Tools for modeling and control of freeway networks

by

Ajith Muralidharan

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering - Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Roberto Horowitz, Chair
Professor Francesco Borrelli
Professor Pravin Varaiya

Fall 2012

Tools for modeling and control of freeway networks

Copyright 2012
by
Ajith Muralidharan

Abstract

Tools for modeling and control of freeway networks

by

Ajith Muralidharan

Doctor of Philosophy in Engineering - Mechanical Engineering

University of California, Berkeley

Professor Roberto Horowitz, Chair

This dissertation presents algorithmic tools that are useful to transportation engineers for freeway traffic modeling and control. A modeling framework that utilizes the link-node cell transmission model (LN-CTM) to simulate traffic dynamics on a chosen freeway network is presented here. A data driven approach, which utilizes available detector measurements on the freeway network to calibrate and specify the model is also illustrated. Flow measurements in ramps, which are needed to specify demands and routing characteristics for the freeway, are usually not available. Two novel imputation algorithms which estimate the missing ramp flows in the freeway network are presented. These algorithms employ a model based estimation procedure, that calculates the unknown on-ramp flows and off-ramp split ratios which can be fed into the model to match the observed mainline density and flow measurements. A detailed analysis of the convergence of these algorithms is presented, along with the advantages of these individual approaches. The final model, specified with the imputed ramp flows is able to replicate the traffic dynamics with good accuracy, as seen by error rates around 5-8% for density/flows contours, and the accurate replication of the bottleneck locations. These imputation algorithms, used within our modeling framework, enables a user to build a freeway model simulating multiple days of freeway behavior, within a week.

A model based optimal predictive controller for freeway congestion control, which utilizes the LN-CTM as its underlying model is also presented. The approach searches for solutions represented by a combination of ramp metering and variable speed limits. The optimization problem corresponding to the optimal control problem based on the LN-CTM is non-convex and non-linear. A relaxation method is presented to solve this problem efficiently using an equivalent linear program, before generating the solution to the original problem using a new mapping algorithm. The predictive controller is also extended to cover situations when ramp weaving and/or capacity drop exists in the freeway network. In this case, a set of heuristics are presented and the optimal control problem is solved using a sequence of linear programs, before mapping the solutions back to the original problem.

To my family,

Contents

List of Figures

List of Tables

Acknowledgments

This has been a wonderful journey, thanks to the support of my advisor, family and friends. My graduate life at Berkeley has been full of great experiences and happy memories.

Roberto Horowitz has been a great advisor. His patience, dedication and unlimited enthusiasm have been instrumental in shaping me as a researcher. He has been a perfect mentor, giving me freedom to pursue my research and intellectual interests along with good guidance and advise when necessary. I am truly amazed by his ability to mentor students and conduct research despite his other assignments.

I have also been extremely fortunate to work with Pravin Varaiya. His critical thinking and depth of knowledge has always amazed me. I admire his ability to infuse both practicality and rigorousness in defining research problems, and this is one of his many qualities that I aspire to emulate. I greatly appreciate his advice and encouragement over these years.

I take this opportunity to thank Professor Francesco Borrelli, for serving in my dissertation and qualifying exam committee. I am also indebted to professors and teachers at Berkeley and IIT Madras, for imparting their knowledge through lectures, discussions and seminars.

I thank all my friends in the TOPL team for their support over the years. Gabriel Gomes and Alex Kurzhanskiy have been great mentors and friends. Rene Sanchez and Gunes Dervisoglu have been wonderful friends on and off work. I will always cherish our weekly TOPL meetings, which have been great source of inspiration and intellectual discussions.

This research was funded by Caltrans and NSF through the TOPL project, and I am grateful for the financial assistance. I also acknowledge John and Janet McMurtry Fellowship and the John G. Maurer Fellowship for providing support to conduct part of this research.

This journey would not be possible without the love of my parents. They have been instrumental in encouraging me to pursue graduate studies. I have also been fortunate to have Arvind and Sukanya living nearby, and I am grateful for their support and company. I also thank my grandparents for their love. Finally, I thank my many friends at Berkeley over the years who have made my life outside work very memorable.

Chapter 1

Introduction

Traffic congestion can be encountered in metropolitan areas during various time periods across the day or sometimes during the night. Congestion levels have also been increasing over the last decade, due to ever increasing demand. An average commuter experiences recurrent congestion during his commute due to presence of system bottlenecks. In addition, non-recurrent events, both planned (road work, public events) and unplanned (accidents) contribute increasingly to the unreliability in commute times. The 2011 annual urban mobility report [15] compiled by the Texas Transportation institute calculated that the average commuter experienced 34 hours of delay in 2010, up from 14 hours in 1982. In 2010, congestion costs accumulate over $100 billion dollars, which is more than $750 per commuter.

The easiest way to combat congestion is through infrastructure expansions. Adding additional lanes and new freeways are not always feasible due to economic and environmental concerns. In many cases, freeway and roadway expansions might be infeasible due to lack of construction space. As a result, transportation engineers increasingly rely on intelligent operational management of the existing infrastructure to improve system efficiency. Over the years, different measures have been implemented, ranging from tolls, congestion pricing, introduction of HOV lanes and freeway control techniques like ramp metering. Simultaneously, transportation authorities also rely on less direct measures like improving transit, providing better information etc to combat congestion.

Transportation authorities can plan and execute strategies over various time scales: long term, medium/short term and real-time. In the long term, transportation systems planning is intertwined with urban planning, policies etc.. Transportation planners project future population patterns and transportation demands to plan for long term capacity expansions. In the medium/short term, planners and engineers evaluate and design operational management strategies to combat current as well as short term projected congestion. For example, system operators might install and enable ramp meters to control freeway traffic, decide on tolls/congestion pricing. Finally, in real-time operations, traffic engineers are expected to take short term measures and counter measures to combat congestion, traffic incidents, as well as planned events. For example, traffic engineers might warn users using variable message signs regarding incidents on their commute route, as well as possible reroute options. They might also deploy incident specific countermeasures to decrease their impact.

Transportation planners, system operators as well as traffic engineers are increasingly relying on traffic flow simulators to aid in the planning and operational management. Traffic simulators provide a cheap and non-intrusive method to plan and study operational strategies before their implementation in the field. Transportation planners frequently use traffic simulators to perform cost-benefit analysis among these various options available to justify short/medium term strategies. Traffic simulation based planning/decision support tools are also seen as essential for improved real time management of transportation systems.

