the habitual departure time,
- 50 minutes prior to the habitual departure time, and
- 60 minutes prior to the habitual departure time.

The selection of the information reception time values is such that, for all three alternatives, the information is received by the driver early enough so that departure time change to all intervals in the choice set can be considered.

(iv) Generation of population of drivers

The combination of two habitual routes, four travel time profiles, three information reception times, two public transit travel time patterns and three values of time results in a total of 144 choice scenarios. For each of these exercises, the probability that each alternative will be chosen was subjectively set. Although this is a subjective process, efforts were made to assign values that reflect reasonable decisions under the particular scenario.

A sample of 2114 individuals was created. The goal of the sample generation procedure was to generate 20 individuals per scenario, a total of 2880 individuals. To achieve that, the 20 individuals were assigned to the alternatives based on the probability. The numbers of individuals were rounded to the closest integer. Nevertheless, since many individual sets were rounded down, the process produced a smaller number of individuals. Each driver was assigned the attributes defined by the scenario and the chosen alternative. The resulting data set was used for the estimation of the model. Due to the artificial character of the data and consequently its possible limitations in embodying some of the more subtle decision processes a multinomial model was preferred over the more complex nested structure. The values of the estimated parameters are presented in Table 10.

Variable	Value	t-test
Travel time (in minutes), users with low value of time	-0.0118	-5.649
Travel time (in minutes), users with medium value of time	-0.0187	-8.597
Travel time (in minutes), users with high value of time	-0.0246	-10.28
Time that the user has received information (in minutes)	0.0389	9.428
Early deviation from the expected habitual arrival time (in minutes)	0.0060	1.040
Early deviation from the expected habitual arrival time (in minutes)	-0.1197	-17.49

Table 9. Estimated coefficients for the behavioral model

The travel time coefficients have the correct sign and their relative values are reasonable. Users with higher value of time have a higher (in absolute value) travel time coefficient.

Similarly, the information reception time coefficient also has the expected sign. This coefficient appears only in the *do not change* alternative, and, therefore, when the time that the information has been received is larger the utility of the *do not change* alternative is larger. This makes intuitive sense, since the users tend to consider older information less up-to-date and therefore less reliable. The tendency of the users to change their travel behavior based on this information is therefore lower, which is represented by a larger *do not change* utility.

The last two coefficients correspond to the early and late arrival deviations. These values should both be negative, since they actually correspond to penalties associated with deviations from the desired arrival time (see for example Small, 1995). The sign of late deviation penalty is correct. The early deviation coefficient sign is not. This is attributed to the lack of real data and the estimation of the model based on an artificial data set. It must be noted that the t-test value for that coefficient suggests that the value is not significantly different from zero in the model for a 95% confidence interval (t-test = 1.040). A model without this variable is worth estimating. However, in this thesis this model is adopted.

Constant	Value	t-test
Leave two intervals early constant	2.1377	0.049
Leave one interval early constant	1.95	3.624
Leave one interval late constant	0.794	7.092
Leave two intervals late constant	0.0221	8.633
Change path constant	1.267	5.301
Leave two intervals early and change path constant	2.404	0.069
Leave one interval early and change path constant	1.9906	3.400
Leave one interval late and change path constant	0.84225	7.168
Leave two intervals late and change path constant	0.0157	7.722
Change mode constant	0.14204	0.626

Table 10. Estimated constants for the behavioral model

The estimated constants which are associated with all the change alternatives have positive signs. This implies a tendency to change from the habitual travel pattern. This may not be realistic in some cases. Furthermore, this emphasizes the need for real data, which would capture the drivers' travel decisions better.

The behavioral parameters presented in Table 9 and Table 10 comprise behavioral scenario D1. In order to provide input for the sensitivity analysis scenario a perturbation is achieved by independently multiplying each parameter with a random number drawn from a uniform distribution in $[1-\epsilon, 1+\epsilon]$. Two alternative values of ϵ are used:

- D2: 5%, and
- D3: 10%.

5.1.5 En route behavior (E)

En route decisions are simulated by the traffic simulator MITSIM (Yang and Koutsopoulos, 1996) and, therefore, are captured by the route choice model incorporated in it. MITSIM has been developed for modeling traffic networks with advanced traffic control, route guidance and surveillance systems. MITSIM represents networks in detail and simulates individual vehicle movements using car following, lane changing, and traffic signal responding logic. A probabilistic route choice model is used to capture

drivers' route choice decisions in the presence of real time traffic information provided by route guidance systems. It is a multinomial logit model which considers the ratio of path travel time and shortest path travel time in choosing a route. The choice set includes all the outgoing links at the downstream node of the current link that may take the vehicle closer to its destination. Based on the route choice probabilities individual vehicles choose their next link at each node. It must be noted at this point, that the assumption is made that the traffic simulator perfectly replicates reality.

5.1.6 Guidance (F)

Guidance is represented in the form of travel times for all alternatives in the users' choice set. These travel times are generated from the simulation of the true demand (described in Section 5.1.2) by MITSIM, described in Section 5.1.5. Based on the occurrence of an incident in the network, two guidance scenarios are generated:

- F1: No incident (A1), and
- F2: Incident in the Third Harbor (Ted Williams) Tunnel (A2).

Therefore, in defining scenarios, F1 can only be paired with A1, and F2 can only be paired with A2.

5.1.7 Habitual travel time data (G)

For the determination of the habitual behavior of the drivers a set of habitual travel times is also required, besides the guidance. These travel times are generated from the simulation of the historical demand by MITSIM, described in Section 5.1.5. Three scenarios of habitual travel time data are generated, based on historical demand scenarios:

- G1: Historical demand B1,
- G2: Historical demand perturbed by 5% (B2), and
- G3: Historical demand perturbed by 10% (B3).

Therefore, in defining a scenario G1 can only be paired with B1, G2 with B2, and G3 with B3.

5.1.8 Link Counts (H)

Link counts are available from the traffic simulator, described in Section 5.1.5, which simulates the true demand, described in Section 5.1.2. The simulator outputs the

number of vehicles that cross each sensor at each time interval. This output is used to derive the link counts.

5.1.9 Assignment matrices (I)

The assignment matrices are computed using the traffic simulator and the historical demand described in Section 5.1.2, by tracking the movements of vehicles. Indeed, the traffic simulator keeps track of the departure time, origin and destination of each vehicle that crosses each sensor in the network. Three sets of assignment matrices are generated, depending on the input historical demand:

- I1: Historical demand (B1),
- I2: Historical demand perturbed by 5% (B2), and
- I3: Historical demand perturbed 10% (B3).

Alternatively, the assignment matrices could have been generated based on the true demand.

5.1.10 Transition matrices (J)

As mentioned earlier, transition matrices consist of elements which are essentially measures of the effects of lagged deviations in a flow on deviations in that flow. It is reminded that the approximation lies at the assumption that deviations in the r th OD flows are most affected by those in the few *preceding OD flows* and that contributions from other OD pairs are insignificant in comparison. The deviations were generated in order to be on average equal to 10% of the total demand, with a variance of 10%.

Using these deviations, a linear regression is performed on Equation 30 to obtain the transition matrices. An autoregressive process with degree 2 is assumed. This means that the effect of the flows of only two prior intervals in the flows of the current interval are taken into account. Under the approximation mentioned above, the transition matrices are expected to be diagonal.

5.1.11 Error covariance matrices (K)

The system error covariance matrices are estimated next. The matrix Q of equation 36 is obtained by taking the covariance of the residual vectors from an OLS regression on the transition equation (Equation 30). Similarly, the matrix R is obtained by taking the covariance of the residual vectors from the measurement equation (Equation

31), when the demand, the respective link counts and the estimated assignment matrices (cf. Section 5.1.9) are used. This procedure is described in more detail by Ashok (1996).

5.1.12 Initial state vector covariance (L)

The initial covariance of the state vector estimate is a measure of the certainty by which the initial vector estimate is known. Instead of estimating the matrix Σ analytically, a different approach is followed. A large covariance of the state vector is initially assumed. Each iteration of the OD estimation algorithm updates Σ , improving the certainty by which the state vector is known. Several iterations of the OD estimation algorithm are performed and the covariance of the state vector is thus estimated. The iterations are stopped when the decrease in the values of the matrix becomes small, i.e. convergence is achieved.

5.2 Impact of Behavioral Update

As it has been mentioned before, the proposed demand simulator extends the OD estimation process with a behavioral update in order to incorporate explicitly the effect of information on the demand. The impact of the behavioral update of the historical demand is explored in this section. Although the entire period from 7:00 to 8:15 is simulated, the results are only analyzed for interval 3.

A more ambitious evaluation aimed at assessing the value of this behavioral update in better estimating actual demand, however, is reserved for further research. A framework for such an evaluation is, therefore, presented in Chapter 6.

In this section, the process is described, the results are presented and analyzed, and conclusions are drawn.

5.2.1 Description of the process

The framework that is used is presented in Figure 24.

