BEHAVIOR-CONSISTENT DEPLOYABLE REAL-TIME TRAFFIC ROUTING
UNDER INFORMATION PROVISION

A Dissertation
Submitted to the Faculty
of
Purdue University
by
Alexander Paz

In Partial Fulfillment of the
Requirements for the Degree
of
Doctor of Philosophy

August 2007
Purdue University
West Lafayette, Indiana
Dedicated with love and gratitude to my family, Maricela, Jairo, and Monica
Completing this dissertation represents the most challenging technical experience of my life. It has tested my mental and physical strength, and significantly contributed to the development of my methodological skills. It would not have been possible without the support from many wonderful people! First of all, I would like to thank to my family for their unconditional and sustained support, encouragement, and belief in education.

My sincere gratitude is to my advisor, Dr. Srinivas Peeta, for his invaluable advice, interest, and contribution. His lasting commitment to quality and excellence has greatly influenced my professional attitude. It has been very beneficial and challenging working with such an excellent and demanding researcher.

Special thanks to the members of my dissertation committee, Dr. Kumares C. Sinha, Dr. Fred L. Mannering, and Dr. Daniel DeLaurentis, for their participation and collaboration in this academic pursuit. I wish I had more time to continue learning from each of them. I would also like to thank Dr. Yi-Chang Chiu for his friendship, kindness, and collaboration on dynamic traffic assignment; Dr. Fred L. Mannering for his teachings and insightful suggestions on discrete choice models; Dr. Dan Simon for his guidance and collaboration on optimal control theory; Dr. William A. Crossley for his teachings on optimization and his collaboration during the early stages of my studies at Purdue; and Dr. Samuel Labi and Dr. Daniel DeLaurentis for facilitating for me invaluable support through teaching and research assistant positions.

Many thanks to my dear roommates, Guillermo Cedeño and Nils Soyke, for their precious friendship and interesting discussions in engineering, but especially about life. My experience at Purdue would not have been as productive and beneficial without their friendship and support. I know that we will always be friends. My sincere appreciation goes to my friend and classmate Salvador Hernandez for his great friendship, gentleness,
and mind-opening discussions about life; and to my friend and colleague Pengcheng Zhang for his kindness and collaboration in the early stages of my Ph.D. studies.

My profound gratitude to the Borror family, Bruce, Tami, Tracie, Andy, and their extended families Nelda, Maury, Janie, and Jerry for their friendship, support, hospitality, and teachings about America. Special thanks to my dear friend Barbara Crowe for her companionship and the happy memories together! I am also thankful to my friends Juan Carlos Ramirez and Jose Carlos Colon for their help during my initial days in the United States.

Many thanks to my friends from Economics Stefano Gubellini and Francesco Ruscitti for our discussions in mathematics, economics, and game theory. Thanks to my friends at Purdue; Weimin Zhou, Sanjeev Kanekal, Marty Aparicio, Alberto Figueroa, Rafael Perez, Natalie Villwock, Edward Smaglik, Marcela Rodriguez, Juan Diego Velasquez, Georgios Kalafatas, Abhishek Bhargava, Jung Eun Oh, Kong Siew Hwee, Anuj Sharma, and Andrew Nicols, for the happy memories at graduate school. My gratitude to the Society of Hispanic Professional Engineers, the Latino Cultural Center, and the Joint Transportation Research Program at Purdue University for sponsoring, through travel grants and scholarships, my trips to conferences.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF TABLES</td>
<td>x</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xi</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>1</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Background</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Objectives of the Dissertation</td>
<td>3</td>
</tr>
<tr>
<td>1.3 Organization of the Dissertation</td>
<td>4</td>
</tr>
<tr>
<td>2. INFORMATION-BASED NETWORK CONTROL STRATEGIES CONSISTENT WITH ESTIMATED DRIVER BEHAVIOR</td>
<td>7</td>
</tr>
<tr>
<td>2.1 Introduction</td>
<td>7</td>
</tr>
<tr>
<td>2.2 Modeling of Information Characteristics</td>
<td>10</td>
</tr>
<tr>
<td>2.2.1 Information Type</td>
<td>10</td>
</tr>
<tr>
<td>2.2.2 Information Class Modeling</td>
<td>11</td>
</tr>
<tr>
<td>2.3 Problem Description</td>
<td>12</td>
</tr>
<tr>
<td>2.4 Problem Formulation</td>
<td>15</td>
</tr>
<tr>
<td>2.4.1 Notation and Terms</td>
<td>15</td>
</tr>
<tr>
<td>2.4.1.1 Notation</td>
<td>15</td>
</tr>
<tr>
<td>2.4.1.2 Definition of Terms</td>
<td>17</td>
</tr>
<tr>
<td>2.4.2 Problem Definition</td>
<td>18</td>
</tr>
<tr>
<td>2.4.3 Formulation</td>
<td>19</td>
</tr>
<tr>
<td>2.4.3.1 Objective Function</td>
<td>21</td>
</tr>
<tr>
<td>2.4.3.2 Controller-estimated Driver Behavior Constraints</td>
<td>21</td>
</tr>
<tr>
<td>2.4.3.3 Demand Conservation Constraints</td>
<td>23</td>
</tr>
<tr>
<td>2.4.3.4 Information-based Network Control Constraints</td>
<td>23</td>
</tr>
<tr>
<td>2.4.3.5 0-1, Qualitative, and Non-negativity Variable Constraints</td>
<td>26</td>
</tr>
<tr>
<td>2.5 Problem Solution</td>
<td>26</td>
</tr>
</tbody>
</table>
2.5.1 Fuzzy Control Model ................................................................. 28
  2.5.1.1 Variables and Notation .......................................................... 29
  2.5.1.2 Input ..................................................................................... 30
  2.5.1.3 Decision Processing Component ....................................... 31
    2.5.1.3.1 Control If-then Rules .................................................. 31
    2.5.1.3.2 Membership Functions .............................................. 32
    2.5.1.3.3 Decision Process ..................................................... 33
  2.5.1.4 Output .................................................................................. 35
  2.5.2 Iterative Search Based Optimization Procedure .................. 36

2.6 Experiments .................................................................................. 38
  2.6.1 Experimental Setup ................................................................. 38
    2.6.1.1 Network Characteristics .................................................. 38
    2.6.1.2 Behavior Characteristics ................................................. 39
    2.6.1.3 Information Characteristics ............................................ 39
  2.6.2 Experiments: Prescriptive Information Only ....................... 40
    2.6.2.1 Specific Objectives and Design ....................................... 40
    2.6.2.2 Experiment Results and Analysis ..................................... 40
  2.6.3 Experiments: Descriptive Information Only ....................... 41
    2.6.3.1 Specific Objectives and Design ....................................... 41
    2.6.3.2 Experiment Results and Analysis ..................................... 42
  2.6.4 Experiments: Prescriptive, Descriptive, Prescriptive and Descriptive, and No Information ................................................................. 43
    2.6.4.1 Specific Objectives and Design ....................................... 43
    2.6.4.2 Experiment Results and Analysis ..................................... 43

2.7 Summary and Insights ................................................................. 44

3. BEHAVIOR-CONSISTENT REAL-TIME TRAFFIC ROUTING UNDER INFORMATION PROVISION ................................................................. 61
  3.1 Introduction .................................................................................. 61
  3.2 Problem Description .................................................................... 66
  3.3 Problem Formulation ................................................................. 68
    3.3.1 Notation .................................................................................. 68
    3.3.2 Problem Definition ............................................................... 70
    3.3.3 Formulation ............................................................................ 71
      3.3.3.1 Objective Function ....................................................... 74
      3.3.3.2 Controller-estimated Driver Behavior ......................... 75
      3.3.3.3 Demand Conservation Constraints .............................. 75
      3.3.3.4 Information-based Network Control Constraints .......... 76
      3.3.3.5 Flow Modeling Constraints ......................................... 77
      3.3.3.6 Flow Conservation Constraints at Nodes and Links .......... 78
      3.3.3.7 Definitional Constraints ................................................ 78
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Behavioral <em>if-then</em> rules for the controller-estimated driver behavior model</td>
<td>48</td>
</tr>
<tr>
<td>2.2 Control <em>if-then</em> rules used by the fuzzy control model to determine prescriptive and/or descriptive information</td>
<td>49</td>
</tr>
<tr>
<td>6.1 Behavioral <em>if-then</em> rules for the controller-estimated driver behavior model for prescriptive information</td>
<td>174</td>
</tr>
<tr>
<td>6.2 Calibration control <em>if-then</em> rules used</td>
<td>175</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Conceptual framework: (a) traditional DTA-based approach, (b) proposed behavior-consistent approach</td>
</tr>
<tr>
<td>2.2</td>
<td>Solution framework for the behavior-consistent real-time traffic routing problem under information provision</td>
</tr>
<tr>
<td>2.3</td>
<td>Rolling horizon framework</td>
</tr>
<tr>
<td>2.4</td>
<td>Iterative search procedure and fuzzy control model for the determination of the behavior-consistent information-based control strategies</td>
</tr>
<tr>
<td>2.5</td>
<td>Conventional and fuzzy design methodologies</td>
</tr>
<tr>
<td>2.6</td>
<td>Membership functions used by the fuzzy control model to determine prescriptive and descriptive information</td>
</tr>
<tr>
<td>2.7</td>
<td>Borman network showing the sets of driver-preferred routes (zigzag lines) and controller-desired routes (dashed lines) for a single O-D pair</td>
</tr>
<tr>
<td>2.8</td>
<td>Results for 100% prescriptive information case: (a) controller-estimated proportion of drivers taking routes, (b) proportion of drivers that must be recommended to take specific routes</td>
</tr>
<tr>
<td>2.9</td>
<td>Results for 100% descriptive information case: (a) controller-estimated proportion of drivers taking routes, (b) messages to provide to drivers</td>
</tr>
<tr>
<td>2.10</td>
<td>Results for 25% prescriptive, 25% descriptive, 25% descriptive and prescriptive, and 25% no information case: controller-estimated proportion of drivers choosing routes</td>
</tr>
<tr>
<td>2.11</td>
<td>Results for 25% prescriptive, 25% descriptive, 25% descriptive and prescriptive, and 25% no information case: (a) proportion of drivers that must be recommended to take routes, (b) messages to provide to drivers</td>
</tr>
<tr>
<td>3.1</td>
<td>Conceptual framework for the behavior-consistent traffic routing problem under information provision</td>
</tr>
<tr>
<td>3.2</td>
<td>Solution framework for the behavior-consistent traffic routing problem under information provision</td>
</tr>
<tr>
<td>3.3</td>
<td>Borman expressway corridor network</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>3.4</td>
<td>Cumulative system travel time savings under the less responsive behavior benchmarked against the no-information case (base-case)</td>
</tr>
<tr>
<td>3.5</td>
<td>Compliance rates under less responsive behavior</td>
</tr>
<tr>
<td>3.6</td>
<td>Cumulative system travel time savings under the more responsive behavior benchmarked against the no-information case (base-case)</td>
</tr>
<tr>
<td>3.7</td>
<td>Compliance rates under more responsive behavior</td>
</tr>
<tr>
<td>4.1</td>
<td>An example of default and optimized membership functions</td>
</tr>
<tr>
<td>4.2</td>
<td>Filter-based optimization of fuzzy control model parameter</td>
</tr>
<tr>
<td>4.3</td>
<td>Training progress of the filtering approach</td>
</tr>
<tr>
<td>4.4</td>
<td>Trajectory of the controller-estimated proportions of drivers choosing routes for the default and optimized fuzzy control models</td>
</tr>
<tr>
<td>4.5</td>
<td>Computational savings relative to the default fuzzy control model</td>
</tr>
<tr>
<td>4.6</td>
<td>Cumulative system travel time savings benchmarked against the no-information scenario</td>
</tr>
<tr>
<td>4.7</td>
<td>Cumulative system travel time savings benchmarked against the no-information scenario: (a) parameter B2 varies, (b) parameter B1 varies</td>
</tr>
<tr>
<td>5.1</td>
<td>Cumulative system travel time savings under the less responsive behavior benchmarked against the no-information case (base-case)</td>
</tr>
<tr>
<td>5.2</td>
<td>Cumulative system travel time savings under the more responsive behavior benchmarked against the no-information case (base-case)</td>
</tr>
<tr>
<td>5.3</td>
<td>Compliance rates for the less responsive behavior case under the standard DTA and behavior-consistent approaches</td>
</tr>
<tr>
<td>5.4</td>
<td>Percentage of driver-preferred routes matching UE and SO routes</td>
</tr>
<tr>
<td>5.5</td>
<td>Compliance rates for more and less responsive behaviors</td>
</tr>
<tr>
<td>5.6</td>
<td>Cumulative system travel time savings under the less responsive behavior relative to the base-case for the DOV paradigms</td>
</tr>
<tr>
<td>5.7</td>
<td>Cumulative system travel time savings under the more responsive behavior relative to the base-case for the DOV paradigms</td>
</tr>
<tr>
<td>5.8</td>
<td>System travel time savings under the less responsive behavior relative to the base-case versus the DOV</td>
</tr>
<tr>
<td>5.9</td>
<td>Cumulative system travel time savings under the less responsive behavior relative to the base-case for route type paradigms</td>
</tr>
<tr>
<td>5.10</td>
<td>Cumulative system travel time savings under the more responsive behavior relative to the base-case for route type paradigms</td>
</tr>
<tr>
<td>6.1</td>
<td>Conceptual framework for the behavior-consistent real-time traffic routing and calibration problem</td>
</tr>
<tr>
<td>6.2</td>
<td>Membership functions for the controller-estimated driver behavior model</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>6.3</td>
<td>Conceptual framework for the controller-estimated driver behavior model</td>
</tr>
<tr>
<td>6.4</td>
<td>Network loading for roll period of stage $\sigma$ using the controller-</td>
</tr>
<tr>
<td></td>
<td>estimated driver behavior model</td>
</tr>
<tr>
<td>6.5</td>
<td>Conceptual framework for the fuzzy on-line calibration model</td>
</tr>
<tr>
<td>6.6</td>
<td>Average percentage difference between the observed and estimated/SO traffic</td>
</tr>
<tr>
<td></td>
<td>counts for the 1st day for the BC-info-CS scenario</td>
</tr>
<tr>
<td>6.7</td>
<td>Cumulative system travel time savings benchmarked against the no-information</td>
</tr>
<tr>
<td></td>
<td>case (base-case) for the 1st day</td>
</tr>
<tr>
<td>6.8</td>
<td>Weights for behavioral if-then rule 11a for the 1st day</td>
</tr>
<tr>
<td>6.9</td>
<td>Proportion of drivers that must be recommended to take specific routes</td>
</tr>
<tr>
<td></td>
<td>for “less responsive” drivers</td>
</tr>
<tr>
<td>6.10</td>
<td>Average percentage difference between the observed and estimated/SO traffic</td>
</tr>
<tr>
<td></td>
<td>counts for the 2nd day for the BC-info-CS scenario</td>
</tr>
<tr>
<td>6.11</td>
<td>Cumulative system travel time savings benchmarked against the no-information</td>
</tr>
<tr>
<td></td>
<td>case (base-case) for the 2nd day</td>
</tr>
<tr>
<td>6.12</td>
<td>Weights for behavioral if-then rule 11a for the 2nd day</td>
</tr>
</tbody>
</table>
This research proposes a new methodological perspective to address the real-time information-based control of vehicular traffic systems problem. Existing approaches do not realistically consider the interdependencies between driver response behavior and the information provision strategies. The proposed methodology determines effective and realistically deployable information strategies by explicitly factoring the controller’s objectives and its estimation of driver response behavior in the generation of these strategies, thereby circumventing realism issues with existing dynamic traffic assignment (DTA) models that pre-specify driver behavior. This leads to a behavior-consistent deployable traffic routing approach that can enhance system performance in light of the actual driver behavior. It illustrates the need for explicit demand-supply integration. The proposed approach uses a controller-estimated driver behavior model based on aggregate behavioral if-then rules to estimate driver route choices under information provision. This circumvents the need for data at the individual driver level, and the calibration is based on measurable traffic data. The proposed methodology enables the classification of routes based on their relevance to the drivers and controller. This leads to the definition of controllable routes which provides a realistic deployment mechanism to simultaneously enhance system performance and driver compliance in a behavior-consistent manner. The concept of behavior-consistency gap is developed to illustrate the behavioral inadequacies of DTA models. Experimental insights suggest that the behavior-consistent approach provides superior performance compare to DTA
models. This is because lack of behavior-consistency can potentially deteriorate system performance as the controller may over- or under-recommend, or recommend routes that are not considered by the drivers. A key insight from this study is that there are trade-offs between the number of driver-preferred routes considered by the controller for routing and the quality of routes relative to the controller objective. That is, even when routing is performed in a behavior-consistent manner, higher compliance rates need not necessarily translate to better system performance as it also depends on the quality of the routes. From a computational standpoint, the behavior-consistent approach can be parallelized at the origin-destination level enabling real-time deployment. In addition, model parameters are optimized to further enhance the computational efficiency of the proposed solution framework.
1. INTRODUCTION

1.1 Background

Sustained increases in traffic congestion levels motivate the need for the efficient utilization of existing transportation infrastructure and resources. Integrated advances in computing, telecommunications, and informatics have enabled the development of Advanced Traveler Information Systems (ATIS) as a mechanism to alleviate traffic congestion. ATIS can enhance the efficiency of existing network capacity utilization by enabling drivers to make more informed decisions.

A critical aspect for the successful deployment of ATIS strategies is the explicit consideration of driver behavior in the determination of the information to provide to drivers. This is because driver routing decisions are a major determinant of network performance. Information strategies developed using approaches that are behaviorally restrictive and limited in their ability to incorporate drivers’ likely response behavior can result in misleading control strategies, and potentially deteriorate network performance.

Route guidance related information strategies are typically determined by solving some variant of a Dynamic Traffic Assignment (DTA) problem which achieves individual and/or system-wide objectives by assigning drivers to the associated routes to their destinations. Though methodologically elegant, most DTA models focus on robustly capturing the traffic flow dynamics and tend to be restrictive in terms of modeling driver response to information. They mostly pre-specify driver response behavior and/or assume artificial compliance rates. In reality, information provision and content can be used as control mechanisms to only influence driver behavior but cannot
imply the acceptance of the advisories or the prescribed routes, as is predominantly assumed in the DTA arena. This constitutes the primary motivation for this dissertation.

This study seeks to contribute to the effective real-time deployment of ATIS traffic routing strategies by assigning primacy to the intricate interdependencies between driver behavior and the information strategies. Beyond these independencies, the problem is characterized by spatio-temporal interactions, linguistically labeled information-related inputs/outputs, incomplete data, and computational concerns.

This research seeks to develop behavior-consistent information-based network traffic control strategies. Behavior-consistent strategies are defined here as information-based strategies that factor the likely response of drivers in their determination. This implies a fixed point problem that highlights the interdependencies between information strategies and driver behavior. Behavior-consistent strategies are more realistic in a deployment context vis-à-vis enhancing system performance compared to the solution of traditional DTA models. Behavior consistency is enabled by defining the concept of controllable routes. These are routes that address the system controller objectives while simultaneously belonging to the preferred route sets of drivers. To improve deployment effectiveness, various paradigms are defined to enhance the set of controllable routes. The behavior-consistent approach uses an iterative search based optimization procedure to push the system close to the controller-desired states. This procedure consists of a fuzzy control model that provides the step-size and move direction, and a controller-estimated driver behavior model that estimates the likely reactions of drivers, as part of an iterative framework. To enhance the computational efficiency for real-time deployment, a filtering-based optimization approach is used to optimize the fuzzy control model parameters. Finally, an on-line calibration approach is proposed to calibrate the parameters of the controller-estimated driver behavior model to enhance state consistency and increase the quality of the information strategies.
1.2 Objectives of the Dissertation

The primary objective of this dissertation is to propose a behavior-consistent approach to determine information strategies to achieve real-time operational control of large-scale congested traffic networks. The proposed behavior-consistent approach is motivated by the need to explicitly factor the drivers’ likely response behavior while determining the information strategies, and to periodically update model parameters based on actual on-line traffic flow measurements to enhance the actual effectiveness of the resulting strategies. The specific problems addressed to achieve the above objective are:

(i) Develop an iterative search based optimization procedure to address the interdependencies between the information strategies and driver behavior to determine behavior-consistent information-based network control strategies. The primary focus is to solve the fixed-point problem from the dependence between information strategies and driver behavior. Hence, a primary contribution of the dissertation is the explicit consideration of driver behavior while determining the information to provide, thereby circumventing realism issues with existing models that pre-specify driver response behavior.

(ii) Integrate, in a rolling horizon stage based deployment framework, the iterative search procedure and a DTA model to enable the determination and deployment of real-time information strategies. Therefore, a fundamental contribution of the dissertation is to enable the on-line determination of the behavior-consistent information strategies.

(iii) Develop a filtering-based optimization approach to enhance the computational performance of the proposed solution framework. This is important from the deployment perspective because the information strategies are required in sub-real-time.
(iv) Investigate deployment paradigms to enhance the practical applicability of the behavior-consistent approach. The emphasis is on the analysis of the proposed approach under various objectives, and the real-world applicability of the associated solution framework.

(v) Develop the controller-estimated driver behavior model and its associated online calibration model. The controller-estimated driver behavior model has a fuzzy multinomial logit structure where the systematic utility component is obtained using aggregate behavioral if-then rules. The calibration model minimizes the difference between estimated and actual traffic flow measurements in terms of link traffic counts by calibrating the parameters of the controller-estimated driver behavior model. A key contribution is the mechanism to calibrate behavioral parameters. Previous calibration/consistency studies have focused on traffic flow and origin-destination (O-D) demand parameters.

1.3 Organization of the Dissertation

This dissertation is divided into seven chapters. Chapter 2 develops a fuzzy control based methodology to determine behavior-consistent information-based network control strategies. The controller seeks behavior consistency by solving a fixed-point problem that estimates drivers’ likely reactions to the controller-proposed information strategies while determining them.

Chapter 3 addresses the real-time deployment of the behavior-consistent information strategies. It proposes a framework to simultaneously consider driver behavior and the controller objectives in a behavior-consistent manner. Experiments are performed to compare the effectiveness of the proposed approach with traditional DTA based approaches for deployment. The results show the importance of incorporating driver behavior realistically in the determination of the information strategies. Significant differences in terms of system travel time savings and compliance to the information strategies can be obtained when the proposed approach is compared to the traditional
approaches. The results suggest that if driver behavior is not realistically incorporated, the information strategies can potentially deteriorate system performance.

Chapter 4 presents an H-infinity filtering approach that optimizes the fuzzy control model used to determine the behavior-consistent information strategies. By adjusting the associated membership function parameters to better respond to nonlinearities and modeling errors, the approach is able to enhance the computational performance of the fuzzy control model. Computational efficiency is an important aspect in this problem context because the information strategies are required in sub-real time to be real-time deployable. Experiments are performed to evaluate the effectiveness of the approach. The results indicate that the optimized fuzzy control model can determine the behavior-consistent strategies in significantly less computational time than when the default controller is used.

Chapter 5 develops practical approaches to enhance deployment effectiveness of the behavior-consistent approach. It analyzes alternative controller desired objectives, controllable route paradigms, and the effects of augmenting the driver-preferred route set. The results suggest that the behavior-consistent approach can perform better than standard dynamic traffic assignment models while directing the system towards the desired state. They also illustrate the effectiveness of considering driver-preferred routes in developing the information strategies. Further, they suggest that driver-preferred route choice set augmentation and the associated route types can have differential impacts on performance. Also, performance is influenced by trade-offs between the number of driver-preferred routes considered by the controller for routing and the quality of routes relative to the controller objective. The results suggest that higher compliance rates may not translate to better performance and question the justification of user equilibrium solutions for route guidance on the ground that a system optimal strategy is not behaviorally sustainable or implies unfair routing recommendations.

Chapter 6 develops the controller-estimated driver behavior model and its associated on-line calibration model that enhances the consistency of the controllers’ estimation of driver behavior, and consequently contributes to the enhanced effectiveness of the behavior-consistent information-based network control strategies. The practical
deployment of the calibration model is aided by the structure of the controller-estimated driver behavior model which uses aggregate level if-then rules. This circumvents the need for data at the individual driver level, and the calibration can be based on measurable traffic data. The parameters of the controller-estimated driver behavior model are calibrated on-line using the unfolding link traffic counts. The results highlight the importance of calibrating these parameters in a deployment context.

Chapter 7 summarizes the overall insights from the research, identifies significant contributions, and discusses potential future research directions.
2. INFORMATION-BASED NETWORK CONTROL STRATEGIES CONSISTENT WITH ESTIMATED DRIVER BEHAVIOR

2.1 Introduction

Deployment effectiveness of information-based network control strategies in congested vehicular traffic networks entails the robust modeling of traffic flow dynamics and driver behavior. Existing approaches, broadly addressed under the umbrella of DTA model, primarily focus on robustly capturing traffic flow dynamics (Peeta and Ziliaskopoulos, 2001). However, their driver behavioral assumptions can be restrictive for real-time deployment (Peeta and Yu, 2006). This motivates the development of a route guidance paradigm that integrates information-based network control strategies and realistic driver behavior representation.

Driver behavior is a fundamental factor and a key source of complexity in predicting traffic network states unfolding over time. However, most DTA models are based on a rigid framework; they either pre-specify behavior of drivers and/or assume rigid compliance characteristics. Few DTA models consider heterogeneity among drivers. Even these models assume that driver behavior classes can be pre-specified. In addition, they assume *a priori* knowledge of the driver behavior class fractions in the ambient traffic stream. This rigidity raises issues related to the realistic modeling of the driver behavior and consequently, of the effectiveness of the information-based network control strategies. A detailed discussion of the behavioral limitations of DTA models is presented in Peeta and Yu, (2006).

Incorrect prediction of traffic system states based on the aforementioned assumptions can negatively impact the validity and effectiveness of the information-based network control strategies and potentially deteriorate system performance. In reality, driver route
choice decisions, even under information provision, are based on the driver’s innate behavioral tendencies, past experience, situational factors (such as time-of-day, weather conditions, and trip purpose), and the ambient traffic conditions encountered (Peeta and Yu, 2004). This is true irrespective of the type of information, the strategy used to deploy it, or whether drivers receive no information.

While information provision and content can be used as control variables to influence system performance, they cannot imply perfect or pre-specified rates of compliance by the drivers to the supplied information, as is predominantly done in the DTA arena. From the traffic controller perspective, providing personalized, generic or class-specific information based on a better understanding of driver response tendencies and ambient traffic conditions could generate a more effective control paradigm. It would determine what information to provide to whom, based on the system controller objectives and the controller’s estimation of the driver behavior.

This study is motivated by the issues raised heretofore that reflect a practical need: how do we bridge the realism gap between existing DTA models proposed for network-level deployment and the need to incorporate driver response behavior adequately while reconciling them with reasonable expectations in terms of data availability? We propose to address this through a conceptual extension of the traditional DTA-based approach, labeled behavior-consistent real-time traffic routing. Behavior-consistent information-based control strategies imply that the likely (controller-estimated) response behavior of drivers to these strategies is explicitly factored in determining them. The use of a controller-estimated driver response behavior model in this study is a formal recognition of the limitations on data availability in a deployment context. Ideally, the controller would like to have full knowledge of each driver’s behavior. But, this may not be practically possible for a variety of reasons. However, the controller can construct an estimated model of driver behavior based on historical data, field sensor data, surveys, and insights from past behavioral studies. Such a model can further be fine-tuned over time.
Existing DTA models solve for some controller objective(s) under pre-specified driver response behavior characteristics and use the resulting route assignment proportions directly as the information-based control strategies to be deployed (Bottom, 2000; Peeta and Yu, 2006). By contrast, since the behavior-consistent approach accounts for the likely driver response in determining the control strategies, the associated route assignment proportions recommended to drivers are different than those based on the specific DTA objective. For example, the DTA approach for a system optimal (SO) objective will use the SO route proportions “as is” to provide route recommendations to drivers. Under the behavior-consistent approach, the SO route proportions tend to become the controller’s goal, and the controller recommends, based on an estimated driver behavior model, more or less proportions of drivers to take specific routes so as to approach as close as possible to the SO proportions. This implies a fixed-point problem where the information-based control strategies depend on the estimated driver response and vice versa.

Figure 2.1 conceptually shows the traditional DTA-based and the proposed behavior-consistent approaches. Figure 2.1(a) indicates that the traditional DTA-based approaches use the DTA solutions directly as information-based network control strategies (for example, Peeta and Mahmassani, 1995; Lo et al., 1996; Nakayama et al., 1999). As discussed earlier, this approach has behavioral limitations in the deployment context. Figure 2.1(b) illustrates that the proposed approach uses a fuzzy control mechanism to determine the behavior-consistent information-based network control strategies based on a DTA solution and the controller’s estimation of driver behavior. This enables the controller to ensure consistency between its objectives, the information strategies, and the drivers’ likely reactions to the information provision. Hence, unlike the traditional DTA deployment strategy, the proposed approach prevents the under- or over-recommendation of routes, or the recommendation of routes that are not considered by the drivers. This is because the controller factors the drivers’ likely reactions to the information strategies while determining them. It should be noted here that the controller could use other objectives, such as the user equilibrium (UE) solution, as the desired goal instead of the SO objective within this framework.
The remainder of this chapter is organized as follows. Section 2.2 summarizes some characteristics of information strategies and uses them to define driver information classes. Section 2.3 describes the problem and Section 2.4 formulates it. Section 2.5 presents the solution concept used to determine the behavior-consistent information-based control strategies. Section 2.6 discusses experiments and analyzes their results. Section 2.7 presents some concluding comments.

2.2 Modeling of Information Characteristics

2.2.1 Information Type

From the information type perspective, information can be categorized as: (i) descriptive information where instantaneous or projected traffic conditions are provided, and (ii) prescriptive information where specific routes are recommended to the drivers, typically UE or SO routes based on instantaneous or projected travel times. In this context, information can also be categorized as: (a) quantitative information which consists of numeric information related to the network and/or route conditions such as expected travel times, and (b) qualitative information which consists of linguistic labels describing route conditions. Therefore, we could consider information-based control strategies in terms of descriptive quantitative information, descriptive qualitative information, and prescriptive information.

Most DTA models view and/or pre-specify driver behavior in terms of objectives such as UE, SO, stochastic user equilibrium (SUE), or bounded rationality (BR) under the descriptive information type, and in terms of compliance characteristics under the prescriptive information type. As discussed in Section 2.1, such a modeling approach is restrictive in depicting realistically both information characteristics and driver response behavior. In addition, these models are unable to adequately handle and process linguistic variables (Peeta and Yu, 2004). In our study, behavior is not pre-specified or restrictive, and the information-based network control strategies are modeled to be consistent with the real-world information types.
DTA models typically provide link/route travel times from a descriptive perspective or the recommended route in a prescriptive context. The study approach provides more realistic information content. That is, the models used to determine information strategies and estimate the driver’s likely response behavior enable the determination and processing of linguistic messages such as “heavy traffic ahead” under the descriptive qualitative information type, specific route recommendations under the prescriptive information type, or both simultaneously. Hence, we focus on personalized information that can be descriptive qualitative and/or prescriptive. Both these information types are simultaneously determined by the behavior-consistent approach, as discussed in Section 2.3 and later. Descriptive qualitative information implies linguistic messages describing traffic conditions downstream of the current location for the current set of routes that a driver is considering to his/her destination. Prescriptive information implies the specific route recommended to the driver.

### 2.2.2 Information Class Modeling

In our study, only drivers with suitably-equipped devices can receive personalized information; other drivers receive no information. The personalized information received by the equipped drivers is viewed as being part of an information service market which provides prescriptive information, descriptive linguistic information, or both as three different subscribed products. If a driver subscribes to prescriptive information, he/she may or may not be provided a route at various decision points by the behavior-consistent approach as discussed in Section 2.1. By contrast, a driver subscribing to descriptive linguistic information always receives it at various decision points though it is also determined by the behavior-consistent approach. In this context, the behavior-consistent approach determines whether a stronger or weaker linguistic message achieves the desired proportions. Based on the above discussion, we define four driver information classes.

The first class of drivers \((u = 1)\) subscribe to prescriptive information only. These drivers may receive specific routes at times and no information at other times during
their trip depending on the time-dependent behavior-consistent network control strategy used. In a pre-trip context, a subset of these drivers is recommended to take specific routes based on the proportions suggested by the behavior-consistent strategy. The remaining prescriptive class drivers are recommended pre-trip routes based on the controller-estimated driver behavior model.

The second class of drivers \((u = 2)\) subscribe to descriptive linguistic information only. Drivers in this class receive time-dependent linguistic information about downstream conditions for their current sets of alternative routes. The third class of drivers \((u = 3)\) subscribe to both prescriptive and descriptive linguistic information, and can process both types of information simultaneously. As in class 1, prescriptive information may or may not be provided to specific drivers depending on the behavior-consistent strategy. However, as in class 2, these drivers always receive linguistic information. The fourth class of drivers \((u = 4)\) do not receive information implying that these drivers have not subscribed to an information service.

2.3 Problem Description

The behavior-consistent information-based network control strategies problem is defined as follows. A system controller (or information service provider) seeks to determine information-based network control strategies that are consistent with driver behavior while addressing its objective of enhancing system performance. The approach used by the controller is to influence driver route choice decisions by providing routing information (both linguistic and prescriptive) in such a way that the proportions of drivers taking specific routes are close to the corresponding proportions under a system-wide objective, for example the SO or UE solution. In the thesis, the SO solution represents the primarily controller objective. However, in Chapter 5 the analysis of the proposed approach is also conducted for the UE objective. Thereby, the SO routes are defined here as the controller-desired routes, and the corresponding route assignment proportions are labeled controller-desired proportions. To achieve this consistency, the controller estimates the driver route choice decisions using an estimated driver route choice behavior model, and uses it to determine the appropriate behavior-consistent
information-based network control strategies. The methodology to obtain these strategies is the focus of this chapter. Hence, this chapter addresses a key sub-problem of the broader problem addressed in Chapter 3 where the objective is to minimize system travel time while minimizing the difference between the controller-desired and actual proportions of drivers choosing routes.

This study adopts a perspective that by directing the system, to the extent possible, towards a time-dependent objective (e.g., the SO or UE DTA state), the objective of the controller to enhance system performance can be achieved in a behaviorally more realistic manner than that under the traditional DTA approaches. It should be reiterated here that behavior-consistent route proportions that move the system closer to the SO state are provided through our approach, and not the standard SO solution route proportions obtained by solving the DTA problem itself. It is well-known in the literature that the SO solution is not behaviorally sustainable. Hence, SO routes that are not considered by the drivers are not used by the controller to determine the information strategies and therefore are not recommended to the drivers. The validity of the proposed perspective is tested in Chapter 3, where the approach is expanded to capture the network level interactions in time and space for the real-time information-based control of a vehicular traffic network. Those results illustrated the importance of incorporating driver behavior realism in the determination of the information-based network control strategies. Significant differences in terms of system travel time savings are obtained when the behavior-consistent approach is compared to the traditional approaches (UE, SO, etc.).

Figure 2.2 shows the flowchart of the proposed solution framework for the broader traffic routing problem in the context of real-world deployment. It uses a rolling horizon stage-based approach to deploy the information-based control strategies in real-time to enhance system performance for a pre-determined planning horizon. In stage number $\sigma$ of the rolling horizon, the SO DTA solution for the next stage $\sigma+1$ is generated based on the current field network conditions and the corresponding projected O-D demand. An iterative search based optimization procedure involving the controller-estimated driver behavior model and a fuzzy control model is then used to solve the fixed-point problem
described in Section 2.1, to determine the behavior-consistent information-based control strategies \((\theta, \phi)\) to provide route guidance to drivers in the roll period of the next stage. In the next stage, the controller uses these strategies to provide routes and/or descriptive information to adequately equipped drivers. The drivers use the available information and their innate behavioral tendencies to make route choice decisions. The current field network conditions resulting from the actual driver decisions and the associated traffic flow interactions are then measured through sensors to complete the loop.