Tools for operations planning (TOPL) [67] is a Caltrans/NSF funded project started at PATH and UC Berkeley on April 2006, to provide simulation based support for operational planning and real time management of freeway traffic corridors. Freeway corridors usually include freeways and nearby arterials, providing a self contained road system that can be consistently analyzed when the chosen operational strategies are implemented. TOPL provides a quantitative assessment of the effects of operational strategies designed to improve traffic congestion in freeway corridors. The main elements of such strategies can be classified into

Traffic control - Employ congestion control through the use of ramp metering, and possibly variable speed limits.

Demand Management - Reduce or redistribute current road network demand in space and time.

Incident Management - Alleviate congestion related to planned and unplanned incidents (road work, accidents etc).

Traveler Information - Provide up to date current and predictive information of traffic conditions for trip planning and routing.

At the centerpiece of TOPL is a fast and trusted simulator, Aurora. This software executes simulations magnitudes faster than real-time, providing the operator the ability to simultaneously execute multiple operational management strategies and predict their effect in real time. Geometric models of corridors are built using Network editor [68], which has been built to leverage maps provided by **Google Maps**.

This dissertation is motivated, and at many times, guided by the development of TOPL. All of the theory and algorithms developed in this dissertation has been implemented as a part of various tools which are a part of TOPL. The first part of this dissertation is related to building calibrated models of freeways using measured data from the freeway. Calibrated models of the managed freeway sections are essential to ensure that realism of the simulations used as a part of TOPL. We will describe algorithms that aid in building these models from measured data, under conditions that some of the data is usually missing.

The second part of this dissertation presents the design of model based optimal control of freeway network, used as an operational strategy for congestion management. Freeway control, using ramp metering and variable speed limits are being increasingly adopted as the first step to combat congestion in freeway networks. Optimal controllers provide the “best” achievable congestion reduction, and we present solution techniques that be used to solve the optimal control problem efficiently. As we will explain later, this optimal controller is a useful tool for simulation based operational management, and also for implementation in the field.

Dissertation Outline

This dissertation is organized as follows. In Chapter 2, we review different concepts and previous research related to the material presented in this dissertation. First, we review the vehicle detection technologies and the traffic measurements available from commonly deployed vehicle detectors. We also review the Performance Measurement System (PeMS), before highlighting the quality of data available and data imputation schemes used to fill in missing values. We also review traffic flow models including the Cell Transmission Model. Finally we describe the commonly used freeway control mechanisms: ramp metering and variable speed limits before describing goals of freeway control as well as the metrics for evaluation.

In Chapter 3 we introduce the Link Node Cell Transmission Model (LN-CTM), along with the dynamic equations that can be used to model traffic dynamics in freeway networks. We also describe the methodology used to build a model of a chosen freeway, including geometry specification and parameter calibration. In this chapter, we motivate the necessity and importance of developing imputation algorithms for estimation of on-ramp flows and off-ramp splits in the freeway. We present simulation results obtained using a calibrated model of the I-80E freeway, constructed using the procedure explained in this chapter.

In Chapter 4, we describe an imputation algorithm based on the Asymmetric Cell Transmission Model (ACTM). The ACTM is a simplified model based on the CTM, described by piecewise affine dynamic equations. This simplified dynamics allowed the development of the first provably convergent imputation algorithm that can be used to estimate on-ramp flows and off-ramps split ratios which are not measured.

In Chapter 5, we present an imputation algorithm based on the LN-CTM, which is a more accurate representation of freeway dynamics when merging flows from on-ramps are appreciable. The LN-CTM contains state as well as input non-linearity, and these present new challenges as compared to the imputation algorithm based on the ACTM. The new imputation algorithm estimates the unknown ramp flows in two steps: first matching the densities along the freeway, and then matching the available flow measurements. We present, in detail, the convergence properties of the algorithm, and also demonstrate the application of the algorithm.

We present model based predictive controllers for traffic congestion control in Chapters 6 and 7. These controllers regulate traffic flow in the freeway through the use of ramp metering and variable speed limits. In Chapter 6, we present the optimal control problem, utilizing the LN-CTM model presented in Chapter 3. The solution of the original optimal control problem involves nonlinear optimization, due to the presence of nonlinear, non-convex constraints. We present a relaxed linear program, whose solution can be mapped to a feasible solution of the original optimal control problem. We prove that this solution is the globally optimal solution of the original optimization problem.

We extend the optimal controller in Chapter 7 when the freeway network experiences weaving and/or capacity drop. We supplement the LN-CTM by integrating a node based weaving model and a discontinuous capacity drop model. The solution technique presented in Chapter 6 cannot be directly applied due to the presence of capacity drops, and we present a set of heuristics that allow us to solve the optimal control problem using a sequence of linear programs. We demonstrate the application of the model predictive controller on a simulated example, and discuss the characteristics of the controller.

A preliminary version of the results in Chapter 4 have been presented in [41]. [40] describes our first imputation algorithm utilizing the LN-CTM. The algorithm presented in Chapter 5 has been modified in view of obtaining favorable convergence properties. Preliminary analysis of a part of this algorithm is published in [42]. We have also presented some of the results described in Chapters 6, 7 in [43, 44].

Chapter 2

Review of Related Work

In this chapter, we review concepts and literature related to the material covered in this dissertation. Additional references are also provided for readers interested in exploring the material in detail.

2.1 Traffic detection and data archival

Traffic state measurement and data archival are crucial components of any intelligent transportation infrastructure designed for operational management of traffic networks. Transportation planners frequently require a rich source of historical data to plan for long term as well as medium term projects. A good quality data set will allow these planners to budget and invest resources in projects that have high benefit to cost ratio. Traffic engineers also use real time data in the day to day operations of the traffic network. For example, traffic engineers informed of unforseen congested traffic conditions can plan countermeasures in real time to better manage the traffic system.

Vehicle Detection

A broad range of vehicle detection technologies exist to measure the traffic properties in road networks. These may be classified as intrusive or non-intrusive, depending on whether they need to be embedded or installed in the road pavements. We list a few commonly found detection technologies below.

Inductive loops Inductive loops use induced eddy currents to determine the presence of vehicles on top of the detectors. These are the most commonly deployed vehicle detection system.

While their detection accuracy is high, their deployment causes significant traffic disruption and they are highly susceptible to malfunction.