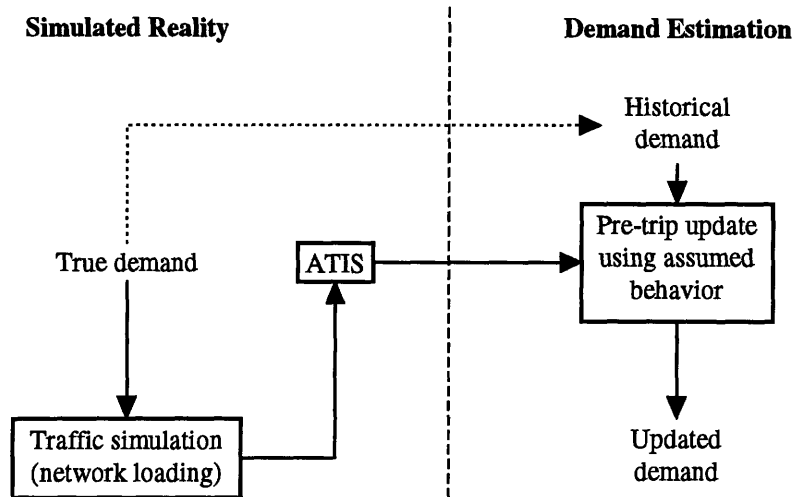


Figure 24. Impact of behavioral update process overview

Besides the demand estimation, this framework also includes a simulation of the reality. This is required for the generation of the inputs required by the demand simulation, in particular the guidance. As part of the simulated reality, the traffic simulation is run using the true demand, described in Section 5.1.2, in order to generate a guidance, in the form of travel time information, as described in Section 5.1.6. In order to assess the impact of the behavioral update on the historical demand, two guidance scenarios are generated, as described in Section 5.1.6:

- F1: No incident, and
- F2: Incident in the Third Harbor Tunnel (see Figure 21).

This guidance is used by the behavioral update in order to update the historical demand. This process also uses the behavioral parameters (D1) and the travel times produces from the historical demand (G1), which are used as habitual travel times in the update model.

Using the conventions that have been used to classify the alternative values of the dimensions that define the scenarios, the two scenarios can be characterized as:

- A1-B1-C-D1-F1-G1, and
- A2-B1-C-D1-F2-G1.

5.2.2 Analysis of the Results

The behavioral update component of the demand simulator is expected to play a more important role in the demand, under adverse traffic conditions in the network. This is derived from the observation that users of the network are more likely to update their

travel behavior under traffic conditions significantly different that the ones they usually experience.

The percent change of the updated demand (relative to the historical demand) for each case is presented in Figure 25. This figure shows the impact of the behavioral update on the demand both on average for all OD pairs and for each OD pair individually. The impact of the behavioral update is more significant in the presence of the incident. This is consistent with the intuitive expectation, since in the case of an incident the traffic conditions are significantly different than the historical and, therefore, users tend to change their travel pattern to respond to that. In particular, drivers would either delay their departure or leave earlier in order to avoid adverse traffic conditions, or even switch to public transit.

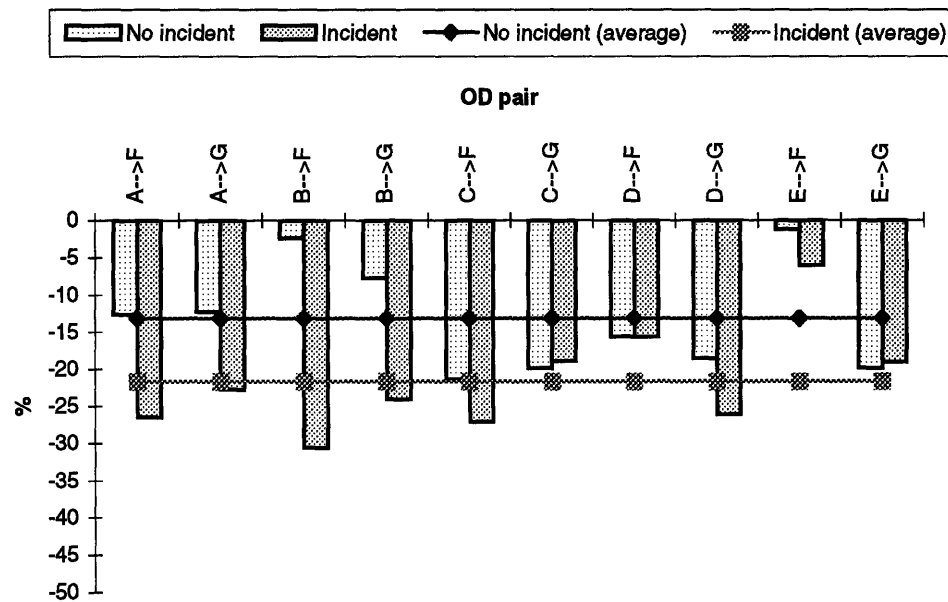


Figure 25. Deviation of updated from historical demand

The relatively high deviation of the updated demand is consistent with the behavioral model constants presented in Section 5.1.4. These constants imply a tendency of the drivers to change their travel behavior.

Moreover, in the context of this evaluation exercise, the demand pattern presents a peak during the third interval (which, as shown in Figure 23, is the interval of interest). Therefore, the traffic conditions on the network are generally worsening until the third interval and improving thereafter. This pattern results in a tendency of the guided drivers,

whose habitual departure time was within the third interval, to depart earlier or later. This tendency is larger than that of drivers, whose habitual departure time was in other intervals, to switch to the third time interval. Therefore, a decrease in the demand for the third interval is indeed expected.

5.2.3 Conclusions

This case study shows that the impact of the behavioral update is stronger in the presence of an incident. In particular, when the conditions that are prevailing in the network resemble the habitual traffic conditions, experienced by the drivers, a smaller number of drivers decide to change their travel behavior. On the contrary, when the conditions worsen and the travel times become larger, then the drivers exhibit a larger tendency to change their habitual travel pattern. These results are consistent with the motivations of the demand simulator design and the underlying assumptions. Therefore, they encourage further investigations on the value of the pre-trip decisions in response to guidance. An evaluation framework is proposed in Chapter 6.

5.3 Stochasticity Analysis

The second case study in the evaluation of the demand simulator relates to the assessment of the stochasticity of its output. As it has already been mentioned, the behavioral update component of the simulator introduces stochasticity with the use of statistical disaggregate models. This case study attempts to assess the extent to which this stochasticity is reflected in the aggregate output of the demand simulator. It is repeated here that stochasticity is not undesirable per se. Nevertheless, the existence of stochasticity would mean that in order to obtain a consistent estimate, the simulator would have to be run several times and its output would have to be averaged. Therefore, since the demand simulator is designed for use in a real-time context, high stochasticity, requiring a large number of repetitions, would be a disadvantage.

Like the previous case, although the entire period is simulated, the results are only analyzed for interval 3.

5.3.1 Description of the Process

The framework for the stochasticity analysis is presented in Figure 26.

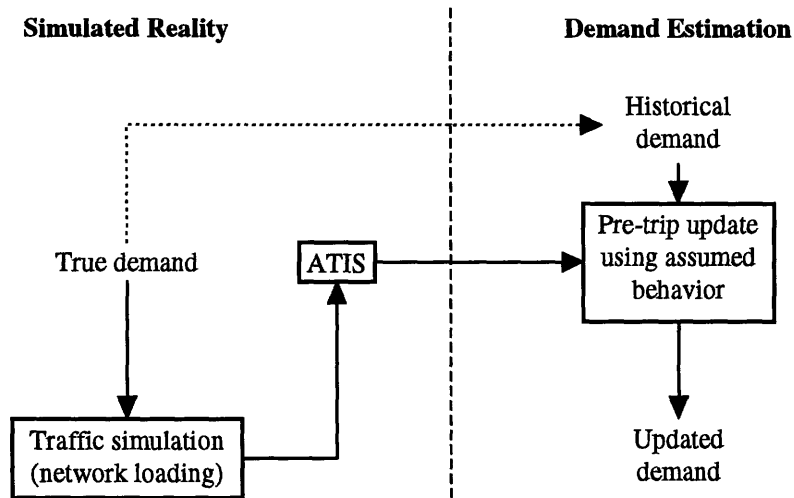


Figure 26. Stochasticity analysis process overview

Like the first case, this framework also includes a simulation of the reality, besides the demand estimation process. This is required for the generation of the inputs required by the demand simulation, in particular the guidance. As part of the simulated reality, the traffic simulation is run using the true demand, described in Section 5.1.2, in order to generate a guidance, in the form of travel time information, as described in Section 5.1.6. This guidance is used by the behavioral update in order to update the historical demand. The process also uses the behavioral parameters (D1) and the travel times produces from the historical demand (G1), which are used as habitual travel times in the update model. The inputs that were used are the ones generated for the no-incident scenario for the first case study (Section 5.2). The behavioral update component of the demand simulator is run 129 times, using the same guidance, and the variability in the output updated demand is observed. Indeed this is a component that is directly affected by this stochasticity. A similar analysis about estimated demand is left to future research.