The iterative search based optimization procedure, which is used to solve the behavior-consistent information strategies sub-problem of the broader problem, is the focus of this chapter. It is represented by the non-shaded box located in the middle of the flowchart in Figure 2.2. The sub-problem, addressed by the system controller, is the determination of the proportions of drivers that should be recommended specific routes and/or the set of linguistic messages describing route conditions so that when drivers make their decisions according to the controller-estimated driver behavior model, close to SO route proportions are obtained during the roll period of the next stage.

Figure 2.3 provides the details of the implementation of the rolling horizon approach. Each stage consists of \(h\) discrete time intervals of length \(\Delta\) time units. \(\tau\) denotes the departure time interval. From an implementation perspective for computational efficiency, a stage is also divided into discrete assignment intervals \(w\) in which the route assignment proportions are assumed constant within each assignment interval of the stage, though they vary across these intervals. This facilitates, without loss of generality, the formulation description and solution implementation. Further, each assignment interval consists of \(l\) time units. The first assignment interval is also the roll period of the stage. Thus, the stage length is a multiple of the roll period length. The SO DTA solution is computed for the length of the next stage resulting in different SO proportions for each assignment interval of that stage. However, the information strategies for only the next roll period are determined using the corresponding SO assignment proportions. The computation of the SO DTA solution for the entire stage captures the effects of the projected O-D demand and the network level interactions on the information strategies for the roll period. This is because the SO proportions corresponding to the roll period of
the next stage are affected by the projected conditions and/or assignments for the rest of the stage. The next section presents the formulation of the problem using this approach.

2.4 Problem Formulation

2.4.1 Notation and Terms

2.4.1.1 Notation

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>Set of nodes in the network</td>
</tr>
<tr>
<td>$A$</td>
<td>Set of links in the network</td>
</tr>
<tr>
<td>$a$</td>
<td>Subscript for a link in the network, $a \in A$</td>
</tr>
<tr>
<td>$\omega$</td>
<td>Subscript for a linguistic message</td>
</tr>
<tr>
<td>$I$</td>
<td>Set of origins in the network</td>
</tr>
<tr>
<td>$J$</td>
<td>Set of destinations in the network</td>
</tr>
<tr>
<td>$i$</td>
<td>Subscript for an origin node, $i \in I$</td>
</tr>
<tr>
<td>$j$</td>
<td>Subscript for a destination node, $j \in J$</td>
</tr>
<tr>
<td>$\rho(\sigma)$</td>
<td>Roll period of stage $\sigma$; corresponds to $\tau = (\sigma-1) \cdot l+1, \ldots, \sigma \cdot l$</td>
</tr>
<tr>
<td>$\tau$</td>
<td>Superscript for a departure time interval for the next roll period, $\tau = \sigma \cdot l+1, \ldots, \sigma \cdot l+l$</td>
</tr>
<tr>
<td>$\varphi$</td>
<td>Number of time units (in terms of $\Delta$) required to compute the SO solution and information strategies for $\rho(\sigma+1)$</td>
</tr>
<tr>
<td>$\upsilon$</td>
<td>Superscript for the time interval in which the computation of the SO solution and information strategies for the next roll period begins, $\upsilon = \sigma \cdot l - \varphi$</td>
</tr>
<tr>
<td>$K_{ij}$</td>
<td>Set of routes connecting O-D pair $ij$</td>
</tr>
<tr>
<td>$k$</td>
<td>Subscript for a route in the network, $k \in K_{ij}$</td>
</tr>
<tr>
<td>$U$</td>
<td>Set of driver classes in terms of information availability, $U \equiv {1, 2, 3, 4}$</td>
</tr>
<tr>
<td>$u$</td>
<td>Superscript for driver information class, $u \in U$</td>
</tr>
<tr>
<td>$\hat{R}_{ij}^{\tau u}$</td>
<td>Forecasted new O-D demand for the next roll period, expressed as the set of drivers of class $u$ who wish to depart from $i$ to $j$ in time interval $\tau$, $\tau = \ldots$</td>
</tr>
</tbody>
</table>
\[ \sigma \cdot l + 1, \ldots, \sigma \cdot l + l \]

Forecasted intermediate O-D demand for the next roll period, expressed as the set of drivers of class \( u \) who are forecasted to depart from \( i \) to \( j \) in time interval \( \tau \), \( \tau = \sigma \cdot l + 1, \ldots, \sigma \cdot l + l \)

\( r \)
Superscript for an individual driver in the network, \( r \in \{ \hat{R}^u, \hat{S}^u \} \)

\( P^r \hat{K}_{ij} \)
Controller-estimated set of preferred routes connecting O-D pair \( ij \) for driver \( r \), \( P^r \hat{K}_{ij} \subseteq K_{ij} \)

\( P^r \hat{K}_{ij} \)
Controller-estimated set of driver-preferred routes connecting O-D pair \( ij \),

\[ P^r \hat{K}_{ij} = \left\{ \bigcup_r P^r \hat{K}_{ij} \right\} \subseteq K_{ij} \]

\( DK_{ij}^{\rho(\sigma)} \)
Set of controller-desired (SO) routes connecting O-D pair \( ij \) in roll period \( \rho(\sigma) \) of stage \( \sigma \), \( DK_{ij}^{\rho(\sigma)} \subseteq K_{ij} \)

\( CK_{ij}^{\rho(\sigma)} \)
Set of controllable routes connecting O-D pair \( ij \) in roll period \( \rho(\sigma) \) of stage \( \sigma \),

\[ CK_{ij}^{\rho(\sigma)} = \{ DK_{ij}^{\rho(\sigma)} \cap PK_{ij} \} \]

\( \Omega_{ur} \)
Driver-information class relationship; 1 if driver \( r \) belongs to class \( u \), and 0 otherwise

\( SO_{ijk}^{\rho(\sigma)} \)
SO proportion of drivers assigned to route \( k \) in roll period \( \rho(\sigma) \) of stage \( \sigma \), \( k \in DK_{ij}^{\rho(\sigma)} \)

\( \hat{\delta}^r_{ijk} \)
Controller-estimated route choice dummy; 1 if driver \( r \) leaving from \( i \) to \( j \) in time interval \( \tau \) is estimated to take route \( k \), and 0 otherwise, \( k \in P^r \hat{K}_{ij} \)

\( \delta^r_{ijk} \)
Dummy variable for current route of driver; 1 if driver \( r \) is traveling on route \( k \) from \( i \) to \( j \) in time interval \( \nu \), and 0 otherwise, \( k \in P^r \hat{K}_{ij} \)

\( E_{ijk}^{\rho(\sigma)} \)
Controller-estimated behavior-consistent proportion of drivers taking route \( k \) in roll period \( \rho(\sigma) \) of stage \( \sigma \), \( k \in PK_{ij} \)

\( \phi_{ijk}^{\rho(\sigma)} \)
Descriptive qualitative information defined as the linguistic message describing traffic conditions for route \( k \) in roll period \( \rho(\sigma) \) of stage \( \sigma \), \( k \in CK_{ij}^{\rho(\sigma)} \)

\( \theta_{ijk}^{\rho(\sigma)} \)
Prescriptive information defined as the proportion of drivers that must be recommended to take route \( k \) in roll period \( \rho(\sigma) \) of stage \( \sigma \), \( k \in CK_{ij}^{\rho(\sigma)} \)

\( \Phi_\omega \)
Linguistic message; “Very Light Traffic” if \( \omega = 1 \), “Light Traffic” if \( \omega = 2 \), “Moderate Traffic” if \( \omega = 3 \), “Heavy Traffic” if \( \omega = 4 \), and “Very Heavy
<table>
<thead>
<tr>
<th><strong>Traffic</strong> if $\omega = 5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Phi$</td>
</tr>
<tr>
<td>$Y_{ijk}^{rr}$</td>
</tr>
<tr>
<td>$Y_{ijk}^{rv}$</td>
</tr>
<tr>
<td>$Z_{ijk}$</td>
</tr>
<tr>
<td>$\hat{X}_{ijk}$</td>
</tr>
<tr>
<td>$\hat{F}$</td>
</tr>
</tbody>
</table>

### 2.4.1.2 Definition of Terms

**Controller-Desired Routes ($DK$):** These are routes that the controller would like the drivers to choose. Depending on the controller objective, they are the time-dependent SO or UE DTA routes, which are obtained using current network conditions and projected demand by solving a deterministic DTA problem for the appropriate time duration (represented by the stage length in this dissertation).

**Driver-Preferred Routes ($PK$):** These routes are preferred by the drivers and are likely to be accepted by them. The estimation of the driver-preferred route set is a key step for any route choice model. From a technological standpoint, these route sets can be obtained in a straightforward manner for drivers with personalized information/communication devices through two-way communication. More generally, they are estimated (Bekhor et al., 2006) based on historical data collected through travel surveys and/or technologies such as two-way communication systems and global position systems.

**Controllable Routes ($CK$):** These routes belong to both controller-desired and driver-preferred route sets. In the behavior-consistent approach, they represent the set of routes...
used by the controller to influence system performance.

Behavior-Consistency Gap: The behavior-consistency gap for controllable route \( k \) connecting O-D pair \( ij \) is defined as the difference between the controller-desired proportion of drivers \( SO_{ijk}^{\rho(\sigma)} \) that should choose route \( k \) and the proportion of drivers \( \theta_{ijk}^{\rho(\sigma)} \) that must be recommended route \( k \) in order to achieve the controller-desired proportion. Hence, more/less proportions of drivers may have to be recommended controllable routes to achieve the controller-desired proportions depending on the traffic system dynamics and driver behavior.

2.4.2 Problem Definition

Consider a traffic network represented by a directed graph \( G(N,A) \) where \( N \) is the set of nodes and \( A \) the set of directed arcs. A node can represent a trip origin, a destination and/or just a junction of physical links. A network with multiple origins \( i \in I \) and destinations \( j \in J \) is considered for generality. We are given the SO solution for the next roll period, the time-dependent O-D demand forecasts for the next roll period, the forecasted intermediate O-D demand for the next roll period and its associated routes, the controller-estimated set of driver-preferred routes and their attributes, the information class of each driver, and the controller-estimated driver behavior model. We seek the behavior-consistent information-based network control strategies \( \theta_{ijk}^{\rho(\sigma+1)} \) and \( \phi_{ijk}^{\rho(\sigma+1)} \) for the next roll period that minimize the absolute difference between the SO proportions \( SO_{ijk}^{\rho(\sigma+1)} \) and the controller-estimated proportions \( E_{ijk}^{\rho(\sigma+1)} \), \( \forall \ i, j, k \in CK_{ij}^{\rho(\sigma+1)} \).
2.4.3 Formulation

This section formulates the sub-problem described in Section 2.3 and defined in Section 2.4.2. The formulation is also a representation of the non-shaded box in the Figure 2.2 flowchart.

Given:
(i) \( G(N,A) \)
(ii) \( D_{ij}^\rho(\sigma+1) \); \( \forall i, j \)
(iii) \( S_{ijk}^\rho(\sigma+1) \); \( \forall i, j, k \in D_{ij}^\rho(\sigma+1) \)
(iv) \( \hat{K}_{ij}^u \); \( \forall i, j, u, \tau = \sigma l+1, \ldots, \sigma l+l \)
(v) \( \hat{S}_{ij}^u \); \( \forall i, j, u, \tau = \sigma l+1, \ldots, \sigma l+l \)
(vi) \( P_{ij}^r \); \( \forall i, j, r \in \{ \hat{K}_{ij}^u \cup \hat{S}_{ij}^u \} \)
(vii) \( \hat{X}_{ij}^r \); \( \forall i, j, k \in P_{ij}^r, r \in \{ \hat{K}_{ij}^u \cup \hat{S}_{ij}^u \}, \tau = \sigma l+1, \ldots, \sigma l+l \)
(viii) \( Y_{ijk}^r \); \( \forall i, j, k \in P_{ij}^r, r \in \hat{S}_{ij}^u \)
(ix) \( \delta_{ijk}^r \); \( \forall i, j, k \in P_{ij}^r, r \in \hat{S}_{ij}^u \)
(x) \( \Omega_{ij}^u \); \( \forall u, r \in \{ \hat{K}_{ij}^u \cup \hat{S}_{ij}^u \} \)
(xi) \( \hat{F} \)

Objective function (controller objective):
\[
\text{Min. } \sum \sum \sum |S_{ijk}^\rho(\sigma+1) - E_{ijk}^\rho(\sigma+1)(\theta_{ijk}^\rho(\sigma+1), \phi_{ijk}^\rho(\sigma+1))| \quad (2.1)
\]

Subject to:

Controller-estimated driver behavior
\[
\delta_{ijk}^{rt} = \hat{F}(\hat{X}_{ijk}^r, Y_{ijk}^r, Z_{ijk}^r); \forall k \in P_{ij}^r \quad (2.2)
\]
\[
E_{ijk}^\rho(\sigma+1) = \frac{\sum_{\tau=\sigma l+1}^{\sigma l+l} \sum_{r \in CK_{ij}^\rho(\sigma+1)} \delta_{ijk}^{rt}}{\sum_{\tau=\sigma l+1}^{\sigma l+l} \sum_{r \in CK_{ij}^\rho(\sigma+1)} \sum_{k \in P_{ij}^r} \delta_{ijk}^{rt}}; \forall i, j, k \in CK_{ij}^\rho(\sigma+1) \quad (2.3)
\]

Demand conservation constraints
\[
\sum_{r \in S_{ij}^u} \sum_{k \in P_{ij}^r} [\delta_{ijk}^{rt} \cdot \Omega_{ij}^u] = |\hat{S}_{ij}^u|; \forall i, j, u, \tau = \sigma l+1, \ldots, \sigma l+l \quad (2.4)
\]
\[
\sum_{\tau} \sum_{\kappa \in PK_y} [\delta_{ijk} R_{ij} \Omega_{\mu}] = \left| \hat{R}_{ij} \right|; \quad \forall i, j, u, \tau = \sigma l+1, \ldots, \sigma l+l \tag{2.5}
\]

**Information-based network control constraints**

\[
\{ \theta_{ijk}^{(\rho+1)}, \phi_{ijk}^{(\rho+1)} \} = g(\theta_{ijk}^0, E_{ijk}^{(\rho+1)}(\theta_{ijk}^{(\rho+1)}, \phi_{ijk}^{(\rho+1)})); \quad \forall i, j, k \in CK_{ij}^{\rho(\rho+1)} \tag{2.6}
\]

\[
Y_{ijk}^{ur} = g_Y(\theta_{ijk}^{(\rho+1)}, \phi_{ijk}^{(\rho+1)}, r(i_{ij} R_{ij} \hat{Y}_{ijk}, \Omega_{\mu})); \quad \forall i, j, k \in PK_y, r \in \{ \hat{R}_{ij} \cup \hat{S}_{ij} \},
\tau = \sigma l+1, \ldots, \sigma l+l \tag{2.7}
\]

\[
\sum_{k \in CK_{ij}^{\rho(\rho+1)}} \theta_{ijk}^{(\rho+1)} \leq 1; \quad \forall i, j \tag{2.9}
\]

\[
\theta_{ijk}^{(\rho+1)} = 0; \quad \forall i, j, k \notin CK_{ij}^{\rho(\rho+1)} \tag{2.10}
\]

\[
Z_{ijk}^{ur} = \phi_{ijk}^{(\rho+1)} \Leftrightarrow \left\{ \sum_{u=2}^{3} \Omega_{\mu} = 1 \right\}; \quad \forall i, j, k \in CK_{ij}^{\rho(\rho+1)}, r \in \{ \hat{R}_{ij} \cup \hat{S}_{ij} \},
\tau = \sigma l+1, \ldots, \sigma l+l \tag{2.11}
\]

\[
k \in PK_y \Leftrightarrow k \in \bigcup_{r} PK_y^r \}; \quad \forall i, j \tag{2.12}
\]

\[
k \in CK_{ij}^{\rho(\rho+1)} \Leftrightarrow k \in \big\{ DK_{ij}^{\rho(\rho+1)} \cap PK_y \}; \quad \forall i, j \tag{2.13}
\]

**0-1 variable constraints**

\[
\delta_{ijk}^{ur} = 0 \text{ or } 1; \quad \forall i, j, k \in PK_y, r \in \{ \hat{R}_{ij} \cup \hat{S}_{ij} \}, \tau = \sigma l+1, \ldots, \sigma l+l \tag{2.14}
\]

\[
\delta_{ijk}^{uv} = 0 \text{ or } 1; \quad \forall i, j, k \in PK_y, r \in \hat{S}_{ij} \tag{2.15}
\]

\[
\Omega_{\mu}^{ur} = 0 \text{ or } 1; \quad \forall u, r \in \{ \hat{R}_{ij} \cup \hat{S}_{ij} \} \tag{2.16}
\]

\[
Y_{ijk}^{ur} = 0 \text{ or } 1; \quad \forall i, j, k \in PK_y, r \in \{ \hat{R}_{ij} \cup \hat{S}_{ij} \}, \tau = \sigma l+1, \ldots, \sigma l+l \tag{2.17}
\]

\[
Y_{ijk}^{uv} = 0 \text{ or } 1; \quad \forall i, j, k \in PK_y, r \in \hat{S}_{ij} \tag{2.18}
\]

**Linguistic variable constraints**

\[
\phi_{ijk}^{(\rho+1)} \in \Phi; \quad \forall i, j, k \in CK_{ij}^{\rho(\rho+1)} \tag{2.19}
\]

**Non-negativity constraints**

all quantitative variables \( \geq 0 \) \tag{2.20}

The above formulation is a non-linear mixed integer model with some stochastic \( \delta_{ijk}^{ur} \) and linguistic \( \phi_{ijk}^{(\rho+1)} \) variables. It has several contributions to the route guidance literature. A primary contribution is that the formulation explicitly estimates drivers’
likely reactions to the information-based network control strategies while determining them, thereby circumventing realism issues with existing models that pre-specify driver response behavior. Another key contribution is the concept of route classification based on the relevance of routes to the drivers and the controller. It leads to the definition of controllable routes, which provides a realistic deployment mechanism to enhance driver compliance in a behavior-consistent manner. These two aspects enable the development of the behavior-consistent approach for information-based network control. Another contribution is the simultaneous determination of prescriptive and linguistic information that are consistent with each other. Finally, the approach enables the identification of priorities to determine whom to provide information, a significant deployment issue.

The decision variables are the set of information-based network control strategies \( \theta_{ijk}^{\rho(\sigma+1)} \) and \( \phi_{ijk}^{\rho(\sigma+1)} \), \( \forall i, j, k \in CK_{ij}^{\rho(\sigma+1)} \). The formulation explicitly recognizes that the set of controller-desired routes \( DK_{ij}^{\rho(\sigma+1)} \) may differ from the set of driver-preferred routes \( PK_{ij}^{\rho(\sigma+1)} \), \( \forall i, j, r \in \{ \hat{R}_{ij}^{ur} \cup \hat{S}_{ij}^{ur} \} \), leading to the concept of controllable routes.

2.4.3.1 Objective Function

The controller objective (2.1) is to minimize the absolute difference between the SO proportions \( SO_{ijk}^{\rho(\sigma+1)} \) for the next roll period and the corresponding controller-estimated proportion of drivers taking routes, \( E_{ijk}^{\rho(\sigma+1)} \), \( \forall i, j, k \in CK_{ij}^{\rho(\sigma+1)} \). The controller achieves its objective by influencing \( E_{ijk}^{\rho(\sigma+1)} \) through information provision to approach \( SO_{ijk}^{\rho(\sigma+1)} \), \( \forall i, j, k \in CK_{ij}^{\rho(\sigma+1)} \).

2.4.3.2 Controller-estimated Driver Behavior Constraints

Function \( \hat{F} \) in Constraint (2.2) denotes the controller-estimated driver behavior model used to estimate individual driver route choices. The controller-estimated route
choice for driver $r$ (represented through dummy $\hat{\delta}_{ijk}^{rr}$) is a function of the controller-estimated route attributes $\hat{X}_{ijk}^{rr}$, the route recommendation dummy $Y_{ijk}^{rr}$, and the linguistic message $Z_{ijk}^{rr}$, $\forall i, j, k \in PK_{ij}^{'}$, $r \in \{ \hat{R}_{ij}^{ar} \cup \hat{S}_{ij}^{ar} \}$, $\tau = \sigma l + 1, ..., \sigma l + l$. Since $Y_{ijk}^{rr}$ is a function of $\phi_{ijk}^{\rho(\sigma+1)}$ and $Z_{ijk}^{rr}$ depends on $\phi_{ijk}^{\rho(\sigma+1)}$, the constraint also implies that $\phi_{ijk}^{\rho(\sigma+1)}$ and $\phi_{ijk}^{\rho(\sigma+1)}$ simultaneously influence $E_{ijk}^{\rho(\sigma+1)}$. $\hat{F}$ can denote any model structure, such as econometric, rule-based, or hybrid. Hence, the proposed approach is independent of the behavior model structure.

This study uses a hybrid multinomial logit model as part of the controller-estimated driver behavior model, where the systematic component of the utility is determined using simple behavioral if-then rules. The systematic component of the utility for a route is obtained using a fuzzy logic procedure which aggregates the contribution of each route attribute to the utility. The resulting route choice probabilities/proportions are translated into the individual route choices of drivers using Monte Carlo simulation. Hence, $\hat{F}$ represents the combination of the hybrid multinomial logit model and the Monte Carlo simulation. Table 2.1 shows the set of the behavioral if-then rules used in this study while Chapter 6 develops the model along with its on-line calibration procedure based on link traffic counts. Here, the route attributes $X$ are its expected travel time $TT$ and number of nodes $NN$.

Constraint (2.3) is a definitional constraint denoting that the controller-estimated proportion of drivers $E_{ijk}^{\rho(\sigma+1)}$ taking controllable route $k$ connecting O-D pair $ij$ during the next roll period is equal to the controller-estimated number of drivers taking this route divided by the total controller-estimated number of drivers making route choice decisions over all of their corresponding preferred routes ($k \in PK_{ij}^{'}$), for that roll period, $\forall i, j, k \in CK_{ij}^{\rho(\sigma+1)}$. 
2.4.3.3 Demand Conservation Constraints

Constraints (2.4) and (2.5) represent the conservation of the O-D demand for the next roll period. As discussed earlier, this demand is the sum of the numbers of previously assigned drivers that are still in the network (\(|\hat{S}^u_{ij}|\)) and the newly forecasted O-D demand (\(|\hat{R}^u_{ij}|\)). Constraint (2.4) indicates that the summation, over all drivers in \(\hat{S}^u_{ij}\) and the set of controller-estimated driver-preferred routes \(P\hat{K}^r_{ij}\), of the product of the controller-estimated route choice dummy \(\hat{\delta}^r_{ijk}\) and the driver-information class relationship \(\Omega^u_r\), \(\forall\ i, j, u, \tau = \sigma l + 1, \ldots, \sigma l + l\), should equal the cardinality of \(\hat{S}^u_{ij}\). Here, the product \(\hat{\delta}^r_{ijk} \cdot \Omega^u_r\) takes value 1 if driver \(r\) belongs to class \(u\) and the controller estimates that he/she takes route \(k\) in time interval \(\tau\), and 0 otherwise. Similarly, Constraint (2.5) indicates that the summation, over all drivers in \(\hat{R}^u_{ij}\) and the controller-estimated set of driver-preferred routes \(P\hat{K}^r_{ij}\), of the product of the controller-estimated route choice dummy \(\hat{\delta}^r_{ijk}\) and the driver-information class relationship \(\Omega^u_r\), \(\forall\ i, j, u, \tau = \sigma l + 1, \ldots, \sigma l + l\), should equal the cardinality of \(\hat{R}^u_{ij}\).

2.4.3.4 Information-based Network Control Constraints

Constraints (2.6)-(2.13) represent the information-based network control constraints. Constraint (2.6) has a fixed point structure and denotes that the information strategies \(\theta_{ijk}^{\rho(\sigma+1)}\) and \(\phi_{ijk}^{\rho(\sigma+1)}\) are the outcome of a procedure \(g_{\theta,\phi}\) that relates them to the SO proportions \(SO_{ijk}^{\rho(\sigma+1)}\) and the controller-estimated proportions (obtained using \(\hat{F}\)) of drivers taking routes \(E_{ijk}^{\rho(\sigma+1)}\), \(\forall\ i, j, k \in CK_{ij}^{\rho(\sigma+1)}\). Constraints (2.2), (2.3), (2.7), and (2.11) together indicate that \(E_{ijk}^{\rho(\sigma+1)}\) is a function of \(\theta_{ijk}^{\rho(\sigma+1)}\) and \(\phi_{ijk}^{\rho(\sigma+1)}\), implying the
fixed point structure of (2.6). Constraint (2.6) also indicates that $\theta_{ijk}^{\rho(\sigma+1)}$ and $\phi_{ijk}^{\rho(\sigma+1)}$ are interdependent. Hence, different combinations of $\theta_{ijk}^{\rho(\sigma+1)}$ and $\phi_{ijk}^{\rho(\sigma+1)}$ may minimize the objective function, implying the potential for multiple solutions.

In this study, the fuzzy control model in Figure 2.2 represents $g_{\theta\phi}$. An advantage of the fuzzy logic methodology in this context is that it facilitates the simultaneous determination of prescriptive $\theta_{ijk}^{\rho(\sigma+1)}$ and descriptive $\phi_{ijk}^{\rho(\sigma+1)}$ information strategies. This is because it can enable a many-to-many mapping from the SO solution, the controller-estimated driver behavior, and the information provided to the drivers, to the information strategies.

Constraint (2.7) states that the value of the dummy variable $Y_{ijk}^{rt}$ for the route recommended by the controller to driver $r$ is the result of the discretization of the aggregate proportions $\theta_{ijk}^{\rho(\sigma+1)}$ through the procedure $g_Y$. It is also dependent on $\phi_{ijk}^{\rho(\sigma+1)}$ because the recommended proportions for a route should be consistent with the linguistic message provided for it. It further depends on the past route recommendation (up to interval $\nu$) for a driver $Y_{ijk}^{\nu}$ and the route taken by that driver $\delta_{ijk}^{\nu}$ as these characteristics can be used to devise a behavior-consistent priority scheme for the future route recommendation. For example, drivers who subscribe to a premium information provision service and request information in the next roll period from the controller can receive the highest priority. Given that these drivers are requesting information, they are more likely to accept the provided controllable routes.

In the priority scheme used in this study, drivers considered to receive recommendation for route $k$ are categorized in priority subgroups based on their existing routes, prior route recommendations, and their responses to these recommendations. The first sub-group consists of drivers that were recommended to take route $k$ in the previous stage and are currently traveling on it ($Y_{ijk}^{\nu}=1$, $\delta_{ijk}^{\nu}=1$). This is because the controller seeks to prevent route switching, if possible, to enhance driver valuation of information; frequent switch recommendations may cause drivers to increasingly ignore the
recommendations as time progresses. The second priority sub-group consists of drivers that were not recommended to take route $k$ in the previous stage and are not currently traveling on it ($Y_{ijk}^{ru}=0$, $\delta_{ijk}^{ru}=0$). This is designed to attract drivers who are currently traveling on one of their preferred routes which are not controllable. The drivers of the third sub-group are those that were not recommended to take route $k$ in the previous stage but are traveling on it ($Y_{ijk}^{ru}=0$, $\delta_{ijk}^{ru}=1$). Akin to first sub-group, this is to prevent route switching if possible. Within a sub-group, the selection of drivers is performed randomly. Finally, for drivers not belonging to any of these sub-groups, this selection is randomly done.

Constraint (2.8) ensures that no more than one route is recommended to a driver, as per the strategy employed in this study. That is, $Y_{ijk}^{rt}$ can only take value 1 for at most one route in $CK_{ij}^{\rho(\sigma+1)}$, depending on whether the controller chooses to recommend a route to that driver based on the behavior-consistent approach.

Consistent with Constraint (2.8), Constraint (2.9) indicates that the total proportion of drivers receiving route recommendations cannot exceed 1. That is, the controller cannot recommend routes to more than hundred percent of the drivers.

Constraint (2.10) states that routes that do not belong to the set of controllable routes ($k \notin CK_{ij}^{\rho(\sigma+1)}$) are not recommended to drivers.

Constraint (2.11) indicates that the linguistic message $Z_{ijk}^{rt}$ for route $k$ provided to driver $r$ in time interval $t$ is equal to the descriptive information $\phi_{ijk}^{\rho(\sigma+1)}$ for route $k$ if and only if the driver has access to such information ($u = 2$ or 3). If the driver does not have access to descriptive information, $Z_{ijk}^{rt}$ is defined by the null set $\{\}$.

Constraint (2.12) states that route $k$ belongs to the controller-estimated set of driver-preferred routes $PK_{ij}$ if and only if it belongs to the set represented by the union of the preferred route sets of all individual drivers going from $i$ to $j$. Constraint (2.13) states that route $k$ belongs to the set of controllable routes $CK_{ij}^{\rho(\sigma+1)}$ if and only if it belongs to
both the controller-desired and driver-preferred route sets. Constraints (2.2), (2.3), (2.12) and (2.13) together enable the control of a system where the set of driver-preferred routes may vary over the population of drivers. That is, the set of controllable routes and the corresponding controller-estimated proportion of drivers taking routes are defined considering the entire set of driver-preferred routes.

2.4.3.5 0-1, Qualitative, and Non-negativity Variable Constraints

Constraints (2.14)-(2.18) restrict specific variables to take a value 0 or 1. Constraint (2.19) indicates that the descriptive information $\phi^{(\sigma+1)}_{ijk}$ for controllable route $k$ must belong to the set of available linguistic messages. Constraint (2.20) is the non-negativity constraint for all quantitative variables.

2.5 Problem Solution

It is difficult to solve the formulation described in Section 2.4 using traditional hard computing techniques such as non-linear optimization or traditional control theory. A key issue is their limited ability to handle the imprecision, uncertainty and subjectivity associated with incomplete data and/or qualitative/linguistic variables ($\phi$). Linguistic variables are important in this problem context because they enable the modeling of information provision strategies used in the real world; qualitative messages such as “heavy traffic ahead” or “minor delays.”

In this study, the formulation is solved using a fuzzy logic based optimization framework. Fuzzy logic allows some tolerance to imprecision, uncertainty and/or partial truth, while enabling a more tractable and computationally efficient solution mechanism (Tsoukalas and Uhrig, 1997). Computational efficiency is important in the deployment context as the control strategies are needed in sub-real time. Other advantages of using a fuzzy logic framework in this problem context include: (i) the knowledge/experience of traffic control personnel can be incorporated in the control if-then rules, and (ii) the
framework enables the simultaneous processing/determination of quantitative and qualitative traffic information.

An iterative search based optimization procedure, briefly mentioned in Section 2.3 and illustrated by the non-shaded box in Figure 2.2, is used to solve the formulation (2.1)-(2.20). It is shown in detail in Figure 2.4 and consists of the controller-estimated driver behavior model and a fuzzy control model in an iterative search process for an O-D pair. It seeks to determine the information-based control strategies that minimize the difference between the SO proportions and the corresponding controller-estimated proportions of drivers taking routes.

First, the controller-estimated driver behavior model (Figure 2.4) is used to forecast driver route choice decisions using an initial set of information-based control strategies (described in Section 2.5.2.1), and the prioritization scheme (described in Constraint (2.7)) and driver information class. If these controller-estimated proportions in relation to their corresponding SO proportions satisfy a convergence criterion (illustrated in Section 2.5.2.3), the search procedure terminates. If convergence is not yet achieved, the fuzzy control model (illustrated by the non-shaded boxes in Figure 2.4 and described in Section 2.5.1) is used to update the information-based control strategies \((\theta, \phi)\) for the next iteration so as to further reduce the difference between the SO and controller-estimated proportions. Hence, the fuzzy control model represents the update mechanism (direction-finding and step-size) for the optimization framework. The iteration counter is updated and the current information-based control strategies are used to determine the controller-estimated route proportions to close the loop. This iterative search procedure is summarized in Section 2.5.2.

Although the search procedure is conducted for all O-D pairs within a rolling horizon stage as shown in Figure 2.2, the time and O-D pair dimensions (superscript rho(\(\cdot\)) and subscripts \(ij\)) are ignored hereafter without loss of generality to simplify the notation.
2.5.1 Fuzzy Control Model

Figure 2.5 illustrates a sequence of design steps required to develop a control model using a conventional and fuzzy approach. In a conventional approach, the first step is to understand the physical system and its control requirements. The second step is to develop a model of the system under control. The third step is to use control theory in order to determine a simplified version of the control model, such as the parameters of a proportional integral derivative (PID) controller. The fourth step is to develop an algorithm for the simplified control model. The last step is to simulate the design including the effects of non-linearity, noise, and parameter variations. If the performance is not satisfactory then it is necessary to modify the system modeling, re-design the control model, re-write the algorithm, and repeat the sequence of steps (FuzzyNet Online, 2000).

In a fuzzy logic approach, the first step is to understand the physical system and its control requirements. The second step is to directly design the control model using simple logic (control if-then) rules, which describe the principles of the control model regulation in terms of the relationship between its inputs and outputs. The last step is to simulate and validate the design. If the performance is not satisfactory, only some fuzzy rules need to be modified, and the sequence of steps are repeated (FuzzyNet Online, 2000).

Although the two design methodologies are similar, the fuzzy-based methodology substantially simplifies the design loop because the entire process is governed more by observed patterns and knowledge of the problem than by a precise model of the physical system. This results in some significant benefits, such as reduced development time, simpler design, and easier calibration and update. In addition, the approach is computationally efficient because it requires less computational power and demands less operational memory than the conventional PID controller (FuzzyNet Online, 2000).

In a conventional control model such a PID controller, the physical system or process being controlled is modeled, while in fuzzy control models the aim is to incorporate expert human knowledge or observed patterns into the control algorithm.
That is, the control model becomes a model of the thinking processes (Tsoukalas and Uhrig, 1997).

The fuzzy control model consists of three components, as shown by the non-shaded boxes in Figure 2.4. The first component is the input (denoted by the dotted box). The second component represents the decision processing steps (denoted by the three solid boxes) and consists of the control if-then rules based inference step, the aggregation step, and the defuzzification step. The third component is the output (denoted by the dashed box). The model is described in detail hereafter.