Magnetometers Magnetometers detect vehicles using the changes in magnetic fields caused by their presence on top of the detector. Increasingly, these are being deployed to replace loop detectors. They offer advantages in reducing deployment times, and also provide built in failure detection features.

Pneumatic Tubes Pneumatic tubes, deployed perpendicular to the road, detect vehicle axles when tires run over the tubes. These are non-intrusive, quick to install and mostly deployed for short term studies. Tube wear prevents their successful deployment for long term traffic monitoring.

Video Detection Video cameras can be used to detect vehicles by analyzing successive images to detect vehicle presence and movement. These can be used for vehicle presence detection or vehicle tracking. Their main advantage is that a single camera can mimic the operation of multiple loops. However, they are susceptible to bad weather, shadows, vehicle occlusion and also require routine cleaning and maintenance.

Microwave Radars Microwave radars transmit energy towards an area of the roadway and measure reflected energy to detect vehicle presence and track their movement. They have better performance over video cameras as they are not susceptible to weather or light conditions.

Some of the other specialized detectors deployed are license plate readers, toll tag readers, weigh-in motion detectors. The traffic detector handbook [35] is a extensive reference for various detection technologies commonly deployed in the field.

Traffic state measurement

The detectors mentioned previously provide a rich source of traffic measurements. The most commonly obtained traffic measurements are

Occupancy Occupancy is the fraction of time when the detection zone of the sensor is occupied by a vehicle. Inductive loops, magnetometers, microwave radars and video detection systems provide this measurement. In case case of radars and video, it is possible to configure multiple detection zones (of variable sizes) and measure traffic occupancies simultaneously.

Volume Volume is defined as the total number of vehicles that pass over the detection zone of the sensor during a chosen interval of time. Inductive loops, magnetometers, microwave radars and video detection systems provide direct measurement of volume counts. Pneumatic tubes, on the other hand, can only be used to estimate volume counts indirectly, since they provide axle counts.

Speed Detectors are also deployed to measure point speeds of vehicles as they pass the detection zone of the sensors. Dual inductive loops/magentometers, separated by a known distance, are commonly deployed to measure speeds indirectly by noting the activation times of each detectors. Video detection systems also employ loop emulation to measure speeds, but new systems are also capable of tracking vehicles. Microwave radars, based on doppler principle, can directly measure vehicle speeds.

Some detectors also provide specialized traffic information. Microwave and video detection equipment can be deployed in intersections to obtain queue length estimates of vehicles waiting at the signals. Toll tag readers, bluetooth readers and license plate readers are commonly used to re-identify vehicles from one point to another, and these measurements are commonly used to derive origin destination demand samples as well as travel time samples. Vehicle re-identification has also been successfully demonstrated for short distances though the use of magnetometers, to measure travel time in arterials [34]. A recent rich source of traffic data has been GPS measurements from mobile smartphones and dedicated GPS devices. These devices can be used to obtain measurements of vehicle speeds and travel times. Details of a recent academic effort in this area can be found in [1].

Traffic data is collected from these detectors by local controllers which usually process and use this data for local control actions. For example, an actuated intersection may use detection events to trigger traffic signal light changes. If the controller is connected by a network (wireless or wired), then the data is transmitted to the Traffic management center (also known as a traffic operations center in some states in the U.S), where it can be processed and used for daily operations. This data may also be aggregated in a database for archival.

The Performance Measurement System

The Performance Measurement System (PeMS) is a traffic data archival system used in the state of California [7, 57]. This system collects real time data from over 25000 detectors in California (as of 2012), spanning over multiple freeways located in the major metropolitan areas of California. Data from other sources like weigh-in motion stations, tag readers and CHP (California Highway patrol) incident feeds are also aggregated in this system. The collected data is filtered, processed and stored in databases, and users can access data sources going back to 1998. This data is used to calculate performance measures of the freeways as well as access the traffic conditions both historically as well as in real-time. PeMS has been extensively used by traffic engineers, planners, traveler information services as well as academics over its years of operation.

In the case of loop detectors, PeMS receives 30s occupancy and flow values from the vehicle detector stations (vds) which house these loop detectors. In the case of dual loops, PeMS also receives velocities. PeMS aggregates the data into 5-minute intervals, and also estimates vehicle speeds for single loops through the use of g-factors [27]. PeMS has detailed diagnostic measures to ascertain the operation of the detector and the quality of the data. PeMS also has an imputation algorithm, which fills in the missing data with plausible values. Finally, this data is used to compute aggregate performance measures, including Vehicle Miles Traveled (VMT), Vehicle Hours Traveled (VHT), Congestion Delay, and the productivity ratio ‘Q’ (ratio of VMT over VHT) interpreted as the average speed in the region of analysis.

Detector health and data quality

Good detector health and data quality are important to ensure that the acquired data can be used for modeling and performance analysis. PeMS has a bank of filters to evaluate the quality of data and flag failed detectors. PeMS expects loop detectors in California to report occupancy and volume counts every 30 seconds. Additionally, PeMS screens the data for sample quality metrics and decides whether the detector is functioning or malfunctioning. Particulary PeMS screens and identifies the following common issues .

Data never received Many of the detectors never report data, possibly due to communication line, power or detector failure.

Too few samples Very limited number of samples were received, which indicate the the data feed is active, but the detector is functioning intermittently.

High Values and/or constant occupancy Threshold detectors are used to indicate whether the detector reports unrealistic flows and occupancies. Constant occupancies indicate that the detector is stuck on high.

Zero flow/occupancy and/or flow-occupancy mismatch Data tests in 5 min aggregated data can be used to indicate too many samples with zero occ/flow.

Based on these simple data quality filters, it is not uncommon to find that 40% of the detectors which report data are flagged by PeMS as failed detectors [59]. Moreover, there are a lot of sensors which are active, but unlisted in PeMS. Particularly, detectors at on-ramps and off-ramps are usually found missing in the PeMS system for many freeways.

Imputation of missing/bad data

Imputation is the process of filling in missing data with plausible values. The problem of estimation of missing detector counts is commonly addressed by various data imputation techniques, which allows the use of partially complete data for performance analysis and modeling. For detectors along the freeway mainline, simple imputation schemes typically replace missing data by spatial averages (of nearby detectors) or historical averages of the detector data. Modern techniques place higher emphasis on developing a good statistical model for estimating missing data. Dailey [14] introduced an Kalman filtering based data smoothing, error detection and imputation.