Using the conventions that have been used to classify the alternative values of the dimensions that define the scenarios, this case uses the following dimensions:

- A1-B1-C-D1-F1-G1.

5.3.2 Analysis of the results

The measures that are used to assess the impact of the stochasticity in the behavioral update in the updated demand are the mean and the variance of the updated demand.

Figure 27 shows the evolution of the mean updated demand over realizations and Figure 28 shows the histogram of the generated values of the total updated demand.

Figure 29 shows the evolution of its standard deviation. The following three measures are shown in

Figure 27:

- Value of total updated demand for each iteration,
- Mean updated demand over all iterations, and
- Moving mean of updated demand of past iterations, that is the mean of the updated demand for all previous iterations.

The results can be summarized in the following points:

- The maximum deviation from the mean is approximately 5% of the mean.
- The standard deviation corresponds to 2% of the mean.
- As seen in Figures 27 and 29 the convergence of the estimate of the mean and standard deviation is slow.
- The histogram shown in Figure 28 indicates that 41% of realizations lie within less than 1% of the mean and 69% of realizations lie within less than 2% of the mean.

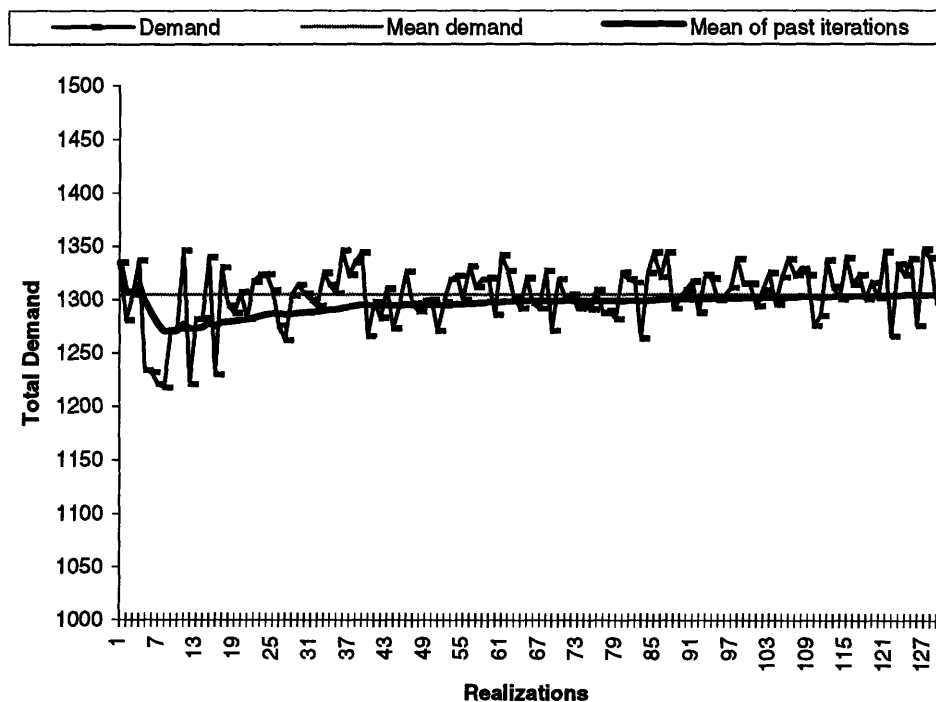


Figure 27. Total updated demand

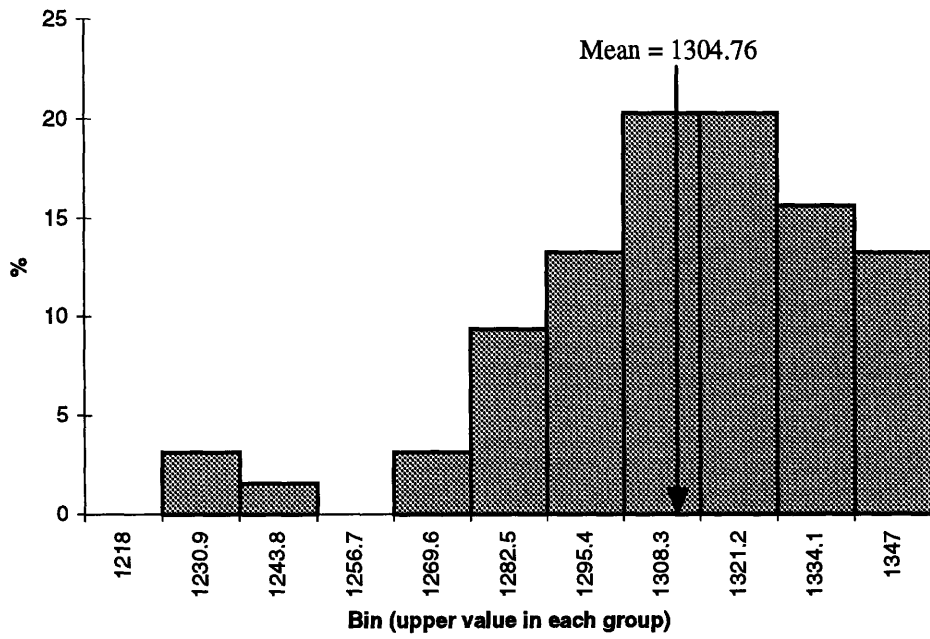


Figure 28. Distribution of total updated demand values

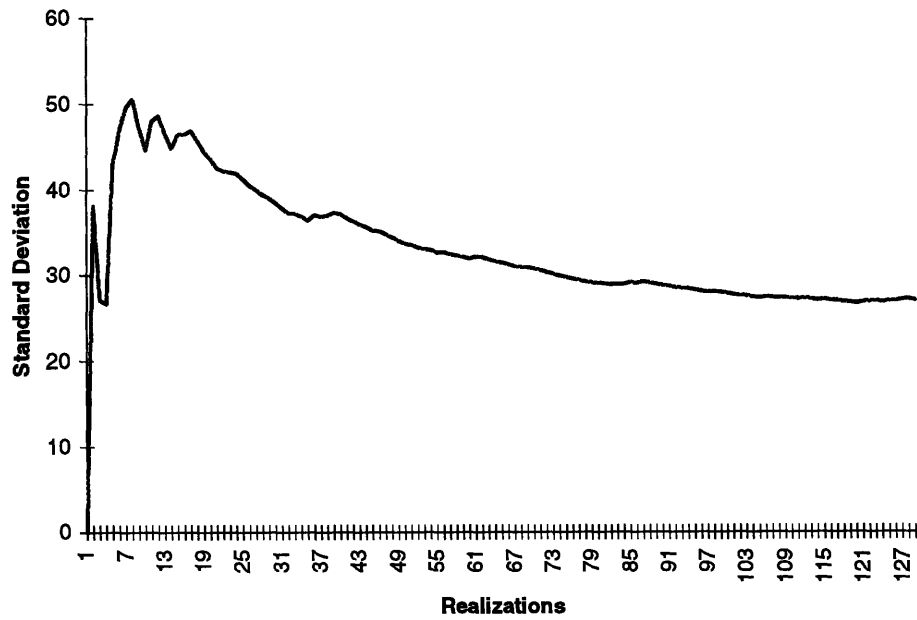


Figure 29. Total updated demand standard deviation

5.3.3 Conclusions

As expected, the stochasticity introduced in the behavioral update affects the updated demand. However, the impact on the total demand does not seem unreasonable. Further analysis should also be conducted in individual cells of the OD matrix, and in the final estimated demand. This is left to future research.

5.4 Sensitivity to Input

The purpose of the second evaluation case is to assess the sensitivity of the demand simulator's performance to variations in its input. In this evaluation case, the values of two important sets of inputs to the demand simulator are perturbed and their effect in the demand is discussed. Like the first two cases, although the entire period is simulated, the results are only analyzed for interval 3.

5.4.1 Description of the Process

The sensitivity of the output of the demand simulator to perturbations in the following inputs is assessed:

- Historical demand, and
- Behavioral parameters.

The sensitivity of both the updated demand and the estimated demand are examined in this analysis.

The followed process is presented in Figure 30.