2.5.1.1 Variables and Notation

Additional variables used in the iterative search based optimization procedure are:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{\eta}$</td>
<td>Number of iterations in the iterative search procedure</td>
</tr>
<tr>
<td>$\eta$</td>
<td>Superscript to denote the iteration number, $\eta = 1, \ldots, \hat{\eta}$</td>
</tr>
<tr>
<td>$RP$</td>
<td>Number of control if-then rules for prescriptive information</td>
</tr>
<tr>
<td>$RD$</td>
<td>Number of control if-then rules for descriptive information</td>
</tr>
<tr>
<td>$R$</td>
<td>Total number of control if-then rules, $R = RP + RD$</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Superscript to denote a control if-then rule, $\alpha = 1, \ldots, R$</td>
</tr>
<tr>
<td>$E_k^\eta$</td>
<td>Controller-estimated proportion of drivers taking route $k$ in iteration $\eta$, $k \in PK$</td>
</tr>
<tr>
<td>$e_k^\eta$</td>
<td>Error in iteration $\eta$ defined as the difference between $SO_k$ and $E_k^\eta$, $k \in CK$</td>
</tr>
<tr>
<td>$\Delta e_k^\eta$</td>
<td>Change in error in iteration $\eta$, defined as the difference between the current iteration error $e_k^\eta$ and the previous iteration error $e_k^{\eta-1}$, $k \in CK$</td>
</tr>
<tr>
<td>$\theta_k^\eta$</td>
<td>Proportion of drivers that must be recommended (prescriptive information) to take route $k$ in iteration $\eta$, $k \in CK$</td>
</tr>
<tr>
<td>$\phi_k^\eta$</td>
<td>Linguistic descriptive information for route $k$ in iteration $\eta$, $k \in CK$, $\phi_k^\eta \in \Phi$</td>
</tr>
<tr>
<td>$\tilde{\phi_k}^\eta$</td>
<td>Crisp value associated with descriptive information for route $k$ in iteration $\eta$, $k \in CK$</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>$\Delta \theta_k^\eta$</td>
<td>Change in the proportion of drivers that must be recommended (prescriptive information) to take route $k$ in iteration $\eta$, $k \in CK$</td>
</tr>
<tr>
<td>$\Delta \phi_k^\eta$</td>
<td>Change in the crisp value associated with descriptive information for route $k$ in iteration $\eta$, $k \in CK$</td>
</tr>
<tr>
<td>$\Delta \theta_k^{a\eta}$</td>
<td>Fuzzy outcome for change in prescriptive information for route $k$ in iteration $\eta$ obtained by fuzzyfying the inputs using rule $a$, $k \in CK$, $a = 1, \ldots, RP$</td>
</tr>
<tr>
<td>$\Delta \phi_k^{a\eta}$</td>
<td>Fuzzy outcome for change in descriptive information for route $k$ in iteration $\eta$ obtained by fuzzyfying the inputs using rule $a$, $k \in CK$, $a = RP+1, \ldots, R$</td>
</tr>
<tr>
<td>$\Delta \theta_k^{a \ast \eta}$</td>
<td>Fuzzy outcome for change in prescriptive information for route $k$ at iteration $\eta$ resulting from the aggregation of the fuzzy outcomes of all the rules ($a = 1, \ldots, RP$), $k \in CK$</td>
</tr>
<tr>
<td>$\Delta \phi_k^{a \ast \eta}$</td>
<td>Fuzzy outcome for change in descriptive information for route $k$ at iteration $\eta$ resulting from the aggregation of the fuzzy outcomes of all the rules ($a = RP+1, \ldots, R$), $k \in CK$</td>
</tr>
<tr>
<td>$\gamma^\alpha_k$</td>
<td>Degree at which rule $\alpha$ for route $k$ is activated in iteration $\eta$, $k \in CK$</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Membership function for a fuzzy set</td>
</tr>
<tr>
<td>$S(\cdot)$</td>
<td>Function to calculate the area of the membership function for a fuzzy set</td>
</tr>
<tr>
<td>$\bar{\theta}^\alpha$</td>
<td>Centroid of the fuzzy set for prescriptive information given by the consequent (right hand side) of rule $\alpha$, $\alpha = 1, \ldots, RP$</td>
</tr>
<tr>
<td>$\bar{\phi}^\alpha$</td>
<td>Centroid of the fuzzy set for descriptive information given by the consequent (right hand side) of rule $\alpha$, $\alpha = RP+1, \ldots, R$</td>
</tr>
</tbody>
</table>

### 2.5.1.2 Input

The vectors of inputs for iteration $\eta$ are defined by:

$$
e_k^\eta = SO_k - E_k^\eta \quad \text{and} \quad \Delta e_k^\eta = e_k^\eta - e_{k}^{\eta-1} \quad \forall \ k \in CK \quad (2.21)$$

They are used to determine the update ($\Delta \theta$, $\Delta \phi$) to the current solution. The role of $\Delta e_k^\eta$ is to smoothen the search process by precluding potential oscillatory behavior in the
decision variables \((\theta, \phi)\) that can occasionally arise by considering only the current error \(e^n_k\) in the update mechanism.

### 2.5.1.3 Decision Processing Component

The processing component can be summarized as follows. In the first step, the inputs are mapped to appropriate membership functions to obtain the fuzzy outcomes according to the control \textit{if-then} rules. The second step aggregates the outcomes of all fired (used) rules. In the final step, a defuzzification scheme is used to determine updates to the decision variables. Sections 2.5.1.3.1 and 2.5.1.3.2 describe the \textit{if-then} rules and the corresponding membership functions, respectively. The three steps of the decision process are described in Section 2.5.1.3.3.

#### 2.5.1.3.1 Control If-then Rules

\textit{If-then} rules are logical statements where the \textit{if} part is called the "antecedent" and the \textit{then} part is called the "consequent". They can entail multiple dimensions to enable the mapping of many inputs to many outputs. In this study, these rules are simple two-dimensional rules obtained from observed patterns and problem characteristics. For example, if the error is positive for a given route (antecedent), the number of drivers taking this route should increase implying that the route must be recommended to more drivers (consequent). Since the controller does not know the ideal level of response \((\Delta\theta, \Delta\phi)\) in relation to the magnitude of the inputs \((e, \Delta e)\), there is uncertainty on the response magnitudes based on the inputs. Hence, multiple rules are defined to account for the various input-response possibilities. The fuzzy control logic is used to identify the rules which are fired and the degree of their contribution so as to elicit the best response through the iterative search based optimization procedure. The relative magnitudes (positive small, negative large, etc.) of the inputs and outputs are handled using membership functions whose parameters can be field-calibrated through optimization (Chapter 4). However, as discussed in Section 2.5.1.3.2, these parameters need not be
calibrated as they only influence computational efficiency and not the update magnitudes ($\Delta \theta, \Delta \phi$).

An example of a control *if-then* rule is as follows:

\[
\text{if \[e \text{ is NS and } \Delta e \text{ is PL}\] then \[\Delta \theta \text{ is PS}\]}
\]

In this rule (also shown in Figure 2.4), if the error $e$ is negative small (NS) and the change in error $\Delta e$ is positive large (PL), then the information strategy $\theta$ is increased by a positive small (PS) quantity $\Delta \theta$. This outcome is aggregated along with the outcomes of all other rules to generate the crisp composite output.

The antecedents or left hand side (LHS) of the rules correspond to the inputs and the consequents or right hand side (RHS) to the outputs. The LHS and RHS are characterized by the following five fuzzy sets: “Negative Large (NL)”, “Negative Small (NS)”, “Zero (Z)”, “Positive Small (PS)” and “Positive Large (PL)” error and change in error. Thus, $e_k^\eta$, $\Delta e_k^\eta$, $\Delta \theta_k^\eta$ and $\Delta \phi_k^\eta$ $\in$ [“NL”, “NS”, “Z”, “PS”, “PL”]. Table 2.2 presents the set of control *if-then* rules used by the fuzzy control model in this study.

2.5.1.3.2 Membership Functions

The membership functions ($\mu$) are used to handle the imprecision, uncertainty and/or partial truth of inputs and their associated consequences on the outputs. Fuzzy logic uses membership functions to enable reasoning with variables that are vague in nature, such as language-based descriptors (e.g. congestion ahead, the error is NL). Corresponding to the five fuzzy sets, there are five triangular membership functions each for $e$, $\Delta e$, $\Delta \theta$, and $\Delta \phi$ as indicated in Figure 2.6; these functions are independent of $\eta$ and $k$. In addition, there are five membership functions associated with the five messages for the descriptive information $\phi$. Based on this, there are three membership functions associated with each control *if-then* rule, one each for the inputs $e_k$ and $\Delta e_k$, and one for the output (either $\Delta \theta_k$ or $\Delta \phi_k$).
The set of membership functions associated with an input/output are designed to cover the range of its domain as illustrated in Figure 2.6; in this study, their parameters evenly cover the range. For \( e \), \( \Delta e \), and \( \Delta \theta \), the corresponding domains [-1,1] have direct physical interpretations based on the values they can take. For \( \Delta \phi \), the domain is divided into five equal parts, each of which corresponds to a linguistic message. The advantage of using a triangular shape is its simplicity which aids computational efficiency as the membership function can be fully defined using only three parameters, its modal point, and its lower and upper half-widths. Chapter 4 develops an off-line H-infinity filter based approach to optimize the membership function parameters specifically to enhance on-line computational efficiency. That is, the optimized parameters provide the same solution as the default parameters but in lesser computational time.

**2.5.1.3.3 Decision Process**

For each iteration \( \eta \), the max-min composition operator and Larsen product implication operator are used for the fuzzy inference step, and the center of gravity method is used for the defuzzification step (see Tsoukalas and Uhrig, 1997). The current inputs, \( e^n_k \) and \( \Delta e^n_k \), are matched against the \( R \) control if-then rules to determine the corresponding degrees of activation. The degree at which each rule is activated is obtained using the relevant components of \( e^n_k \) and \( \Delta e^n_k \), and the max-min operator:

\[
\gamma^n_k = \max_{z \in Z} \min(\mu^\alpha_e(z), \mu^\alpha_{\Delta e}(z)) \quad \forall \, k \in CK, \alpha
\]

(2.22)

where \( Z \) represents the universe of the domains of the fuzzy sets \( e^n_k \) and \( \Delta e^n_k \). The membership functions of the fuzzy outcomes \( \Delta \theta^n_k \) and \( \Delta \phi^n_k \) for each rule are then obtained using the Larsen product implication operator as:
\[
\mu_{\Delta \theta^*_{k\eta}} = \gamma_k^{\alpha \eta} \cdot \mu_{\Delta \theta}^\alpha \\
\forall k \in CK, \alpha = 1, \ldots, RP \tag{2.23}
\]

and

\[
\mu_{\Delta \phi^*_{k\eta}} = \gamma_k^{\alpha \eta} \cdot \mu_{\Delta \phi}^\alpha \\
\forall k \in CK, \alpha = RP + 1, \ldots, R \tag{2.24}
\]

To aggregate the outcomes from all rules for each route \(k\), the following scheme (Zadeh, 1996) is used:

\[
\mu_{\Delta \theta^*_{k\eta}} = \sum_{\alpha = 1}^{RP} \mu_{\Delta \theta^*_{k\eta}}^{\alpha} \\
\forall k \in CK \tag{2.25}
\]

and

\[
\mu_{\Delta \phi^*_{k\eta}} = \sum_{\alpha = RP + 1}^{R} \mu_{\Delta \phi^*_{k\eta}}^{\alpha} \\
\forall k \in CK \tag{2.26}
\]

The center of gravity method is then used to defuzzify the fuzzy aggregate outcomes \(\Delta \theta^*_{k\eta}\) and \(\Delta \phi^*_{k\eta}\) represented by the membership functions in (2.25) and (2.26), respectively, to generate the crisp outcomes of the decision variables as follows.

\[
\Delta \theta^\eta_{k} = \sum_{\alpha = 1}^{RP} \frac{\bar{\theta}^\alpha \cdot S(\mu_{\Delta \theta^*_{k\eta}}^{\alpha})}{\sum_{\alpha = 1}^{RP} S(\mu_{\Delta \theta^*_{k\eta}}^{\alpha})} \\
\forall k \in CK \tag{2.27}
\]

and

\[
\Delta \phi^\eta_{k} = \sum_{\alpha = RP + 1}^{R} \frac{\bar{\phi}^\alpha \cdot S(\mu_{\Delta \phi^*_{k\eta}}^{\alpha})}{\sum_{\alpha = RP + 1}^{R} S(\mu_{\Delta \phi^*_{k\eta}}^{\alpha})} \\
\forall k \in CK \tag{2.28}
\]
2.5.1.4 Output

The crisp results, \( \Delta \theta_k^{\eta} \) and \( \Delta \phi_k^{\eta} \), are used to update the information-based traffic control strategies in iteration \( \eta \):

\[
\theta_k^{\eta} = \theta_k^{\eta-1} + \Delta \theta_k^{\eta} \quad \forall \; k \in CK
\]

(2.29)

and

\[
\phi_k^{\eta} = \phi_k^{\eta-1} + \Delta \phi_k^{\eta} \quad \forall \; k \in CK
\]

(2.30)

where \( \theta_k^{\eta-1} \) and \( \phi_k^{\eta-1} \) are the crisp values for the information strategies in the previous iteration \( (\eta - 1) \). Hence, \( \Delta \theta_k^{\eta} \) and \( \Delta \phi_k^{\eta} \) represent the combined search direction and step size of the iterative search based optimization procedure.

For prescriptive information, \( \theta_k^{\eta} \) is directly used as output from the fuzzy control model. However, since descriptive information is linguistic, an additional step is required to transform the continuous crisp value \( \phi_k^{\eta} \) to a discrete message:

\[
\phi_k^{\eta} = \{ \Phi_{\omega} \mid \min_\omega (|\phi_k^{\eta} - \Phi_{\omega}|) \quad \forall \; \omega = 1, \ldots, 5 \} \quad \forall \; k \in CK
\]

(2.31)

It corresponds to selecting the fuzzy set (linguistic message) with the largest mapping with \( \phi_k^{\eta} \) (degree of membership) among the possible fuzzy sets; it implies that the selected fuzzy set has the closest centroid \( (\Phi_{\omega}) \) to \( \phi_k^{\eta} \). Here, the use of the continuous variables \( \phi_k^{\eta} \) in the fuzzy control model rather than the direct use of the discrete linguistic messages is to achieve smooth convergence by reducing jumps in the objective function that can result from the use of the discrete variables. Hence, the descriptive information variable is viewed in this study as the outcome of continuous crisp values.
As indicated in the decision process, both types of information strategies are computed simultaneously and for all controllable routes as they are mutually dependent. This is necessary and adds several dimensions of complexity to the problem. Some drivers may have access to both types of information and use them to make their routes choice decisions. Therefore, the effect of one strategy influences the effect of the other on the entire set of drivers choosing routes. Further, information on a route directly affects the proportion of drivers taking that route as well as the other routes because the information results in driver switching from some routes to others. These interdependencies are illustrated through the experiment results presented in Section 2.6.

2.5.2 Iterative Search Based Optimization Procedure

First, the set of controllable routes for each O-D pair for the next roll period are determined. It is possible that no controllable routes exist for some O-D pairs, in which case no search is conducted for them. For the next roll period $\rho(\sigma + 1)$ and for an O-D pair $ij$ with controllable routes, the algorithmic steps of the iterative search based optimization procedure (represented by Figure 2.4) for the next roll period are as follows.

Step 0: initialization

Set the iteration counter, $\eta = 1$. If the set of controllable routes for the next roll period (based on $SO_{ij}^{\rho(\sigma+1)}$) are identical to those in the current roll period, initialize the information strategies $(\theta_{k}^{\eta}, \phi_{k}^{\eta})$ for them to the ones in the current roll period. If these controllable route sets are different, set $\theta_{k}^{\eta} = 0$ and $\phi_{k}^{\eta} = \Phi_{3}$, $\forall k \in CK_{ij}^{\rho(\sigma+1)}$.

Step 1: controller-estimated behavior-consistent proportions
Use the controller-estimated driver behavior model F to compute the controller-estimated behavior-consistent proportions of drivers taken routes $E_k^\eta$ based on the information-based network control strategies $\theta_k^\eta$ and $\phi_k^\eta$, $\forall k \in CK_{ij}^{\rho(\sigma+1)}$.

Step 2: convergence check

Check for convergence using (2.32). $\chi$ is the number of iterations used for averaging to check for convergence. First, compute the difference between the $SO_{ijk}^{\rho(\sigma+1)}$ and $E_k^\eta$ to generate $e_k^\eta$. Then, use it along with $\bar{e}_k^\eta$, the average value of the error over the last $\chi$ iterations for route $k$ in iteration $\eta$, and the errors for the last $\chi$ iterations ($e_m^k$ is the error in iteration $m$ for route $k$), to determine the value to compare with $\sigma$, a pre-specified small constant indicating the required accuracy. If the number of iterations at convergence is less than $\chi$, the corresponding number of iterations is used for the averaging.

$$\sqrt{\frac{1}{\chi} \left( \sum_{m=\eta-\chi+1}^{\eta} (e_m^k - \bar{e}_k^\eta) \right)^2} < \sigma \quad \forall k \in CK_{ij}^{\rho(\sigma+1)}$$  \hspace{1cm} (2.32)

Terminate the iterative search procedure if the inequality in (2.32) is satisfied for all controllable routes for O-D pair $ij$. Otherwise, go to Step 3.

Step 3: update the information strategies

Use the fuzzy control model to update the information strategies $\theta_k^\eta$ and $\phi_k^\eta$ based on the SO proportions $SO_{ijk}^{\rho(\sigma+1)}$ and the corresponding controller-estimated behavior-consistent proportions $E_k^\eta$, $\forall k \in CK_{ij}^{\rho(\sigma+1)}$. Update the iteration counter, $\eta = \eta + 1$, and go to Step 1.
2.6 Experiments

Experiments are designed to evaluate the performance of the fuzzy control model and illustrate the significance of behaviorally-consistent approaches to determine information-based network control strategies. Three sets of experiments are conducted to evaluate the performance of the fuzzy control model under various driver classes (in terms of information type, information access, and their level of responsiveness to information).

2.6.1 Experimental Setup

2.6.1.1 Network Characteristics

The Borman expressway corridor network shown in Figure 2.7 is used to conduct the experiments. Located in northwest Indiana, it consists of a sixteen-mile section of interstate I-80/94 (called the Borman expressway), I-90 toll freeway, I-65, and the surrounding arterials and streets. It has 197 nodes, 460 links, and 43 zones (with centroids that represent origins/destinations). The Borman expressway is a highly congested facility with a large number of truck traffic. An advanced traffic management system has been installed on the network to provide drivers with real-time traffic information, especially during incidents. A potential alternative to divert traffic is the Indiana toll road I-90, which operates parallel to the Borman expressway. Depending on the destination, other potential major alternative routes also exist.

While the proposed methodology can be used to determine the information strategies for multiple O-D pairs where different drivers have different sets of preferred routes, a single O-D pair and a single set of driver-preferred routes (all driver have the same set of preferred routes) are used here to illustrate the key methodological insights associated with the behavior-consistent approach. As shown in Figure 2.7, there are four driver-preferred routes (zigzag lines) and four controller-desired routes (dashed lines) connecting the selected O-D pair, but only three of them fully overlap. The controller
seeks to achieve the SO proportions only on the set of controllable routes (the three
routes that fully overlap). Thus, controller-desired routes 1, 2 and 3 are defined as the
controllable routes in these experiments. The SO proportions are 49%, 26% and 10% for
routes 1, 2 and 3, respectively.

2.6.1.2 Behavior Characteristics

As shown in Table 2.1, two types of drivers are considered based on their level of
responsiveness to information. The first type of drivers, categorized as “less responsive”
to information strategies, are drivers that are slightly influenced by the information
provided. To make route choice decisions, these drivers rely more on past experience
and behavioral tendencies than on information. The second type of drivers, labeled as
“more responsive” to information strategies, are more influenced by information
compared to the “less responsive” drivers. Drivers that are not influenced at all by the
information are viewed here as drivers without information.

Note that the actual driver behavior may be different. However, as discussed earlier
and illustrated in Figure 2.2, the actual driver behavior is addressed only in the overall
framework of Figure 2.2, and not in the sub-problem addressed in this chapter. Chapter 3
analyzes the performance of the overall framework.

2.6.1.3 Information Characteristics

In terms of drivers’ access to prescriptive and/or descriptive information, four driver
classes are considered as discussed in Section 2.2. For the experiments involving
descriptive information, $\omega = 1, \ldots, 5$ is used to represent the linguistic messages defined
in Section 2.4.1.1.

The experiments are conducted for only the first stage of the rolling horizon; this is
based on the objectives of this chapter of investigating the effectiveness of the behavior-
consistent approach rather than a network-level analysis. In the figures illustrating the
results, which correspond to the first iteration of the first stage, the points on the y-axis
are based on the initial information-based control strategies ($\theta^0_k = 0$ and $\phi^0_k = \Phi_3$, $\forall k \in CK_{ij}^{\rho(\sigma+1)}$).

2.6.2 Experiments: Prescriptive Information Only

2.6.2.1 Specific Objectives and Design

The objective of these experiments is to evaluate the ability of the fuzzy control model to generate effective prescriptive information strategies under the two classes of driver responsiveness to information. To illustrate insights, it is assumed that all drivers have access to prescriptive information, but only a subset of them receive route recommendations depending on the behavior-consistent strategy used (and the priority scheme discussed in Constraint (2.7)). The remaining drivers do not receive information, and hence, their route choice decisions are assumed to be without the influence of information for the roll period of that stage. The decision variable is the vector $\theta$ that represents the proportions of drivers that must be recommended to take specific routes.

2.6.2.2 Experiment Results and Analysis

Figure 2.8 presents the results of these experiments. Figure 2.8(a) shows the controller-estimated proportion (fraction) of drivers taking routes in each iteration of the search procedure under the currently calculated vector of information strategies. It can be noticed that the controller can achieve close to the SO proportions (shown by the three horizontal lines in the figure for the three routes) through its information provision strategies. However, it achieves a faster convergence when all drivers are more responsive to the information strategies. This is because when drivers are more likely to make route choice decisions consistent with the recommendation, the controller can achieve its objective with fewer recommendations. Figure 2.8(b) shows the proportion of drivers that must be recommended to take each route in order to achieve the desired (SO)
proportions. The values of the information strategies at convergence indicate that more recommendations are required for one of the three routes under the more responsive behavior scenario when compared to the less responsive behavior scenario. This may seem counterintuitive since it is expected that fewer recommendations are necessary to achieve the desired proportions under more responsive behavior. However, note that under this type of behavior (more responsive), the iterative search procedure achieves its objective in fewer (about 5) iterations. After 5 iterations in Figure 2.8(a), the estimated proportions are almost constant, but the recommended proportions in Figure 2.8(b) still have substantial variability. This implies the existence of multiple solutions, due to the interdependencies discussed in Section 2.5.1.4. For example, in Figure 2.8(b), in the neighborhood of 6 iterations (when the desired proportions are achieved for the more responsive case), the controller still needs to provide less information under the more responsive case compared to the less responsive case for route 1, which is the route that requires more recommendations around iteration 50 for the more responsive behavior.

The results from Figure 2.8(b) indicate that there are significant behavior-consistency gaps in all cases. That is, there are significant differences between the controller-desired proportion of drivers choosing routes and the proportions of drivers that must be recommended to take the routes in order to achieve the desired proportions. Some of the behavior-consistency gaps are negative, while others are positive. Hence, the experiment results highlight the importance of using a behavior-consistent approach to determine the information-based network control strategies to achieve the controller-desired proportions.

2.6.3 Experiments: Descriptive Information Only

2.6.3.1 Specific Objectives and Design

The objective of these experiments is to evaluate the ability of the fuzzy control model to generate effective descriptive information under the two classes of driver responsiveness to information. Here, all drivers receive descriptive information only.
The decision variable here is the vector $\phi$ that represents linguistic labels describing route conditions.

2.6.3.2 Experiment Results and Analysis

Figure 2.9 shows the experiment results. Figure 2.9(a) shows the controller-estimated proportion of drivers choosing routes for each iteration of the search procedure under the currently calculated vector of information strategies. As indicated in this figure, the controller can achieve close to the desired proportions. The model achieves a faster rate of convergence and values slightly closer to the desired proportions when all drivers are less responsive. This is because when all drivers receive information, the change from one message to another produces a larger discrete effect in the proportion of drivers choosing routes under the more responsive case. Therefore, typically the controller has reduced ability to get closer to the desired proportion due to the large effects of information provision under more responsive behavior.

Figure 2.9(b) shows the vector of information values $\phi$, the set of messages that the controller provides to the drivers. As illustrated, the procedure converges to a stable set of messages. The messages at convergence indicate that stronger messages are required under less responsive behavior compared to those under the more responsive case. This result is intuitive because stronger messages are needed to compensate the fact that drivers are less influenced by the messages in the less responsive behavior case.

It is not possible to define behavior-consistency gaps for linguistic information because each message represents an unknown proportion of drivers choosing routes. This is another important reason to use a behavior-consistent approach to determine the information-based network control strategies. Traditional approaches cannot incorporate the linguistic nature of information strategies.
2.6.4 Experiments: Prescriptive, Descriptive, Prescriptive and Descriptive, and No Information

2.6.4.1 Specific Objectives and Design

The objective of these experiments is to evaluate the ability of the fuzzy control model to simultaneously generate effective prescriptive and descriptive information under the two classes of driver responsiveness to information. In these experiments, 25% of the drivers can only access prescriptive information; 25% of the drivers only receive descriptive information; 25% of the drivers have access to prescriptive information and receive descriptive information; and the remaining 25% of the drivers cannot access prescriptive information and do not receive descriptive information. Hence, the information-based traffic control strategies here are the vector $\phi$ of messages describing routes conditions and the vector $\theta$ of proportions of drivers that must be recommended to take routes.

2.6.4.2 Experiment Results and Analysis

Figure 2.10 shows the controller-estimated proportion of drivers taking routes for each iteration of the search procedure under the currently calculated vectors of information strategies. For both levels of responsiveness, the controller achieves close to the desired proportions. However, it achieves a smoother convergence when all drivers are less responsive to the information strategies. This is because the linguistic messages have a weaker switching effect for these drivers, reducing jumps in the objective function.

Figure 2.11(a) shows the results of these experiments for the prescriptive vector of information strategies $\theta$, the proportion of drivers that must be recommended to take specific routes. For both levels of responsiveness, the procedure converges to a relative stable set of values. The vector of prescriptive information strategies at convergence indicates that more recommendations are required for two of the three routes under the
more responsive behavior case when compared to the less responsive case. The reasons for this are the same as in the first set of experiments because as shown in Figure 2.10, the estimated vector of drivers taking routes reaches the SO proportions in the early iterations of the search procedure with fewer recommendations than the ones around iteration 50 for the more responsive behavior.

Figure 2.11(b) shows the results of these experiments for the descriptive vector of information strategies $\phi$, the set of messages that the controller provides to the drivers. In both cases, the procedure converges to a stable set of messages, and the set of messages is almost identical. However, the trajectories to achieve the final messages are different; a stronger message is required for route 2 under the less responsive behavior.

The information strategies are the outcome of complex processes resulting from the mutual dependency of prescriptive and descriptive information, as well as the presence of multiple driver classes in terms of information accessibility. These experiments highlight the complexity of the problem faced by the controller and show the effectiveness and robustness of the fuzzy control modeling approach to address the multidimensionality and nonlinearity of the problem.

2.7 Summary and Insights

This study is the first in the literature to propose a methodology to determine behavior-consistent information-based network control strategies, by factoring the controller’s estimation of driver route choice behavior in generating these strategies. It proposes the concept of behavior-consistency gap to illustrate the need for such strategies and to highlight the behavioral inadequacies of existing DTA modeling approaches and the consequent deployment paradigms. Existing deployment mechanisms are primarily categorized as reactive (Hawas and Mahmassani, 1997; Pavlis and Papageorgiou, 1999) or anticipatory (Peeta and Mahmassani, 1995; Peeta and Zhou, 2002). While reactive approaches do not suffice for capturing the dynamics of network spatio-temporal interactions, existing anticipatory mechanisms mostly focus on the effects of high-fidelity traffic flow dynamics combined with rudimentary behavior
dynamics (see Peeta and Yu, 2004, 2006). The proposed behavior-consistent approach represents an anticipatory mechanism that is robust in terms of both the traffic flow and behavioral aspects. The study also proposes the concept of controllable routes to formally incorporate driver route consideration behavioral preferences under information provision. Further, the controllable routes approach circumvents a key deployment concern expressed for traditional DTA models, the possibility that drivers are “lied to” by the traffic control center (due to its system-level objectives) and provided sub-optimal routes thereby affecting their level of trust and credibility in relation to the provided information.

The use of a fuzzy logic methodology based on simple if-then rules has key implications for modeling realism, deployment convenience, and computational efficiency. It simplifies the controller design and results in a computational efficient approach which is a desirable characteristic for real-time operations. The adequacy of aggregate level generic if-then rules based on system observation and problem characteristics for both the controller-estimated behavior modeling and the generation of information strategies circumvents many data needs that would otherwise be required at the level of the individual driver. For the generation of the information strategies, a synergistic advantage is that the calibration of the associated membership function parameters is not required for solution accuracy; such calibration only affects computational efficiency in terms of convergence rate. This characteristic is confirmed in Chapter 4 where an off-line H-infinity filtering methodology optimizes the membership function parameters of the fuzzy control model leading to significant computational savings. The fuzzy control model also enables the simultaneous consideration of quantitative and qualitative variables, an important characteristic of information-based route guidance. Peeta and Yu (2004) show that such a fuzzy logic based framework can also capture information-related behavioral phenomena over multiple timescales in a unified manner.

The study results highlight the complexity of the problem faced by the controller and show the effectiveness of the fuzzy control modeling approach to address the multidimensionality and nonlinearity of the problem. They also indicate the importance
of using a behavior-consistent approach to determine the information-based control strategies. The iterative search procedure was found to converge always to a stable solution in terms of the proportions of drivers that must be recommended to choose routes and/or the linguistic message to provide. A detailed analysis of the experiment results suggests that many driver-preferred routes tend to have large behavior-consistency gaps because large numbers of drivers take these routes independent of information provision. This implies that to direct the system towards desired proportions of drivers choosing routes, the controller may have to recommend more or less drivers to take some routes depending on the network dynamics and driver behavior tendencies. That is, the effects of driver response behavior to information provision may require more meaningful strategies than those provided under the traditional DTA models to have a reliable estimate/control of system performance. The direct use of the solutions from traditional DTA models (proportions of drivers to be assigned to various routes) may not result in the controller-desired solutions due to the behavior-consistency gap, and can possibly worsen conditions compared to the “no information” scenario.

The problem addressed in this chapter is a conceptual sub-problem of the broader traffic routing problem that seeks to minimize system travel time in congested traffic networks. Chapter 3 illustrates the effectiveness of the proposed behavior-consistent approach in a rolling horizon based deployment context that captures the network-level interactions in terms of traffic flow and driver behavior. They suggest that behavior-consistent information-based control strategies are superior and entail greater compliance compared to standard DTA-based UE or SO strategies. The current study requires controllable routes to have a full overlap between the controller-desired and driver-preferred routes. Chapter 5 proposes alternative paradigms to relax this requirement to provide more flexibility in developing practical information-based control strategies. It also explore insights on directing the system towards the UE state rather than the SO state as UE routes are more likely to overlap with driver-preferred routes.

The behavior-consistent framework focuses on personalized information. As generic information is a popular information dissemination mechanism, it would be useful to
extend the framework to disseminate multiple types of information. An advantage of the proposed fuzzy logic based methodology is its amenability to incorporating personalized and generic information simultaneously.
Table 2.1 Behavioral *if-then* rules for the controller-estimated driver behavior model

<table>
<thead>
<tr>
<th>Category</th>
<th>Rule #</th>
<th>LHS</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controller-estimated driver-expected travel time <em>(TT)</em></td>
<td>1</td>
<td>If TT is Very Low (VL)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>If TT is Low (L)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>If TT is Medium (M)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>If TT is High (H)</td>
<td>Driver will probably not choose the alternative (PN)</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>If TT is Very High (VH)</td>
<td>Driver will not choose the alternative (N)</td>
</tr>
<tr>
<td>Route complexity <em>(NN)</em></td>
<td>6</td>
<td>If NN is Very Low (VL)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>If NN is Low (L)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>If NN is Medium (M)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>If NN is High (H)</td>
<td>Driver will probably not choose the alternative (PN)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>If NN is Very High (VH)</td>
<td>Driver will not choose the alternative (N)</td>
</tr>
<tr>
<td>Prescriptive information <em>(Y)</em> for more responsive drivers</td>
<td>11a</td>
<td>If Y is Route is Recommended (RR)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>12a</td>
<td>If Y is Route Was Recommended (RWR)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>13a</td>
<td>If Y is Route is Not Recommended (RNR)</td>
<td>Driver will not choose the alternative (N)</td>
</tr>
<tr>
<td>Prescriptive information <em>(Y)</em> for less responsive drivers</td>
<td>11b</td>
<td>If Y is Route is Recommended (RR)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>12b</td>
<td>If Y is Route Was Recommended (RWR)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>13b</td>
<td>If Y is Route is Not Recommended (RNR)</td>
<td>Driver will not choose the alternative (N)</td>
</tr>
<tr>
<td>Descriptive information <em>(Z)</em> For more responsive drivers</td>
<td>14a</td>
<td>If Z is “Very Light Traffic” (VLT)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>15a</td>
<td>If Z is “Light Traffic” (LT)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>16a</td>
<td>If Z is “Moderate Traffic” (MT)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>17a</td>
<td>If Z is “Heavy Traffic” (HT)</td>
<td>Driver will probably not choose the alternative (PN)</td>
</tr>
<tr>
<td></td>
<td>18a</td>
<td>If Z is “Very Heavy Traffic” (VHT)</td>
<td>Driver will not choose the alternative (N)</td>
</tr>
<tr>
<td>Descriptive information <em>(Z)</em> for less responsive drivers</td>
<td>14b</td>
<td>If Z is “Very Light Traffic” (VLT)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>15b</td>
<td>If Z is “Light Traffic” (LT)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>16b</td>
<td>If Z is “Moderate Traffic” (MT)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>17b</td>
<td>If Z is “Heavy Traffic” (HT)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>18b</td>
<td>If Z is “Very Heavy Traffic” (VHT)</td>
<td>Driver will probably not choose the alternative (PN)</td>
</tr>
</tbody>
</table>
Table 2.2 Control *if-then* rules used by the fuzzy control model to determine prescriptive and/or descriptive information

<table>
<thead>
<tr>
<th>Change in Error (Δe)</th>
<th>NL</th>
<th>NS</th>
<th>ZR</th>
<th>PS</th>
<th>PL</th>
</tr>
</thead>
<tbody>
<tr>
<td>NL</td>
<td>NL</td>
<td>NL</td>
<td>NS</td>
<td>NS</td>
<td>NS</td>
</tr>
<tr>
<td>NS</td>
<td>NL</td>
<td>NS</td>
<td>ZR</td>
<td>ZR</td>
<td>ZR</td>
</tr>
<tr>
<td>ZR</td>
<td>NL</td>
<td>NS</td>
<td>ZR</td>
<td>PS</td>
<td>PL</td>
</tr>
<tr>
<td>PS</td>
<td>ZR</td>
<td>ZR</td>
<td>ZR</td>
<td>PS</td>
<td>PL</td>
</tr>
<tr>
<td>PL</td>
<td>PS</td>
<td>PS</td>
<td>PS</td>
<td>PL</td>
<td>PL</td>
</tr>
</tbody>
</table>

where:
- NL = Negative large
- NS = Negative small
- ZR = Zero
- PS = Positive small
- PL = Positive large
Figure 2.1 Conceptual framework: (a) traditional DTA-based approach, (b) proposed behavior-consistent approach
Determine information-based control strategies for roll period of stage $\sigma + 1$

SO DTA Solution
Controller-desired routes and proportions for stage $\sigma + 1$

Projected time-dependent O-D demand for stage $\sigma + 1$

Controller-estimated network characteristics
Controller-estimated set of driver-preferred routes
Controller-estimated vector of route characteristics

Controller-estimated driver behavior model
Controller-estimated behavior-consistent proportions ($E$)

Fuzzy control model
Information-based network control strategies ($\theta, \phi$)

Update $e, \Delta e$

Convergence ?
Yes
Stage $\sigma = \sigma + 1$

Controller Information provision for roll period of stage $\sigma$

Drivers Driver discrete route choices during roll period of stage $\sigma$

Rolling horizon approach
Stage $\sigma$
Roll period
Stage $\sigma + 1$
Computation Roll period

End of planning horizon ?
Yes
End

Figure 2.2 Solution framework for the behavior-consistent real-time traffic routing problem under information provision
Estimates of O-D demand from $\tau = (\sigma - 1)l + 1$ to $\tau = \sigma l + h$ required in time interval $\nu = \sigma l - \phi$, so that the information strategies can be computed before the start of stage $\sigma + 1$.