Many commonly used imputation techniques employ the standard techniques for imputation proposed in statistics [63]. Some of these include algorithms based on Expectataion-Maximation [16], Multiple imputation [45] or Principle component analysis[58]. PeMS uses the imputation algorithm developed by Chen et al. [9] to replace missing data counts. Given large databases of historical data, linear regression models are built to predict missing data in freeway mainline loop detector stations using data from detectors upstream/downstream or in nearby lanes. Most of these techniques perform best when data is missing for short time intervals and also when sufficient historical data is available. Apart from these statistical methods, model based imputation has been explored in [23]. The authors use traffic flow theory by applying the Lighthill-Whittam Richards (LWR) first-order model. This model is used to generate missing measurements in locations along the freeway mainline using detector data available at an upstream and downstream locations.

All of the techniques presented in literature address the imputation of missing measurements in detectors present on the mainline. However, it is very frequently observed that on-ramp/off-ramp data is missing. In some cases, detectors are not available to measure these flows while in some cases, the data feeds have not been set up. These measurements form an important input without which freeway modeling studies cannot be undertaken. The techniques presented for freeway loop data imputation is not suitable for ramp flow imputation. It is difficult to build accurate statistical models for ramp flow imputation, since one cannot guarantee a high correlation of data between neighboring ramp loop detector stations (or a nearby freeway detector). In addition, for many ramps, archived data may not be available to build models with. In this dissertation, we provide a model-based method to impute flow data for ramps, using the available measurements in the freeways.

2.2 Traffic Models

There are two different approaches to traffic flow modeling - microscopic models and macroscopic models. In microscopic models, individual vehicles are modeled along with their interaction with other vehicles and the road network. These individual vehicles adjust their speeds and lanes and the interaction of all vehicles models the resulting traffic in the network. Macroscopic models ignore these individual vehicle interactions and represent the aggregate dynamic properties of a group of vehicles, usually represented as a continuum. Most macroscopic models represent traffic as a compressible fluid, and describe the density, flow and speed evolution using dynamic equations.

Macroscopic models offer various benefits in comparison to microscopic models. These models run significantly faster than microscopic models, since they do not simulate individual vehicles in the network. This is particularly beneficial when the simulation platform is used in real-time to assist the traffic operators. The process of calibration (i.e. the specification of the parameters in the models) is usually simpler in macroscopic models, since the model variables can be directly observed from measurements (i.e. flow, speeds and occupancy). In comparison, calibration of microscopic models require the user to infer individual driver characteristics from macroscopic measured variables like flow, speeds and occupancy.

In the definitions that follow, the space coordinate is represented by x, which denotes the distance along the traffic flow direction. At any cross-section, traffic properties are assumed to be uniform, and the models we will present will not capture lane change behaviors. t corresponds to the time coordinate. We will use the following definitions, adopted from the Highway Capacity Manual [69] in the materials that follow.

Speed ($v(x,t)$): Rate of motion defined as distance per unit time. Space mean speed is defined as the average speed in an infinitesimal segment around x $(x-\delta x/2, x+\delta x/2)$ at time t. The speed referenced in this section will correspond to this definition. In comparison, time mean speed is the average speed of vehicles observed passing a given point, which is usually reported by detectors.

Flow ($f(x,t)$): Total number of vehicles that pass the point x during an infinitesimal time interval $(t-\delta t, t)$, divided by the length of the time interval δt. It is obtained from volume measurements and usually expressed as an hourly rate.

Density ($\rho(x,t)$): Number of vehicles occupying a length of freeway about point x at instant t. Its measurement is difficult because it requires the observation of a stretch of road. Instead, it is often approximated from measurements of flow and speed by $\rho(x,t) = f(x,t)/v(x,t)$.

Demand: Number of vehicles (or number of vehicle occupants) who desire to use the facility during the specified period of time.

Capacity: Maximum hourly rate at which vehicles can be reasonably expected to traverse a point or a uniform section of a lane or roadway during a given time period under prevailing roadway, traffic, and control conditions.

Bottleneck: Any road element where demand exceeds capacity. Freeway bottlenecks sometimes appear near heavy on-ramps, where a localized increase in demand is combined with a decrease in capacity due to lane changing.

Macroscopic models define the evolution of density, speed and flow over space and time using a set of partial differential equations (PDEs), together with other constituent relationships. Each continuous model has a basic vehicle conservation equation (which captures the fact that vehicles cannot be created or destroyed), along with other PDEs depending on the order of the model. We review a few important models in this section.

Lighthill Whitham Richards (LWR) Model

The Lighthill Whitham Richards model, commonly known as the LWR model [36, 60], is a first order model described by the vehicle conservation equation

$$\frac{\partial \rho(x,t)}{\partial t} + \frac{\partial f(x,t)}{\partial x} = 0 \tag{2.1}$$

and the static flow-density relationship

$$f(x,t) = \rho(x,t)v(x,t) = \Phi(\rho(x,t)) \tag{2.2}$$

where the function $\Phi(\rho)$ is the fundamental diagram of traffic flow. When $\Phi(\rho)$ is differentiable, the conservation equation can also be represented as

$$\frac{\partial \rho(x,t)}{\partial t} + \Phi'(\rho(x,t))\frac{\partial \rho(x,t)}{\partial x} = 0 \tag{2.3}$$

The main assumption in the first order models is the existence of a static density flow relationship, which also implies a static speed-density relationship. Greenshields was the first to propose a parabolic fundamental diagram from observations of traffic along a two lane highway [22]. Figure 2.1 shows some of the common fundamental diagrams used in practice. In general, the fundamental diagram has the following characteristics.

1. $\Phi(0) = \Phi(\rho^J) = 0$, where ρ^J is the maximum density in the freeway which is known as the jam density.
2. $\Phi(\rho) \geq 0$ is a concave continuous function.
3. $\Phi(\rho)$ attains a maximum (F) at ρ^c, which is known as a critical density. The maximal flow F is known as the capacity.

The critical density separates the fundamental diagram into two sections - the free flow regime when $\rho \leq \rho^c$ and the congested regime $\rho > \rho^c$. Empirical measurements are well represented by a straight line during free-flow, while measurements are usually scattered in the congestion region. Some researchers have observed that there is a difference in the maximal flow (capacity) in a few sections depending on whether the freeway section is in free-flow or congestion [24, 6]. This change in flows is known as the capacity drop, and researchers estimate that it is usually in the region of 5-10% when present.