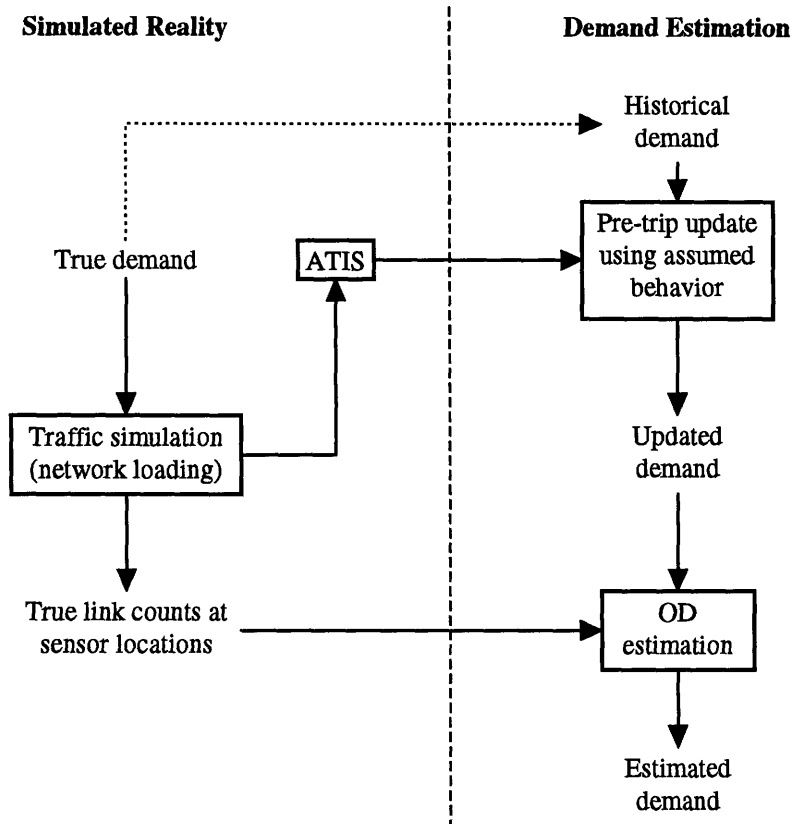


Figure 30. Sensitivity process overview

For each scenario, the true demand is input to the traffic simulator. The traffic simulator provides guidance in the form of travel times for all alternative paths and intervals, and link counts. The historical demand is then updated by the behavioral update, using the guidance and the appropriate behavioral parameters and historical travel time data. The updated demand is then used as input by the OD estimation, along with the true link counts. The OD estimation also requires a set of assignment matrices. These are generated using the appropriate historical demand according to the methodology described in Section 5.1.9. For each historical demand scenario (B1, B2, or B3) a different set of assignment matrices are generated (I1, I2, and I3).

This case is performed for 5 scenarios. In the first, the historical demand is B1 and the behavioral parameters are D1. This is the referent scenario for both sensitivity exercises. In the next two scenarios the historical demand is kept unchanged (B1), but the behavioral parameters are changed to D2 and D3. Similarly, in the last two scenarios the behavioral parameters are kept unchanged (D1), but the historical demand is changed to B2 and B3. Therefore, using the conventions that have been used to classify the alternative values of the dimensions that define the scenarios, the scenarios can be characterized as:

- A1-B1-C-D1-E-F1-G1-H-I1-J-K-L (no perturbation),
- A1-B1-C-D2-E-F1-G1-H-I1-J-K-L (behavioral parameters perturbed 5%)
- A1-B1-C-D3-E-F1-G1-H-I1-J-K-L (behavioral parameters perturbed 10%)
- A1-B2-C-D1-E-F1-G2-H-I2-J-K-L (historical demand perturbed 5%)
- A1-B3-C-D1-E-F1-G3-H-I3-J-K-L (historical demand perturbed 10%)

For each of these scenarios, the deviation (at the OD level) of the output for the perturbed input from the output for the original input is computed. Furthermore, at the aggregate demand level the relative error of the output when the perturbed inputs are used, relative to the original output, is computed. The value of the relative error of an estimated measure \hat{x} (updated or estimated demand using perturbed inputs) from a referent value x (updated or estimated demand using inputs without perturbation) is computed as:

$$\varepsilon_r = \frac{\|\hat{x} - x\|_p}{\|x\|_p} \quad x \neq 0 \quad (46)$$

where p is the norm that is used. For the purposes of this evaluation exercise, the norm 2 is used, which is given by:

$$\|x\|_2 = \left(|x_1|^2 + \dots + |x_n|^2 \right)^{1/2} \quad (47)$$

5.4.2 Analysis of the results

The first scenario that is examined for this case is the perturbation of the behavioral parameters. For the purposes of this case, the impact of the perturbation in the updated and the estimated demand is examined. This is presented in terms of deviation of the updated or estimated demand generated using perturbed behavioral parameters from the respective demand in the case where no perturbation was done in the behavioral parameters (Figure 31 and Figure 32). Furthermore, the relative errors of the updated and estimated total demand obtained from perturbed behavioral parameters relative to the updated and estimated total demand obtained when the original behavioral parameters were used are presented in Figure 33.

A similar analysis is performed for the impact of the perturbation of the historical demand in the updated and the estimated demand. Figure 34 shows the deviation of the updated demand for the perturbed historical demand scenarios from the original updated

demand, whereas Figure 35 shows the deviation of the estimated demand for the perturbed historical demand scenarios from the original estimated demand. Figure 36 shows the relative error of the estimated demand when the historical demand is perturbed from the original estimated demand.

In all cases total demand is stable. Nevertheless, a few OD pairs exhibit behavior that is somewhat not compatible with the anticipated. In particular, the deviation in the updated and estimated demand in a few OD pairs is equal or even larger than the perturbation in the behavioral parameters. Also, in two OD pairs the deviation is inversely related with the perturbation. The explanation for these observations probably lies in the following facts:

- Simulator is stochastic, and
- Perturbations were stochastic and independent, because they reflect errors in input data.

Based on the results presented in the previous section, the level of error due to the perturbations is of the same order of magnitude as the error due to stochasticity. The stochasticity of the behavioral update process contributes to the random pattern that these figures show. As it has already been explained in Section 3 and has been exhibited in Section 5.3.3, the pre-trip behavioral update incorporates stochasticity, which is to some degree reflected in the updated demand. Therefore, the values obtained from a single run of the simulator may deviate from the values that a sufficient number of iterations would converge to. These observations motivate the need of sensitivity tests based on several runs instead of just one.

The stochasticity in the perturbation means that the value of some parameters in the first case (5% perturbation) may be further away from the original value than the value that was obtained when the parameter is perturbed in the second case (10%). Similarly, since the perturbation may be both additive or subtractive, the 5% perturbed value for a parameter may be 5% less than the original value and the 10% perturbed value may be 10% higher than the original value, thus resulting in a 15% difference between the two values.

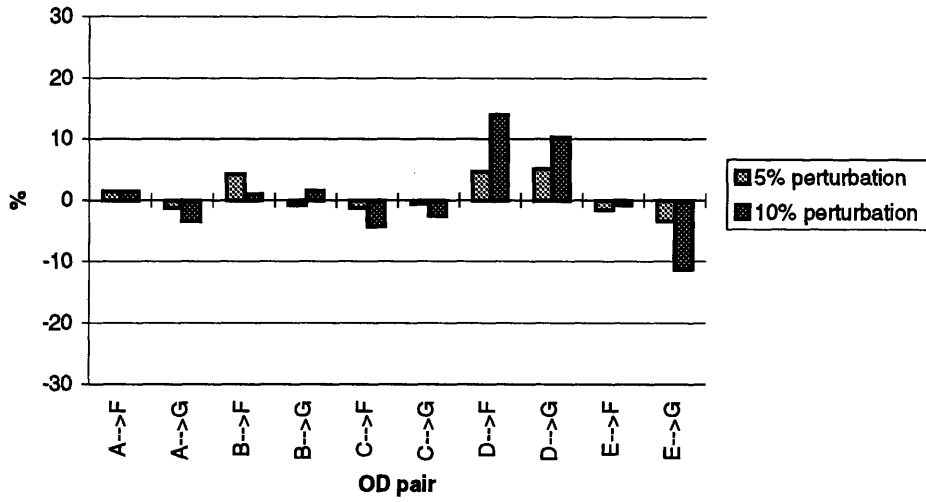


Figure 31. Deviation of updated demand due to perturbation of the behavioral parameters

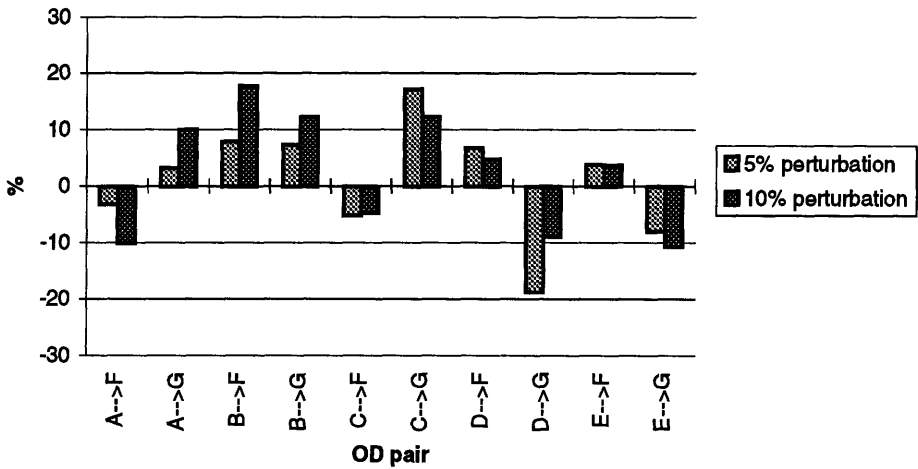


Figure 32. Deviation of estimated demand due to perturbation of the behavioral parameters