Figure 2.3  Rolling horizon framework
Figure 2.4  Iterative search procedure and fuzzy control model for the determination of the behavior-consistent information-based control strategies
Conventional design methodologies

Start

Understand physical system and control requirements

Develop a model of system under control which includes its sensors and actuators

Determine a simplified controller from control theory

Develop an algorithm for the controller

Simulate, debug and implement the design

Is the performance satisfactory?

No

Yes

Deploy

Fuzzy-based design methodologies

Start

Understand physical system and control requirements

Design the controller using fuzzy rules

Simulate, debug and implement the design

Is the performance satisfactory?

No

Yes

Deploy

Figure 2.5 Conventional and fuzzy design methodologies
Figure 2.6 Membership functions used by the fuzzy control model to determine prescriptive and descriptive information
Figure 2.7  Borman network showing the sets of driver-preferred routes (zigzag lines) and controller-desired routes (dashed lines) for a single O-D pair
Figure 2.8  Results for 100% prescriptive information case: (a) controller-estimated proportion of drivers taking routes, (b) proportion of drivers that must be recommended to take specific routes
Figure 2.9  Results for 100% descriptive information case: (a) controller-estimated proportion of drivers taking routes, (b) messages to provide to drivers
Figure 2.10 Results for 25% prescriptive, 25% descriptive, 25% descriptive and prescriptive, and 25% no information case: controller-estimated proportion of drivers choosing routes
Figure 2.11 Results for 25% prescriptive, 25% descriptive, 25% descriptive and prescriptive, and 25% no information case: (a) proportion of drivers that must be recommended to take routes, (b) messages to provide to drivers
3. BEHAVIOR-CONSISTENT REAL-TIME TRAFFIC ROUTING UNDER INFORMATION PROVISION

3.1 Introduction

The state-of-the-art uses DTA models to enhance in real-time the performance of vehicular traffic systems. These models predict the time-dependent network states and determine the corresponding information-based network control strategies. However, the current DTA literature does not factor driver behavior realistically or adequately in the determination of these strategies. Thereby, existing models can achieve system-wide objectives for assumed, though not necessarily realistic, scenarios that require the prespecification of driver response behavior to the information strategies (Peeta and Yu, 2006). This implies a single level decision-making structure where the system controller determines: (i) in a prescriptive context, the vehicular route that each driver must take with the assumption that the driver follows this route, (ii) in a descriptive context, the estimated choice of the driver, or (iii) in a formal but inadequate consideration of the effect of behavior, the set of routing alternatives from which the driver chooses a route.

Most existing literature addresses the determination of deployable information-based network control strategies by focusing on generating consistent anticipatory route guidance and/or employing online consistency checking techniques. Here, “consistent” implies that traffic conditions used to generate the guidance must be similar to the realized conditions once the guidance is deployed. Doan et al. (1999) and Peeta and Bulusu (1999) identify various sources that can cause inconsistencies between the predicted and realized conditions. Peeta and Bulusu (1999) propose a generalized singular value decomposition approach to enable consistency between the predicted and realized network states in terms of link traffic counts and the number of users on various
Ben-Akiva et al. (2001) suggest that consistent guidance entails the formulation and solving of a fixed point problem. Bottom (2000) proposes a conceptual framework for the analysis of the consistent route guidance problem. It identifies the principal elements of the problem and their key relationships, and proposes some solution methods to the problem. It explicitly recognizes the need to estimate how drivers will react to the information provided to them. The solution methods are based on the idea of solving a fixed point problem formed by three alternative composite maps: (i) route assignment fractions to network states, (ii) network states to guidance messages, and (iii) guidance messages to route assignment fractions. He indicates that the solution to this fixed point problem leads to consistency. However, the problem is computationally intensive for real-world networks and this may preclude the deployment of the information in a timely manner. In addition, the framework focuses on the algorithmic and computational aspects of the problem while driver behavior is still represented using a traditional DTA approach. To alleviate the computational intensity of Bottom’s framework (2000), Crittin and Bierlaire (2001) propose a heuristic method based on an approximated objective function. Zhou and Mahmassani (2005) address origin-destination demand consistency checking in conjunction with addressing network state consistency.

While the online consistency problem recognizes the need for an estimation of driver behavior, existing formulations and solution frameworks do not explicitly and/or realistically estimate and represent driver behavior while determining the information strategies. In reality, information provision and content can be used as control mechanisms to only influence driver behavior but cannot imply perfect or pre-specified partial compliance rates, as is predominantly assumed in the DTA arena. This is because drivers make route choice decisions based on several factors related to behavior, information, and traffic conditions, and the information provided by the controller is only one aspect. It implies the need for a bi-level framework that captures the interactions between the controller objectives and driver decisions. Thereby, there is the need for the controller to factor the drivers’ likely response to information-based control strategies while determining these strategies, suggesting a fixed-point problem structure.
That is, deployable information strategies need to be more carefully constructed and are not as straightforward as suggested by the standard DTA deployment approaches. Also, even after such strategies, labeled behavior-consistent in this study, are determined and disseminated by the controller, there is no guarantee that the recommended route will be taken. However, as will be illustrated in Section 3.1.2, the likelihood of compliance increases as only routes that belong to the set of preferred routes of the individual drivers are recommended. Hence, this paper focuses on behavior-consistent strategies which are more acceptable to drivers and simultaneously are more likely to meet controller objectives.

The various limitations of DTA models to realistically represent driver route choices under information provision have motivated the development of new paradigms aimed at bridging functional gaps between driver behavior models and DTA models in terms of predicting the time-dependent network traffic flow patterns. In this context, Peeta and Yu (2006) develop a behavior-based consistency-seeking (BBCS) modeling approach. The approach uses a hybrid probabilistic-possibilistic behavior model to consistently address day-to-day learning and within-day dynamics of driver behavior in a single framework. It avoids assumptions of a priori knowledge of driver behavior class fractions as it is able to determine them in real-time based on link volumes. The BBCS models can be used to develop alternatives to DTA models to deploy information-based control strategies that are more realistic.

While the BBCS approach has modeling richness in the context of driver response behavior to information, the role of the controller is limited to the consistency-seeking process whereby the parameters of the driver behavior model are updated across multiple timescales based on the day-to-day and within-day experiences of the drivers. The logical next step is to involve the controller to develop behavior-consistent information-based network control strategies. It addresses the question: what information-based network control strategies should the controller adopt so that the drivers behaving the way they do also enable the controller to achieve its system-wide objectives? As a first step to addressing this question, Chapter 2 develops a fuzzy logic based approach to determine information-based strategies that are consistent with the
controller-estimated response behavior of drivers to the information provided. It entails solving a fixed-point problem where these strategies depend on the controller-estimated driver response behavior and vice versa. In this chapter, we close the loop by extending the framework to enable real-time deployment, and analyze the effectiveness of behavior-consistent information-based control strategies at the network level using a rolling horizon stage based approach where the actual driver behavior model is independent of the controller-estimated behavior model. The study also illustrates the limitations of standard DTA-based strategies.

The explicit consideration of driver behavior leads to a new dimension of complexity in predicting traffic states which is further complicated by the need to adequately capture the traffic flow dynamics that represent the network-level spatio-temporal interactions of driver route choice decisions. Typical DTA objectives (such as UE and/or SO), inherited from static traffic assignment concepts, have a single-level optimization structure which is at the controller level. However, in an operational context, using UE as a behavioral paradigm, or UE and SO as the information-based network control strategies with partial or perfect compliance rates, is an inherently restrictive approach from behavioral and information-related standpoints. Peeta and Yu (2004, 2006) illustrate the limitations of these strategies and the need to capture information-related driver learning processes and consider situational factors. Further, from an information standpoint, these strategies make strong assumptions on drivers’ real-time knowledge about dynamic network conditions and their abilities to process the information provided by the controller in real-time, both of which can significantly influence driver route choice decisions.

This chapter addresses the bi-level interactive decision-making process where the controller determines behavior-consistent information-based network control strategies and the drivers’ route choice decisions are influenced by several factors. Figure 2.1 illustrates conceptually the difference between the traditional DTA-based and the proposed behavior-consistent approaches. The traditional approach (part a) uses the DTA solutions directly as the information strategies to provide to drivers, while the behavior-consistent approach (part b) uses a fuzzy control based search procedure
(Section 2.5.2) to determine these strategies using the standard DTA solution and a controller-estimated driver behavior model.

This chapter integrates several components in a rolling horizon framework to analyze the bi-level interactive decision-making process: a DTA model (Peeta, 2004), an iterative search based optimization procedure involving a fuzzy control model and a controller-estimated driver behavior model (Section 2.5.2), a traffic flow simulator DYNASMART (Mahmassani, 2001) as a proxy for field conditions, and a hybrid multinomial logit model to represent actual driver behavior. The latter two models are not required in the real-world deployment context as field data is available. The DTA model computes the ideal proportion of drivers who should choose specific routes for the objective considered; for example, the SO solution. The optimization procedure is used to determine behavior-consistent strategies that direct the traffic system as close as possible to the DTA objective. The traffic flow simulator is used to capture the dynamic network level interactions and evaluate the system performance.

Given the tradeoffs that exist between the computational needs for real-time operations and the need to incorporate various problem dimensions adequately (for example, traffic flow and behavior modeling), the proposed solution framework does not explicitly solve the fixed point problem proposed by Bottom (2000) involving the three alternative composite maps. Instead, it takes advantage of an explicit estimation of driver behavior, and leverages the beneficial characteristics of the rolling horizon procedure and the projected SO solution, to determine the information strategies. The approach adopted here is to direct the system, through behavior-consistent information strategies, as close as possible to the projected SO DTA solution within a rolling horizon framework. It has four synergistic characteristics. First, and the primary contribution of this paper, is that behavior consistency is explicitly incorporated. Second, the determination of the SO DTA solution for a specific roll period using projected traffic conditions provides a desirable goal for the controller to achieve through information provision. Third, the incorporation of evolving field traffic conditions through the rolling horizon framework from one roll period to the next significantly limits potential error propagation that can result from the non-projection of traffic conditions after the
behavior-consistent strategies to be provided to drivers are determined. Finally, and consequent to the second and third characteristics, the determination of computationally very expensive iterative real-time DTA solutions to estimate projected traffic conditions is avoided. This enables the determination, in a timely manner, of effective and behaviorally more realistic information strategies that can lead to an enhancement in system performance. The remainder of this chapter is organized as follows. Section 3.2 describes the problem and Section 3.3 formulates it. Section 3.4 presents the solution concept. Section 3.5 discusses experiments and analyzes their results. Section 3.6 presents some concluding comments.

3.2 Problem Description

The problem being addressed here is the behavior-consistent control of a vehicular traffic network for the period of interest, typically a peak traffic period. It is labeled behavior-consistent real-time traffic routing under information provision (BCRTRIP), and can be described as follows. A controller seeks to continuously optimize network performance by providing real-time traffic routing information to drivers where the drivers’ likely response behavior is factored in determining the information. Hence, the problem being addressed here requires the determination of information-based network control strategies that are, to the extent possible, simultaneously consistent with the controller-estimated driver behavior and the objectives of the controller. After the information strategies are generated, they are disseminated to the drivers to influence their route choice decisions, and consequently system performance. We assume that every node (with alternative routing options) on the current route of a driver is a potential decision point, implying that en-route re-routing is possible. The performance of the system under the information strategies is continuously measured in real-time and new information strategies are computed based on the field data measurements.

Figure 3.1 shows the conceptual flowchart for the BCRTRIP problem. The problem is represented using a rolling horizon stage based framework due to its deployment characteristic and the uncertainty associated with future time-dependent demand and
network conditions. The planning horizon of interest, taken here as the peak traffic period, is divided into stages. Each stage is divided into a roll period and a tail period. Using the field network conditions in the roll period of the current stage \( \sigma \), and the projected time-dependent O-D demand for the next stage \( \sigma+1 \), the behavior-consistent information-based network control strategies for the next roll period are determined at some point during the current roll period. At the end of the current roll period, the stage counter is incremented by one. In the next stage, the controller uses these information strategies to provide route recommendations to drivers. Each driver uses his/her behavioral logic (based on inherent behavioral tendencies, ambient traffic conditions, and the information provided by the controller) to select a route from his/her preferred set of alternatives. The aggregation of all individual driver route choice decisions determines the network performance. The rolling horizon framework terminates if the end of the planning horizon is reached. If not, the controller tracks the filed network conditions (system state) using sensor data, and repeats the rolling horizon deployment process.

It should be noted here that the route selected by a driver can coincide with that recommended by the controller or differ from it partially or fully. Since the information provided by the controller factors the drivers’ likely reactions to the information, the likelihood that a driver chooses the route recommended by the controller increases.

Details on the implementation of the rolling horizon approach illustrated in Figure 2.3 are provided in Section 2.3. The next section discusses the formulation.
3.3 Problem Formulation

3.3.1 Notation

Additional variables used in this chapter are:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>Index for a node in the network, $n \in N$</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>Superscript denoting a departure time interval up to the end of the current roll period, $\kappa = 1, \ldots, \sigma l$</td>
</tr>
<tr>
<td>$\tau$</td>
<td>Superscript for a departure time interval for the next stage, $\tau = \sigma l+1, \ldots, \sigma l+h$</td>
</tr>
<tr>
<td>$t$</td>
<td>Superscript for the current time interval</td>
</tr>
<tr>
<td>$\rho(\sigma)$</td>
<td>Roll period indicator for stage $\sigma$; corresponds to $\kappa = (\sigma-1)l+1, \ldots, \sigma l$</td>
</tr>
<tr>
<td>$i^*$</td>
<td>Subscript for the origin node of a driver who departed up to time interval $\sigma l$ and does not reach his/her destination in the current roll period, $i^* \in I$</td>
</tr>
<tr>
<td>$U$</td>
<td>Set of driver classes in terms of information availability, $U = {1, 2}$</td>
</tr>
<tr>
<td>$u$</td>
<td>Superscript for driver information class, $u \in U$; $u = 1$ if driver can receive information, and $u = 2$ if driver cannot receive information</td>
</tr>
<tr>
<td>$\hat{R}_{ij}^{u, \tau}$</td>
<td>Forecasted new O-D demand for the next stage, expressed as the set of drivers of class $u$ who wish to depart from $i$ to $j$ in time interval $\tau$, $\tau = \sigma l+1, \ldots, \sigma l+h$</td>
</tr>
<tr>
<td>$R_{ij}^{u, \tau}$</td>
<td>Actual new O-D demand for the next stage, expressed as set of drivers of class $u$ who wish to depart from $i$ to $j$ in time interval $\tau$, $\tau = \sigma l+1, \ldots, \sigma l+h$</td>
</tr>
<tr>
<td>$Q_{\kappa ij}^{su}$</td>
<td>Set of drivers of class $u$ that departed origin $i^*$ to destination $j$ in time interval $\kappa = 1, \ldots, \sigma l$ who have not reached their destinations at the end of the current roll period, and are on link $a$ in time interval $\sigma l$</td>
</tr>
<tr>
<td>$\hat{S}_{i^*ij}^{s\kappa}$</td>
<td>Set of drivers of class $u$ that departed origin $i^*$ in time interval $\kappa = 1, \ldots, \sigma l$ who have not reached their destinations at the end of the current roll period, and are forecasted to depart from the first intermediate node $i$ to destination $j$ in time interval $\tau$ in the next stage, $\tau = \sigma l+1, \ldots, \sigma l+h$</td>
</tr>
</tbody>
</table>
### Notations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_{ij}^{ur}$</td>
<td>Set of drivers of class $u$ that departed origin $i^*$ in time interval $\kappa = 1, \ldots, \sigma l$ who have not reached their destinations at the end of the current roll period, and depart from the first intermediate node $i$ to destination $j$ in time interval $\tau$ in the next stage, $\tau = \sigma l+1, \ldots, \sigma l+l$</td>
</tr>
<tr>
<td>$\hat{S}_{ij}^{ur}$</td>
<td>Forecasted intermediate O-D demand for the next stage, expressed as the set of drivers of class $u$ who are forecasted to depart from $i$ to $j$ in time interval $\tau$, $\tau = \sigma l+1, \ldots, \sigma l+h$</td>
</tr>
<tr>
<td>$S_{ij}^{ur}$</td>
<td>Intermediate O-D demand for the next roll period, expressed as the set of drivers of class $u$ who depart from $i$ to $j$ in time interval $\tau$, $\tau = \sigma l+1, \ldots, \sigma l+l$</td>
</tr>
<tr>
<td>$PK_r^{ij}$</td>
<td>Set of preferred routes connecting O-D pair $ij$ for driver $r$, $PK_r^{ij} \subseteq K_{ij}$</td>
</tr>
<tr>
<td>$PK_{ij}$</td>
<td>Set of driver-preferred routes connecting O-D pair $ij$, $PK_{ij} = \bigcup_r PK_r^{ij} \subseteq K_{ij}$</td>
</tr>
<tr>
<td>$\hat{\delta}_{ijk}^{r}$</td>
<td>Controller-estimated route choice dummy; 1 if driver $r$ is leaving from $i$ to $j$ in time interval $\tau$ and is estimated to take route $k$, and 0 otherwise, $k \in \hat{PK}_{ij}^{r}$</td>
</tr>
<tr>
<td>$X_{ijk}^{r\tau}$</td>
<td>Vector of attributes for route $k$, excluding information, that influence the route choice decision of driver $r$ in time interval $\tau$, $k \in PK_{ij}^{r}$</td>
</tr>
<tr>
<td>$T_{ijk}^{r\tau}$</td>
<td>Travel time experienced during the next roll period by driver $r$ leaving node $i$ at some point during that roll period for node $j$ on route $k$ in time interval $\tau$, $k \in PK_{ij}^{r}$</td>
</tr>
<tr>
<td>$\xi_{ijk}^{\tau ar}$</td>
<td>Time-dependent driver spatio-temporal variable; 1 if driver $r$ choosing route $k$ (connecting O-D pair $ij$) in time interval $\tau$ is on link $a$ in time interval $t$, and 0 otherwise</td>
</tr>
<tr>
<td>$d_{ijk}^{\tau ar}$</td>
<td>Number of drivers of class $u$ traveling from $i$ to $j$ on route $k$ in time interval $\tau$ who enter arc $a$ in time interval $t$</td>
</tr>
<tr>
<td>$m_{ijk}^{\tau ar}$</td>
<td>Number of drivers of class $u$ traveling from $i$ to $j$ on route $k$ in time interval $\tau$ who exit link $a$ in time interval $t$</td>
</tr>
<tr>
<td>$x_{ta}$</td>
<td>Number of drivers on link $a$ at the beginning of time interval $t$</td>
</tr>
</tbody>
</table>
3.3.2 Problem Definition

Consider a directed graph \( G(N,A) \) representing a traffic network with \( N \) nodes, \( A \) directed arcs, origins \( i \in I \), and destinations \( j \in J \). An origin, a destination and/or just a junction of physical links can be represented by a node. We are given the time-dependent O-D demand forecasts for the next stage, the number of previously assigned drivers who are present in the network at the beginning of the next stage and their current routes, the associated forecasts for the intermediate demand, the controller-estimated set of driver-preferred routes and their attributes, the information class of each driver, and the controller-estimated driver behavior model. The controller determines towards the end of the current roll period the behavior-consistent information-based network control strategies \( \theta_{ijk}(\sigma+1) \) for the next roll period to provide to the O-D demand route recommendations so as to minimize the system travel time for the next roll period.
3.3.3 Formulation

Given:
(i) \( G(N,A) \)
\( \hat{R}^{ut} \);
(ii) \( \forall i, j, u, \tau = \sigma l + 1, \ldots, \sigma l + h \)
(iii) \( Q_{r,j}^{au} \);
\( \forall i^*, j, u, \kappa = 1, \ldots, \sigma l \)
(iv) \( \hat{S}_{i^*j}^{au} \);
\( \forall i^*, i, j, \kappa = 1, \ldots, \sigma l, u, \tau = \sigma l + 1, \ldots, \sigma l + h \)
(v) \( P \hat{K}_{ij}^r \);
\( \forall i, j, r \in \{ \hat{R}^{ut} \cup \hat{S}_{i^*j}^{au} \} \)
(vi) \( \hat{X}_{ijk}^r \);
\( \forall i, j, k \in P \hat{K}_{ij}^r, r \in \{ \hat{R}^{ut} \cup \hat{S}_{i^*j}^{au} \}, \tau = \sigma l + 1, \ldots, \sigma l + h \)
(vii) \( Y_{ij}^r \);
\( \forall i, j, k \in P \hat{K}_{ij}^r, r \in \hat{S}_{i^*j}^{au} \)
(viii) \( \hat{\delta}_{ijk}^{rv} \);
\( \forall i, j, k \in P \hat{K}_{ij}^r, r \in \hat{S}_{i^*j}^{au} \)
(ix) \( \Omega_{ur}^{\tau} \);
\( \forall u, r \in \{ \hat{R}^{ut} \cup \hat{S}_{i^*j}^{au} \} \)
(x) \( \hat{f} \)

Objective function (controller objective):

Min. \( \{ \sum \sum \sum \sum \sum \sum \sum \sum \sum \sum \Delta \cdot \sum \sum \sum \sum \sum \sum \sum \sum \sum \sum \} \)
\( \sum \sum \sum \sum \sum \sum \sum \sum \sum \sum \Delta \cdot \sum \sum \sum \sum \sum \sum \sum \sum \sum \sum \} \)
\( \sum \sum \sum \sum \sum \sum \sum \sum \sum \sum \Delta \cdot \sum \sum \sum \sum \sum \sum \sum \sum \sum \sum \} \)
\( \sum \sum \sum \sum \sum \sum \sum \sum \sum \sum \Delta \cdot \sum \sum \sum \sum \sum \sum \sum \sum \sum \sum \} \)

Subject to:

Controller-estimated driver behavior
\( \hat{\delta}_{ijk}^{ur} = \hat{f}(\hat{X}_{ijk}^{r(t)}, \hat{X}_{ijk}^{l(t)}), \forall k \in P \hat{K}_{ij}^r \); \( \forall i, j, k \in P \hat{K}_{ij}^r, r \in \{ \hat{R}^{ut} \cup \hat{S}_{i^*j}^{au} \}, \tau = \sigma l + 1, \ldots, \sigma l + h \) (3.2)

Demand conservation constraints
\( \hat{S}_{ij}^{ru} = \bigcup_{k=1}^{l} \hat{S}_{ij}^{ru} \);
\( \forall i, j, u, \tau = \sigma l + 1, \ldots, \sigma l + h \) (3.3)
\( \hat{S}_{ij}^{au} = \bigcup_{k=1}^{l} \hat{S}_{ij}^{au} \);
\( \forall i, j, u, \tau = \sigma l + 1, \ldots, \sigma l + h \) (3.4)
\( \sum \sum |\hat{\delta}_{ijk}^{ru} \cdot \Omega_{ur}^{\tau}| = |\hat{S}_{ij}^{ru}| \);
\( \forall i, j, u, \tau = \sigma l + 1, \ldots, \sigma l + h \) (3.5)
\[
\sum_{r \in \mathbb{R}_{ijk}^l} \sum_{k \in PK^r_{ij}} [\delta^r_{ijk} \cdot \Omega^{sr}] = |R^h_{ij}|; \quad \forall i, j, u, \tau = \sigma l + 1, \ldots, \sigma l + l
\]  
(3.6)

**Information-based network control constraints**

\[
\theta_{ijk}^{\sigma(+1)} = g_y[(G(N, A), \hat{R}^r_{ij}, \hat{S}^u_{ij}, \hat{R}^s_{ij}, \hat{Y}^r_{ij}, \hat{X}^r_{ij}, \gamma_{ijk}, \Omega^u, \delta^r_{ijk}(F)), \forall i, j, k \in PK^r_{ij}, u, r, \tau = \sigma l + 1, \ldots, \sigma l + h]; \quad \forall i, j, k \in CK^{\rho(\sigma+1)}
\]  
(3.7)

\[
Y^r_{ijk} = g_y(\theta_{ijk}^{\sigma(+1)}, Y^r_{ijk}, \delta^r_{ijk}, \Omega^u); \quad \forall i, j, k \in PK^r_{ij}, r \in \{ R^u_{ij} \cup S^u_{ij} \}, \tau = \sigma l + 1, \ldots, \sigma l + l
\]  
(3.8)

**Flow modeling constraints**

\[
\zeta^r_{ijk} = f[(X^r_{ijk}, Y^r_{ijk}), \forall k \in PK^r_{ij}]; \quad \forall i, j, k \in PK^r_{ij}, r \in \{ R^u_{ij} \cup S^u_{ij} \}, \tau = \sigma l + 1, \ldots, \sigma l + l
\]  
(3.9)

\[
\zeta^r_{ijk} = g_y[(\mathcal{R}^u_{ij}, \mathcal{Q}^a_{ij}, \delta^r_{ijk}), \forall i, j, k, \in PK^r_{ij}, \kappa = 1, \ldots, \sigma l, u, a, r \in \{ R^u_{ij} \cup Q^a_{ij} \}, \tau = \sigma l + 1, \ldots, \sigma l + l]; \quad \forall i, j, k \in PK^r_{ij}, \tau
\]  
(3.10)

\[
x^{a|l} = \sum_{i j k} \sum_{r t} \sum_{\tau} \sum_{l_{ijk}} \zeta^r_{ijk} \cdot \Delta; \quad \forall t, a \in \mathcal{A}
\]  
(3.11)

\[
T^{a|l}_{ijk} = \sum_{t=\sigma l+1}^{\sigma l+1} \sum_{\tau} (x^{a|l}_{ijk} \cdot \Delta); \quad \forall i, j, k \in PK^r_{ij}, r \in \{ R^u_{ij} \cup S^u_{ij} \}, \tau = \sigma l + 1, \ldots, \sigma l + l
\]  
(3.12)

\[
x^{a|l+1} = \sum_{\kappa=1}^{\sigma l} \sum_{i j k} \sum_{r} \mathcal{Q}^a_{ijk}; \quad \forall a \in \mathcal{A}
\]  
(3.13)

**Flow conservation constraints at nodes and links**

\[
\sum_{b} \mathcal{D}^b_{cb} = \sum_{c} \mathcal{D}^c_{cb} + L^t_{n} - O^t_{l}; \quad \forall t, n \in \mathcal{L}, b \in \mathcal{B}(n), c \in \mathcal{C}(n)
\]  
(3.14)

\[
x^{a|l} = x^{a|l} + \mathcal{D}^{l-1} - m^{l-1, a}; \quad \forall t, a \in \mathcal{A}
\]  
(3.15)

**Definitional constraints**

\[
d^{a|l} = \sum_{i j k u} \sum_{r} \sum_{\tau} d^{a|l}_{ijk}; \quad \forall t, a \in \mathcal{A}
\]  
(3.16)

\[
m^{a|l} = \sum_{i j k u} \sum_{r} \sum_{\tau} m^{a|l}_{ijk}; \quad \forall t, a \in \mathcal{A}
\]  
(3.17)

\[
L^t_{n} = \sum_{j u} \sum_{\tau} R^a_{uj} + \sum_{j u} \sum_{\tau} S^a_{uj}; \quad \forall t, n \in \mathcal{L}
\]  
(3.18)

\[
O^t_{l} = \sum_{i j c} \sum_{\tau} \sum_{k u} m^{a|l}_{ijk}; \quad \forall t, n \in \mathcal{J}
\]  
(3.19)

**0-1 variable constraints**

\[
\zeta^r_{ijk} = 0 \text{ or } 1; \quad \forall i, j, k \in PK^r_{ij}, t, a \in \mathcal{A}, r \in \{ R^u_{ij} \cup S^u_{ij} \}, \tau = \sigma l + 1, \ldots, \sigma l + l
\]  
(3.20)

\[
\delta^r_{ijk} = 0 \text{ or } 1; \quad \forall i, j, k \in PK^r_{ij}, r \in \{ \hat{R}^u_{ij} \cup \hat{S}^u_{ij} \}, \tau = \sigma l + 1, \ldots, \sigma l + l
\]  
(3.21)

\[
\delta^r_{ijk} = 0 \text{ or } 1; \quad \forall i, j, k \in PK^r_{ij}, r \in \{ Q^a_{ij} \}, \tau = \sigma l + 1, \ldots, \sigma l + l
\]  
(3.22)

\[
\Omega^u = 0 \text{ or } 1; \quad \forall u, r \in \{ R^u_{ij} \cup Q^a_{ij} \}
\]  
(3.23)
This formulation is a non-linear mixed integer model with some stochastic variables $(\hat{\delta}_{ijk}, \delta_{ijk})$. It integrates in a stage-based rolling horizon framework several components that are required to adequately model and address the BCRTRIP problem. A primary contribution to the literature is that it explicitly considers network dynamics and driver behavior. That is, the system states and the information strategies depend on both driver behavior and traffic flow dynamics resulting from individual driver route choice decisions. Another key contribution is that the controller does not pre-specify driver behavior but rather estimates their likely behavior under information provision. Hence, the formulation includes two driver behavior models; one ($\hat{F}$) is used to explicitly estimate driver behavior while the other (F) is used to represent actual driver behavior. It is important to note that in a real-world deployment context the controller does not know the actual driver behavior a priori. Consequently, the information strategies are determined using only the estimation of the driver behavior. The information strategies are used to provide routing information to the drivers who make route choice decisions based on their behavioral tendencies and the controller-provided information. This denotes the bi-level interactive decision-making structure discussed in Section 3.1. The formulation also uses the concept of route classification based on the relevance of routes to the drivers and the controller, as defined in Section 3.3.1.2. This concept is developed in Chapter 2 and is used here to provide a realistic deployment mechanism to enhance driver compliance in a behavior-consistent manner. Further, the approach determines who to provide information based on the identification of priorities (Section 2.4.3.4). These contributions together enable the development of the behavior-consistent approach.
The decision variables are the set of information-based network control strategies \( \theta_{ijk}^{\rho(\sigma+1)} \), \( \forall i, j, k \in CK_{ij}^{\rho(\sigma+1)} \). The set of controller-desired routes \( DK_{ij}^{\rho(\sigma+1)} \) are explicitly differentiated from the controller-estimated set of driver-preferred routes \( PK_{ij} \) leading to the concept of controllable routes \( CK_{ij}^{\rho(\sigma+1)} \), \( \forall i, j \). There are different time scales associated with: (i) the projection of network conditions to determine the information strategies, and (ii) the evaluation of system performance under these strategies. Depending on the case, the superscript \( \tau \) is defined to take values corresponding to the length of a stage or the length of a roll period.

### 3.3.3.1 Objective Function

Equation (3.1) represents the controller’s objective, the minimization of the system travel time for the next roll period. This travel time is equal to the summation of all the realized individual driver travel times during this period, and can be computed using three components. The first component is the travel time that drivers who have not reached their destinations before the end of the current roll period \( (r_{\sigma}^{\text{end}} \in Q_{r\sigma j}^{\text{end}} ) \) spend traveling in the next roll period before reaching their first intermediate node. It is possible that a driver may not reach such a node in the next roll period. This component is divided in two sub-components. The first sub-component is a constant term equal to the number of drivers in the set \( Q_{r\sigma j}^{\text{end}} \) times the number of time intervals in the next roll period, multiplied by \( \Delta \), resulting in the travel time that those drivers would spend during the next roll period if they did not reach their first intermediate node. The second sub-component computes the cumulative number of drivers in \( Q_{r\sigma j}^{\text{end}} \) that reach their first intermediate node in each successive time interval of the next roll period, and multiplies them with \( \Delta \), to determine the travel time that those drivers would spend traveling during the next roll period after reaching their first intermediate node. Hence, the difference between these two sub-components gives the travel time that drivers in \( Q_{r\sigma j}^{\text{end}} \) spend traveling in the next roll period before reaching their first intermediate node. The second
component is the travel time of the intermediate demand \((r \in S^\text{int}_{ij})\) drivers from their first intermediate node in the next roll period. Similarly, the third component computes the travel time of the new demand \((r \in R^\text{int}_{ij})\) drivers from their origin in the next roll period.

### 3.3.3.2 Controller-estimated Driver Behavior

Constraint (3.2) denotes the controller-estimated route choice for driver \(r\) (represented through \(\hat{\delta}_{jk}^r\)) as a function \((\hat{F})\) of the estimated route attributes \(\hat{X}_{ijk}^r\), and the route recommendation 0-1 dummy \(Y_{jk}^r\). A hybrid multinomial logit model is used to represent the controller-estimated driver behavior model. Its systematic component is determined using simple if-then rules. \(\hat{X}_{ijk}^r\) consists of the controller-estimated expected route travel times \(TT\) and the number of nodes \(NN\) for each route. Table 2.1 shows the set of behavioral if-then rules used in this study while details on function \(\hat{F}\) are provided in Chapter 6.

### 3.3.3.3 Demand Conservation Constraints

Constraints (3.3) and (3.4) represent intermediate demand conservation constraints. Constraint (3.3) states that the intermediate O-D demand for the next stage forecasted towards the end of the current roll period is equal to the aggregation of \(\hat{S}^\text{int}_{ij}\). This forecasted demand is used along with the forecasted new demand to generate information strategies for the next roll period towards the end of the current stage. Constraint (3.4) indicates that the actual intermediate O-D demand for the next roll period is equal to the aggregation of \(S^\text{int}_{ij}\).
Constraints (3.5) and (3.6) denote actual intermediate and new demand conservation constraints, respectively. They are used to ensure that all drivers in $S_{ij}^{ur}$ and $R_{ij}^{ur}$ have chosen a route to their corresponding destinations. Here, the product $\delta_{ijk}^{rt} \cdot \Omega^{ur}$ takes value 1 if driver $r$ of class $u$ chooses route $k$ in time interval $\tau$, and 0 otherwise.

3.3.3.4 Information-based Network Control Constraints

Constraints (3.7) and (3.8) represent the information-based network control constraints. Constraint (3.7) states that the behavior-consistent information-based network control strategies are a function of the time-dependent O-D demand forecasts for the next roll period, the number of previously assigned drivers who are present in the network at the beginning of the next stage and their current routes, the controller-estimated set of driver-preferred routes and their attributes, the information class of each driver, and the controller-estimated driver behavior model. Here, $g_\theta$ denotes a procedure used to determine the information strategies for the next roll period; in this study we use the iterative search based optimization procedure described in Section 2.5.2 as part of the solution algorithm described in Section 3.4 The iterative search based optimization procedure is a sub-problem of the broader problem discussed in this paper. It only solves for the information strategies but does not address the broader problem depicted in Figure 3.2 and described in Section 3.4.1.

Constraint (3.8) denotes the discretization of the information strategies to determine the specific routes to recommend to a subset of drivers selected according to the behavior-consistent strategy. It uses a priority scheme where drivers considered to receive route recommendations are categorized in priority subgroups based on their existing routes, prior route recommendations, and their responses to these recommendations. Constraints (3.2), (3.7) and (3.8) together indicate that $\hat{\delta}_{ijk}^{rt}$ is a function of $\theta_{ijk}^{r(\sigma+1)}$ and vice versa, implying the fixed-point structure of (3.7). Details of the priority scheme are provided in Section 2.4.3.4.
3.3.3.5 Flow Modeling Constraints

Constraints (3.9)-(3.13) represent the flow modeling constraints. Constraint (3.9) states that the route choice for driver $r$ (represented through dummy $\delta_{ijk}^{rt}$) is a function (F) of the route attributes $X_{ijk}^{rt}$ (such as past experience, inertia, and route complexity) and the route recommendation (information) dummy $Y_{ijk}^{rt}$. Function F symbolically represents individual driver behavior and is not an explicit model/procedure. In reality, the actual driver behavior mechanism is unknown to the controller and manifests through the realized network conditions. In the study experiments, in the absence of field data, a specific model (discussed in Section 3.5.1.3) is used to represent F.