The solution to the LWR model can be given in terms of characteristics [36], which are trajectories in the space time plot whose evolution is defined using ordinary differential equations (ODEs). Flows and densities are constant along these characteristics, whose slope is given by $Q'(\rho)$. It can be seen that the characteristics have a positive slope during free-flow and a negative slope during congestion. Thus, when well-posed initial and boundary conditions are given, the

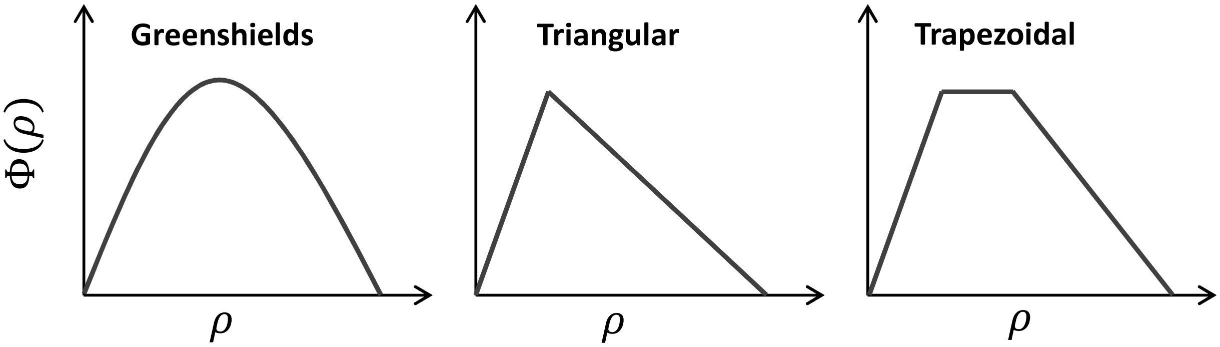

Figure 2.1: Commonly used fundamental diagrams

value of the density or flow at any point is equal to a boundary or initial condition to which the point is connected by a characteristic. The boundary value problem is well-posed if the family of characteristics emanating from the initial and boundary conditions spans the entire time/space plane. The main complication arises, when characteristics intersect in location known as shocks (discontinuities), leading to multiple values at the point of intersection. In this situation, the PDE admits only weak solutions, which satisfy a integral form of Eq. (2.3). The speed of the shock is given by

$$v_s = \frac{f_2 - f_1}{\rho_2 - \rho_1} \tag{2.4}$$

where subscripts $2/1$ denote the state of traffic infinitesimally upstream/downstream of the shock.

Cell Transmission Models

The Cell Transmission Model (CTM) was developed by Daganzo [11] as a first order discrete dynamic model which is consistent with the hydrodynamic theory of the LWR model. The CTM can be interpreted as the discretization of the LWR model with a time step of T_s and uniform sections with length L, according to $L = T_s v_f$, where v_f is the free-flow speed. The uniform sections are known as cells, and they are increasingly numbered from upstream to downstream. Figure 2.2 shows a uniform stretch of roadway divided into cells. The density of cell i (represented by $n_i(k)$) can be represented by the conservation equation

$$n_i(k+1) = n_i(k) + f_{i-1}(k) - f_i(k) \tag{2.5}$$

where $f_i(k)$ is the flow moving from cell i to cell $i+1$ during the time step k (each time step corresponds to T_s seconds). In the CTM, this flow is obtained by comparing the sending and receiving flows, also known as the demand and supply, as

$$\begin{aligned} f_i(k) &= \min\left(D_i(k), S_{i+1}(k)\right) \\ D_i(k) &= \min(F_i, n_i(k)V_i) \\ S_i(k) &= \min\left(F_i, W_i(n_i^J - n_i(k))\right) \end{aligned} \tag{2.6}$$

In the equations above, we have assumed a trapezoidal fundamental diagram. The fundamental diagrams can be different for each cell, and the parameters are indexed by the cell number i. The trapezoidal fundamental diagram is characterized by the free-flow speed (V_i, the slope of the free-flow part of the diagram), the capacity (F_i, the maximum flow), the congestion wave speed (W_i, the slope of the congested region) and the jam density (n_i^J, the maximal density). The demand function $D_i(k)$ captures the flow that can be sent from the upstream cell, while the supply function $S_i(k+1)$ specifies the maximum flow that can be received by the downstream section. The flow can be obtained by taking the minimum of the demand and the supply. From the flow equation, we say that the flow conditions in the boundary of cell i and cell $i+1$ are congested, when the demand function exceeds the supply, and otherwise the boundary is in free-flow.

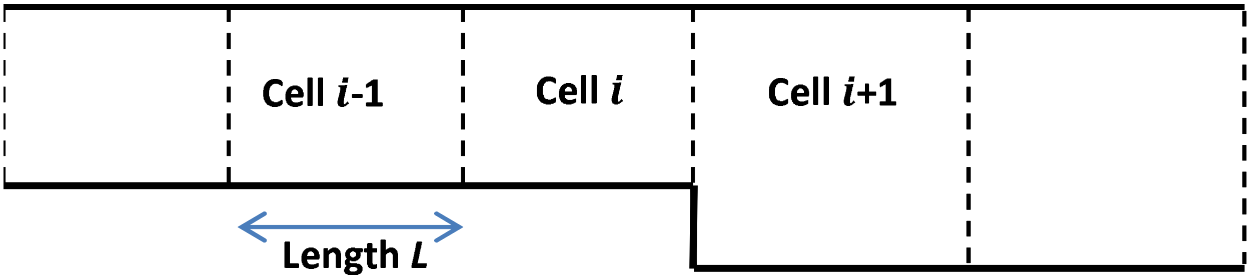

Figure 2.2: A road segment divided into cells of equal length

The CTM was extended to simulate traffic dynamics in a network with a more general topology, by introducing models for merging and diverging flows [12]. This model was also further adapted to accommodate nonuniform cell lengths and other continuous, piecewise differentiable fundamental diagram [12]. This resulted in a density-based model for the conservation equations. Despite the relative simplicity, it still captures the shock behavior predicted by the LWR.