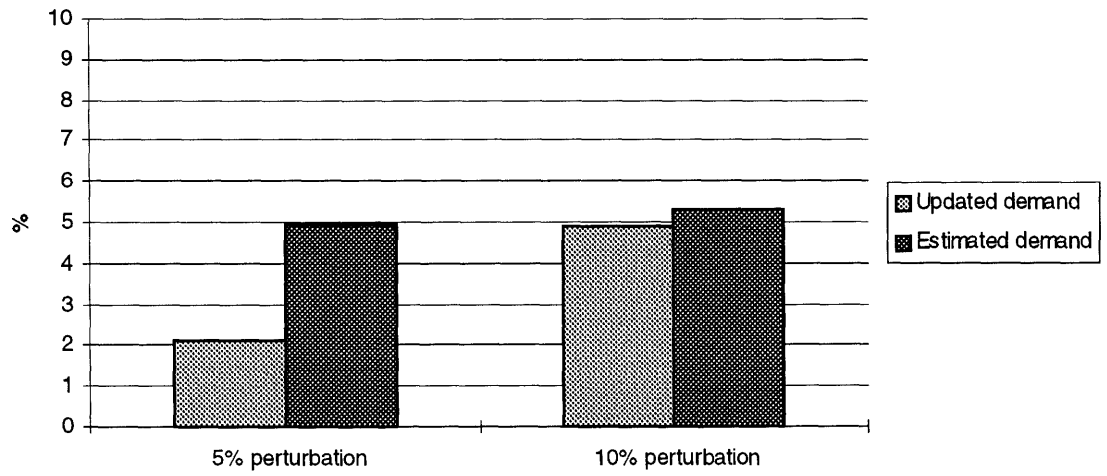


Figure 33. Relative error in the updated and estimated demand due to perturbations in the behavioral parameters

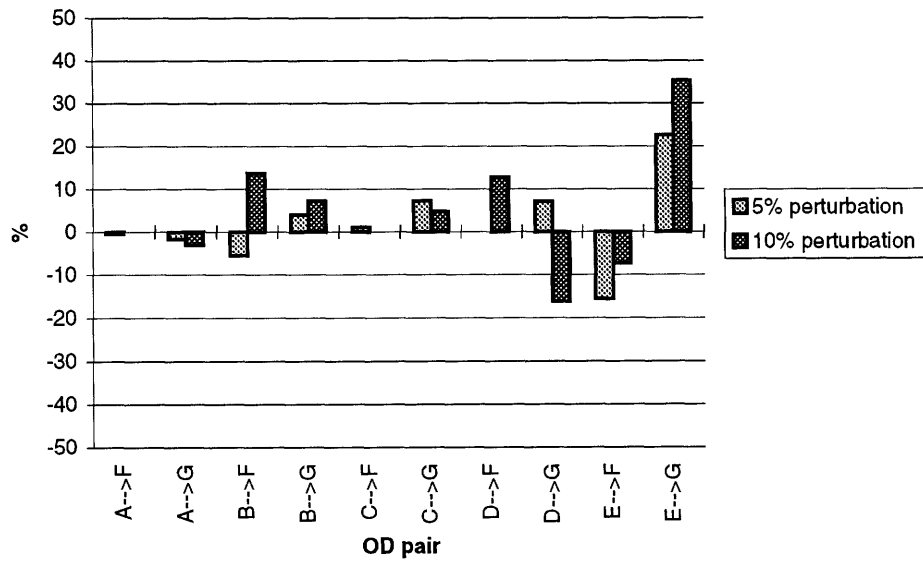


Figure 34. Deviation in updated demand due to perturbation of the historical demand

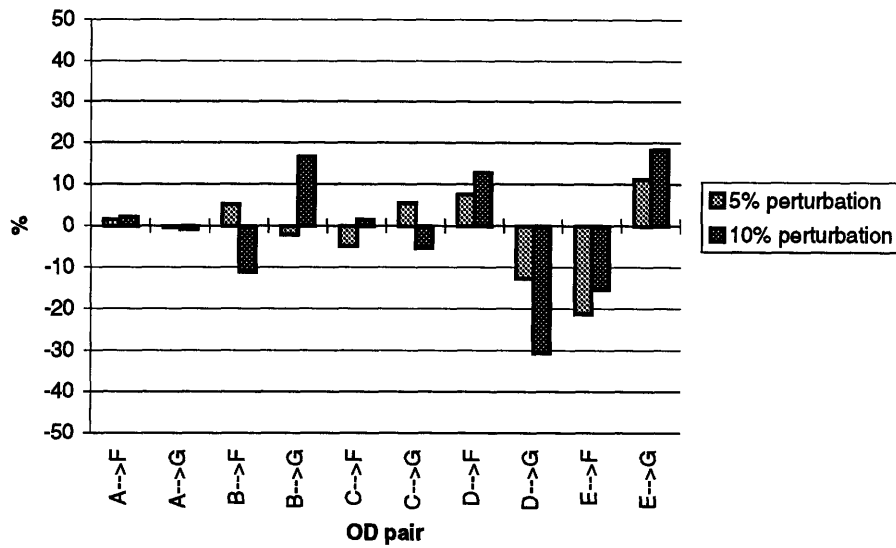


Figure 35. Deviation of estimated demand due to perturbation of the historical demand

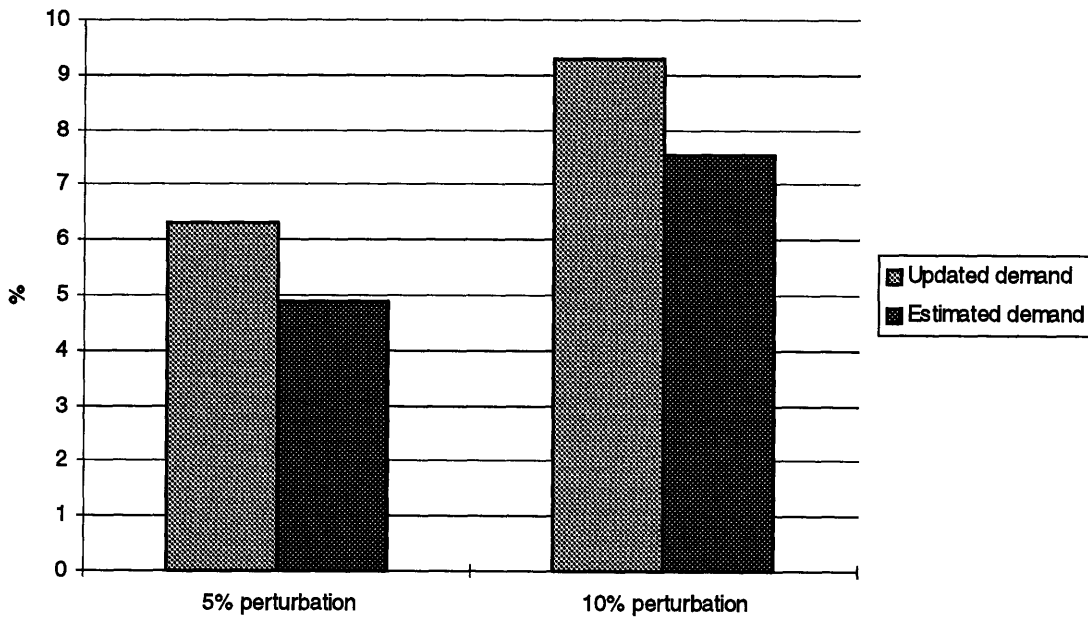


Figure 36. Relative error in the updated and estimated demand due to perturbations of historical demand

It is also interesting to analyze the sensitivity of the link counts to perturbations in the inputs. Figure 37 presents the relative errors in the estimated link flows due to perturbation in the historical OD flows. The effect of the perturbation of the historical demand on the estimated link flows is very small.