Constraints (3.10)-(3.12) incorporate time-dependent driver spatio-temporal variables $\tau_{ijk}\xi$ to represent traffic flow evolution as a function of the driver route choices ($\delta_{ijk}^{rt}$). Constraint (3.10) uses the time-dependent driver spatio-temporal variable $\zeta_{ijk}^{tar}$ to track the driver; it indicates if driver $r$ leaving from $i$ to $j$ choosing route $k$ in time interval $\tau$ is on link $a$ in time interval $t$. Function $g_{\xi}$ symbolically represents the traffic flow evolution in the network and captures the complex nonlinear spatio-temporal interactions among vehicles. It is typically modeled using a traffic flow simulator. However, in the real-world deployment context, $\zeta_{ijk}^{tar}$ is obtained from the sensor data which represents the realized network conditions.

Existing DTA models do not incorporate the driver spatio-temporal variables which involve driver decision-making as an integral element of these variables. Instead, they use time-dependent link-path incidence variables to determine the presence of a vehicle on a given link across time intervals. These incidence variables are typically the outcome of an assignment process undertaken by a system controller where full or partial compliance to the information provided is assumed. Hence, the incidence variables do not adequately represent driver behavior and the resulting models are behaviorally restrictive and limited in their ability to model driver response to information provision. By contrast, the time-dependent driver spatio-temporal variables seamlessly incorporate
driver behavior and the network-level interactions. That is, the \( \tau_{ijk} \) variables enable the actual network dynamics to be the result of the actual driver route choices.

Constraint (3.11) computes the number of drivers on link \( a \) at the beginning of time interval \( t \) using the driver spatio-temporal variable values from interval \( t-1 \). Constraint (3.12) states that the individual route travel times are the summation of the number of time intervals in which a 0-1 driver spatio-temporal variable (for a given \( i, j, k \) and \( \tau \)) takes a value 1, multiplied by \( \Delta \). It implies the number of discrete time intervals that the corresponding driver \( r \) spends in the system. Constraint (3.13) is a boundary conservation constraint that ensures that all drivers who have not reached their destinations before the end of the current roll period are present at the beginning of the next roll period where they are located at the end of the current roll period.

3.3.3.6 Flow Conservation Constraints at Nodes and Links

Constraints (3.14) and (3.15), respectively, represent the conservation of vehicles at nodes and links. Constraint (3.14) states that at time \( t \) on node \( n \), the number of vehicles entering all links incident from the node should equal the sum of the number of vehicles exiting from all links incident to that node and the net demand. Constraint (3.15) states that at the beginning of time interval \( t \), the number of vehicles on link \( a \) is the sum of the number of vehicles on the link at the beginning of the previous time interval (\( t-1 \)) and vehicles entering the link in the previous time interval, minus the vehicles exiting the link in the previous time interval.

3.3.3.7 Definitional Constraints

Constraints (3.16)-(3.19) are definitional constraints. Constraints (3.16) and (3.17) are, respectively, the definitional constraints for the number of vehicles entering and exiting links. Constraint (3.18) states that the number of vehicles representing the demand at node \( n \) in time interval \( t \) is the sum of the corresponding new and intermediate
demand. Constraint (3.19) is the definitonal constraint for the number of vehicles exiting the network at node \( n \) in the time interval \( t \).

### 3.3.3.8 0-1, Temporal Correctness, and Non-negativity Constraints

Constraints (3.20)-(3.25) restrict specific variables to take a value 0 or 1. Constraints (3.26) are the temporal correctness constraints that restrict the departure time interval \( \tau \) to be at most the current time interval \( t \). Constraints (3.27) indicate the non-negativity requirement for all variables.

### 3.4 Solution Concept

A key difference of the behavior-consistent approach compared to most traditional DTA approaches is that the controller only recommends routes to (a subset of) drivers, and does not assume compliance, whether complete or based on an artificial rate. The system states are determined by the driver decisions, but these decisions can be influenced by the information provided by the controller. Thus, the controller only has limited “control” on the system through information provision. Therefore, the controller objective is to “guide” the system, to the extent possible, towards a desired state in each stage (for example, the SO solution) by adjusting its information provision strategies. The route chosen by a driver is decided by his/her (actual) behavior.

#### 3.4.1 Solution Framework

Figure 3.2 shows the solution framework for the BCRTRIP problem. The controller uses a rolling horizon stage-based deployment framework and seeks to direct the system towards the time-dependent SO DTA state. It should be noted here that the UE DTA state or any other controller objective could also be used in this framework without loss of generality. Given the field network conditions in the current roll period \( \rho(\sigma) \) and the projected time-dependent O-D demand for the next stage \( \sigma+1 \), the corresponding SO
DTA solution for the next stage is generated. For computational efficiency, the SO route assignment proportions are assumed constant within each assignment interval of the stage, though they vary across these intervals. The controller then uses the SO proportions and an iterative search procedure to determine the behavior-consistent information-based network control strategies to provide route guidance to the drivers, so that the actual driver decisions in the next roll period result in close to SO route proportions. The iterative search optimization procedure (Section 2.5.2), represented by the non-shaded box located in the middle of the flowchart in Figure 3.2, involves a controller-estimated driver behavior model and a fuzzy control model. The fuzzy control model represents the search mechanism (direction and step size). The iterative search optimization procedure determines the behavior-consistent route proportions for the next roll period that should be recommended to the drivers. At the end of the current roll period, the stage counter is incremented by one. In the next roll period, the controller uses the behavior-consistent route proportions to provide route recommendations to a prioritized subset of drivers (Section 2.4.3.4). The system states for that roll period are a function of the driver routes, which include decisions by the subset of drivers that receive information. If the end of the current roll period does not represent the end of the planning horizon, the controller measures the field network conditions (system state) using sensor data, and repeats the process for the next roll period. Otherwise, the rolling horizon framework is terminated.

In this framework, the routes that are recommended to the drivers are only those that are simultaneously driver-preferred and move the system closer to the SO state. That is, the SO routes which are not considered by the drivers (do not belong to their preferred choice set) are not recommended to them. Some O-D pairs may not have routes that are simultaneously SO and driver-preferred, in which case no search is done for them. Chapter 5 develops alternative paradigms where the controller uses routes that match SO routes to a significant degree. This introduces deployment flexibility by enabling the practical implementation of the behavior-consistent strategies.
The solution approach computes the SO proportions for each assignment interval of the next stage by solving the SO DTA problem for the length of that stage. However, the information strategies are determined only for the next roll period using the corresponding SO proportions. The effects of the projected O-D demand and the network level interactions on the information strategies for the next roll period are captured to some extend through the computation of the SO DTA for the length of the stage. This is because the SO proportions in each assignment interval are interdependent with the projected conditions and/or assignments for the rest of the stage.

3.4.2 Algorithmic Solution Framework

The algorithmic steps of the solution framework are briefly described hereafter. It uses off-line and on-line components. The off-line component, which is represented by Step 0, determines the driver-preferred route sets and the corresponding controller-estimated expected travel times for drivers. The on-line components, represented by Steps 1-6, are used to determine and deploy the information-based network control strategies.

**Step 0: estimation of preferred routes and expected travel times for the current day**

Estimate the driver-preferred route sets and their corresponding time-dependent controller-estimated expected travel times for the current day. These can be done through a combination of historical data, travel surveys, and/or technologies such as two-way communication systems and global position systems. In this chapter, a heuristic approach is used to generate the route sets and the expected travel times for the experiments, as discussed in the experimental setup.
Step 1: initialization

To initialize the stage-based solution framework for the current day, set $\sigma = 1$ and $\theta_{ijk}^{\rho(\sigma+1)} = 0$, $\forall\ i, j, k$.

Step 2: determination of the SO states

Given the network conditions for the roll period $\rho(\sigma)$ and the projected time-dependent O-D demand for the next stage $\sigma+1$, the time-dependent SO DTA solution is computed for the various assignment intervals of the next stage. The SO DTA solution provides the controller-desired route sets $DK_{ijk}^{\rho(\sigma+1)}$ and the corresponding SO proportions $SO_{ijk}^{\rho(\sigma+1)}$ of drivers assigned to these routes during the next roll period, $\forall\ i, j, k \in DK_{ijk}^{\rho(\sigma+1)}$.

Step 3: iterative search based optimization procedure

This step consists of sub-steps 3.1-3.3 that represent an iterative search procedure. The iteration counter is set to 1.

Step 3.1: controller’s estimation of driver behavior

The controller-estimated driver behavior model $\hat{F}$ is used to compute the controller-estimated proportions of drivers, $E_{ijk}^{\rho(\sigma+1)}$, taking routes for the next roll period based on the corresponding information-based network control strategies $\theta_{ijk}^{\rho(\sigma+1)}$, $\forall\ i, j, k \in CK_{ijk}^{\rho(\sigma+1)}$. 
Step 3.2: update of the information strategies

The fuzzy control model is used to adjust the information-based network control strategies $\theta^{\rho(\sigma+1)}_{ijk}$ based on the difference between the SO proportions $SO^{\rho(\sigma+1)}_{ijk}$ and the controller-estimated route choice proportions $E^{\rho(\sigma+1)}_{ijk}$, $\forall i, j, k \in CK^{\rho(\sigma+1)}_{ijk}$.

Step 3.3: convergence check

Check for convergence. Convergence is achieved when the controller-estimated proportions $E^{\rho(\sigma+1)}_{ijk}$ do not change from one iteration to the next by more than a pre-specified threshold value, $\forall i, j, k \in CK^{\rho(\sigma+1)}_{ijk}$. If convergence is achieved, the set of behavior-consistent information-based network control strategies $\theta^{\rho(\sigma+1)}_{ijk}$ for the next roll period are available; go to Step 4. If convergence is not achieved, the iteration counter is updated by 1; go to Step 3.1.

Step 4: stage update and dissemination of information

At the end of the roll period, the projection horizon is rolled forward by $l$ time units to obtain the next stage ($\sigma = \sigma + 1$). Route recommendations $Y^{\tau \tau}_{ijk}$ are provided to the drivers during the roll period $\rho(\sigma)$ using the behavior-consistent information-based network control strategies $\theta^{\rho(\sigma)}_{ijk}$.

Step 5: evaluation of system performance

The system performance and the field conditions for the roll period are determined by the driver route choice decisions (based on behavioral tendencies, the routes characteristics $X^{\tau \tau}_{ijk}$, and the information provided by the controller $Y^{\tau \tau}_{ijk}$).
Step 6: check for termination

The solution framework terminates if the end of the planning horizon for the day is reached. If not, go to Step 2 of the algorithm.

3.5 Experiments

Simulation experiments are conducted for the BCRTRIP problem to address two primary objectives: (i) to provide insights in terms of the ability of the iterative search based optimization procedure to determine, at the network-level and in real-time, robust behavior-consistent information-based network control strategies, and (ii) to compare the performance of the behavior-consistent strategies with that of the traditional DTA-based strategies.

3.5.1 Experimental Setup

The experimental setup follows the design in Section 3.5.1. Figure 3.3 illustrates the Borman expressway corridor network and Section 2.6.1.2 describes it. Experiments are conducted using this network and synthetic data. Behavior characteristics are described in Section 2.6.1.3. Additional aspects for the experimental setup are discussed hereafter.

3.5.1.1 Driver-preferred Routes and their Controller-estimated Expected Travel Times

The driver-preferred route sets and their corresponding time-dependent controller-estimated expected travel times are estimated using an off-line two-step approach. The initial driver-preferred route sets are estimated using a two-step approach. First, a UE DTA problem is solved for the entire planning horizon using an average time-dependent demand matrix. These UE routes represent the initial set of routes used as input in the second step. Second, the initial route set and the controller-estimated driver behavior model are used to determine the route choice proportions. Then, several simulation runs are conducted in a sequential manner as follows. A random-number generator is used to
allocate a route to a driver in each run consistent with the route choice proportions. The actual route taken by the driver is based on the allocated route information, the controller-estimated driver behavior model, and the controller-estimated expected travel time for each driver. Hence, each run can generate several new routes for each O-D pair. The output from a run, in terms of the actual route chosen by the driver and the updated expected link travel times for the UE routes (which update the UE route choice proportions for random allocation in the next run), are used sequentially for the next run. This process is repeated several times. Then, the top five routes taken by the set of drivers for an O-D pair and their associated time-dependent travel times are assumed to represent, respectively, the driver-preferred route set and the associated time-dependent controller-estimated expected travel times. This approach is designed to represent the learning process that most drivers experience over time in the context of the determination of their preferred route choice set and their corresponding expected travel times. This is based on the premise that a driver considers only a subset of possible O-D routes based on past experience and imperfect/incomplete current knowledge of the traffic network.

3.5.1.2 Actual Driver Behavior

In the absence of field data, the actual behavior of the drivers is represented here by model. It should be noted here that the actual behavior at the individual driver level is currently an inferred quantity in the real-world, though technologies such as global positioning systems can substantially aid in modeling it. That is, in the future, when these technologies are adequately deployed and privacy-related policies are developed, the ability to track individual drivers can provide robust models of actual behavior as well as controller-estimated behavior. In this study, the actual behavior model is deliberately assumed to have a different structure compared to the controller-estimated model. This is to ensure that the study insights are based on conservative analyses and to imply that the actual behavior model is unknown to the controller. However, the
controller can estimate the linkages between various factors and aggregate level behavior using past studies and historical data.

Equation (3.28) shows the specification for the random coefficients path-size multinomial logit model used in this study to represent the actual behavior or the drivers. The path-size component (Equation (3.29)) corresponds to the general specification proposed by Ben-Akiva and Bierlaire (1999) and extended by Ramming (2002). It accounts for links being common to different routes. That is, the path-size component is an approximated measure of the amount of overlap of a route with all other routes in the choice set. Ignoring the effects of link overlaps across the choice set can result in unrealistic volumes over the set of common links.

It is assumed in this study that the distributions of the coefficients ($\beta$) are identical across all drivers. However, as indicated in Equation (3.28), the values of these coefficients vary across individual drivers to represent random taste variations across drivers. This study assumes a 10% random variation with respect to the mean of the coefficients.

$$U_{ijr} = \beta_{ET} \cdot ET_{ijr} + \beta_C \cdot C_{ijr} + \beta_Y \cdot Y_{ijr} + \beta_{PS} \cdot \ln(PS_{ijr}) + \beta_{SW} \cdot SW_{ijr} + \epsilon_{ijr}; \forall i, j, k \in PK_r, r, t \quad (3.28)$$

where,

$$PS_{ijr} = \sum_{a \in \Gamma_{ijk}} \left( \frac{l_a}{L_{ijk}} \right) \frac{1}{\sum_{m \in PK_j} \left( \frac{L_{ijk}^2}{L_{ijm}^2} \cdot \Theta_{ajm} \right)}; \forall i, j, k \in PK_r, r \quad (3.29)$$

$$SW_{ijr} = 1 \Leftrightarrow \left[ \frac{\sum_{n \in \Pi_{ijk}}}{\sum_{n \in \Pi_{ijm}}} = 1 \right], 0 \text{ otherwise}; \forall i, j, k \in PK_r, r, t \quad (3.30)$$

$U_{ijr}$ is the utility of route $k$ for driver $r$ leaving node $i$ for node $j$ in period $t$, $k \in PK_r$

$\beta_x^r$ is the coefficient of variable/function $x$ for driver $r$
$ET_{r}^{ijk}$ is the driver-expected travel time on route $k$ for driver $r$ leaving node $i$ for node $j$ in period $t$, $k \in PK_{ij}^r$

$C_{ijk}$ is the number of nodes on route $k$ connecting $ij$, $k \in PK_{ij}^r$

$PS_{r}^{ijk}$ is the path-size component for driver $r$ and route $k$ connecting $ij$, $k \in PK_{ij}^r$

$\Gamma_{ijk}$ is the set of links on route $k$ connecting $ij$, $k \in PK_{ij}^r$

$l_{a}$ is the length of link $a$, $a \in A$

$L_{ijk}$ is the length of route $k$ connecting $ij$, $k \in PK_{ij}^r$

$\Theta_{aijm}$ is the link-route incidence dummy; 1 if route $m$ connecting $ij$ uses link $a$, and 0 otherwise

$\lambda$ is a path-size model parameter

$\Pi_{ijk}$ is the set of nodes on route $k$ connecting $ij$, $k \in PK_{ij}^r$

$SW_{r}^{ijk}$ is a route switching dummy; 1 if by choosing route $k$ connecting $ij$ in period $t$, driver $r$ is not making a route switching from his/her current route $\tilde{m}$, and 0 otherwise, $k \in PK_{ij}^r$

$\epsilon_{r}^{ijk}$ is an i.i.d. extreme value disturbance or random component for driver $r$ in time period $t$ for route $k$ connecting $ij$, $k \in PK_{ij}^r$

The mean values of the coefficients were selected based on previous studies and problem characteristics. For example, a slightly higher value for $\beta_{r}$ is used to represent more responsive behavior in comparison to the less responsive behavior.

In the study experiments, it is assumed that the approach adopted in Section 3.5.1.1 provides reasonable estimates for the controller-estimated expected travel times, which are then multiplied by a uniform random number (between 0.95 and 1.05) to determine the driver-expected travel times ($ET$). This is to represent the notion that the controller may not have perfect knowledge of the driver-expected travel times.

In addition, it is important to reiterate here that the model used to represent driver behavior is completely different from the model used to estimate it. In the real world deployment context, the actual driver behavior replaces the model (Equation (28)) used
to determine it and the controller-estimated driver behavior model is calibrated using the observed data.

The route choice probabilities obtained from this model are converted to $\delta_{ijk}$ using the following approach. First, the probability range for a driver is demarcated into smaller ranges according to the choice probabilities. For example, if there are three routes with estimated choice probabilities 0.2, 0.3, and 0.5, the ranges associated with them are 0.0-0.2, 0.2-0.5, and 0.5-1.0, respectively. Second, a uniform random number generator is used to generate a value between 0 and 1. Third, if the generated random number falls in the range associated with a specific route, that route is assigned to the driver. The same approach is used to determine $\hat{\delta}_{ijk}$ from the values obtained through $F$.

### 3.5.1.3 Traffic Flow Simulation-assignment Model

A traffic simulation-assignment model, DYNA SMART, is used here to achieve two objectives: (i) to determine in each stage the time-dependent SO solution (Step 2 of the solution framework) using the DTA module of DYNA SMART, and (ii) to evaluate the system performance in each roll period under the time-dependent demand and driver route choice decisions (Step 5 of the solution framework) using the traffic flow simulator module of DYNA SMART.

An overview of the capabilities of DYNA SMART is provided by Chiu (2002). Pre-trip routing and en-route re-routing capabilities are enabled by embedding in DYNA SMART the model used to represent the actual driver behavior (Equation (28)). As illustrated in Equation (31), the actual compliance is a function $g_{\psi}$ of the route recommendation provided to individual drivers and the driver route choice behavior. The compliance variables $\psi_{ijk}$ take value 1 if route $k$ is recommended to driver $r$ in time interval $t$ and he/she chooses this route; and 0 otherwise.
\[
\psi_{ijk}^{rt} = g_\psi(\delta_{ijk}^{rt}, Y_{ijk}^{rt}); \quad \forall \ i, j, k \in PK_y^r, r, t = \sigma l + 1, \ldots, \sigma l + l
\] (3.31)

\(\delta_{ijk}^{rt}\) is the route choice dummy that takes value 1 if driver \(r\) chooses route \(k\) connecting O-D pair \(ij\) in time interval \(t\); and 0 otherwise.

3.5.1.4 Assumptions

Without loss of generality, the study experiments assume that: (i) the controller-forecasted demand is the same as the actual demand, (ii) all drivers with the same O-D pair have the same set of driver-preferred routes, (iii) except for the no-information case, all drivers have capabilities to receive personalized information, and (iv) the controller-estimated set of driver-preferred routes is the same as the actual set of driver-preferred routes. These assumptions ensure that the focus of the experiments is on analyzing the effectiveness of the behavior-consistent strategy relative to existing DTA strategies.

3.5.1.5 Computational Aspects

In all scenarios, 120,000 drivers are loaded during the first 60 minutes of analysis. The behavior-consistent information strategies are computed for all node-destination pairs, implying 8471 (197 x 43) O-D pairs. This represents a significant computational load but enables the provision of information at any point in time and space. In all scenarios, each stage has a length of 20 minutes and a roll period of 5 minutes. The experiments were conducted with a single Pentium 4 extreme edition processor running at 4.0 GHz. To the extent that the focus of this chapter is on developing a behavior-consistent paradigm, the computational aspects are not analyzed here. However, the proposed solution framework lends itself to a significant amount of parallelization at the O-D pair level. Further, Chapter 4 develops an off-line H-infinity filtering approach that optimizes the parameters of the fuzzy control model resulting in significant additional
computation savings. Hence, the parallelization in conjunction with the optimization of parameters can be used to enhance computation efficiency.

3.5.1.6 Scenarios

Six scenarios are evaluated in the experiments to investigate system performance under different information-based network control strategies. These scenarios are as follows.

Scenario I (no information): No information is provided to the drivers (NO-info). It is the do-nothing strategy and represents the base-case. Here, drivers make route choice decisions based only on past experience.

Scenario II (SO DTA): In this scenario, all drivers are assumed to fully comply with the recommended SO routes (SO DTA). By definition, it represents the best possible system performance.

Scenario III (SO-based information): Here, route guidance is provided using the SO solution route proportions (SO-info). SO routes (DK) are recommended to the drivers based on the proportion of drivers that are required to take each SO route. The SO route recommended to a driver may or may not belong to his/her driver-preferred route set. If the recommended route is not in the driver-preferred set, the driver completely ignores the information provided by the controller in the route choice decision-making process. If the recommended route is a driver-preferred route (PK), the driver uses the information in his/her decision-making process. If so, the likelihood of choosing the recommended route is increased.

Scenario IV (SO-based information only about controllable routes): In this scenario, only controllable (CK) routes based on their proportions in the SO solution are recommended (SO-CK-info). Hence, if no driver-preferred route exists in the SO solution, the controller does not recommend a route to that driver. This approach increases the likelihood that drivers comply with the controller recommendation.

Scenario V (UE-based information): Here, route guidance is provided using the UE solution route proportions (UE-info). This is conceptually similar to Scenario III; UE
routes \((DK)\) are recommended by the controller to the drivers based on the proportion of drivers that are required to take each UE route. The UE route recommended to a driver may or may not belong to his/her driver-preferred route set. If the recommended route is not in the driver-preferred set, the driver completely ignores the information provided by the controller in the route choice decision-making process.

Scenario VI (behavior-consistent information): In this scenario, routes are recommended based on the behavior-consistent information-based network control strategy (BC-SO-info). Akin to Scenario IV, only controllable routes are recommended to the drivers. However, these recommendations are based on the proportion of drivers that must be recommended to take these routes so as to approach as close as possible to the time-dependent SO solution proportions. This can imply the controller recommending routes in higher or lower proportions than the corresponding SO solution proportions so that the actual proportions achieved after the driver decision-making process come close to the SO solution proportions.

3.5.2 Results and Analysis: Less Responsive Behavior

Figure 3.4 shows the percentage cumulative system travel time savings (over the horizon of interest) under the five information-based network control strategies for the less responsive behavior case relative to the base-case (NO-info) where no information is provided. By definition, Scenario II (SO DTA) has the highest cumulative system travel time savings. Hence, it represents the benchmark for comparing the performance under the other strategies.

The results show that all information strategies result in significant improvements to the system performance compared to the NO-info case. However, the behavior-consistent information-based network control strategy (BC-SO-info) outperforms both the SO-based information strategies (SO-info, SO-CK-info) and the UE-based information strategy (UE-info). By estimating drivers’ likely reactions and only recommending routes that are behavior-consistent, the controller is able to move the system closer (to the extent possible given driver behavior) to the ideal SO state. In
addition, there is a region of negative travel time savings in the early stages of the planning horizon. The negative values indicate that the base-case results in better system performance for the relevant duration. This implies that for some levels of demand and network dynamics, the SO-based and UE-based information strategies can potentially deteriorate system performance. Hence, the common practice of assigning the DTA solutions directly to the O-D demand may overestimate the system performance.

The significance of ensuring behavioral consistency in the controller-recommended routes is also reflected in Figure 3.5, where higher compliance rates are obtained for the BC-SO-info strategy compared to the other strategies. Figure 3.5 also illustrates that the compliance rates are perceptible even for the SO- and UE-based strategies (between 45% to 56%), though not as much as for the behavior-consistent strategy (around 65%). In all of the information-based network control strategies, the route recommended by the controller is considered by the drivers only if the recommended route is a driver-preferred route. Hence, the likelihood of choosing the recommended route is increased if it is also a driver-preferred route. In addition, the values for compliance rates indicate that there are many preferred routes ($PK$) that fully overlap with desired routes ($DK$). This may overestimate the performance of the SO- and UE-based strategies because they do not consider the likely response behavior to the route recommendations.

3.5.3 Results and Analysis: More Responsive Behavior

Figure 3.6 shows the percentage cumulative total travel time savings (over the horizon of interest) under the five information-based network control strategies for the more responsive behavior case relative to the base-case (NO-info). The results show that both the SO- and UE-based information strategies perform worse than even the NO-info strategy. This is consistent with the trends identified in previous studies involving system performance under large market penetration levels of personalized information provision.

In these experiments, the travel times are significantly increased because the controller is over-recommending routes to highly responsive drivers. There are some
controllable-routes with large numbers of drivers choosing them even when no information is given about them. This is because drivers are familiar with these routes, and favor them based on past experience. When the controller recommends these routes, they become even more attractive. If the proportions of drivers choosing these routes are higher than the SO proportions for these routes, they become congested leading to higher total travel times. This situation is circumvented under the behavior-consistent information-based network control strategy (BC-SO-info) because it takes into account the drivers’ likely reactions to the information strategies and hence does not over-recommend those routes. Hence, as seen in Figure 3.6, the behavior-consistent information-based network control strategy not only outperforms the SO- and UE-based strategies, but also significantly improves overall system performance. By estimating drivers’ likely reactions and only recommending routes that are behavior-consistent, the controller is able to move the system (to the extent possible under driver behavioral tendencies) in the SO direction. Hence, the BC-SO-info strategy improves performance and increases compliance rates (as seen in Figure 3.7).

In general, it should be noted that some drivers are likely to experience travel times that are longer than anticipated if they comply with the recommendations of strategies that are not behaviorally consistent (such as SO-info or UE-info strategies). This makes it less likely that they will comply with the controller recommendation in the long-term (Peeta and Yu, 2006).

3.6 Summary and Insights

From the controller’s perspective, ideally all drivers are equipped to receive personalized information, and follow the SO routes provided to them. However, such behavioral simplicity is not realistic. Different drivers may have different preferences in terms of route choice, and may have different responses to the same information. Further, drivers may have different levels of capabilities to receive information. This study is the first to develop a behavior-consistent approach for information-based network-level control. It explicitly factors the drivers’ likely response behavior while
determining the information that directs the system as close as possible to the SO solution. Thereby, the resulting information strategies address the controller and driver objectives simultaneously, and are more likely to be accepted by drivers.

The study experiments illustrate the benefits of the behavior-consistent information-based network control strategies. In all cases, the system travel time savings are significantly higher for the behavior-consistent approach compared to those of the no-information, the SO-based and UE-based information strategies. In addition, compliance rates are higher for the behavior-consistent strategy compared to those for the SO- and UE-based strategies. These insights suggest that factoring driver behavior while determining the controller route recommendations can further enhance performance as well as driver compliance. A detailed analysis of the results suggests that most of the preferred routes of the drivers tend to have large behavior-consistency gaps because large numbers of drivers take these routes independent of information provision. That is, to achieve the ideal route assignment percentages (whether UE or SO, obtained through the standard DTA approaches), the controller may have to recommend more or less users to take those routes depending on the network dynamics and driver behavior tendencies.

Under the proposed framework, only routes that are simultaneously desired by the controller and preferred by the drivers are recommended. Chapter 5 extends this framework by considering routes that overlap mostly, but not fully, with the controller-desired routes. It enhances deployment realism by providing additional routing options for the controller to recommend to drivers. Further, it analyzes behavior-consistent routes determined using the UE DTA solution as UE routes are more likely to overlap with driver-preferred routes.

In this chapter, the controller-estimated driver behavior model is calibrated off-line using the actual driver behavior model. Chapter 6 proposes a fuzzy on-line calibration model that fits within the solution framework of the behavior-consistent approach to calibrate the parameters of the controller-estimated model. It enables the simultaneous on-line determination of behavior-consistent information strategies and the calibration of the controller-estimated model parameters.
Figure 3.1 Conceptual framework for the behavior-consistent traffic routing problem under information provision
Figure 3.2 Solution framework for the behavior-consistent traffic routing problem under information provision
Figure 3.3  Borman expressway corridor network
Figure 3.4 Cumulative system travel time savings under the less responsive behavior benchmarked against the no-information case (base-case)
Figure 3.5  Compliance rates under less responsive behavior
Figure 3.6  Cumulative system travel time savings under the more responsive behavior benchmarked against the no-information case (base-case)
Figure 3.7 Compliance rates under more responsive behavior
4. FUZZY CONTROL MODEL OPTIMIZATION FOR BEHAVIOR-CONSISTENT TRAFFIC ROUTING UNDER INFORMATION PROVISION

4.1 Introduction

Fuzzy control through a rule-based mechanism can be used to determine behavior-consistent information-based control strategies for route guidance to robustly respond to the performance enhancement objectives of a system controller in a dynamic vehicular traffic system. Behavior-consistent (BC) strategies explicitly factor the likely driver response behavior to information provision in determining the controller-proposed route guidance strategies. In this context, fuzzy control provides a convenient mathematical handle to treat the uncertainty and vagaries associated with human decision-making in general, and additionally, the subjective interpretation of the linguistic attributes of the information provided. The fuzzy control model defines a fuzzy system that continuously seeks real-time information-based control strategies to improve the overall vehicular traffic system performance. Chapter 2 develops the fuzzy control model and Chapters 2 and 3 show its effectiveness.

The performance of a fuzzy system depends on its rule base and membership functions. The rule base is a collection of relations in the form of control if-then rules that are used to determine the control actions based on the current and desired system states. Mathematically, the membership functions are used to represent and handle the vagueness of inputs and their associated consequences on the outputs; and the operations of fuzzy sets facilitate building the logical framework for reasoning with variables that are vague in nature, such as language-based descriptors (e.g. congestion ahead, the error is negative large).
Given a rule base, the membership functions can be trained (optimized) to enhance the performance of the fuzzy system. The initial membership functions can be constructed based on experience, and they can later be trained to capture nonlinearities and modeling errors. Hence, trained membership functions result in better computational performance because the system can better respond to the nonlinearities and modeling errors. In the context of information-based real-time control of vehicular traffic systems, both nonlinearities and modeling errors are significant elements. Nonlinearities are a consequence of the intricate network-level interactions and dynamics as well as the human behavior component. Modeling errors are mainly a reflection of the imperfections and/or limitations underlying the driver behavior models, traffic flow models and the relationships between the control inputs and outputs. Nonlinearities and modeling errors are always present given the complexity of human behavior and the associated consequences on network interactions and dynamics.

Methods to train fuzzy membership functions can be broadly divided into derivative-based or derivative-free methods (Simon, 2005). Derivative-based methods include among others the gradient descent, Kalman filtering, simplex method, least squares, and backpropagation. Derivative-free methods range from genetic algorithms to heuristic methods.

H-infinity and Kalman filters are derivative-based methods that can be used to estimate and optimize system states that cannot be observed directly. The Kalman filter works well only under certain characteristics for the noise process. The process and measurement noises in the system need to have zero means. This zero mean property must hold at each time instant and across the entire time history of the process. That is, the expected value of noise at each time instant must equal zero. From a data needs standpoint, the Kalman filter requires knowledge of the distribution of the noise processes, and uses the associated covariance matrices as design parameters. The attractiveness of the Kalman filter is that it is the minimum variance estimator; it results in the smallest possible standard deviation of the estimation error.
The H-infinity (Minimax) filter assumes that nature is throwing the worst possible noise at the estimator, which can be limiting if untrue. Thereby, it minimizes the worst case estimation error. The optimization of fuzzy membership function parameters involves high levels of nonlinearities; the H-infinity filter has been shown to provide better results than the Kalman filter in this context (Simon and El-Sherief, 1996; Simon, 2001). This is because the H-infinity filter is often more robust than the Kalman filter to system noise, modeling errors and nonlinearities (Simon and El-Sherief, 1996; Simon, 2001). In our problem context, these characteristics makes the H-infinity filtering approach suitable to train the fuzzy membership functions that are used by an information-based fuzzy control model to improve the performance of a vehicular traffic system.

Given the difficulties in obtaining data on the covariance matrices for the Kalman filter in our context, and the more restrictive nature of its assumptions for the noise processes, this study proposes an H-infinity filtering approach to optimize the proposed fuzzy control model. By adjusting membership function parameters to minimize the worst case estimation error to better respond to nonlinearities and modeling errors, the approach is able to enhance the computational performance of the fuzzy control model. Results from experiments suggest that the optimized fuzzy control model determines the behavior-consistent information-based control strategies using significantly less computational time than the default fuzzy control model. This is synergistic with the real-time route guidance problem as information strategies are required in sub-real time to be deployable.

The remainder of this chapter is organized as follows. Section 4.2 describes the problem. Section 4.3 expands on important details of the fuzzy control model to explicitly identify the parameters to optimize. Section 4.4 discusses the H-infinity methodology proposed to address the problem. Section 4.5 describes the study experiments and analyzes the results. Section 4.6 presents some concluding comments.
4.2 Problem Description

The problem addressed here focuses on optimizing the computational efficiency of a fuzzy control model in which a controller seeks to improve the performance of a vehicular traffic system via real-time behavior-consistent information provision to drivers. It involves the optimization of the membership function parameters used by the controller to determine the information strategies. This leads to a faster convergence to the information strategies to provide to drivers.

Section 3.4 discusses the solution framework for the broader problem, the behavior-consistent traffic routing problem under information provision. The problem addressed in this chapter is a sub-problem of the broader problem, and consist on the off-line parametric optimization of the fuzzy control model to enhance on-line computational efficiency. Section 2.5 and Figure 2.4 provide the details of the iterative search based optimization procedure involving the fuzzy control model and a controller-estimated driver behavior model for determining the information strategies. While the iterative search occurs within a rolling horizon stage $\sigma$ as discussed in Section 3.4, the time dimension associated with the discrete time intervals within this stage is ignored hereafter without loss of generality to simplify the notation.

4.3 Fuzzy Control Model

The fuzzy control model is explained in detail in Section 2.5.1. Section 4.3.1, provides additional details required to explicitly identify the membership function parameters. Section 4.3.2 describes the decision process of the fuzzy control model using the parameters identified in Section 4.3.1.

4.3.1 Membership Functions

The development of the default membership functions is explained in detail in Section 2.5.1.3.2. Figure 4.1 shows an example of the sets of membership functions associated
with the control if-then rules. The top part of the figure depicts the default membership functions which are identical for all O-D pairs. The bottom part depicts an example of an optimized set of membership functions for a specific controllable route for an O-D pair.