Higher order models

In first order models, fundamental diagrams specify the static function that exists between the density and the realized flow, which results in a static speed-density relationship. As a result, the resulting density dynamics might result in large speed variations as drivers travel along the freeway, especially in the locations with shocks. Higher order models capture the fact that drivers cannot respond to speed changes instantaneously. These models augment the conservation equations with the dynamics of the space-mean speeds to provide better descriptions of the traffic dynamics. These additional PDEs, which represent the speed dynamics are often referred to as the momentum equations.

Payne [55] was the first to propose a second order traffic flow model, which he implemented in FREFLO [56], a macroscopic simulator. Paynes original model was demonstrated to allow negative speeds (wrong way travel) and to allow vehicles to be influenced by traffic upstream [13]. This model was modified and improved by various researchers (Zhang [75], Liu et al. [37], among others), to improve its traffic realism. Papageorgieu et al. [51, 30, 52] extended Payne's model to develop the METANET model for network traffic simulation. This is a one of the popular second order models reported in literature.

While second-order traffic models are claimed to be more accurate for representing traffic dynamics, they suffer from additionally complexity which makes model calibration difficult. In contrast, as shown in the next chapter, the calibration of models with the Cell Transmission Models are relatively simple. The LWR and the CTM model are generally accurate for reproducing congestion phenomena and the propagation of jams. The relatively simple traffic dynamics allows us to perform theoretical analysis and also design convergent estimation and control algorithms. In this dissertation, we use the first order models, particularly some extensions of the CTM.

2.3 Freeway traffic control

Traffic congestion in metropolitan areas has been increasing over the last decade, leading to large losses in productivity due to increased commute times. Due to significant investments involved, infrastructure expansions are not always feasible even though they provide the best means to tackle traffic congestion. As a result, transportation engineers rely on intelligent operational management of the existing infrastructure to increase system efficiency. The most commonly used operational management strategy is traffic control.

Goals and Metrics of evaluation

Before going into the control strategies, it is beneficial to review the goals of freeway control and also the metrics of evaluation. The main goals of ramp metering is to reduce freeway congestion and increase the efficiency of the freeway. One natural way to capture this is to consider the Total Travel Time (TTT), which is the sum of travel times of all the users of the freeway system. Given constant demands, any controllers performance can be captured by the magnitude of decrease of the TTT experienced by all the freeway users. The condition of constant demand can be roughly captured using the Total Travel Distance (TTD), which is the sum of distances traveled by all the users in the freeway. For reasons explained later, the application of ramp metering or control actions might change the demand patterns and the TTD of the freeways, and in this case a combination of TTT and TTD is used to evaluate the adopted control measures. Another useful metric for evaluation is the Total Congestion Delay (TCD), which is the additional time spent by all users under congested conditions as compared to conditions under which the traffic is in free-flow.

Given the density and flow along the freeway, TTT, TTD and TCD can be measured as

$$\begin{aligned} TTT &= \int_{t_1}^{t_2} \int_{x_1}^{x_2} \rho(x,t) \\ TTD &= \int_{t_1}^{t_2} \int_{x_1}^{x_2} f(x,t) \\ TCD &= TTT - \frac{TTD}{v_f} \end{aligned} \tag{2.7}$$

In the above formula for delay calculation, we have assumed a constant free flow speed v_f through the freeway. The formulaes can also be extended to capture the delay, and time spent by users in the ramps, as the wait to enter the freeway. It can be seen that it is easy to measure these quantities using detectors along the freeways. In this dissertation, we calculate all system utility/performance measures as the sum of individual vehicle utility/performance measures, as opposed to considering individual passengers. One advantage of this choice is that we can directly measure these performance measures using the detection systems commonly employed in the freeway.

Care must be taken to define the boundaries of the network chosen for computing these performance measures so that we can correctly evaluate the controller performance. Ideally, the network should be chosen such that the application of the controller does not change the boundary conditions (for eg. density and flows at the boundaries) as compared to the case against which it is compared. This is difficult to achieve in reality and it is generally acceptable that the boundaries of the chosen system are in free-flow.

To understand the working principle of the controllers, we rewrite TTT from Eq. (2.7) along the lines explained in [46].

$$TTT = (t_2 - t_1) \int_{x_1}^{x_2} \rho(x,t_1) + \int_{t_1}^{t_2} (t_2 - t) f^{in}(t) - \int_{t_1}^{t_2} (t_2 - t) f^{out}(t) \tag{2.8}$$

Here, $f^{in}(t)$ ($f^{out}(t)$) is the sum of all flows entering (exiting) the system. In the above equation, the first term corresponds to the initial conditions, which cannot be affected by freeway control. Likewise, the second term corresponds to the demands entering the system, which we assume are not affected by the controller, since the controllers we design do not change demand profiles. Thus the controllers affect the performance of the system through the third term, which is a weighted integral of the flows exiting the system. To decrease the TTT, the controller must operate the system such that vehicles exit the system as soon as possible. The least TTT can be realized when vehicles entering the system travel at free-flow speeds and then exit the system.

Ramp Metering

Ramp metering is a control method in which vehicles entering the freeways through the on-ramps are controlled. A traffic light, present at the ramp entrance regulates the traffic entering the freeway when the controller is active. Most traffic lights in ramps allow 1 or 2 cars per green. Therefore, regulating the frequency of green lights leads to the indirect control of the rate at which vehicles enter the freeway.

Ramp metering algorithms can be classified depending on the scope of their action as either local or co-ordinated. As the name suggests, local ramp metering algorithms adjust the metering rates independent of other controllers which are active. Generally, their scope and objective is limited to relieving congestion present locally around the region of the ramp where the meter is installed. In coordinated controllers, several ramp meters coordinate their actions to regulate traffic simultaneously. Ramp meters can also be classified as traffic responsive, or fixed time, depending on whether the metering actions are dependent on the traffic conditions or not.