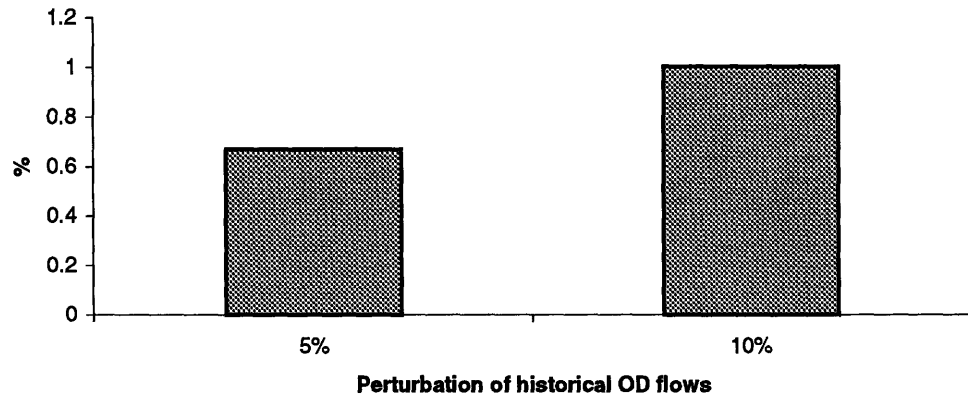


Figure 37. Relative errors in estimated link flows due to perturbation in the historical OD flows

5.4.3 Conclusions

As a conclusion, it can be stated that although the aggregate results seem pretty stable with respect to noise in the input, additional sensitivity analysis in specific disaggregate outputs is desirable. Also, sensitivity analysis should be performed over many replications, since stochasticity has turned out to be relevant.

6. Conclusions

This chapter provides a summary of the thesis and suggests a number of topics for further research.

6.1 Summary

The demand simulator developed in this thesis uses a number of inputs to estimate demand. These inputs include:

- Aggregate historical demand,
- Real-time information and guidance, and
- Link counts.

The demand simulator incorporates the effect of pre-trip information and guidance provision to update the historical demand prior to OD estimation, in order to capture the drivers' response to real-time information available at the pre-trip stage. Although the OD estimation model is applied on aggregate OD matrices, the individual choice of each driver is captured by disaggregate behavioral models. Thus, variations of travel behavior can be captured at the individual driver level. This is important because it allows the simulator to use individual driver characteristics to capture travel behavior in a potentially more accurate fashion, rather than being limited in capturing behavior at the OD level. In order to be able to use disaggregate models, though, the demand simulator disaggregates the historical OD matrices into a population of drivers, which are updated and subsequently aggregated to produce the updated OD matrices that are then used as input to the OD estimation model.

Therefore, besides the two main functions of the demand simulator:

- Travel behavior update in response to information, and
- Dynamic OD estimation and prediction,

two more components are required by the demand simulator:

- Disaggregation of the aggregate historical demand, and
- Aggregation of the updated demand.

Finally, depending on the nature and the requirements of each application of the demand simulator, its output can be either:

- Aggregate demand, or
- Disaggregate demand.

If aggregate demand is required as output, then no further operation needs to be performed and the estimated and predicted OD matrices are the desired output. On the other hand, if disaggregate demand is required, then the estimated and predicted OD matrices are disaggregated to a list of drivers by an additional disaggregation component.

In the generation of the individual drivers from the historical OD matrices, the disaggregation component assigns to each driver a number of socioeconomic characteristics, such as value of time, and trip characteristics, such as trip purpose. Information from the OD matrices that are used for the generation of the drivers is used to assign origin, destination and habitual departure time interval to these drivers. Also, car is assigned to all individuals as habitual mode. Furthermore, a habitual behavior model is applied to each driver in order to generate habitual travel behavior, i.e. path, based on historical information. The behavioral model that is used to provide choice probabilities for each path is the C-logit model, proposed by Cascetta (1993). The habitual behavioral model uses the socioeconomic and trip characteristics of each driver, and historical travel time information to generate choice probabilities for all paths in the driver's choice set. A set of enumerated paths is assumed for each OD pair. Monte Carlo simulation is used to select one of these paths, based on the generated probabilities, and assign it to the driver as the habitual route.

At this stage of the demand simulation process, a population of individual drivers with corresponding socioeconomic and trip characteristics, as well as habitual choices on departure time, mode and route, is available. These choices reflect a priori decisions which can be updated based on real-time information and guidance. The pre-trip behavior update applies a behavioral model to each individual driver in the historical population to capture their travel behavior in response to available information. The drivers may decide to change departure time, path, mode, a combination of these, or even cancel their trip. Nevertheless, it is assumed that the drivers' destination is fixed and it cannot be changed in response to available information. Different behavioral models are used depending on whether prescriptive or descriptive information is provided to the driver.

At this point, the effect of pre-trip information in the drivers' travel behavior has been captured, and a disaggregate updated population is available. The OD estimation, though, needs aggregate demand as input. An aggregation component is used to generate updated OD matrices by aggregating the updated population of drivers. The updated OD

matrices are used as input for the OD estimation model. The aggregation is based on the departure time interval, the origin and the destination of the drivers.

The next step is the OD estimation and prediction. The OD estimation and prediction model is a model for real-time OD estimation and prediction, proposed by Ashok (1996). The model is a state-space formulation that uses deviations of OD flows from historical values as unknown variables. The solution of the model is given by a *square root* Kalman Filtering algorithm (Chui and Chen, 1987).

The demand simulator is implemented using the Object Oriented paradigm. The programming language of choice is C++ with the Standard Template Library (STL). The system is implemented as a client/server distributed application. Finally, the Object Modeling Technique (OMT) has been selected as the object design methodology.

The following general objectives were set and followed during the implementation stage:

- Object oriented (OO) design,
- Computational efficiency, and
- Numerical robustness.

The objects that comprise the demand simulator are designed in a way that they provide:

- Flexibility in the design,
- Efficient access of their elements, and
- Efficient storage.

A number of implementation issues concerning the components of the demand simulator are also discussed in detail.

Finally, a number of evaluation exercises were performed. These were focusing on specific aspects of the performance of the demand simulator and dealt with the following issues:

- Impact of the behavioral update to the demand simulator's output under different traffic conditions,
- Effect of stochasticity inherent in the models required for the behavioral update on the stochasticity of its output, and
- Sensitivity of the simulator's output to perturbations in some of its inputs.

These tests provided valuable information. Although this information can not validate the simulator alone, it provides indicative evidence that its performance is compliant to the design. In particular:

- Behavioral update plays a stronger role in the demand simulation, when adverse traffic conditions are observed in the network. This is consistent with the intuitive expectations, since the need to update the habitual travel behavior is greater in the case that the traffic conditions are significantly different than those usually experienced by the drivers.
- Stochasticity inherent in the behavioral update of the simulator is indeed reflected at the aggregate output level. Nevertheless, its impact does not seem unreasonable.
- Small changes in the inputs of the demand simulator reflecting input inaccuracies are not reflected dramatically in its outputs.

Further research is required on the topic of the evaluation of the demand simulator. These conclusions have to be verified and extended in a complete evaluation framework, and the simulator's overall performance has to be assessed. Therefore, most of the topics suggested for further research, presented in the next section, are associated with the evaluation of the simulator.

6.2 Further Research

This section presents a number of topics for potential research in this area. Of course, it does not attempt to capture the entire spectrum of further research that could be derived from this work. Nevertheless, this could be used as a starting point for the selection of a relevant topic.

6.2.1 Extensions of Performed Tests

A number of evaluation exercises have been performed in Chapter 5. These evaluation exercises led to some interesting conclusions about the demand simulator and pointed to additional exercises that could provide valuable information. In relation to the assessment of the impact of the behavioral update, presented in Section 5.2, a more general framework is proposed and, therefore, is presented in a separate section (Section 6.2.3). In this section, proposed extensions to stochasticity and sensitivity analysis are summarized.

In Section 5.3, the impact of the stochasticity of the demand simulator on the total updated demand was assessed. Future work should also address the following issues:

- Impact of the stochasticity in the estimated demand: although the stochasticity is introduced by the behavioral update, it is also important to assess its impact at the estimated demand, which is the final output of the simulator.
- Analysis of the stochasticity at the individual OD cell level: the impact of the stochasticity has only been analyzed at the total demand level. It is possible that the stochasticity analysis at the individual OD cell level will provide more information about the effect of stochasticity on the performance of the demand simulator.

In Section 5.4, the sensitivity of the simulator in two of its inputs (behavioral parameters and habitual demand) has been investigated. There are more inputs that may not be perfectly known, thus compromising the ability of the demand simulator to replicate demand accurately. Therefore, it is important to know to what extent imperfection in the knowledge of some input would affect the simulator's output. Further research could include the investigation of the sensitivity of the demand simulator to perturbations in additional inputs, including:

- Link counts, and
- Transition matrices.

Finally, although the sensitivity analysis, performed in Section 5.4, was based in a single run, future sensitivity analysis should be based on several runs, for the same interval, in order to limit the effect of stochasticity in the results.

6.2.2 Assessment of Real-time performance

As it has already been mentioned, the demand simulator has been developed in a real-time context. The assessment of the real-time performance of the simulator would be an interesting topic for further research. This topic should include, but not be limited to, the following issues:

- Execution speed: the speed of the simulator should be such that it allows its use in a real-time environment. In particular, in a DTA application the selection of the time intervals of operation depends on the execution speed of its components. Slow components could result in large intervals which

are undesirable, as they may have negative impact on the ability of the DTA to react promptly to changing conditions in the network.