Consider the $g^{th}$ fuzzy membership function of the $f^{th}$ input $z_{kf}$ for route $k$ (here, $z_{k1} = e_k^n$ and $z_{k2} = \Delta e_k^n$). Its modal point is denoted as $c_{kgf}$, its lower half-width as $b^-_{kgf}$, and its upper half width as $b^+_{kgf}$. The membership function is equal to 1 when the input is $c_{kgf}$. As the input increases or decreases from $c_{kgf}$, the membership function value decreases linearly to 0 at $c_{kgf} + b^+_{kgf}$ and $c_{kgf} - b^-_{kgf}$, respectively. For input values less than $c_{kgf} - b^-_{kgf}$ or higher than $c_{kgf} + b^+_{kgf}$ the membership function is equal to 0. Hence, the degree of membership of the $f^{th}$ crisp input for route $k$, $z_{kf}$, in its $g^{th}$ fuzzy set is given by:

$$\mu_{kgf}(z_{kf}) = \begin{cases} 
1 + (z_{kf} - c_{kgf})/b^-_{kgf} & \text{if } -b^-_{kgf} \leq (z_{kf} - c_{kgf}) \leq 0 \\
1 - (z_{kf} - c_{kgf})/b^+_{kgf} & \text{if } 0 \leq (z_{kf} - c_{kgf}) \leq b^+_{kgf} \\
0 & \text{otherwise.}
\end{cases}$$

The fuzzy control model has two outputs for each route $k$, the prescriptive ($\theta_k$) and descriptive ($\phi_k$) information strategies. Prescriptive information corresponds to route recommendations for individual drivers while descriptive information corresponds to qualitative descriptions of routes (linguistic variables) provided through mass dissemination devices.

The numbers of rules corresponding to the prescriptive and descriptive information are $RP$ and $RG$, respectively. Hence, there are a total of $R=RP+RG$ rules in our fuzzy system. The consequent of the $\alpha^{th}$ rule for output $\omega$ for route $k$ is a triangular fuzzy set with modal point $d_{k\omega\alpha}$, lower half-width as $\beta^-_{k\omega\alpha}$, and upper half width $\beta^+_{k\omega\alpha}$. Hence, the fuzzy set of the consequent of the $\alpha^{th}$ rule for route $k$ for output $\omega$ is given by:
where \( \omega \) takes value 1 for prescriptive information and value 2 for descriptive information.

### 4.3.2 Decision Process

The decision process is explained in detail in Section 2.5.1.3.3. The centroid of \( \mu_{\Delta\theta_k^{\alpha\eta}} \), the fuzzy outcome for change in prescriptive information for route \( k \) in iteration \( \eta \) obtained by fuzzyfing the inputs using rule \( \alpha \), is defined as:

\[
\bar{\theta}_k^\alpha = \frac{\int y \cdot \mu_{\Delta\theta_k^{\alpha\eta}}(y) \cdot dy}{\mu_{\Delta\theta_k^{\alpha\eta}}(y) \cdot dy} \quad \forall \ k \in CK, \ \alpha
\]

where \( y \) represents the domain of the membership function. After substituting (4.2) into the above equation, the centroid of each fuzzy set for prescriptive information can be defined in terms of its membership function parameters as follows:

\[
\bar{\theta}_k^\alpha = \frac{\beta_{k1}^+ \cdot (3 \cdot d_{k1} + \beta_{k1}^-) + \beta_{k1}^- \cdot (3 \cdot d_{k1} - \beta_{k1}^-)}{3 \cdot (\beta_{k1}^+ + \beta_{k1}^-)} \quad \forall \ k \in CK, \ \alpha
\]

Similarly, the centroid of each fuzzy set for descriptive information can be defined as follows:

\[
\bar{\varphi}_k^\alpha = \frac{\beta_{k2}^+ \cdot (3 \cdot d_{k2} + \beta_{k2}^-) + \beta_{k2}^- \cdot (3 \cdot d_{k2} - \beta_{k2}^-)}{3 \cdot (\beta_{k2}^+ + \beta_{k2}^-)} \quad \forall \ k \in CK, \ \alpha
\]
Thus, the outputs of the fuzzy control model are completely defined in terms of the crisp inputs and the parameters of the membership functions, as required for the optimization approach discussed in Section 4.4.

4.4 Fuzzy Control Model Optimization Via H-infinity Filtering

4.4.1 Error Function for Optimization

The optimization of the membership function parameters of the fuzzy control model can be viewed as a weighted least-squares error minimization problem, where the error vector is the difference between the controller-estimated and SO proportions of drivers taking routes. This is done by using the derivatives of the weighted least-squares error function ($\Omega$) to solve the optimization problem. The expressions for these derivatives with respect to the half-widths and modal points of the membership functions can be obtained using Equations (4.1)-(4.6). The derivative formulas for this type of error functions are given in (Simon, 2002). The multidimensional error function is defined as follows:

$$\Omega_k = \frac{1}{2 \cdot W} \sum_{\sigma=1}^{W} \sum_{\eta=1}^{N} \lambda^\eta \left( E_k^{\rho(\sigma)\eta} (\theta_k^\eta, \phi_k^\eta) - SO_k^{\rho(\sigma)} \right)^2 \quad \forall \ k \in CK \quad (4.6)$$

where $W$ is the number of stages of the rolling horizon approach, $N$ is the number of iterations in a stage that the fuzzy control model uses to determine the information strategies, $E_k^{\rho(\sigma)\eta}$ are the controller-estimated proportions of drivers taking route $k$ in the roll period $\rho(\sigma)$ of stage $\sigma$ based on the information strategies $(\theta_k^\eta, \phi_k^\eta)$ in iteration $\eta$ of the search procedure, and $\lambda^\eta$ is a weighting function. $\lambda^\eta$ is defined here as $N/\eta$ and weighs the initial iterations higher to induce faster convergence.
4.4.2 H-infinity Filtering

As will be discussed hereafter, we apply the H-infinity filter to a nonlinear system defined by the membership function parameters of the fuzzy control model. The controller-estimated proportions of drivers taking routes constitute the output from this nonlinear system.

Consider the nonlinear time-invariant finite dimensional discrete system defined by the inputs and outputs of the fuzzy membership functions of the fuzzy control model:

\[ X_{k}^{m+1} = f_{X}(X_{k}^{m}) + B \cdot w_{k}^{m} + \delta_{k}^{m} \quad \forall \ k \in CK \]  \hspace{1cm} \text{(4.7)}

\[ SO_{k}^{\alpha(\sigma)} = h_{X}(X_{k}^{m}) + \nu_{k}^{m} \quad \forall \ k \in CK \]  \hspace{1cm} \text{(4.8)}

where \( m \) represents an iteration \((m=1,...,M)\) in the recursive process used to solve the filter (also called the “training” iteration), the vector \( X_{k}^{m} \) is the state of the system for route \( k \), \( f_{X}(\cdot) \) is the identity mapping, \( h_{X}(\cdot) \) is the nonlinear mapping between the fuzzy control model membership function parameters and its output in terms of the controller-estimated proportions of drivers \((E_{k}^{\alpha(\sigma)})\) taking route \( k \), \( B \) is a tuning parameter, \( w_{k}^{m} \) and \( \nu_{k}^{m} \) are white noise sequences, and \( \delta_{k}^{m} \) is an arbitrary noise sequence.

The augmented noise vector \( \varepsilon_{k}^{m} \) and the estimation error \( \tilde{x}_{k}^{m} \) are defined as follows:

\[ \varepsilon_{k}^{m} = [w_{k}^{m} \quad \nu_{k}^{m}]^{T} \quad \forall \ k \in CK \]  \hspace{1cm} \text{(4.9)}

\[ \tilde{x}_{k}^{m} = X_{k}^{m} - \hat{X}_{k}^{m} \quad \forall \ k \in CK \]  \hspace{1cm} \text{(4.10)}

We now develop the expression for the system state. In Figure 4.1, the fuzzy control model has 5 fuzzy sets for each input and each output. Following the definitions in Section 4.3.2, the state of the nonlinear system is expressed as:
\[ X_k = \begin{bmatrix} b^{-}_{k11} & b^{+}_{k11} & c_{k11} & \cdots & b^{-}_{k51} & b^{+}_{k51} & c_{k51} \\ b^{-}_{k12} & b^{+}_{k12} & c_{k12} & \cdots & b^{-}_{k52} & b^{+}_{k52} & c_{k52} \\ \end{bmatrix} \forall k \in CK \] (4.11)

The H-infinity filter finds an estimate \( \hat{x}_k^m \) such that the infinity norm of the transfer function \( G \) from the augmented noise vector \( \varepsilon \) to the estimation error \( \bar{x} \) is bounded by a user-defined quantity \( V \).

\[ \| G_{\bar{x}} \|_\infty < V \] (4.12)

The estimate \( \hat{x}_k^m \) is obtained through the following recursive H-infinity estimator (Simon, 2005; Yaesh and Shaked, 1992).

\[
F_k^m = \frac{\partial f_X(X)}{\partial X} \bigg|_{X=\hat{x}_k^m} \\
H_k^m = \frac{\partial h_X(X)}{\partial X} \bigg|_{X=\hat{x}_k^m} \\
Q_k^m (I - [H_k^m]^T \cdot H_k^m \cdot P_k^m) = (I - Q_k^m / B_2^2) P_k^m \\
Q_{k+1}^m = F_k^m \cdot P_k^m \cdot [F_k^m]^T + B_1 \cdot B_1^T \\
K_k^m = F_k^m \cdot P_k^m \cdot [H_k^m]^T \\
\hat{x}_{k+1}^m = F_k^m \cdot \hat{x}_k^m + K_k^m (\Theta_k^{\rho(\sigma)} - h_X(\hat{x}_k^m))
\] (4.13)

where \( B1 \) and \( B2 \) are tuning parameters proportional to the magnitudes of the artificial noise processes. \( Q \) and \( P \) are assumed to be nonsingular sequences of matrices. \( K \) is known as the H-infinity gain. \( Q_0^m \) is the initial state covariance matrix. \( H \) is the partial derivative of the error function (\( \Theta \)) with respect to the membership function parameters, and \( F \) is the identity matrix.
After obtaining the $\hat{X}_k^n$, they are used by the fuzzy control model to determine the information strategies ($\theta_k, \phi_k$). In summary, tuning the parameters of the fuzzy control model leads to faster convergence to the information strategies.

### 4.4.3 Recursive Solution Scheme

Figure 4.2 illustrates conceptually the implementation of the recursive approach used to solve the H-infinity filter discussed heretofore. It consists of an inner loop that is an extended version of the iterative search procedure illustrated in Figure 2.4.

The additional component is the step used to calculate the partial derivatives of the error function ($\Omega$) with respect to the current estimated vector $\hat{X}_k^n$ of membership function parameters. After the extended search procedure is completed (when $\eta = N$), updated values of the derivatives are obtained which are then used in the H-infinity estimator (4.13) of the outer loop to determine new values for the fuzzy membership function parameters. This recursive scheme of the outer loop is repeated until $M$ training iterations are completed. The outcome of the recursive scheme is the set of optimized membership function parameters.

### 4.5 Experiments

Two sets of experiments, one at the O-D pair level and the other at the network level, are conducted to illustrate the advantages of using optimized fuzzy control model parameters to determine the information-based control strategies. In all experiments, it is assumed that all drivers have capabilities to receive personalized information.
4.5.1 Experimental Setup

The general experimental setup follows the design in Sections 2.6.1.2-2.6.1.4 and Sections 3.5.1.1-3.5.1.5. The convergence criteria used to terminated the iterative search based optimization procedure are as indicated in Section 2.5.2.

4.5.2 Experiments: O-D Level

In these experiments, a single O-D pair is used to illustrate methodological insights of the proposed H-infinity filter based optimized fuzzy control model. Figure 2.7 illustrates the selected O-D pair as well as the four driver-preferred routes (zigzag lines) and the four controller-desired routes (dashed lines) that connect this O-D pair, where only three of them fully overlap. The controller seeks to achieve the SO proportions only for the set of controllable routes (the three routes that fully overlap). Thus, routes 1, 2 and 3 are defined as the controllable routes, with the corresponding SO proportions being 44%, 33%, and 11%, respectively.

Figure 4.3 depicts the progress of the training of the fuzzy membership function parameters for this O-D pair using the H-infinity filtering approach. At the end of the process, the filter is able to capture a significant portion of the error. The consequence of this is illustrated in Figure 4.4 where the trajectories of the controller-estimated proportions of drivers choosing routes are depicted under the optimized and default search procedures. In the first iteration, the controller recommends routes based on the SO proportions. Each iteration of the search procedure seeks information strategies (behavior-consistent proportions) so that the controller-estimated proportions of drivers choosing controllable routes gets closer to the corresponding SO proportions. When the search procedure converges, the corresponding behavior-consistent proportions are recommended to drivers. The results show a significant reduction in the number of iterations required to achieve convergence under the optimized fuzzy control model. In this example, the convergence of the optimized search procedure requires only 8 iterations while the default search procedure requires 15 iterations. It represents an improvement of more than 45%.
These experiments evaluate the performance and effectiveness of the H-infinity filter at the network level. They are conducted using the broader framework of the behavior-consistent real-time traffic routing problem illustrated in Figure 3.2. In the experiments, the stage length is 20 minutes and the roll period is 5 minutes.

Figure 4.5 depicts the computational savings under the optimized parameters relative to the default search procedure for various numbers of O-D pairs. It emphasizes the significant value in using optimized parameters. The computational requirements increase with the number of controllable routes, the length (number of nodes) of the routes, and the associated volumes. Hence, the number of O-D pairs is not a direct proxy for computational load. This aspect is illustrated in the figure, where the rate of increase in computational savings is higher initially as the associated O-D pairs tended to be mostly connected by controllable routes and had higher traffic volumes.

Figure 4.6 shows the cumulative system travel time savings under the ideal SO DTA and the behavior-consistent real-time traffic routing (BC-SO-info) strategies relative to the no-information scenario. By definition, the SO DTA strategy has the highest cumulative system travel time savings. Hence, it represents the benchmark for comparing the performance of other strategies. The BC-SO-info strategy results in significant improvements to the system performance relative to the no-information scenario. This figure is shown here to indicate that the use of the H-infinity filter does not adversely affect the capability of the (optimized) fuzzy control model to determine behavior-consistent traffic routing strategies, and results in savings identical to those under the default fuzzy control model.

An important aspect of a filtering approach is the need to fine-tune its model parameters. Fine-tuning of parameters is still an open area for research (Simon, 2005). The parameters of the H-infinity estimator (4.13) that are subject to fine-tuning are $B_1$ and $B_2$. Figures 4.7a and 4.7b show the cumulative system travel time savings obtained when the fuzzy control model uses membership parameters that have been optimized using different values for $B_1$ and $B_2$, respectively. As these figures illustrate, system
performance depends highly on \( B1 \) and \( B2 \). This implies that different values for these parameters may lead to different values for the optimized membership function parameters, and some of them can lead to less effective information strategies. Further, some values for \( B1 \) and \( B2 \) can lead to erroneous information strategies that may deteriorate system performance. Hence, caution must be taken when optimizing these parameters to prevent inconsistent information strategies. This can be achieved by testing for relative improvements in system performance using the default and optimized parameters.

4.6 Summary and Insights

This research shows the effectiveness of the proposed off-line H-infinity filtering scheme to optimize the membership function parameters of the fuzzy control model used in the approach to generate behavior-consistent traffic routing strategies for real-time deployment. The optimized fuzzy membership function parameters can better respond to nonlinearities and noise uncertainty, leading to enhanced computational performance. This ability to significantly improve computational efficiency is critical from a practical standpoint because the information strategies are required in sub-real time under the rolling horizon deployment approach. Although the H-infinity filtering methodology can provide significant benefits in this problem context, the fine-tuning of its model parameters is a critical aspect. There is potential for significant computational savings using optimized membership function parameters.
Figure 4.1  An example of default and optimized membership functions
Figure 4.2  Filter-based optimization of fuzzy control model parameter
Figure 4.3  Training progress of the filtering approach
Figure 4.4 Trajectory of the controller-estimated proportions of drivers choosing routes for the default and optimized fuzzy control models
Figure 4.5  Computational savings relative to the default fuzzy control model
Figure 4.6  Cumulative system travel time savings benchmarked against the no-information scenario
Figure 4.7 Cumulative system travel time savings benchmarked against the no-information scenario: (a) parameter $B_2$ varies, (b) parameter $B_1$ varies.
5. DEPLOYMENT PARADIGMS FOR BEHAVIOR-CONSISTENT REAL-TIME TRAFFIC ROUTING UNDER INFORMATION PROVISION

5.1 Introduction

The benefits of real-time traffic network control through information provision using ATIS hinge on the controller’s ability to identify effective routing strategies that entail high levels of acceptability by drivers. Current efforts to deploy information provision strategies are primarily concentrated under the umbrella of DTA models. However, the behavioral foundations of most DTA models are idealistic and insufficient to address real-world driver behavior (Peeta and Ziliaskopoulos, 2001). This is primarily because existing DTA models are not behavior-consistent; they do not realistically factor the drivers’ likely response towards information while generating these strategies. They mostly pre-specify driver behavior. Some assume artificial compliance rates to predict traffic conditions or generate control strategies. Others use the DTA solution route assignment proportions “as is” for route guidance, and use a feedback loop or a consistency-checking procedure to correct for prediction errors. Thereby, most approaches do not have interactive linkages between route recommendations and driver response. Peeta and Yu (2006) propose a consistency-seeking procedure that updates behavior model parameters in an operational context based on unfolding field conditions. However, it is also reactive and does not entail a behavior-consistent paradigm. In summary, DTA models do not simultaneously consider network flow interactions and behavior realism to develop meaningful information-based network control strategies.
To address the behavioral realism gap of traditional DTA models vis-à-vis determining the time-dependent traffic flow patterns, Chapter 2 and 3 propose a behavior-consistent approach to determine and deploy real-time information-based network control strategies. It determines the information strategies by explicitly accounting for the drivers’ likely response to these strategies while determining them. This implies solving a fixed-point problem that arises because the information strategies depend on driver behavior and vice versa. The proposed approach enhances system performance while being consistent with driver behavior. It also has reduced sensitivity to data needs as it is based on aggregate if-then rules that preclude the need for information at the individual driver level. These rules relate the route choice decisions to the routes characteristics, the driver attributes in terms of information availability, and level of responsiveness to the information strategies. As drivers are likely to use simple rules and/or a few factors (Nakayama and Kitamura, 2000; Peeta and Yu, 2005) to make on-line routing decisions due to the associated time constraints, the aggregate if-then rules consist of simple and straightforward one-dimensional left hand side and right hand side components (Paz and Peeta, 2008).

The behavior-consistent approach implements a control mechanism that continuously directs the traffic system towards a desired system state through information provision. That is, the controller directs the system towards a particular objective such as the time-dependent system optimal (SO) state. Thereby, the controller may need to recommend routes for an origin-destination (O-D) pair to more or less drivers than suggested by the SO DTA solution so as to achieve close to SO route proportions. This is done using a controller-estimated if-then rules based driver behavior model. Further, the approach uses the concept of controllable routes to enhance behavior consistency whereby the route recommended to a driver belongs to the controller’s SO (desired) route set and the preferred route set for that driver. This increases the likelihood of the recommended route being accepted by the driver. It also circumvents a key practical concern that potentially arises for ATIS-based information provision. That is, while some researchers have advocated that drivers could be persuaded to use SO
routes, others (such as Hall, 1996) stress the value of “honest” information, and that in the long run drivers will resist SO routes that are not user optimal.

While the notion of controllable routes enhances behavior consistency, it may entail practical limitations. For example, it is possible that an O-D pair may not have a controllable route as no controller-determined SO route coincides with a driver-preferred route. This motivates the consideration of alternative definitions for controllable routes to enable the deployment effectiveness of the behavior-consistent approach. In this paper, alternative controllable route paradigms are proposed that entail significant overlap of the controller-determined SO routes with the driver-preferred routes, but do not require perfect match. This enables the controller to recommend driver-preferred routes that are not necessarily SO routes, as well as target drivers who do not consider SO routes. At a more basic level, such a study can shed light on the interplay between route quality relative to controller objectives and driver real-time route choice decisions.

By definition, the SO solution entails some long routes which may imply fewer common routes with the driver-preferred set. While the alternative degree of overlap paradigm represents one mechanism to increase the controllable route set, another strategy is to use the UE solution as the controller’s objective. This is because UE routes have a more defensible behavioral rationale, possibly having a greater degree of commonality with the driver-preferred route set. In this study, we compare the performance of the behavior-consistent approach under the UE and SO objectives. It should be noted here that the commonly cited advantages of the UE paradigm over the SO benchmark for standard DTA models do not necessarily apply for the behavior-consistent approach. Since the behavior-consistent approach provides a trajectory to approach the desired system state in a manner consistent with individual driver routing decisions, the limitations arising from the behavioral underpinnings of the standard SO strategy relative to the UE strategy are obviated. That is, the compliance rates under the behavior-consistent approach are perceptibly higher than under the standard DTA paradigms. Further, the relative gap in compliance rates between UE and SO under the behavior-consistent approach tends to be smaller than under the standard DTA approach. It suggests that focusing on the SO paradigm can represent a legitimate deployment
alternative with better behavior-consistent performance, rather than the UE centric focus of the current literature based on the behavior rationale. This aspect is analyzed in depth in this chapter.

A long-term phenomenon vis-à-vis driver behavior under information-based traffic routing is the influence of learning effects on driver response. Peeta and Yu (2005) show that several information-related phenomena can manifest over time based on past driving experience and the experience with the provided information. These include familiarity, trust in information, inertia, delusion, freezing, etc. Vaughn et al. (1993) and Bonsall and Joint (1991) present evidence that drivers may not comply with information perceived to be inaccurate. Over time, these effects and experiences can lead to changes in the set of routes preferred by a driver. Nakayama and Kitamura (2000) show that drivers may ignore routes associated with poor travel experience and remove them from their preferred route sets. By contrast, it is also possible that a driver may add new alternatives to his/her preferred route set based on positive experiences with a controller-recommended or a newly-explored route. Hence, the driver-preferred route set can potentially change over time.

The number of routes in the driver-preferred route set is significantly influenced by the driver’s network familiarity. Familiar drivers are likely to have larger preferred route sets compared to unfamiliar drivers. While this research does not consider a day-to-day learning framework, we explore the effect of increasing the driver-preferred route set with alternative route type paradigms, and compare the performance of these paradigms from a deployment perspective.

The remainder of this chapter is organized as follows. Section 5.2 defines relevant terms. Section 5.3 describes the alternative controllable route paradigms proposed in this study. Section 5.4 discusses experiments and analyzes their results. Section 5.5 presents some concluding comments.
5.2 Definition of Terms

While the behavior-consistent approach addresses a real-time problem using a stage-based procedure, the time dimension is ignored hereafter without loss of generality to simplify the notation.

Degree of Overlap ($DOV$): The degree of overlap $DOV_{ijk}$ for a driver-preferred route $k$ ($k \in PK_{ij}$) from origin $i$ to destination $j$, is a fraction defined by the maximum amount of common link length between that route and any controller-desired route $m$ ($m \in DK_{ij}$) divided by the length of the driver-preferred route ($k$):

$$DOV_{ijk} = \frac{1}{L_{ijk}} \left[ \max_{m \in DK_{ij}} \sum_{a \in \Gamma_{ijk}} l_a \cdot \Theta_{aijm} \right] ; \; \forall \; i, j, k \in PK_{ij}$$ (5.1)

where

- $\Gamma_{ijk}$: set of links on route $k$ connecting O-D pair $ij$, $k \in PK_{ij}$
- $l_a$: length of link $a$
- $L_{ijk}$: length of route $k$, $k \in PK_{ij}$
- $\Theta_{aijm}$: link-route incidence dummy; 1 if route $m$ connecting O-D pair $ij$ includes link $a$, and 0 otherwise

Threshold degree of overlap ($\lambda$): Minimum degree of overlap at which a driver-preferred route ($k \in PK$) is accepted as a controllable route ($k \in CK$).

5.3 Control and Controllable Route Paradigms

As stated in Section 5.1, this study focuses on identifying mechanisms to increase the controllable route set to enhance deployment effectiveness of the behavior-consistent approach. Another positive outcome of increasing the controllable route set is that it diffuses the effects of errors in estimating the driver-preferred route sets. This is because the controller has more options to recommend and the likelihood of recommending a
route that does not belong to the actual driver-preferred set decreases. The paradigms discussed hereafter seek to increase the controllable route sets relative to the SO-based behavior-consistent approach.

5.3.1 SO and UE Control Paradigms

Under the SO paradigm developed in Chapters 2 and 3, the controller seeks to direct the system towards the SO DTA state. Hence, for O-D pair \( ij \), the controller-desired route set is its time-dependent SO route set \( SOK_{ij} \), and the controllable route set is the subset of driver-preferred routes for that O-D pair across all drivers that perfectly match routes in \( SOK_{ij} \). This paradigm, labeled as BC-SO-info, is expressed as follows:

\[
k \in CK_{ij} \iff k \in \{SOK_{ij} \cap PK_{ij}\}
\]

(5.2)

The experiments in Chapter 3 show that the behavior-consistent approach using the SO control paradigm results in better performance than traditional DTA-based approaches while being behaviorally realistic.

Under the UE control paradigm, the controller seeks to direct the system towards the UE DTA state. In this case, for O-D pair \( ij \), the controller-desired route set is its time-dependent UE route set \( UEK_{ij} \). The controllable route set is the subset of driver-preferred routes for that O-D pair across all drivers that perfectly coincide with routes in \( UEK_{ij} \). This paradigm, labeled as BC-UE-info, is expressed as follows:

\[
k \in CK_{ij} \iff k \in \{UEK_{ij} \cap PK_{ij}\}
\]

(5.3)

As stated in Section 5.1, the behavioral underpinnings of UE routes typically make them more likely to overlap with driver-preferred routes, leading to a potential increase in the controllable route set compared to the SO control paradigm. However, as illustrated in Section 5.4, this does not necessarily translate into better system performance. Hence,
the controllable route paradigms proposed in Section 5.3.2 are based on the SO control paradigm.

5.3.2 Controllable Route Paradigms

5.3.2.1 Degree of Overlap Paradigms

These paradigms use the DOV to define the controllable route sets.

5.3.2.1.1 1st DOV Paradigm

The 1st DOV paradigm is called the “full overlap paradigm”. This is an all-or-nothing approach where only driver-preferred routes that fully overlap (match) with controller-desired routes are classified as controllable routes. That is, only driver-preferred routes with a degree of overlap equal to 1 are classified as controllable. This paradigm is expressed as follows:

\[ k \in CK_{ij} \iff DOV_{ijk} = 1 \] (5.4)

As discussed in Section 5.1, Chapter 3 uses this paradigm to analyze the performance of the behavior-consistent approach. However, the strict match requirement can preclude the consideration of “good” route alternatives in the driver-preferred route sets that significantly overlap with the controller-desired routes, potentially limiting deployment effectiveness. This represents the motivation for the 2nd and 3rd DOV paradigms. Under these paradigms, the controller uses threshold DOV related rules to treat appropriate driver-preferred routes as controller-desired routes, and consequently as controllable routes.
5.3.2.1.2 2nd DOV Paradigm

The 2nd DOV paradigm is labeled the “threshold degree of overlap paradigm”. Here, a threshold DOV value is pre-specified and only driver-preferred routes with DOV values greater than or equal to this threshold are classified as controllable. This paradigm can be represented as follows:

\[ k \in CK_{ij} \iff DOV_{ijk} \geq \lambda \]  \hspace{1cm} (5.5)

This paradigm potentially provides more deployment options to the controller when compared to the 1st DOV paradigm. However, it precludes the consideration of the effects of alternative combinations of controllable routes beyond the combination that satisfies the pre-specified threshold. Presumably, other combinations could result in a more favorable performance. This motivates the next DOV paradigm.

5.3.2.1.3 3rd DOV Paradigm

The 3rd DOV paradigm is called the “combination degree of overlap paradigm”. It uses various threshold DOV values and an error function to identify the set of controllable routes. The error function computes the total error \( TE_{ij} \) for O-D pair \( ij \), defined as the summation over the set of controllable routes of the absolute difference between the controller-desired proportion of drivers \( SO_{ijk} \) choosing those routes and the corresponding controller-estimated proportion of drivers \( E_{ijk} \) obtained at the convergence of the iterative search procedure. In the iterative search procedure described in Section 5.2.1, the controller-estimated driver behavior model is used to compute \( E_{ijk} \) for each iteration. Hence, \( E_{ijk} \) is the result of estimating individual route choices over the set of driver-preferred routes. \( TE_{ij} \) can be expressed as:

\[ TE_{ij} = \sum_{k \in CK_{ij}} |SO_{ijk} - E_{ijk}| \]  \hspace{1cm} (5.6)
Under this paradigm, two threshold $DOV$ values, $\lambda_L$ and $\lambda_U$ are pre-defined. $\lambda_L$ is a lower threshold value and $\lambda_U$ is an upper threshold value. These threshold values and the total error are used as part of a systematic three-step procedure to identify the controllable route set:

**Step 1:** Driver-preferred routes that perfectly match ($DOV=1$) controller-desired routes are classified as controllable, and represent the initial set of controllable routes $CK^1_{ij}$ for O-D pair $ij$. This is equivalent to the $1^{st}$ $DOV$ paradigm:

$$k \in CK^1_{ij} \iff DOV_{ijk} = 1$$  (5.7)

**Step 2:** In this step, at most one additional driver-preferred route is identified as controllable using the heuristic rules of Equation (5.8). That is, $CK^2_{ij}$, an updated set of controllable routes has at most one route more than $CK^1_{ij}$. This additional route, if it exists, is the one with the lowest $TE_{ij}$ among all driver-preferred routes not in $CK^1_{ij}$ which also has a $DOV$ value greater than or equal to $\lambda_U$. In Equation (5.8), $CK^1_{ij}^*$ denotes the complement of the set $CK^1_{ij}$.

$$k \in CK^2_{ij} \iff (k \in CK^1_{ij}) \cup \{ k \mid (DOV_{ijk} \geq \lambda_U) \cap [TE_{ij}(CK^1_{ij} \cup k) \leq TE_{ij}(CK^1_{ij} \cup m)$$

$$\forall k, m \in (PK_{ij} \cap CK^1_{ij}^*) \}$$  (5.8)

**Step 3:** As in Step 2, at most one additional driver-preferred route is identified as controllable using the heuristic rules of Equation (5.9). That is, $CK_{ij}$, the final set of controllable routes identified using the $3^{rd}$ $DOV$ paradigm has at most one route more than $CK^2_{ij}$. This additional route, if it exists, is the one with the lowest $TE_{ij}$ among all
driver-preferred routes not in $C K^2_0$ which also has a DOV value greater than or equal to $\lambda_L$.

\[ k \in C K_0 \iff (k \in C K^2_0) \cup \{ k \mid (DOV_{ijk} \geq \lambda_L) \cap \left[ TE_{ij}(C K^2_0 \cup k) \leq TE_{ij}(C K^2_0 \cup m) \right. \left. \forall k, m \in (PK_{ij} \cap C K^2_0^\prime) \} \} \] (5.9)

The last two steps of this paradigm partly include implementing the 2nd DOV paradigm for two different threshold values. Since, in each of these steps, the route possibly added to the controllable route set has the smallest total error among all routes that could be considered, this paradigm is more sensitive to the impact on the total error of adding a route.

5.3.2.2 Route Type Paradigms

These paradigms increase the controllable route set by adding a route according to a specific route type rule.

5.3.2.2.1 1st Route Type Paradigm

This paradigm is labeled the “maximum SO route type paradigm”. The controller-desired route to be added to the driver-preferred route set is that route, if it exists, with the highest SO proportion that does not belong to the initial driver-preferred route set $PK^r_{ij}$. The driver-preferred route set according to this paradigm is denoted by $PK^r_{ij}$. It is expressed as follows:

\[ k \in PK^r_{ij} \iff \{(k \in PK^r_{ij}^0) \cup \{ k \mid (SO_{ijk} \geq SO_{ijm} \forall m \in DK_{ij}, m \notin PK^r_{ij}^0) \cap (k \notin PK^r_{ij}^0) \} \} \] (5.10)
The motivation for this paradigm is that by adding a SO route with high routing proportion to the driver-preferred set, the system performance can potentially be enhanced. However, due to the nature of the SO solution, there is a possibility that the added route may be significantly longer than those in $PK_{ij}^{r^0}$, and drivers may ignore it. This is the motivation for the next paradigm.

5.3.2.2.2 2nd Route Type Paradigm

This paradigm is called the “maximum degree of overlap route type paradigm”. It is conceptually similar to the second route type paradigm, but adds the controller-desired (SO) route that does not belong to $PK_{ij}^{r^0}$ and has the highest associated $DOV$ with a driver-preferred route. It can be expressed as follows:

$$k \in PK_{ij}^r \iff \{(k \in PK_{ij}^{r^0}) \cup \left[k \mid (DOV_{ijk} \geq DOV_{ijm} \ \forall k, m \in DK_{ij}) \cap (k \notin PK_{ij}^{r^0})\right]\}$$  \hspace{1cm} (5.11)

Akin to the first two $DOV$ paradigms, this paradigm does not use driver behavior to determine the enhanced controllable route sets. This represents the motivation for the next paradigm.

5.3.2.2.3 3rd Route Type Paradigm

This paradigm is labeled the “combination route type paradigm”. Similar to the 3rd $DOV$ paradigm, this paradigm adds the controller-desired route that does not belong to $PK_{ij}^{r^0}$ and results in the lowest $TE_{ij}$. This paradigm is expressed as follows:

$$k \in PK_{ij}^r \iff \{(k \in PK_{ij}^{r^0}) \cup \left[k \mid TE_{ij}(PK_{ij}^{0} \cup k) \leq TE_{ij}(PK_{ij}^{0} \cup m) \right. \left. \forall k, m \in DK_{ij} \cap (k \notin PK_{ij}^{r^0})\right]\}$$  \hspace{1cm} (5.12)

This paradigm is likely to perform at least as well as the previous two paradigms because
it considers the expected driver response in the determination of the driver-preferred route set while seeking to minimize the total error.

5.4 Experiments

Simulation experiments are conducted using the solution framework for the behavior-consistent approach (Figure 3.1) to address three primary objectives: (i) evaluate the sensitivity of the approach under alternative definitions for the set of controllable routes, (ii) analyze mechanisms to enhance the performance of the approach, and (iii) enhance deployment effectiveness.

5.4.1 Experimental Setup

Experiments are conducted following the experimental setup in Sections 2.6.1.2-2.6.1.4 and Sections 3.5.1.1-3.5.1.5. Particular characteristics are defined as follows.

5.4.1.1 Particular Details for Experiments

Different scenarios are used to evaluate the performance of the behavior-consistent approach under each of the proposed paradigms. All drivers are assumed to have capabilities to receive personalized information in all scenarios except for the no-information case. This is designed to isolate effects and enable equitable comparison of the effects of the different control paradigms.

In all scenarios, other than the SO and UE DTA cases, the same model is used to represent the actual driver behavior (as discussed in Section 3.5.1.2). This is done so as to ensure that the insights are focused on the relative performance of the behavior-consistent approach. Similarly, all drivers with the same O-D pair are assumed to have the same set of driver-preferred routes. A total of 120,000 vehicles are loaded during the first 60 minutes of analysis. Each stage of the rolling horizon has a length of 20 minutes and a roll period of 5 minutes.
5.4.1.2 Benchmark Scenarios for Experiments

Three scenarios are used to benchmark the performance of other scenarios.

Scenario I (base-case): No information is provided to the drivers. It is the do-nothing strategy and represents the base-case. Here, drivers make route choice decisions based only on past experience.

Scenario II (SO DTA): This is the SO DTA solution. By definition, it represents the best possible system performance.