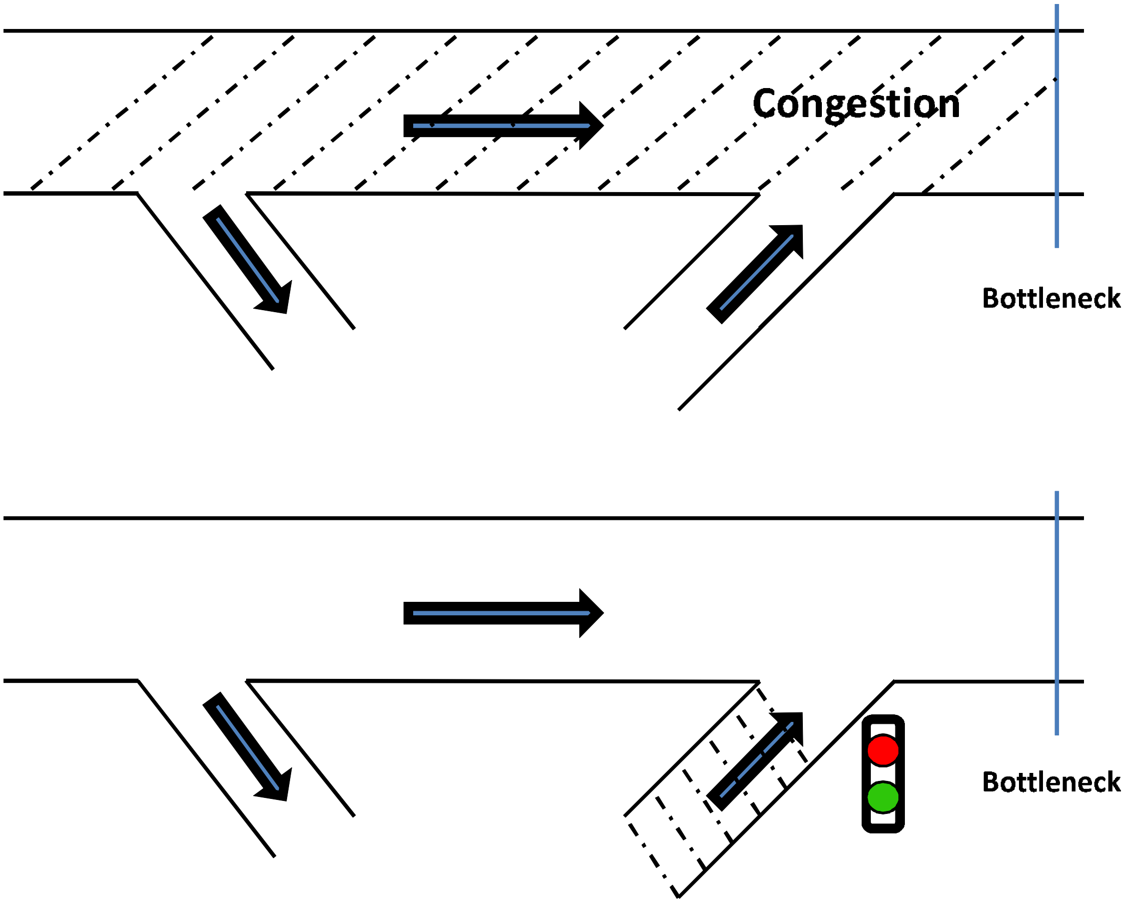

Figure 2.3: Effects of ramp metering in action.

Ramp metering can lead to earlier exit flows (thereby improving TTT) in the freeways through two main mechanisms (1) Avoiding off-ramp (exit) blockage (2) Preventing capacity drop. Figure 2.3 depicts an example of the first scenario, where there is a bottleneck just downstream of the on-ramp. To understand off-ramp blockage, we consider the scenario when ramp metering is not used (Figure 2.3 top), and demand exceeds the capacity at the bottleneck. The congestion propagating from this bottleneck leads to vehicle queues along the mainline that extend quickly past the off-ramps located upstream of the on-ramp. This delays the vehicles exiting through this off-ramp, even though they do not pass through the bottleneck region. Furthermore, these vehicles which cannot exit through the off-ramp also add to the queues and contribute to further congestion upstream. In comparison, when the ramp is metered, the upstream off-ramp blockage can either be delayed or avoided leading to increased earlier exit flows from the system. This leads to a decrease in the TTT of all users of the system, as shown in Eq. (2.8). Ramp metering can also help avoid capacity drop, thereby increasing the exit flows from the system. Figure 2.4 (Top) shows the application of ramp metering to avoid capacity drop. As described before, under congested conditions the queue discharge rate might be below capacity, while flows equal to the capacity can be sustained when the bottleneck section is in free-flow. Ramp metering, in this case, aims to maintain free-flow conditions at this section, leading to a higher throughput while also prevent exit blockage. In comparison, the ramp metering controller can maintain controlled congestion when only mechanism (1) is present, as long as the congestion tail does not reach the upstream off-ramp.

Ramp metering may also have an indirect effect on freeway congestion by inducing changes in routing choices or leading to a temporal shift in demands. Wardrop's first principle [71] states that users with multiple route choices choose the route that minimizes their travel time. This principle is also relevant in describing the temporal model of vehicular demands on the ramps. In order to accurately model these phenomenon we need demand data, segregated by intended destination, along with a description of preferred route choices for each users. Accurate data in this format is not usually available. Also, it is generally expected that ramp metering does not change the choices of users, when the metering leads to bounded queues and bounded increases in waiting delays. The users who are expected to be directly affected would be users who use the freeway for a short commute. In this dissertation, we will not model and take advantage of traffic diversion as a methodology to improve travel times. Thus we generally expect that the performance gains reported by using the model with the controllers may be a lower bound of the performance gains that can be expected in a field implementation.

Ramp metering is often used in conjunction with queue length controllers. The objective of these queue controllers is to prevent excessive queue buildups in the ramps. This is necessary for two reasons (1) Excessive queues might lead to traffic disruptions in the arterial from which the on-ramp originates (2) Large queues might lead to excessive penalization of users of the ramp. To elaborate consider the example from Figure 2.3. When metering is active in the on-ramp, queue starts building up on the on-ramp as the outflow is restricted by the metering algorithm. Limited storage spaces in the ramps mean that continued ramp metering may lead to queue spill-back onto the arterials. Additionally, the travel time of the users on the on-ramp is increased as compared to the case when ramp metering was inactive. This increases the total travel time for the users using the ramps. Queue controllers, which help limit the maximum queue lengths indirectly promote equity. Queue overrides and integral queue regulators are some of the strategies proposed to maintain queue lengths in on-ramps [66].