- **Size of network that can be processed:** a DTA system is more likely to be applied to large networks. The demand simulator stores and processes a large amount of information. The size of the network that can be processed by the demand simulator may be limited by the amount of information that can be handled by the available resources (in particular memory of the computer). This problem is stressed by the fact that the demand simulator must coexist with the rest of the other modules of the DTA and share the same hardware. The distributed design of the demand simulator (cf. Section 4.1) may provide a solution to this problem. Since the system can be distributed to more than one computer, the hardware requirements can be shared by these machines.
- **Tradeoffs between speed and accuracy:** the demand simulator has been designed to provide the best estimate of the demand. Nevertheless, it is possible that compromises that could offer significant computational advantages may not affect its performance significantly. The identification of such potential and the evaluation of the tradeoffs is another interesting topic for future research.

6.2.3 Evaluation of the Demand Simulator

A series of evaluation exercises that provide useful information about the performance of the demand simulator are presented in Chapter 5. Nevertheless, they do not cover the evaluation of the demand simulator completely. This section presents a framework that can be used for a more comprehensive evaluation of the demand simulator. This framework is presented in Figure 38.

The framework proposed here describes an evaluation methodology in a simulated environment. This is especially useful when no real data is available to compare with estimated results. A basic assumption in such an evaluation context is a full knowledge of the (simulated) reality by the analyst. Nevertheless, to ensure the validity of the evaluation, the system to be tested will only be aware of some aspects of this reality, sometimes containing errors.

The simulated reality uses a number of external inputs and generates inputs required for the demand simulator. Furthermore, the simulated reality provides measures that will be used for the evaluation of the performance of the demand simulator.

The inputs to the overall framework are:

- Habitual demand,
- ATIS information,
- Behavioral model parameters,
- Socioeconomic characteristics, and
- Network,

and the output is the estimated demand. The habitual demand refers to one particular day, for which the estimation takes place. This is the “intended” demand, which does not incorporate the effect of the real-time information provided by the ATIS. The ATIS information is assumed to be a scenario, computed externally and provided as input (an even more ambitious and comprehensive evaluation where guidance information is internal to the process can be conceived). Similarly, the behavioral parameters and the socioeconomic characteristics are external inputs. Finally, the network is assumed to be known, as well. All these inputs are restricted to the simulated reality, with the exception of the ATIS information which is also used by the pre-trip update in the demand estimation being evaluated since it should be known to it in reality. The demand estimation also uses data generated from the simulated reality:

- Link counts at sensor locations, and
- Assignment matrices,

as well as data derived from other information. In particular:

- Historical demand is constructed as an average of the estimated true demand over many prior days, and
- Behavioral model parameters for the demand simulator are generated as a perturbation of the model parameters provided for the simulated reality. Indeed, the *true* behavior is not known perfectly by the demand simulator.

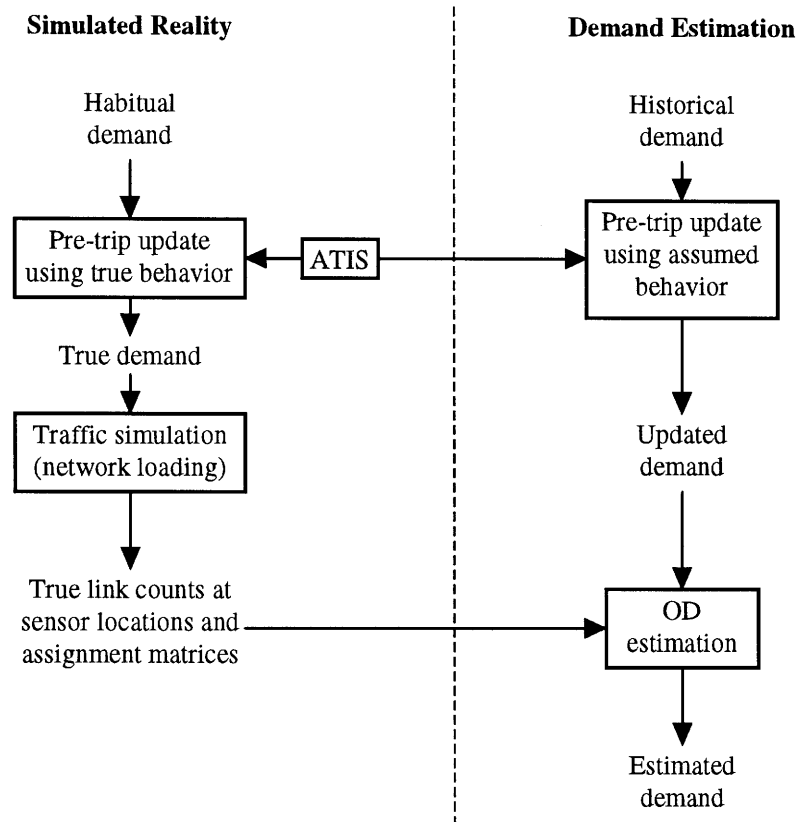


Figure 38. Simulation Based Evaluation of Demand Estimation

The representation of the demand simulator in Figure 38 is the demand simulator that has already been described. The process starts with the historical demand. This historical demand is updated by the pre-trip update to reflect the impact of information. The information is the same information that is provided by the ATIS to the simulated reality. The result of this update is an updated demand, which incorporates the impact of information. This demand, along with link counts at the sensor location and assignment matrices –both available from the simulated reality– is input to the OD estimation. The output of the OD estimation is the estimated demand.

The simulation of the reality follows a similar process. This process starts with a known habitual demand, and given ATIS information, behavioral parameters and socioeconomic characteristics. The habitual demand is updated using the provided information. The demand resulting from this process is considered to be true. It incorporates the effect of information provided to the drivers at the pre-trip stage. This true demand is loaded on the given network from the traffic simulator. As described above, this process provides two important inputs for the demand simulation:

- Link counts at sensor locations, and

- Assignment matrices.

In order to get a good estimate of the historical demand, it is important to run the simulation of the reality for several days and average the corresponding true demand realizations across these days.

In the context of this evaluation framework, a number of interesting tasks can be performed. The first one is the assessment of the ability of the simulator to replicate true demand. The major motivation behind the entire demand simulation is to use a number of known inputs and combine them to get a good estimate of the unknown true demand. With the existence of a simulation environment, in which the *true* demand is known to the analysts performing the evaluation exercise, the ability of the simulator to estimate demand sufficiently close to the true can be assessed. For this exercise, the estimated demand needs to be compared with the *true*, as it was loaded to the network in the simulation of the reality. Additional information may be drawn by comparison of the estimated demand with the historical and the updated demand. This exercise should be conducted for many intervals, since the behavioral models move people through intervals.

7. References

- Ashok, K. (1996), Estimation and Prediction of Time-Dependent Origin-Destination Flows, Ph.D. Thesis, Department of Civil and Environmental Engineering, MIT.
- Ashok, K. and M. Ben-Akiva (1993), Dynamic O-D Matrix Estimation and Prediction for Real-Time Traffic Management Systems, in Daganzo, C.F. (ed), Transportation and Traffic Theory, Elsevier Science Publishing Company Inc.
- Ashok, K and M. Ben-Akiva (1994), Models for Real-Time Estimation and Prediction of Time-Dependent O-D Matrices, Working Paper, Center for Transportation Studies, Massachusetts Institute of Technology, Cambridge.
- Badoe, D. A., and E. J. Miller (1995), Analysis of the Temporal Transferability of Desegrate Work Trip Mode Choice Models, Presented at the 74th Annual TRB Meeting, Washington, D.C.
- Ben-Akiva, M. (1973), Structure of Passenger Travel Demand Models, Ph.D. dissertation, Department of Civil Engineering, MIT.
- Ben-Akiva, M. (1987), Methods to Combine Different Data Sources and Estimate Origin-Destination Matrices, in Gartner, N.H. and Wilson, N.H.M. (eds), Transportation and Traffic Theory, Elsevier Science Publishing Company Inc.
- Ben-Akiva, M., M. J. Bergman, A. J. Daly, and R. Ramaswamy (1984), Modeling Inter-Urban Route Choice Behavior, in Proceedings of the Ninth International Symposium on Transportation and Traffic Theory, VNU Science Press, Utrecht, Netherlands.
- Ben-Akiva, M., E. Cascetta, H. Gunn, S. Smulders, and J. Whittaker (1994), DYNA: A Real-Time Monitoring and Prediction System for Inter-Urban Motorways, Proceedings of the First World Congress on Applications of Transport Telematics and Intelligent Vehicle-Highway Systems, Paris, France.
- Ben-Akiva, M., and I. Kaysi (1991), Dynamic Driver Behavior Modeling for Advanced Driver Information Systems, Paper presented at the 6th international conference on travel behavior, Quebec, Canada.
- Ben-Akiva, M., and S. Lerman (1979), Disaggregate Travel and Mobility Choice Models and Measures of Accessibility, In Behavioral Travel Modelling, D. Henscher and P. Stopher, eds, Croom Helm.