Scenario III (BC-ideal): Under this scenario, the driver-preferred route set is assumed to be identical to the controller-desired (SO DTA) route set. So, the controller recommends routes from the SO DTA route set under the behavior-consistent approach, and the drivers choose routes from the SO DTA route set as well as it represents their preferred-route set. Thereby, though practically unrealistic, this scenario represents the benchmark for the best possible performance for the behavior-consistent approach because the controller can use its “ideal” route set (SO routes) to recommend routes. Hence, it is more meaningful to compare the performance of other behavior-consistent paradigms with this benchmark rather than the SO DTA solution. However, the system performance under this scenario cannot exceed the SO DTA performance because routes are chosen by the drivers based on their behavioral tendencies though they are recommended the SO routes.

5.4.2 Results and Analysis
5.4.2.1 Control Paradigms

Three other benchmark scenarios are evaluated here: one for the UE DTA, and one each for the routing using SO and UE DTA solution routing proportions. Two scenarios related to the control paradigms are analyzed: one each for the behavior-consistent approach directing the traffic system towards the SO or UE states. Figures 5.1 and 5.2 show the percentage cumulative system travel time savings relative to the base-case (where no information is provided) for each of these scenarios under the less and more responsive behavior cases, respectively.

Scenario IV (UE DTA): This is the UE DTA solution. As is well-known in the literature, Figure 5.1 and 5.2 indicate significant travel time savings with respect to the base-case scenario (Scenario I) but not as much as under Scenarios II and III.

Scenario V (SO-info): SO routes and their corresponding proportions are used to provide route guidance. These SO routes may or may not match driver-preferred routes. A driver completely ignores information provided about routes that do not belong to his/her preferred route choice set. This scenario is used to illustrate the implications of directly using the UE DTA solution to provide information. As shown in Figure 5.1, while this scenario results in travel time savings relative to the base-case for the less responsive behavior case, there is a substantial performance gap compared to the SO and UE DTA solutions. The experiments in Chapter 3 illustrate that the SO-info scenario can perform worse than the base-case under more responsive behavior. This illustrates trade-offs between level of compliance and overreaction, implying that lack of behavior consistency can result in poor information-based control strategies.

Scenario VI (UE-info): This scenario is conceptually similar to Scenario V. However, instead of using SO routes, it uses UE routes and their corresponding proportions to provide route guidance. Figure 5.1 shows that while the UE-info scenario has savings over the base-case, its performance is worse than that of the SO-info scenario, indicating the inherent value of the SO objective. However, this difference is small, suggesting the existence of trade-offs between the number of controllable routes and the quality of routes relative to the controller objective. Figure 5.3 illustrates that
the UE-info scenario has higher compliance rates compared to the SO-info scenario. As discussed in Section 5.1, this is because more UE routes match driver-preferred routes compared to SO routes. However, higher compliance rates may not translate to better performance. It questions the justification of the focus on UE for route guidance based on the notion that the SO strategy is not behaviorally sustainable. This point is further illustrated in the next two scenarios where behavior-consistent strategies are used.

Scenario VII (BC-SO-info or 1st DOV paradigm BC-SO-info): Here, the controller uses the behavior-consistent approach to direct the system towards the SO DTA state. This scenario corresponds to basic behavior-consistent approach where information is provided for only the driver-preferred routes that fully match with controller-desired routes. As shown in Figures 5.1 and 5.2, the system performance under this paradigm results in significant travel time savings relative to the SO-info and UE-info scenarios. It highlights the value of developing behavior-consistent strategies, which leads to significantly higher compliance rates (Figure 5.3). However, the BC-SO-info scenario results in fewer savings compared to the idealized SO or UE DTA scenarios which unrealistically assume 100% compliance with the corresponding routing strategies. In reality, the controller has only limited control over the system as drivers make route choice decisions using several factors.

Scenario VIII (BC-UE-info with $\lambda = 1$): Here, the controller uses the behavior-consistent approach to direct the system towards the UE DTA state. Akin to the previous scenario, the controller recommends routes using only driver-preferred routes that fully match with controller-desired routes. The BC-UE-info scenario performs better than the SO-info and UE-info scenarios because of the behavior-consistent approach. It implies that the controller can significantly enhance system performance by directing the system towards either the SO or the UE DTA states in a behavior-consistent manner. However, for the reasons discussed under Scenario VI, it does not perform as well as the BC-SO-info scenario.

Figures 5.1 and 5.2 indicate that the gap between SO DTA and BC-SO-info is larger than the gap between UE DTA and BC-UE-info. As discussed in Section 5.3.1, this is
because there are more controllable routes under BC-UE-info that under BC-SO-info. Hence, the controller has more options to approach to its objective when it seeks the UE state. For the reasons discussed in Scenarios VI and VIII, the BC-SO-info scenario performs better than the BC-UE-info scenario. This is in contrast to most of the current literature that advocates UE-based information strategies over SO-based ones. While many route guidance models (e.g., Mahmassani et al., 1994, Ben-Akiva et al., 1997) include both SO and UE as driver classes, there is a marked bias to considering the UE solution as the preferred approach for route guidance, while the SO solution is typically relegated to being an upper bound or justified only for special cases such as incident management. Stier-Moses (2004) and Jahn et al., (2005) discuss the inefficiency of UE-based information strategies. They propose the use of SO-based constrained routing approaches, where a static SO problem is solved while precluding relatively long routes from being included in the solution to generate a better routing approach compared to UE.

Figure 5.4 illustrates that there are more controllable routes when the behavior-consistent approach seeks to direct the system towards the UE state rather than towards the SO state. This is consistent with the earlier discussion on UE strategies entailing higher compliance rates compared to SO ones. In general, more controllable routes imply higher compliance rates under the behavior-consistent approach. Figure 5.5 shows the compliance rates for the BC-UE-info and BC-SO-info scenarios under the two levels of responsiveness. As expected, the compliance rates are higher for the “more responsive” case under both the UE and SO strategies.

5.4.2.2 Degree of Overlap Paradigms
The 1st DOV paradigm corresponds to Scenario VII. Two scenarios each are evaluated for the 2nd and 3rd DOV paradigms. Figures 5.6 and 5.7 plot the percentage cumulative system travel time savings relative to the base-case scenario for these scenarios for less and more responsive drivers, respectively.

Scenario IX (2nd DOV paradigm BC-SO-info $\lambda = 0.90$): This scenario implements the threshold DOV paradigm with $\lambda$ equal to 0.90. The results indicate that scenario enhances system performance relative to the 1st DOV paradigm. It implies that additional “good quality” controllable routes are available for traffic routing as $\lambda$ is relatively high. Thereby, there are some driver-preferred routes that are almost identical to some controller-desired routes. This is a case where the number of routes added to the controllable set and their quality are significant enough to positively affect performance.

Scenario X (2nd DOV paradigm BC-SO-info $\lambda = 0.80$): When $\lambda$ equal to 0.80 is used, the trade-offs between the number of routes and route quality become apparent. Hence, while this scenario performs better than the base-case, it does worse than the 1st DOV paradigm. The trade-offs are nicely illustrated in Figure 5.8 which shows the system travel time savings relative to the base-case for various $\lambda$ values under the less responsive behavior. As the $\lambda$ value decreases from 1, more routes are identified as controllable though their quality degrades relative to the controller objective. Initially, the presence of more controllable routes improves performance beyond the 1st DOV paradigm, but as $\lambda$ is decreased further, the negative effect of route quality kicks in. In the context of route guidance, this implies that simply increasing the number of routing options does not necessarily imply better performance, and the effectiveness of a route vis-à-vis the controller objective is as important.

Scenario XI (3rd DOV paradigm BC-SO-info $\lambda_U = 0.90 \& \lambda_L = 0.80$): This scenario represents the 3rd DOV paradigm with $\lambda_U$ equal to 0.90 and $\lambda_L$ equal to 0.80. Figure 5.6 shows that this scenario performs worse than Scenario IX and about the same as the 1st DOV paradigm. For the more responsive case (Figure 5.7), it performs similar to Scenario IX. This suggests that there may not be a need for the complex approach represented by the 3rd DOV paradigm, and the 2nd DOV paradigm is sufficient to achieve comparative results. The performance of the 3rd DOV paradigm is not superior because
at most it adds one route each in the last two steps of its approach, while Scenario IX can add several good quality routes as part of its paradigm.

Scenario XII (3rd DOV paradigm BC-SO-info $\lambda_U = 0.85 \& \lambda_L = 0.75$): The savings under this scenario are lower than under Scenario XI. This is expected based on the 2nd DOV paradigm insights relative to the value of $\lambda$ (Figure 5.8); the quality of the additional controllable routes is not significant enough to positively affect performance.

Figures 5.6 and 5.7 also illustrate that the BC-UE-info strategy does not perform as well as Scenarios IX and XI, both of which perform at least as well or better than the BC-SO-info strategy with $\lambda$ equal to 1. This further corroborates the insights discussed in Section 5.4.2.1 on the relative value of SO-based strategies.

In summary, the 2nd DOV paradigm with a high $\lambda$ value provides the best approach to enhance system performance beyond that under the 1st DOV paradigm.

5.4.2.3 Route Type Paradigms

Each BC-SO-info route type paradigm described in Section 5.3.2.2 is associated with one scenario. Figures 5.9 and 5.10 depict the percentage cumulative system travel time savings relative to the base-case for less and more responsive drivers, respectively. It should be noted here that the three scenarios discussed hereafter, Scenarios XIII-XV, have a larger driver-preferred route set (one more route) compared to Scenarios IV-XII, and hence, may have an a priori advantage over them. This is confirmed in Figures 5.9 and 5.10, where the performance of Scenarios XIII-XV is at least as good as or better than the various DOV paradigm based scenarios.

Scenario XIII (1st route type BC-SO-info): This scenario performs better than the 1st DOV paradigm. This is because for each O-D pair one additional controller-desired route is likely to become a driver-preferred route, providing more options to the controller.

Scenario XIV (2nd route type BC-SO-info): The results under this scenario are very similar to the ones under the previous scenario. This implies that adding the SO route that overlaps the most with a driver-preferred route is as effective, in terms of system performance, as adding the route with the highest SO proportion. From the behavior-
consistent approach perspective, this suggests that the 2nd route type paradigm may suffice compared to exploring the 1st route type paradigm as its focus is on the degree of overlap.

Scenario XV (3rd route type BC-SO-info): This scenario performs better than Scenarios XIII and XIV. This is because it uses the estimation of driver behavior to select the controller-desired route to add to the driver-preferred route set that minimizes the total error. It corroborates the importance of behavior-consistent approaches for the development of information-based network control strategies.

The travel time savings illustrated in Figures 5.6, 5.7, 5.9 and 5.10 indicate how various paradigms perform relative to the BC-ideal and full overlap paradigms. The objective of the analysis is to obtain insights on the tradeoffs offered by the various paradigms and specific trends (Figure 5.8) that can aid deployment strategies. As noted before, the BC-ideal case is an idealized strategy and the full overlap paradigm may have deployment limitations. Hence, the experiments seek to explore potential alternative paradigms in terms of their deployment effectiveness.

5.5 Summary and Insights

This study proposes alternative paradigms to enhance the performance and the deployment effectiveness of a behavior-consistent information-based network control approach. It compares the performance and compliance aspects associated with directing the system towards the UE or SO states. It evaluates the sensitivity of the behavior-consistent approach under various definitions for the sets of controllable and driver-preferred routes so as to improve performance and analyze practical aspects to enhance deployment effectiveness.

The study captures the interdependencies between network interactions and driver response to information by explicitly focusing on the acceptability of routes to drivers, the quality of those routes relative to the controller objective, and ensuring behavior consistency in route recommendations. Existing approaches, addressed primarily under the DTA label, tend to mostly focus on adequately capturing the network flow
interactions and dynamics, while making strong assumptions on driver behavior under information provision.

Broadly, the study results confirm the primary finding of the behavior-consistent approach; the simultaneous consideration of controller objectives and driver behavior is essential to identifying realistic and superior information-based control strategies. Here, “realistic” implies that the expectation of the controller in terms of the likely response of drivers to the route recommendations is a reasonable representation of the evolving network conditions. The superior performance is due to the explicit assurance of behavior consistency, thereby preventing the possibility that the controller may over-recommend or under-recommend routes, or recommend routes that are not considered by the drivers.

A key insight from this study is that there are trade-offs between the number of driver-preferred routes considered by the controller for routing and the quality of routes relative to the controller objective. They manifest during the investigation of alternative control (SO and UE) and DOV paradigms. In the control paradigm context, the results suggest that while a larger percentage of UE routes match driver-preferred routes, the inherent quality of the SO solution has intrinsic value. That is, even when routing is performed in a behavior-consistent manner, higher compliance rates need not necessarily translate to better system performance. This questions the justification of UE DTA solutions for route guidance on the ground that a SO strategy is not behaviorally sustainable or implies unfair routing recommendations. A fundamental corollary is that focusing primarily on robustly addressing either driver behavior (enhancing compliance) or controller objective (quality of routes) while representing the other aspect in a rudimentary manner is not adequate to ensure effective performance in the real-world. That is, approaches driven primarily by either network traffic flow modeling or behavior modeling do not suffice, implying the need for models with explicit supply-demand integration.

The trade-offs between the number and quality of routes are further illustrated by the DOV paradigm experiments (as shown in Figure 5.8). In these paradigms, higher
threshold $DOV$ values are associated with better quality of routes, but may lead to fewer controllable routes. Hence, the trade-offs lead to an “optimal” threshold $DOV$ value at which the system performs the best. The system performance deteriorates dramatically as the threshold $DOV$ value decreases from the “optimal” value, and at some point this paradigm may not generate savings over even the base-case.

Over time, the learning processes of drivers vis-à-vis route guidance can alter their preferred-route choice set. The controller can potentially influence this process by providing “new” routes based on its objectives and a recognition of driver preferences. For example, these can be controller-desired routes that highly overlap driver-preferred routes. Hence, they are likely to be accepted as new preferred-route alternatives by the drivers, increasing the controller’s ability to enhance system performance.

The use of the controller-estimated driver behavior model in the behavior-consistent approach to determine the routes to be recommended to drivers and to identify new driver-preferred routes is a key element of the behavior-consistent approach. There are numerous route choice models in the literature that can potentially be used to meaningfully estimate driver route choice decisions and calibrate the associated model parameters. For example, Peeta and Yu (2006) propose a consistency-seeking mechanism for a hybrid route choice model (Peeta and Yu, 2005) that updates the model parameters on-line. Chapter 6 develops a fuzzy on-line calibration model that updates the parameters of the proposed rule-based controller-estimated driver behavior model using field data to minimize the difference between the actual and the estimated network states in terms of link volumes. It closes the loop for the deployment of the behavior-consistent approach by enabling the simultaneous on-line determination of the behavior-consistent information strategies and the calibration of the controller-estimated model parameters.
Figure 5.1  Cumulative system travel time savings under the less responsive behavior benchmarked against the no-information case (base-case)
Figure 5.2  Cumulative system travel time savings under the more responsive behavior benchmarked against the no-information case (base-case)
Figure 5.3 Compliance rates for the less responsive behavior case under the standard DTA and behavior-consistent approaches
Figure 5.4 Percentage of driver-preferred routes matching UE and SO routes
Figure 5.5  Compliance rates for more and less responsive behaviors
Figure 5.6  Cumulative system travel time savings under the less responsive behavior relative to the base-case for the DOV paradigms
Figure 5.7  Cumulative system travel time savings under the more responsive behavior relative to the base-case for the DOV paradigms
Figure 5.8  System travel time savings under the less responsive behavior relative to the base-case versus the DOV
Figure 5.9  Cumulative system travel time savings under the less responsive behavior relative to the base-case for route type paradigms
Figure 5.10 Cumulative system travel time savings under the more responsive behavior relative to the base-case for route type paradigms
6. ON-LINE CALIBRATION OF BEHAVIOR PARAMETERS FOR BEHAVIOR-CONSISTENT ROUTE GUIDANCE

6.1 Introduction

The development of deployable strategies for real-time information-based network control to enhance system performance requires simultaneously addressing significant methodological problems related to traffic flow dynamics, driver behavior, demand modeling, and information processing, typically within a dynamic traffic assignment (DTA) framework. This is because a realistic, effective, and anticipatory solution to the problem requires the estimation of traffic network states resulting from complex interdependencies among network flow spatio-temporal interactions, driver response behavior, supply characteristics (such as the information provided), and related sources of randomness.

A significant body of literature exists for each of the aforementioned methodological problems. DTA models have predominantly focused on robustly capturing the traffic flow dynamics while seeking to achieve some system-wide objectives. However, they mostly tend to pre-specify driver behavior. For example, they categorize driver behavior and/or assume driver classes with restrictive pre-specified objectives such as user equilibrium (UE) or system optimal (SO). However, such simplicity is not consistent with the real-world and is consequently problematic. The traffic network states unfolding over time are fundamentally dependent on driver behavior which is a key source of complexity due to the spatio-temporal interactions that result from driver route choice decisions. Hence, the incorrect modeling of driver behavior can negatively impact the prediction of the traffic network states and the effectiveness of information-based control strategies. Peeta and Yu (2004, 2006) highlight realism issues arising from the
rigid representation of driver behavior under information provision, and the consequent barriers to developing effective operational paradigms for information-based traffic network management.

In contrast to the emphasis on traffic flow modeling in DTA models, traditional route choice models focus primarily on the socio-economic characteristics of drivers and the physical attributes of the routes. In addition, route choice models under information provision also consider some information-related attributes. However, they typically do not consider the spatio-temporal interactions resulting from the individual driver route choice decisions. To address this aspect, route switching decisions are typically modeled through en-route driver behavior models. Mahmassani and Jayakrishnan (1991) seek to capture the network-level traffic flow interactions by simulating driver en-route switching decisions assuming boundedly-rational driver behavior based only on route travel times. Abdel-Aty (1998) develops a nested logit model to predict en-route routing decisions for incident-related congestion under real-time information provision. Srinivasan and Mahmassani (2000) propose a multinomial probit framework specifying compliance and inertia as two factors that influence driver route choices under real-time information. While en-route driver behavior models consider spatio-temporal interactions and driver behavior to predict traffic network states, they are descriptive and do not address the controller objectives of enhancing system performance. Even when controller objectives are addressed in conjunction with a behavior model, the control mechanism does not engage them interactively and is based on a sequential logic. That is, there is no mechanism to ensure that controller-recommended routes are consistent with drivers’ likely route choice decisions.

To ensure consistency between controller objectives and the driver decision-making process, Chapters 2 and 3 develop the behavior-consistent traffic routing approach where the controller factors the drivers’ likely reactions to the information strategies while determining them. The behavior-consistent approach uses a controller-estimated driver behavior model to predict the proportion of drivers taking routes under the potential information strategies that the controller is iteratively seeking to determine. This implies solving a fixed-point problem where the controller-estimated driver behavior depends on
the information strategies and vice versa. Hence, the behavior-consistent approach enables the simultaneous consideration of the controller objectives and driver behavior. Chapter 5 illustrates trade-offs between the controller objectives and the driver acceptability of the controller-recommended routes. They suggest that higher compliance rates by themselves do not necessarily translate to better performance, and that the route quality relative to the controller objectives is as important. Therefore, due to the aforementioned fixed-point relationship, the prediction accuracy of the controller-estimated driver behavior model is a key aspect of the behavior-consistent approach. This is because an incorrect prediction of the drivers’ likely reactions to the information strategies can result in the generation of erroneous information strategies, negatively impacting network performance. From a deployment standpoint, this implies the need to calibrate the controller-estimated driver behavior model and represents the motivation for this research.

Procedures for the on-line calibration of traffic estimation/prediction systems vis-à-vis route guidance typically seek to correct for systematic inconsistencies so as to minimize the gap between the predicted and actual (observed) networks states unfolding in real-time. While there are several potential sources of inconsistency, the characterization and solution methods for the problem have evolved from simple reactive approaches that adjust network-level factors or control strategies to consistency-seeking models (Peeta and Yu, 2006) that assign primacy to the behavioral aspects. Peeta and Bulusu (1999) propose a generalized singular value decomposition based method that adjusts the number of drivers on each route to minimize the error between the prediction and observed system states. They view the inconsistencies as arising from the incorrect prediction of the unequipped driver routes, time-dependent O-D demand, incident characteristics, and route compliance aspects of equipped users. However, these sources are not separated in the method, and aggregate link counts are the basis for the route proportion adjustments. Thereby, systematic inconsistencies in the associated models or their parameters are not addressed.
Initial efforts to calibrate DTA model parameters have concentrated mostly on the supply or demand aspects whereby the traffic flow modeling parameters or the origin-destination (O-D) demand are adjusted on-line based on unfolding system states. Mahmassani et al., (1998) calibrate a DTA model using a proportional-integral-derivative feedback control strategy that reacts to any observed on-line deviations in traffic conditions. The procedure uses a real-time module to adjust the parameters of the travel time function and the flow propagation equations using real-time data on traffic measures such as average speed, inflow, and outflow. Further, an off-line module is used to update the parameters using full information on past conditions so as improve the real-time adjustments.

Recent efforts seek to develop generalized frameworks to simultaneously calibrate supply and demand parameters to capture the critical interactions between these elements. Balakrishna et al., (2007) and Antoniou (2004) propose state-space frameworks to address the off-line and on-line calibration of DTA models, respectively. In the experimental analysis, they consider parameters associated with speed-density relationships, segment capacities, and the time-dependent O-D demand.

Driver behavior is a fundamental determinant of the network state evolution. Hence, a natural next step towards the accurate prediction of traffic network states is the incorporation of behavioral aspects in the consistency-checking procedures. In this context, Peeta and Yu (2006) propose a behavior-based consistency-seeking approach that considers traffic flow dynamics in conjunction with a hybrid probabilistic-possibilistic driver behavior model (Peeta and Yu, 2005) to consistently address driver learning processes so as to predict the time-dependent network states. The associated consistency-seeking problem updates the driver class fractions in the ambient traffic stream based on link traffic counts to reduce the state consistency gap. However, it does not adjust the underlying driver behavior models or their parameters.

In this study, the on-line calibration problem seeks to update the behavior model parameters. It focuses on ensuring a meaningful prediction of driver behavior under information provision, and consequently, on evolving network states. A fuzzy on-line calibration model is proposed to address the problem where the controller seeks to
minimize the difference between the actual and the estimated network states by updating
the controller-estimated driver behavior model parameters using the link traffic counts
unfolding over time. As discussed earlier, the calibration of the controller-estimated
driver behavior model enhances system performance by enabling more effective
information-based network control strategies.

The remainder of this paper is organized as follows. Section 6.2 discusses the on-line
calibration problem. Section 6.3 describes the controller-estimated driver behavior
model. Section 6.4 presents the fuzzy on-line calibration model used to address the on-
line calibration problem as part of the deployment framework for the behavior-consistent
approach. Section 6.5 discusses experiments and analyzes their results. Section 6.6
provides some concluding comments.

6.2 On-line Calibration Problem

The notion that the behavior-consistent routing problem and the associated on-line
calibration problem are addressed in a single on-line deployment framework is
conceptually illustrated in Figure 6.1, where the shaded boxes correspond to the solution
logic for the behavior-consistent approach alone (Chapter 3). The non-shaded boxes
correspond to new components developed in this paper to enable the on-line calibration
of the controller-estimated driver-behavior model. The controller-estimated driver
behavior model is used to estimate the driver routing decisions in the current roll period
in light of the controller-recommended routes. A traffic flow simulator is used to obtain
the controller-estimated traffic network states for the current roll period based on the
estimated driver decisions. In this study, the traffic flow simulator is assumed to be
accurate, and the calibration is focused on the behavior parameters. Thereby, if gaps
exist between the controller-estimated and actual network states in terms of link traffic
counts, a fuzzy on-line calibration model is used to calibrate the controller-estimated
driver behavior model parameters. If state consistency issues do not exist, the field
conditions for the current roll period are used to repeat the behavior-consistent approach
by calculating the SO DTA solution for the next roll period.
While this study addresses state inconsistency by adjusting behavior parameters alone, the proposed fuzzy on-line calibration methodology provides a generalized approach to handle multiple sources of inconsistency through the use of aggregate if-then rules. Thereby, for example, rules associated with traffic flow modeling inconsistency can be seamlessly incorporated along with rules for behavior model inconsistency without any change to the structure of the fuzzy methodology. Another advantage of the methodology is that aggregate level sensor data can be used for addressing the calibration problem. This circumvents the need for disaggregate data (such as individual driver level data), enhancing the ability to practically deploy the methodology.

6.3 Controller-estimated Traffic Network States

Section 6.3.1 describes the controller-estimated driver behavior model used in this study. The output from the model is the set of time-dependent driver routing decisions. Section 6.3.2 illustrates how these drivers are loaded onto the network in the traffic flow simulator to estimate the time-dependent network states for the calibration problem. While the route choice process is time-dependent, its associated time dimension is ignored in Section 6.3.1 without loss of generality to simplify the notation.

6.3.1 Controller-estimated Driver Behavior Model

Over the past two decades, a body of literature has been developed for the on-line estimation and prediction of driver route choice behavior under information provision. Existing models range from econometric (probabilistic) to hybrid (probabilistic-possibilistic) models (Peeta and Yu, 2005). To handle the uncertainty associated with driver behavior, econometric models assume well-defined probability distributions while possibilistic models used fuzzy frameworks that can handle linguistic/qualitative and/or difficult-to-measure variables (Peeta and Yu, 2004). The current study proposes a fuzzy multinomial logit model as the controller-estimated driver behavior model. It uses
simple aggregate-level behavioral *if-then* rules to determine the systematic component of the utilities of the various routes (alternatives). Akin to standard discrete choice logit models, an i.i.d extreme value error component is added to each utility to account for the randomness in driver behavior. Akin to models proposed by Lotan and Koutsopoulos (1993, 1999) for route choice behavior, the decision process is modeled as a non-linear combination of behavioral rules where each rule deals with a different aspect of the overall choice process. The controller-estimated driver behavior model is described hereafter.

### 6.3.1.1 Behavioral If-then Rules

The driver routing decisions are based on a set of behavioral *if-then* rules that relate the decisions to the routes characteristics, the driver attributes in terms of information availability, and level of responsiveness to the information strategies. It is reasonable to expect that drivers do not use very sophisticated rules and/or many factors to make online routing decisions due to the associated time constraints. Hence, simple and straightforward rules consisting of one-dimensional left hand side (LHS) and right hand side (RHS) components are proposed here. In our experiments, it is assumed that travel time, route complexity, and the controller-recommended routes, are the key factors that influence the route choice decision-making process. However, additional factors can easily be added by creating the corresponding behavioral *if-then* rules.

The LHS (antecedent) of the rules deal with travel time, route complexity, and the controller-recommended routes. The RHS (consequent) deals with the propensity to choose a route, but does not represent the route choice itself. Rather, it is used to model the attractiveness of a driver-preferred route based on the conditions described by the LHS. In general, the rules used here are defined as:

"If $A^h_k$, Then $B^h_k$", \hspace{1cm} $h = 1,\ldots, BR$ and $k \in PK_{ij}$
where $A_k^h$ is the LHS component of the $h$th rule that corresponds to a characteristic of route $k$ connecting O-D pair $ij$, $B_k^h$ is the RHS component of the $h$th rule that deals with the attractiveness of route $k$, and $BR$ is the total number of rules. Here, $k$ belongs to the set of driver-preferred routes ($PK_{ij}$).

Table 6.1 summarizes rules grouped using their LHS. Different sets of rules are used to model different levels of responsiveness to the information strategies. The LHS of the rules associated with the controller-estimated expected route travel times $TT$ is characterized by the following five fuzzy sets: “Very Low (VL)”, “Low (L)”, “Medium (M)”, “High (H)”, and “Very High (VH)” travel times. The controller can estimate these expected travel times using historical data. The number of nodes $NN$ for each route is used to estimate the effect of route complexity on the route choice decisions. Here, the LHS is characterized by the following five fuzzy sets: “Very Low (VL)”, “Low (L)”, “Medium (M)”, “High (H)”, and “Very High (VH)” number of nodes. For controller-recommended routes, the LHS corresponding to a route recommendation $Y$ is characterized by the following three fuzzy sets: “the Route is Recommended (RR)”, “the Route is Not Recommended (RNR)”, and the “Route Was Recommended (RWR)” in the previous roll period.

The RHS of the rules characterizes the attractiveness $V$ of a route in terms of the following five fuzzy sets: “the driver will not choose this route (N)”, “the driver will probably not choose this route (PN)”, “the driver is indifferent to choosing this route (I)”, “the driver will probably choose this route (PO)”, and “the driver will choose this route (O)”.

The rules used in this study to capture driver behavior are based on the findings from previous studies and field observations. For example, routes with short travel time are preferred over those with higher travel times. Consistent with fuzzy logic, it is important to note that the inputs for the rules may not necessarily coincide with one of the LHS fuzzy sets described above. Rather, each input belongs to these fuzzy sets with different degrees of membership, and consequently will likely trigger the firing of more than one rule. The degree of membership is determined using membership functions.
6.3.1.2 Membership Functions

Triangular membership functions are used to define the fuzzy sets associated with the behavioral if-then rules. The controller’s expectation of driver route perception can be modeled through the shape, range, and amount of overlap between adjacent sets of the membership functions. For example, if the controller has poor knowledge or high ambiguity with expected route travel times, wide membership functions can be used to represent that aspect. By contrast, narrow membership functions imply that the controller has good estimates. Although the shape of the membership function and its parameter values contribute to prediction accuracy, the behavioral if-then rules used and their associated weights in the fuzzy aggregation process are more critical for accurate route choice estimation. This is because the behavioral if-then rules define which membership functions are used, and their weights affect their contribution to the route attractiveness. Hence, this study uses simple membership functions and focuses on calibrating the weights of the if-then rules.

The membership functions are used to capture the expected degree of mapping $\mu$ between the controller’s expectation for an attribute and the LHS fuzzy sets. For controller-estimated expected travel times, the controller estimates that drivers have a range $(\text{Min}\, TT_k, \text{Max}\, TT_k)$ of possible travel times for each preferred route $k$. The degree of mapping for $TT_k$ is represented by $\mu_{T}(TT_k)$. Five membership functions are defined to cover the range of the controller-estimated travel time. Given the range for each driver-preferred route, a super range covering all routes is defined $(\text{Min}\, TT = \min_{k \in PK_{ij}} \text{Min}\, TT_k, \text{Max}\, TT = \max_{k \in PK_{ij}} \text{Max}\, TT_k)$. This super range is covered evenly using five membership functions, as shown in Figure 6.2. The same approach is used for number of nodes (route complexity) as well.

Figure 6.2 also shows the three functions used to represent the membership functions for the LHS of the rules associated with route recommendations. There is no overlap among them because a route is either recommended or not recommended. Hence, the membership functions associated with the RHS of these rules are directly used by the
procedure; the degree of membership is either 1 or 0 based on whether that route is recommended or not recommended.

For the RHS of the behavioral rules, five membership functions corresponding to five fuzzy sets are used to characterize the route attractiveness. A range (-1, 1) is used to model the relative attractiveness of the routes, and the fuzzy logic decision process discussed hereafter uses only the relative difference in attractiveness over the set of driver-preferred routes to generate the controller-estimated driver route choice.

6.3.1.3 The Fuzzy Logic Decision Process

Figure 6.3 summarizes the fuzzy logic decision process used to obtain the controller-estimated driver route choice. The inputs, \( TT_k \), \( NN_k \), and \( Y_k \), are matched against the \( BR \) behavioral if-then rules to determine the activated (fired) rules and their corresponding fuzzy consequents \( V_{hk}^* \). The membership functions \( \mu \) of the consequents of the behavioral if-then rules are multiplied by their weights \( W \). A fuzzy inference and aggregation mechanism is used to combine the consequences of all rules that are fired, and a defuzzification scheme is used to determine the controller-estimated attractiveness of each route. As in Section 2.5.1, the max-min composition operator and Larsen product implication operator are used for fuzzy inference, and the center of gravity method (CGM) is used for defuzzification (Tsoukalas and Uhrig, 1997). The CGM is given by:

\[
V_k = \frac{\sum_{h=1}^{BR} W_h \cdot V_{hk} \cdot S(\mu_{V_{hk}^*})}{\sum_{h=1}^{BR} W_h \cdot S(\mu_{V_{hk}^*})} \quad \forall k \in PK_{ij}
\]

(6.1)

where \( S(\cdot) \) determines the area of the fuzzy sets \( V_{hk}^* \) whose centroids are defined by \( V_{hk} \), and \( V_k \) represents the attractiveness of route \( k \). This process is repeated for all driver-preferred routes to generate the route attractiveness vector \( V \). Since the controller-estimated driver behavior model needs to identify a discrete route for each driver, a
mechanism is developed to select a route based on the vector $V$ in which the attractiveness of an alternative $V_k$ is treated as the systematic component of a random utility model. The utility of alternative $k$ for driver $r$ is given by:

$$U_k^r = V_k^r + \varepsilon_k^r$$

\[ \forall r, k \in PK_{ij} \tag{6.2} \]

where $\varepsilon_k^r$ is assumed to be an i.i.d. extreme value random component. Thereby, alternative $k$ is chosen by driver $r$ using the resultant fuzzy multinomial logit model with probability:

$$P'(k) = P(V_k^r + \varepsilon_k^r \geq V_l^r + \varepsilon_l^r, \forall l \neq k)$$

\[ \forall r, k \in PK_{ij} \tag{3} \]

The route choice probabilities are converted to discrete route choices using the approach described in Section 3.5.1.2.

6.3.2 Network Loading Mechanism

Figure 6.4 illustrates the network loading mechanism for roll period $\rho(\sigma)$ of stage $\sigma$ using the controller-estimated driver behavior model to determine the initial and en-route controller-estimated driver route choices. The roll period is divided into discrete time intervals of length $\Delta$, denoted by $t$. The initial routes for the new drivers in interval $t$ are determined based on the dynamic inputs for $t$ in terms of information provision and the current route characteristics. Driver en-route route choices in interval $t$ are considered for those drivers who did not reach their destination in interval $t-1$ and who are located at an intermediate node (on their existing route) at the beginning of interval $t$. Intermediate nodes are viewed as potential decision nodes. These drivers are loaded onto the network at the beginning of $t$ based on their en-route route choices using the dynamic inputs. If a driver who did not reach his/her destination at the end of the previous roll period $\rho(\sigma-1)$ is located on a link rather than at an intermediate node at the beginning of the first
interval \( t \) of \( \rho(\sigma) \), he/she is loaded onto the network at the beginning of \( t \) using his/her existing route.

A traffic flow simulator is used to generate the controller-estimated network state for interval \( t \) using the controller-estimated route choices. If \( t \) represents the last interval of \( \rho(\sigma) \), the network loading for this roll period is terminated. Otherwise, the procedure is repeated until the end of \( \rho(\sigma) \).

### 6.4 On-line Parameter Calibration

Section 6.4.1 discusses the formulation for the on-line calibration of the behavioral parameters. Section 6.4.2 describes the fuzzy on-line calibration model to calibrate the weights of the if-then rules in the controller-estimated driver behavior model.

#### 6.4.1 Calibration of Behavioral Parameters

Several factors can contribute to the inconsistency between the controller-estimated network states and the actual conditions unfolding in real-time. Peeta and Bulusu (1999) list the following factors: (i) incorrect estimation of the time-dependent O-D demand, (ii) unexpected traffic incidents, (iii) incorrect traffic flow modeling, (iv) incorrect driver behavior modeling, (v) incorrect assumptions on system-related parameters, (vi) noise/sparsity in measurements, and (vii) failure of advance traveler information systems (ATIS) components.

The fuzzy on-line calibration model proposed in this paper can handle inconsistencies due to modeling errors (related to O-D demand, traffic flow, and behavior). As stated earlier, this study focuses on state inconsistency arising due to inaccurate values for the parameters of the controller-estimated driver behavior model. It assumes that the traffic flow modeling, the O-D demand predictions, and the data used here are accurate. This is done to derive insights on the controller-estimated driver behavior modeling aspects by isolating its effects.
The calibration problem seeks to update the weights of behavioral if-then rules of the controller-estimated driver behavior model so as to minimize the difference between the controller-estimated and actual (observed) network states. The associated formulation for roll period $\rho(\sigma)$ of stage $\sigma$ is as follows.