- Ben-Akiva, M., and S. Lerman (1985), *Discrete Choice Analysis: Theory and Applications to Travel Demand*, The MIT Press, Cambridge, MA.
- Bierlaire, M. (1995), *A Robust Algorithm for the Simultaneous Estimation of Hierarchical Models*, GRT Report 95/3, Department of Mathematics, FUNDP.
- Bierlaire, M., and Y. Vandevyvere (1995), *HieLoW: The Interactive User's Guide*, Transportation Research Group - FUNDP, Namur.
- Bonsall, P. W. (1992), *The Influence of Route Guidance Advice on Route Choice in Urban Networks*, Transportation No. 19.
- Boyce, et al. (1993), *A Combined Dynamic Mode/Departure Time/Route Choice Model with Multiple Groups of Travelers*, 72nd TRB Annual Meeting, Washington D.C.
- Bovy, P .H. L., and E. Stern (1990), *Route Choice Wayfinding in Transportation Networks*, Kluwer Academic Publishers, 1990.
- Bradley, M.A. and A. J. Daly (1991), *Hague Consulting Group, Estimation of Logit Choice Models Using Mixed State Preference and Revealed Preference Information*, Proceedings of the 6th International Conference on Travel Behavior, International Association for Travel Behavior, Quebec, Canada.
- Cascetta, E. (1995), *A Modified Logit Model for Route Choice*, Department of Transport Engineering, Internal Report, Napoli, Italy.
- Cascetta, E., and G. Biggiero (1992), *Analysis and Modeling of Commuters Departures and Route Choices in Urban Networks*, in *Second International Capri Seminar on Urban Traffic Networks*.
- Cascetta, E., M. Di Gangi, K. Linveld, and P. Mijjer (1994), *A Dynamic Traffic Assignment Model for Real Time Applications*, Proceedings of the First World Congress on Applications of Transport Telematics and Intelligent Vehicle-Highway Systems, Paris, France.
- Cascetta, E., D. Inaudi, and G. Marquis (1993), *Dynamic Estimators of Origin-Destination Matrices using Traffic Counts*, *Transportation Science*, Vol. 27, No. 4.
- Cascetta, E., and S. Nguyen (1989), *A Unified Framework for Estimating or Updating Origin/Destination Matrices from Traffic Counts*, *Transportation Research B*, Vol. 22, No. 6, pp. 437-456.

- Cascetta, E., A. Nuzzolo, F. Russo, and A. Vitetta (1996), A Modified Logit Route Choice Model Overcoming Path Overlapping Problems. Specification and Some Calibration Results for Interurban Networks.
- Chang, G., and J. Wu (1994), Recursive Estimation of Time-Varying Origin-Destination Flows from Traffic Counts in Freeway Corridors, *Transportation Research*, Vol. 28B, No. 2.
- Chui, C. K. (1987), G. Chen, *Kalman Filtering with Real-Time Applications*, Springer-Verlag.
- Cremer, M. and H. Keller (1987), A New Class of Dynamic Methods for the Identification of Origin-Destination Flows, *Transportation Research*, Vol. 21B, No. 2.
- de Palma, A., M. Ben-Akiva, C. Lefevre, and N. Litinas (1983), Stochastic Equilibrium Model of Peak Period Traffic Congestion, *Transportation Science*, Vol. 17.
- Fatoohi, R., (1996), Performance Evaluation of Communication Software Systems for Distributed Computing, submitted to the 2nd Conference on Object-Oriented Technologies and Systems (COOTS), February 1996.
- Friesz, T. L., D. Bernstein, T. E. Smith, R. L. Tobin, and B. Wie (1993), A Variational Inequality Formulation of the Dynamic Network User Equilibrium Problem, *Operations Research*, Vol. 41.
- Gamma, E., R. Helm, R. Johnson, and J. Vlissides (1995), *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison Wesley Professional Computing Series.
- Gill, P. E., and W. Murray (1974), Newton-type Methods for Unconstrained and Linearly Constrained Optimization, *Mathematical Programming* 28, pp. 311-350.
- Golub, G. H., Charles F. Van Loan (1983), *Matrix Computations*, North Oxford Academic.
- Inaudi, D., E. Kroes, S. Manfredi, and S. Toffolo (1994), The DYNA On-line Origin-Destination Estimation and Prediction Model, *Proceedings of the First World Congress on Applications of Transport Telematics and Intelligent Vehicle-Highway Systems*, Paris, France.
- Jou, R., and H. Mahmassani (1993), In an Urban Environment: Departure Time and Route Decisions, Presented at the 73rd Annual TRB Meeting, Washington, D.C.

- Jou, R., and H. Mahmassani (1994), Compatibility and Transferability of Commuter Behavior Characteristics Between Cities: Departure Time and Switching Decision, Presented at the 73rd Annual TRB Meeting, Washington, D.C.
- Khattak, A., A. Polydoropoulou, and M. Ben-Akiva (1996), Modeling Revealed and Stated Pre-Trip Response to ATIS, Submitted for Publication in Transportation Research Record.
- Kitamura, R. and P. P. Jovanis (1991), Driver Decisionmaking with Route Guidance Information, paper presented at TRB 70th Annual Meeting.
- Koutsopoulos, H. N., and Yang, Qi, (1996), A Microscopic Traffic Simulator for Evaluation of Dynamic Traffic Management Systems, Submitted for publication in Transportation Research C.
- Lappin, J., S. M. Sloan, and R. F. Church (1994), A Market Analysis of the Commercial Traffic Information Business, Technical Paper, US DOT/Research and Special Programs Administration.
- Lerman, S. (1993), Problem Solving and Computation for Scientists and Engineers. An Introduction Using C, Prentice Hall.
- Lotan, T., and H. N. Koutsopoulos (1993), Approximate Reasoning Models for Route Choice Behavior in the Presence of Information, in Proceedings of the 12th International Symposium on Transportation and Traffic Theory, Berkeley, California.
- Mahmassani, H., and G. L. Chang (1987), On Boundary Rational User Equilibrium in Transportation Systems, Transportation Science, Vol. 21.
- McFadden, D. (1978), Modeling the Choice of Residential Location, In Spatial Interaction Theory and Residential Location, A. Karlquist et al., eds, North Holland, pp.75-96.
- Nelson, M. (1995), C++ Programmer's Guide to the Standard Template Library, IDG Books.
- Nihan, N. L., and G. A. Davis (1987), Recursive Estimation of Origin-Destination Matrices from Input/Output Counts, Transportation Research, Vol. 21B, No. 2.
- Noland, R. B. and K. A. Small (1995), Travel Time Uncertainty, Departure Time Choice, and the Cost of the Morning Commute, Presented at the 74th Annual TRB Meeting, Washington, D.C.

- Okutani, I. (1987), The Kalman Filtering Approach in Some Transportation and Traffic Problems, in Gartner, N. H. and N. H. M. Wilson (eds), Transportation and Traffic Theory, Elsevier, New York. Proceedings of the 10th International Symposium on Transportation and Traffic Theory.
- Organization for Economic Co-Operation and Development (OECD) (1994), Congestion Control and Demand Management, Road Transport Research, Paris.
- Ran, B., D. E. Boyce, and L. J. LeBlanc (1992), Dynamic User-Optimal Route Choice Models Based on Stochastic Route Travel Times, presented at the 2nd International Capri Seminar on Urban Traffic Networks, Capri, Italy.
- Rumbaugh, J., M. Blaha, W. Premerlani, F. Eddy, W. Lorensen (1991), Object-Oriented Modeling and Design, Prentice Hall.
- Schnabel, B. R., E. Eskow (1988), A New Modified Cholesky Factorization, Technical Report, University of Colorado.
- Small, K. (1995), Urban Transportation Economics, In the Regional and Urban Economics section of Fundamentals of Pure and Applied Economics, Arnott, R. ed., Harwood Academic Publishers.
- Sodeikat, H. (1995), Dynamic Route Guidance and Driver Information Services with Infrared Beacon Communication, Proceedings of the Second World Congress on Intelligent Transport Systems, Yokohama.
- Stroustrup, B. (1991), The C++ Programming Language, Addison Wesley.
- Sussman, J. (1992), Intelligent Vehicle Highway Systems, OR/MS Today, December 1992.
- Winston, L. W. (1994), Operations Research: Applications and Algorithms, Duxbury Press.
- Yai, T., and S. Ikawura (1994), Route Choice Modeling and Investments Effects in Metropolitan Rail Network, Working Paper, Department of Civil Engineering, Tokyo Institute of Technology.
- Yang, Q., and H. N. Koutsopoulos (1996), A Microscopic Traffic Simulator for Evaluation of Dynamic Traffic Management Systems, Transportation Research C, Vol. 4, No. 3, pp. 113-129.

5516-10