Minimize: $\left[ \hat{\omega}^{\rho(\sigma)} - \bar{\omega}^{\rho(\sigma)} \right]^2$ \hspace{1cm} (6.4)

Subject to:

$$\hat{\omega}^{\rho(\sigma)} = \hat{\omega}^{\rho(\sigma)}_* + \sum_t (MK \times C^t \times \hat{\delta}^t)$$ \hspace{1cm} (6.5)

$$\hat{\delta}^t = \sum_{r \in R^t} \mathbb{E}[\hat{F}(X'^t, Y'^t (\theta^{\rho(\sigma)}); W^{\rho(\sigma)})]$$ \hspace{1cm} (6.6)

where,

$\hat{\omega}^{\rho(\sigma)}$ is the estimated vector of link traffic counts for roll period $\rho(\sigma)$ of stage $\sigma$

$\bar{\omega}^{\rho(\sigma)}$ is the observed vector of link traffic counts for roll period $\rho(\sigma)$ of stage $\sigma$

$\hat{\omega}^{\rho(\sigma)}_*$ is the estimated vector of link traffic counts for roll period $\rho(\sigma)$ of stage $\sigma$ for drivers who do not reach their destinations during roll period $\rho(\sigma-1)$ of stage $\sigma-1$

$MK$ is the estimated link-route incidence matrix for the driver-preferred routes

$C^t$ is the estimated link-route incidence adjustment matrix for time interval $t$

$\hat{\delta}^t$ is the estimated vector of the number of new O-D desires for interval $t$ taking driver-preferred routes

$\hat{F}$ is the controller-estimated driver behavior model which is used to estimate driver route choices

$R^t$ is the vector of O-D desires in time interval $t$

$X'^t$ is the estimated vector of route characteristics excluding information that influences the route choice decision of driver $r$ in time interval $t$

$Y'^t$ is the route recommended by the controller to driver $r$ in time interval $t$
θρ(σ) is the prescriptive information defined as the proportion of drivers that must be recommended to take specific routes in roll period ρ(σ) of stage σ.

Wρ(σ) is the vector of rule weights (parameters) of the controller-estimated driver behavior model for roll period ρ(σ) of stage σ.

The controller objective (4) is to minimize the square of the difference between the estimated and observed vectors of link traffic counts for roll period ρ(σ). The estimated vector of link traffic counts is determined using the network loading mechanism discussed in Section 6.3.2. It is expressed here by Equation (5) as the summation of the vectors of link counts for existing drivers who did not reach their destinations at the end of ρ(σ-1) and the new O-D desires. MK×C×δt represents the estimated vector of link count contributions from the new O-D desires entering the network in time interval t. The link-route incidence matrix MK is defined by the driver-preferred route sets. This matrix is used here only to generate the initial set of route alternatives for interval t. Unlike for DTA models, it does not define the entire driver route trajectory using a time-dependent link-path incidence matrix. Drivers make pre-trip route choices, and can change these choices en-route at decision nodes based on the ambient driving conditions and the information provided to them. Ct denotes the adjustment to MK to ensure consistency between the observed and estimated link count contributions due to the new O-D desires entering the network in time interval t. Equation (6) defines the estimated vector of the number of new O-D desires for interval t taking driver-preferred routes in terms of the number of times that these routes are chosen by the O-D desires. Here, function g counts the number of times that a driver-preferred route is estimated to be chosen by new drivers. In the study experiments, X consists of the controller-estimated expected travel times and the number of nodes for the driver-preferred route sets.

The on-line calibration is done towards the end of ρ(σ), at which time the observed link counts for all time intervals in this roll period are available. However, it needs to be done before the computation begins (Figure 6.1) for the behavior-consistent strategies for the next roll-period. The unknown variables in the formulation (4)-(6) are the weights Wρ(σ) of the controller-estimated driver behavior model $\hat{F}$. Link traffic counts
averaged across all time intervals in the roll period \( \rho(\sigma) \) for stage \( \sigma \) (up to the point where behavior-consistent strategy computations begin for the next stage) serve as the network state data points to estimate the weights using the fuzzy calibration model described in the next section.

6.4.2 Fuzzy On-line Calibration Model

Figure 6.5 illustrates the fuzzy on-line calibration model which consists of an input step (non-shaded box with dotted borders), a decision-processing step (non-shaded boxes with solid borders), and an output step (non-shaded box with dashed borders).

6.4.2.1 Input

The inputs are the vectors of error \( e_{\rho(\sigma)} \) and change in error \( \Delta e_{\rho(\sigma)} \) defined by:

\[
e_{\rho(\sigma)} = \rho(\sigma) - \hat{\rho}(\sigma) \quad \text{and} \quad \Delta e_{\rho(\sigma)} = e_{\rho(\sigma)} - e_{\rho(\sigma-1)}
\]

(6.7)

where \( \Delta e_{\rho(\sigma)} \) is the difference between the current error \( e_{\rho(\sigma)} \) and the error in the previous stage \( e_{\rho(\sigma-1)} \).

6.4.2.2 Decision-processing Component

Akin to the fuzzy logic decision process described in Section 2.5.1, the decision processing step uses calibration control rules, their associated membership functions, and a fuzzy aggregation, inference, and defuzzification scheme to determine the adjustment to the weights of the behavioral if-then rules.
6.4.2.2.1 Calibration Control Rules

The control if-then rules used by the fuzzy calibration model are two-dimensional rules (two inputs) obtained from observed patterns and problem characteristics. Three sets of control if-then rules are used to calibrate the behavioral if-then rules, one each for the behavioral rule consequent (RHS) implying an increase, decrease or neutrality related to route attractiveness. Hence, for the same route, the weights of some behavioral rules may need to be increased while those of others may need to be decreased. For example, if the error associated with a route is positive, the number of drivers taking this route should be increased. This implies an increase in the weights associated with the behavioral rules that correspond to an increase in the attractiveness of this route. Hence, the calibration model determines how to calibrate the behavioral rule weights using a set of control rules such as:

If $[e \text{ is NL and } \Delta e \text{ is PL}]$, Then $[\Delta w \text{ is NS}]$

In this example, if the error $e$ is negative large (NL) and the change in error $\Delta e$ is positive large (PL), then the weight $w$ is decreased by a negative small (NS) quantity $\Delta w$. Here, the consequent implies a decrease in the attractiveness of a route.

The LHS and RHS of the control rules are characterized by the following five fuzzy sets: “Negative Large (NL)”, “Negative Small (NS)”, “Zero (ZR)”, “Positive Small (PS)”, and “Positive Large (PL)” for error and change in error. Hence, $e$, $\Delta e$, and $\Delta w \in [\text{NL, NS, Z, PS, PL}]$. Table 6.2 shows the set of control if-then rules used by the fuzzy calibration model based on these five fuzzy sets. The total number of control rules is denoted as $CR$. 
6.4.2.2 Membership Functions

Corresponding to the five fuzzy sets, there are five triangular membership functions each for \( e \), \( \Delta e \), and \( \Delta w \). Three membership functions, one for each of the two inputs and one for the output, are associated with each control \( \text{if-then} \) rule. The membership functions evenly cover the range of the domains for the inputs and output. The membership function parameters require off-line calibration to enable consistent on-line calibration of the parameters of the controller-estimated driver behavior model. In the study experiments, the off-line calibration of the membership function parameters was conducted using several iterations of the solution framework shown in Figure 6.1.

6.4.2.3 Decision Process

The max-min composition operator and Larsen product implication operator are used for fuzzy inference to determine the membership function \( \mu \) of the RHS of control rule \( l \) of the weight \( w_{h}^{\rho(\sigma)}l^{\ast} \) for all behavioral rules \( h \) for roll period \( \rho(\sigma) \) of stage \( \sigma \). The center of gravity method is then used for defuzzification to determine the adjustments to the weights:

\[
\Delta w_{h}^{\rho(\sigma)} = \sum_{l=1}^{CR} \frac{\sum_{l=1}^{CR} \overline{w_{h}} \cdot S(\mu_{w_{h}^{\rho(\sigma)}l^{\ast}}))}{\sum_{l=1}^{CR} S(\mu_{w_{h}^{\rho(\sigma)}l^{\ast}}))} \quad \forall \ h = 1, \ldots, BR
\]

where \( S(.) \) determines the area of the fuzzy sets \( w_{h}^{\rho(\sigma)}l^{\ast} \) whose centroids are defined by \( \overline{w_{h}} \cdot \Delta w_{h}^{\rho(\sigma)} \) represents the adjustment to the weight of behavioral rule \( h \). The process needs to be repeated for all behavioral rules and data points (link traffic counts) resulting in a vector of weight adjustments \( \Delta w_{h}^{\rho(\sigma)} \).
6.4.2.3 Output

The calibrated weights used to determine the information strategies for the roll period of the next stage ($\sigma+1$) are defined as:

$$w^{\rho(\sigma+1)} = w^{\rho(\sigma)} + \Delta w^{\rho(\sigma)}$$

(6.9)

6.5 Experiments

Simulation experiments using the framework of Figure 6.1 are designed to evaluate the performance of the fuzzy calibration model in the context of the broader problem which is the on-line determination and deployment of the behavior-consistent information-based network control strategies. They follow the experimental setup in Sections 2.6.1.2-2.6.1.4 and 3.5.1.1-3.5.1.5. Particular details are as follows.

6.5.1 Experiment Details

6.5.1.1 Calibration Cases

The 1st day case: this is the case where the controller does not have any information to determine the weight to assign to each behavioral rule used by the controller-estimated driver behavior model. It is the situation faced by the controller on the first day of deploying the behavior-consistent information-based network control strategies. Hence, the controller initially assigns the same weight to all rules. Therefore, the results obtained under this case are conservative and are affected by the initial values adopted for the weights.

The 2nd day case: this represents the case where there is information on the prior values for the weights associated with the controller-estimated driver behavior model. The information on the values of the weights is available after the first day. Here, the controller initially assigns the values computed at the end of the previous day using the on-line calibration model for the behavioral rule weights. This aids computational
efficiency as the relative values that drivers assign to the different choice attributes are likely to be the same under normal conditions.

6.5.1.2 Scenarios

Two scenarios each are evaluated for each level of responsiveness and day case. In the “BC-info-CS” scenario, the system controller uses the full framework of Figure 6.1 to determine the information strategies. That is, the information strategies are determined using the behavior-consistent approach, and the calibration model is used to calibrate the parameters of the controller-estimated driver behavior model during each stage. In the “BC-info” scenario, the system controller uses only the shaded boxes in Figure 6.1 to determine the information strategies. That is, the calibration model is not used to update the parameters of the controller-estimated driver behavior model. Hence, the two scenarios differ only in terms of whether the calibration is performed or not.

6.5.1.3 Performance Measures

The effectiveness of the calibration model is measured in terms of its ability to accurately estimate the traffic pattern unfolding over time. The performance measure used here is the average percentage difference between the observed and estimated link traffic counts:

\[
\left[ \frac{\sum_{a \in \Gamma} \left| \chi_a^{\rho(\sigma)} - \bar{y}_a^{\rho(\sigma)} \right|}{\chi_a^{\rho(\sigma)}} \right] \frac{100}{|\Gamma|}
\]  

(6.10)

where \( \chi_a^{\rho(\sigma)} \) is the observed count on link \( a \) in roll period \( \rho(\sigma) \), \( \bar{y}_a^{\rho(\sigma)} \) is the estimated count on link \( a \) in roll period \( \rho(\sigma) \), and \( \Gamma \) is the set of links for which real-time measurements are obtained.
In Figure 6.1, Equation (6.10) is also used to determine whether the calibration model needs to be activated in each roll period. In the study experiments, the calibration model is activated if a threshold value of 5% is exceeded for this performance measure.

A second performance measure, the difference between the observed and corresponding SO states, can be computed by replacing \( j_{\sigma}^{(\rho)} \) in Equation (6.10) with the counts obtained from the SO DTA traffic assignment.

### 6.5.1.8 Network States

In the study experiments, the “observed” network states are assumed to be the outcome of the actual driver behavior model (Section 3.5.1.2) in conjunction with the traffic flow simulator. The “estimated” network states are obtained using network loading mechanism described in Section 6.3.2.

### 6.5.2 Results and Analysis

#### 6.5.2.1 The 1st Day Case

Figure 6.6 shows the average percentage differences between observed and estimated traffic counts for the first day on which the calibration model is implemented. Initially, these differences are significant as the behavioral rules are arbitrarily assigned equal weights. However, as the calibration model starts using information from more stages, it is able to significantly reduce these differences. This suggests that the fuzzy calibration model can calibrate the controller-estimated driver behavior model by reducing the estimation errors. As seen in the figure, the information strategies determined using the calibrated parameters are also able to move the system closer to the SO states. This is because the information strategies based on the calibrated parameters are able to generate better field behavior consistency, implying improved observed system performance. However, the calibration lowers the gap more for the behavior-consistent
approach as it focuses on ensuring behavioral realism unlike the idealized SO states by updating the weights for all behavioral rules and all routes.

Figure 6.6 also shows that better state consistency is achieved for the “more responsive” drivers when the observed states are compared to the SO DTA states. However, as illustrated in Figure 6.7, the system travel time savings are larger for the “less responsive” drivers. Since the behavioral parameters are not calibrated before the first day, the calibration model needs a few stages during the first day to adapt the controller-estimated driver behavior model to the unfolding actual behavior. During these initial stages, the system performance may deteriorate. Since, drivers are more likely to accept route recommendations under the “more responsive” scenario, the negative performance effects can be amplified for the “more responsive” drivers. Consequently, though the calibration is enhanced over time, the initial negative effects cannot be compensated for adequately by the “more responsive” behavior. However, if the initially allocated values for the behavioral parameters are more representative of the actual behavior, the “more responsive” scenario could perform better than the “less responsive” one. Figure 6.7 also shows that the BC-info scenario performs worse than the BC-info-CS scenario for both levels of responsiveness. This highlights the importance of calibrating the behavioral parameters. It also reinforces the notion that the determination of information strategies requires a meaningful estimation of driver behavior.

Figure 6.8 shows the weights of the information behavioral if-then rule 11a (in Table 6.1) over time for the O-D pair illustrated in Figure 6.6. It indicates that the weights are continuously updated without reaching convergence except for few routes. However, as illustrated in Figure 6.6, the calibration model is able to reduce the estimation error though the weights do not converge. Figure 6.6 also indicates that the percentage difference has lower variability after about stage 15, suggesting the likelihood of multiple solutions for the weights of the behavioral if-then rules in Figure 6.8. This result is intuitive as only the relative differences in attractiveness (utility) of the alternative routes matters for the controller-estimated driver behavior model. That is, different
combination of values for the weights can result in the same route choice probabilities in Equation (6.3).

The rule 11a in Figure 6.8 increases the attractiveness of a route if that route is recommended. The figure shows that route 1 has higher weights compared to other routes. This implies that a large number of drivers are choosing this route based on the behavior-consistent information. Figure 6.9 indicates that route 1 is significantly recommended to drivers by the controller under the behavior-consistent approach. This suggests that the calibration model consistently adapts the controller-estimated driver behavior model to the observed network states.

6.5.2.2 The 2nd Day Case

Figure 6.10 shows the percentage differences for the second day on which the calibration model is implemented. Their initial values are lower than those for the first day as the controller uses the calibrated parameters from the 1st day at the beginning of the second day. As before, the calibration model can reduce these differences over time. However, the relative improvement is not as significant as on the first day because the initial differences on the second day are smaller. Figure 6.11 shows that the initial negative effects that existed for the first day are no longer present because calibrated parameters from the first day are used. Consequently, as expected, it is observed that the “more responsive” case performs better.

A comparison of Figures 6.9 and 6.12 indicates that the weights are more stable for the 2nd day. This implies that the initial values for the weights can influence their variability. However, our experiments assume that the actual driver behavior is not modified from day 1 to day 2. But, such modification is possible under driver learning processes. Hence, it is possible that there is variability in the values of weights on the second day if that effect were captured. Figure 6.9 also illustrates that fewer drivers are recommended to take route 1 for the second day according to the behavior-consistent approach compared to the first day. This is because using calibrated parameters (obtained at the end of the first day) allows the controller to determine more consistent
proportions to recommend routes, without the need to over-recommend or under-recommend significantly. For example, the initial negative effects in Figure 6.7 are due to over- and under-recommendations of various routes. This is illustrated further in Figure 6.9 where route 1 is recommended consistently more for the first day compared to the second day.

6.6 Summary and Insights

This paper develops a fuzzy on-line calibration model to calibrate the parameters of a controller-estimated driver behavior model to enhance system state consistency in an operational context. The controller-estimated driver behavior model is a key component in the determination of behavior-consistent information-based network control strategies. The proposed calibration model fits seamlessly within a rolling horizon framework to deploy the behavior-consistent approach. Thereby, the framework determines the information strategies and updates the parameters associated to the controller-estimated driver behavior model.

The calibration model minimizes the difference between the observed and estimated network states in terms of link traffic counts. The practical deployment of the associated calibration model is aided by the structure of the controller-estimated driver behavior model which uses aggregate level if-then rules. This circumvents the need for data at the individual driver level, and the calibration can be based on measurable traffic data. In the context of broader route guidance related calibration problem, the proposed model provides a generalized approach to seamlessly incorporate modeling parameters associated with several components such as O-D demand, traffic flow, and driver response behavior. It implies adding calibration control rules for each parameter type and various data sources.

The study results are based on conservative analyses performed by deliberately having different structures for the controller-estimated driver behavior model and the actual driver behavior model. They suggest that the fuzzy on-line calibration model can effectively update the parameters of the controller-estimated driver behavior model,
resulting in significant improvements in terms of the accuracy of the controller-estimated network states. Further, there are substantial benefits in terms of system travel time savings with respect to the no-information scenario when the calibration is performed in conjunction with the behavior-consistent approach. The calibration problem is important for the effective deployment of the behavior-consistent information-based network control strategies. If calibration is not performed, negative effects due to inconsistent driver behavior estimation can be magnified under high levels of driver responsiveness.

In general, the behavior-consistent approach in conjunction with the fuzzy on-line calibration model provides an alternative methodological perspective to address the complex deployment problem associated with the real-time information-based control of vehicular traffic systems. While the study experiment address only behavior model related inconsistencies, it is useful to analyze simultaneously the state inconsistency effects related to traffic flow and O-D demand parameters.
Table 6.1 Behavioral *if-then* rules for the rule-based controller-estimated driver behavior model

<table>
<thead>
<tr>
<th>Category</th>
<th>Rule #</th>
<th>LHS</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controller-estimated driver-expected travel time (<em>TT</em>)</td>
<td>1</td>
<td>If <em>TT</em> is Very Low (VL)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>If <em>TT</em> is Low (L)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>If <em>TT</em> is Medium (M)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>If <em>TT</em> is High (H)</td>
<td>Driver will probably not choose the alternative (PN)</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>If <em>TT</em> is Very High (VH)</td>
<td>Driver will not choose the alternative (N)</td>
</tr>
<tr>
<td>Route complexity (<em>NN</em>)</td>
<td>6</td>
<td>If <em>NN</em> is Very Low (VL)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>If <em>NN</em> is Low (L)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>If <em>NN</em> is Medium (M)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>If <em>NN</em> is High (H)</td>
<td>Driver will probably not choose the alternative (PN)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>If <em>NN</em> is Very High (VH)</td>
<td>Driver will not choose the alternative (N)</td>
</tr>
<tr>
<td>Prescriptive information (<em>Y</em>) for more responsive drivers</td>
<td>11a</td>
<td>If <em>Y</em> is Route is Recommended (RR)</td>
<td>Driver will choose the alternative (O)</td>
</tr>
<tr>
<td></td>
<td>12a</td>
<td>If <em>Y</em> is Route Was Recommended (RWR)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>13a</td>
<td>If <em>Y</em> is Route is Not Recommended (RNR)</td>
<td>Driver will not choose the alternative (N)</td>
</tr>
<tr>
<td>Prescriptive information (<em>Y</em>) for less responsive drivers</td>
<td>11b</td>
<td>If <em>Y</em> is Route is Recommended (RR)</td>
<td>Driver will probably choose the alternative (PO)</td>
</tr>
<tr>
<td></td>
<td>12b</td>
<td>If <em>Y</em> is Route Was Recommended (RWR)</td>
<td>Driver is indifferent to the alternative (I)</td>
</tr>
<tr>
<td></td>
<td>13b</td>
<td>If <em>Y</em> is Route is Not Recommended (RNR)</td>
<td>Driver will probably not choose the alternative (PN)</td>
</tr>
</tbody>
</table>
Table 6.2 Calibration control *if-then* rules

<table>
<thead>
<tr>
<th>Rules for weights 1, 2, 6, 7, 11a, 11b, 12a</th>
<th>Error ($e$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The consequent (RHS) is O or PO</td>
<td>NL</td>
</tr>
<tr>
<td>Change</td>
<td>NL</td>
</tr>
<tr>
<td>in Error</td>
<td>NS</td>
</tr>
<tr>
<td>$(\Delta e)$</td>
<td>ZR</td>
</tr>
<tr>
<td></td>
<td>PS</td>
</tr>
<tr>
<td></td>
<td>PL</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rules for weights 3, 8, 12b</th>
<th>Error ($e$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The consequent (RHS) is I</td>
<td>NL</td>
</tr>
<tr>
<td>Change</td>
<td>NL</td>
</tr>
<tr>
<td>in Error</td>
<td>NS</td>
</tr>
<tr>
<td>$(\Delta e)$</td>
<td>ZR</td>
</tr>
<tr>
<td></td>
<td>PS</td>
</tr>
<tr>
<td></td>
<td>PL</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rules for weight 4, 5, 9, 10, 13a, 13b</th>
<th>Error ($e$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The consequent (RHS) is N or PN</td>
<td>NL</td>
</tr>
<tr>
<td>Change</td>
<td>NS</td>
</tr>
<tr>
<td>in Error</td>
<td>ZR</td>
</tr>
<tr>
<td>$(\Delta e)$</td>
<td>PS</td>
</tr>
<tr>
<td></td>
<td>PL</td>
</tr>
</tbody>
</table>

where:
- NL = Negative large
- NS = Negative small
- ZR = Zero
- PS = Positive small
- PL = Positive large
Figure 6.1 Conceptual framework for the behavior-consistent real-time traffic routing and calibration problem
Figure 6.2  Membership functions for the controller-estimated driver behavior model
Figure 6.3 Conceptual framework for the controller-estimated driver behavior model
Figure 6.4  Network loading for roll period of stage $\sigma$ using the controller-estimated driver behavior model.
Figure 6.5 Conceptual framework for the fuzzy on-line calibration model
Figure 6.6  Average percentage difference between the observed and estimated/SO traffic counts for the 1st day for the BC-info-CS scenario
Figure 6.7  Cumulative system travel time savings benchmarked against the no-information case (base-case) for the 1st day
Figure 6.8  Weights for behavioral if-then rule 11a for the 1st day
Figure 6.9  Proportion of drivers that must be recommended to take specific routes for “less responsive” drivers
Figure 6.10  Average percentage difference between the observed and estimated/SO traffic counts for the 2nd day for the BC-info-CS scenario
Figure 6.11  Cumulative system travel time savings benchmarked against the no-information case (base-case) for the 2nd day
Figure 6.12 Weights for behavioral if-then rule 11a for the 2nd day
7. CONCLUSIONS

This chapter presents concluding comments on this research, highlights its significance, and suggests directions for future research. Section 7.1 summarizes the research and discusses associated conclusions. Section 7.2 highlights the significance of the research, and Section 7.3 discusses possible extensions and directions for future research.

7.1 Summary and Conclusions

This study proposes a behavior-consistent approach to determine real-time information-based network control strategies. It integrates in a stage-based rolling horizon deployment framework several components that are required to adequately model the real-time information-based traffic routing problem addressed by the behavior-consistent approach. It integrates a DTA model, an iterative search based optimization procedure consisting of a fuzzy control model and a controller-estimated driver behavior model, an H-infinity filtering approach, a fuzzy on-line calibration model, a traffic flow simulator model as a proxy for field conditions, and a fuzzy multinomial logit model to represent actual driver behavior. In the real-world deployment context, the last two models are not required as actual field data is available. The DTA model provides the desired system states for the objective considered; for example, the SO solution. The optimization procedure determines the behavior-consistent information strategies in terms of the proportions of drivers that must be recommended to choose routes and/or the linguistic message to provide to individual drivers so as to direct the traffic system as close as possible towards the desired states. The H-infinity approach enhances the computational efficiency of the optimization
procedure. The fuzzy on-line calibration model ensures consistency between the estimated and the realized system states.

The study results highlight the ability of the fuzzy control model to handle the multidimensionality and nonlinearity of the problem. They also illustrate the importance of using a behavior-consistent approach to determine the information-based network control strategies. The results show that the system travel time savings under the behavior-consistent approach are significantly higher than those under standard DTA-based deployment approaches. Further, the associated information strategies entail higher compliance rates compared to the standard DTA approaches. This implies that the behavior-consistent approach generates more effective and behaviorally realistic deployment strategies. A detailed analysis of the experiment results suggests that many driver-preferred routes tend to have large behavior-consistency gaps because large numbers of drivers take these routes independent of information provision. This implies that to direct the system towards a desired state, the controller may have to recommend more or less drivers to take some routes relative to the desired proportions depending on the network dynamics and driver behavior. Further, the results show that the standard DTA approaches may deteriorate the system performance relative to even the no-information case as they tend to pre-specify driver behavior or assume artificial compliance rates.

The study exploits the advantages of fuzzy logic methodology in terms of using simple if-then rules and computational efficiency for deployment. The methodology also enables the simultaneous consideration of quantitative and linguistic information variables, as information strategies can be prescriptive or descriptive. Further, the calibration (optimization) of the membership function parameters of the fuzzy control model is required only to enhance computational efficiency and not the solution accuracy. Hence, even default parameters can provide the desired solution, though at a lower rate of convergence. An off-line H-infinity filtering approach is proposed to enhance computational performance by responding to nonlinearities and noise uncertainty.
Deployment flexibility is enhanced using different types of paradigms to increase the number of driver-preferred routes considered by the controller for routing. The results suggest that the controller objectives, the number and quality of routes used by the controller to recommend routes, and the driver-preferred route choice set augmentation and associated route types, can have varying impacts on performance. The behavior-consistent approach discussed heretofore can have deployment limitations for some O-D pairs as sufficient numbers of controllable routes may not be available due to strict requirement of full overlap between the controller-desired and driver-preferred routes. This requirement is relaxed to develop alternative control paradigms that can increase the set of controllable routes. The associated experiment results illustrate that trade-offs exist between the number of controllable routes and the quality of those routes.

A key aspect in the determination of information strategies is the level of accuracy in the models used to estimate driver behavior. In the behavior-consistent approach, this implies the robust estimation of the parameters of the controller-estimated driver behavior model. This study proposes a fuzzy on-line calibration model to enhance the accuracy of the controller-estimated driver behavior model and consequently the effectiveness of the information strategies. The results highlight the importance of the on-line calibration of the behavioral parameters. They also suggest that inaccurate parameters can potentially magnify negative effects associated to erroneous information strategies.

It should be noted here that the actual behavior at the individual driver level is currently an inferred quantity in the real-world, though technologies such as global positioning systems can substantially aid in modeling it. That is, in the future, when these technologies are adequately deployed and privacy-related policies are developed, the ability to track individual drivers can provide robust models of actual behavior as well as controller-estimated behavior. In this study, the actual behavior model is deliberately assumed to have a different structure compared to the controller-estimated model. This is to ensure that the study insights are based on conservative analyses and to imply that the actual behavior model is unknown to the controller. However, the
controller can estimate the linkages between various factors and aggregate level behavior using past studies and historical data.

7.2 Research Contributions

This research provides an alternative methodological perspective to address the complex deployment problem associated with the real-time information-based control of vehicular traffic systems. It develops a behavior-consistent approach to determine effective real-time information-based network control strategies. The proposed approach represents an anticipatory mechanism that is robust in terms of both the traffic flow and behavioral aspects. The associated information strategies are behavior-consistent because there is an explicit estimation of the drivers’ likely response behavior while determining them. Thereby, the proposed approach circumvents realism issues with existing models that pre-specify driver response behavior and/or assume artificial compliance rates. The concept of behavior-consistency gap is proposed to illustrate the need for such strategies and to highlight the behavioral inadequacies of existing DTA modeling approaches and their deployment paradigms. The behavior-consistent strategies are also consistent with the objectives of a controller of enhancing overall system performance, as they direct the system as close as possible towards an ideal system state (such as a SO or UE DTA solution). Thereby, the resulting information strategies address the controller objectives and driver behavior tendencies simultaneously, and are more likely to be accepted by drivers.

The behavior consistent approach includes several elements that represent important contributions to the literature. It develops the concept of route classification based on the relevance of routes to the controller and the routes considered by the drivers. This leads to the definition of controllable routes, which provides a realistic and effective deployment mechanism to enhance network performance and driver compliance in a behavior-consistent manner. It is designed to circumvent a key deployment concern expressed for traditional DTA models which is the possibility that drivers are provided “sub-optimal” routes from a user perspective, affecting their level of trust with the
controller-recommended routes. Another key contribution is the development of a priority scheme that enables to determine whom to provide information based on the existing routes of drivers, their behavioral characteristics, past controller recommendations, and ambient traffic conditions. This prevents unnecessary route switching and increases the trust in controller recommended routes. The solution framework enables the simultaneous determination of prescriptive and linguistic (descriptive) information that are consistent with each other. This is important in the context of currently deployed ATIS systems which commonly employ linguistic labels (for example, through variable message signs) to deploy the information.

A primary contribution to the literature is that the proposed approach explicitly considers network dynamics and driver behavior. That is, the information strategies and the system states depend on both driver behavior and traffic flow dynamics resulting from individual driver route choice decisions. The simultaneous consideration of controller objectives and driver behavior is essential to identifying realistic and superior information-based control strategies. Here, “realistic” implies that the expectation of the controller in terms of the likely response of drivers to the route recommendations is a reasonable representation of the evolving network conditions. The superior performance is due to the explicit assurance of behavior consistency, thereby preventing the possibility that the controller may over-recommend or under-recommend routes, or recommend routes that are not considered by the drivers.

Computational efficiency is an important aspect in this problem context because the information strategies are required in sub-real time to be real-time deployable. The proposed off-line H-infinity filtering approach enhances the on-line computational performance of the fuzzy control model. The optimized fuzzy control model enables the determination of the behavior-consistent information based control strategies in significantly less computational time than when the default controller is used.

A key insight from this study is that there are trade-offs between the number of driver-preferred routes considered by the controller for routing and the quality of routes relative to the controller objective. They manifest during the investigation of alternative control paradigms. The results suggest that while a larger percentage of UE routes match
driver-preferred routes, the inherent quality of the SO solution has intrinsic value. That is, even when routing is performed in a behavior-consistent manner, higher compliance rates need not necessarily translate to better system performance. This questions the justification of UE DTA solutions for route guidance on the ground that a SO strategy is not behaviorally sustainable or implies unfair routing recommendations. A fundamental corollary is that focusing primarily on robustly addressing either driver behavior (enhancing compliance) or controller objective (quality of routes) while representing the other aspect in a rudimentary manner is not adequate to ensure effective performance in the real-world. That is, approaches driven primarily by either network traffic flow modeling or behavior modeling do not suffice, implying the need for models with explicit supply-demand integration.

This study calibrates on-line the parameters of the controller-estimated driver behavior model by comparing the time-dependent actual and estimated system states in terms of link traffic counts. The proposed calibration model fits seamlessly within a rolling horizon framework to deploy the behavior-consistent approach. Thereby, the framework determines the information strategies and updates the parameters associated with the controller-estimated driver behavior model. The generalized structure of the calibration component enables it to simultaneously incorporate other sources of state inconsistency such as traffic flow and O-D demand model parameters. It provides the ability to more accurately predict drivers’ likely route choices by using aggregate if-then rules, and consequently, aggregate level data. This is attractive in a deployment context as it implies reduced data needs at a disaggregate level, a difficult proposition in the real world.

7.3 Future Research

This research proposes new concepts and consequent contributions to the route guidance literature. Due to its novelty, the primary emphasis of the research is on developmental work to illustrate the proof-of-concept. Hence, there are several venues for future research to extend the proposed approach vis-à-vis deployment,
generalization, and additional functional capabilities. The behavior-consistent approach discussed here addresses personalized information strategies. As generic information is also a popular information dissemination mechanism, it is useful to extend the solution framework to simultaneously determine and disseminate multiple types of information. An advantage of the proposed fuzzy logic based methodology is its amenability to seamlessly incorporating linguistic labels which characterize generic information.

The behavior-consistent approach requires driver-preferred route sets. In this research, these sets are assumed to be static, though Chapter 5 performs some analysis involving augmentation of these sets. In reality, it is likely that these sets are dynamic over a longer timescale. That is, driver learning process can lead to routes entering or exiting the driver-preferred route sets. It will be useful to analyze these route set dynamics within a day-to-day framework.

The day-to-day framework aspect also arises in the context of the evolution of driver behavioral tendencies based on the driver’s experiences over time with the controller-recommended routes. While the proposed calibration model can capture the changes to the behavioral parameters over time based on the evolving actual behavior of drivers, it is nevertheless useful to perform the within-day dynamics and the day-to-day learning in a single unified framework. Such a framework will provide an important analytical tool to compare various types of route guidance strategies in terms of their long-term effectiveness.

In traditional route guidance, drivers are recommended a single route in a prescriptive context. The characteristic of the behavior-consistent approach raise innovative new possibilities of recommending more than one driver-preferred route to a driver. This could lead to a higher likelihood of achieving the desired performance because there is a higher probability that the driver chooses one of these recommended routes. However, this may result in a broader range for system performance. Also, there are trade-offs in terms of the maximum number of routes that can be provided in real-time from a human information processing standpoint. These issues can be explored by extending the proposed behavior consistent approach.
Another paradigm that can be explored by extending the behavior-consistent approach is in terms of modifying how traditional DTA models are used in the solution framework. The SO or UE solutions can be computed by precluding routes with low degrees of overlap from being included in the desired solution. Though these quasi SO or UE solutions are different from and underperform the corresponding standard solutions, the resulting information strategies may result in better system performance under some circumstances when used as part of the behavior-consistent solution framework to represent the desired states. This is because using the quasi SO or UE solutions will most likely lead to more controllable routes. However, further research is required to test this hypothesis as trade-offs between number of controllable routes and their quality tend to exist.

This study assumes that driver departure times are fixed. The proposed approach can be extended to simultaneously consider route choices and departure times.

The on-line calibration focuses only on behavioral parameters. As stated in Chapter 6, since the model can simultaneously address the calibration of traffic flow and demand parameters as well, it will be insightful to study the effects of multiple sources of inconsistencies.
LIST OF REFERENCES
LIST OF REFERENCES


VITA
VITA

Alexander Paz was born in Popayán, Colombia on January 23, 1978, the son of Jairo Paz and Maricela Cruz. After graduating from the T.C.M. Military Academy, Colombia, he entered the University of Cauca, Colombia in July, 1994. He received his Bachelor of Science degree in Civil Engineering from University of Cauca, in October 1999. In January 2000, he entered the Master of Science in Civil Engineering program at the University of Puerto Rico, Mayagüez. In August 2002, he joined the Ph.D. program in the Transportation and Infrastructure Systems area in the School of Civil Engineering at Purdue University.

Permanent address: Calle 3N # 10-33
Villa Paula, Popayán, Colombia