Over the years, many ramp metering algorithms have been developed and deployed. The simplest ones are fixed time of day controllers which specify a fixed rate at any particular time of the day. These metering rates are usually determined from historical data. Percent occupancy control, which is another widely deployed ramp metering strategy, uses occupancy thresholds to determine the metering rates. Alinea is a popular local traffic responsive ramp metering algorithm developed by Papageorgiou et al. [47]. The basic version is an integral controller, which regulates the density downstream of the ramp to be around the target density (which is usually chosen as the critical density). Compared to the percent occupancy scheme, which is a feedforward controller, Alinea is a feedback controller and its field implementations have yielded improved performance [48]. Various versions of ALINEA, including the upstream ALINEA (which uses density measurements upstream of the ramp) and FL-ALINEA (which uses flow measurements) have been developed [65]. Various co-ordinated ramp metering strategies have been presented in literature and deployed in the field. The most popular ones include Compass, Bottleneck algorithm, SWARM (Systemwide adaptive ramp metering), ZONE algorithm and METALINE, among others [76]. Heuristic Co-ordinated ramp metering (HERO), a coordinated ramp metering strategy [54], was recently deployed successfully in the Monash freeway in Australia.

Variable speed limits

Variable speed limits(VSL) is another popular control strategy for traffic regulation in freeways. Variable message signs display the current speed limits, often determined in response to the current road, traffic and weather conditions. In some installations, the posted speeds are advisory, while many require mandatory compliance with enforcement. In most of the installations, the main target objective is to ensure traffic safety, and the VSL's are designed to ensure speed reduction and homogenisation in locations with high traffic incidents [70].

There are very few studies documenting the direct effect of VSL on aggregate traffic flow characteristics. Various researchers have proposed different models for effect of VSL on the fundamental diagrams [25, 5, 4]. Figure 2.5 shows two of the popular models capturing the effect of VSL on fundamental diagrams. There is a general consensus that the reduction of speed limits decrease the total flow throughput at under-critical densities. Heygi et al. [25] suggest that the new critical density is the point of intersection of the new free-flow line with the original fundamental diagram, and the fundamental diagrams overlap beyond the new critical density (Figure 2.5, Left). This differs from the fundamental diagram models proposed by Carlson et al. [4], where different fundamental diagrams result as the speed limits change, and these fundamental diagram often intersect. In general, these models predict higher flows are over-critical densities when speed limits are changed. The models proposed by Carlson et al. can also model increases in capacities as speed limits are decreased. This model was developed in response to the empirical observations made

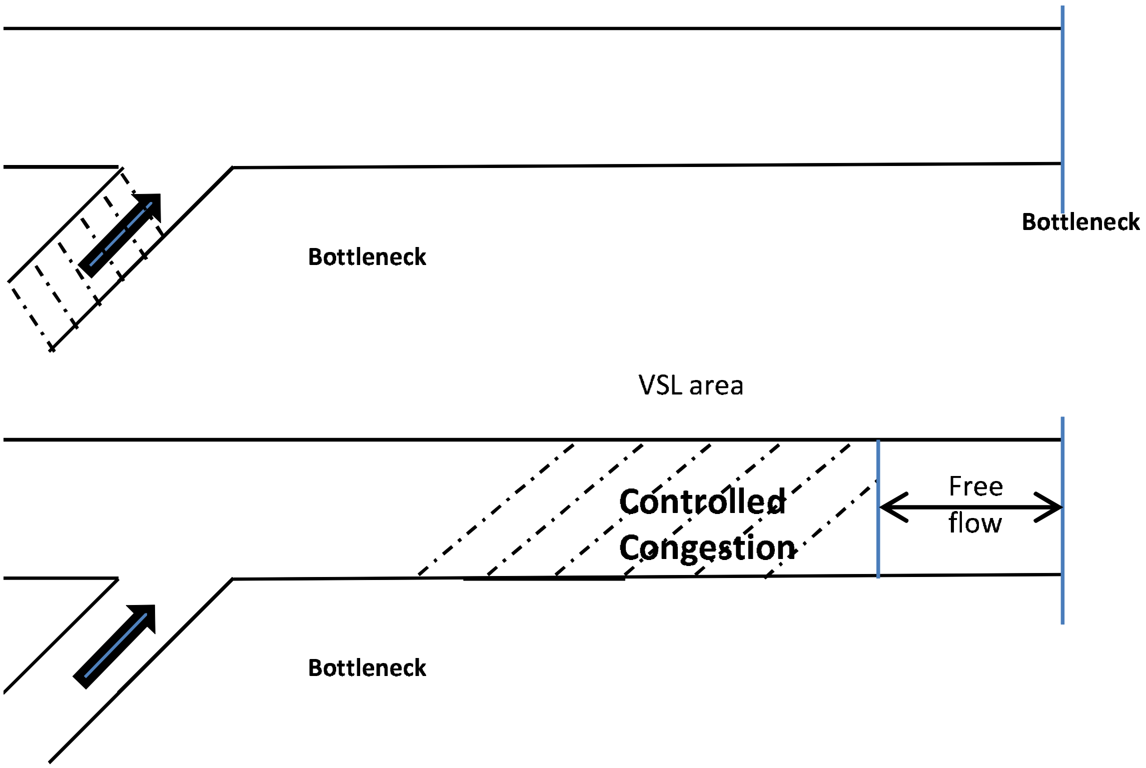

Figure 2.4: Avoiding capacity drop. Top : By ramp metering, Bottom : Using variable speed limits

by Papageorgiou et al. [49]. However, this study does not include a rigorous statistical analysis, presumably due to lack of large amounts of data, to summarily conclude the effects of VSL on the fundamental diagram. Moreover, the effect of VSL on overcritical densities, as modeled by Carlson et al. is debatable. For example, consider that the current state of the traffic flow corresponds to overcritical densities with no VSL in effect. The model suggests that decreasing the speed limit would lead to increase in throughput. In fact, this would be the case even when the speed limit is still above the current speed on the highway. On the other hand, the model proposed by Heygi et al., predict that the VSL is ineffective unless the speed limit imposed is lesser than the current speed on the highway.

Variable speed limits can be used as a mainline traffic control mechanism. Figure 2.4 (Bottom) shows the application of variable speed limits to create controlled congestion to prevent capacity drops downstream. Suppose the bottleneck section experiences a capacity drop once its density exceeds critical density. This capacity drop can be avoided if VSL is applied to the location upstream of the capacity drop section, such that the feeding flows to this section maintain its density below the critical density. VSL offers a couple of advantages over ramp metering for avoiding capacity drop. VSL can be applied to the area directly upstream of the bottleneck section, and there is usually a minimal delay for the effect of the control action. In contrast, if the first upstream on-ramp is located far away, there will be a large time delay when ramp metering based controllers

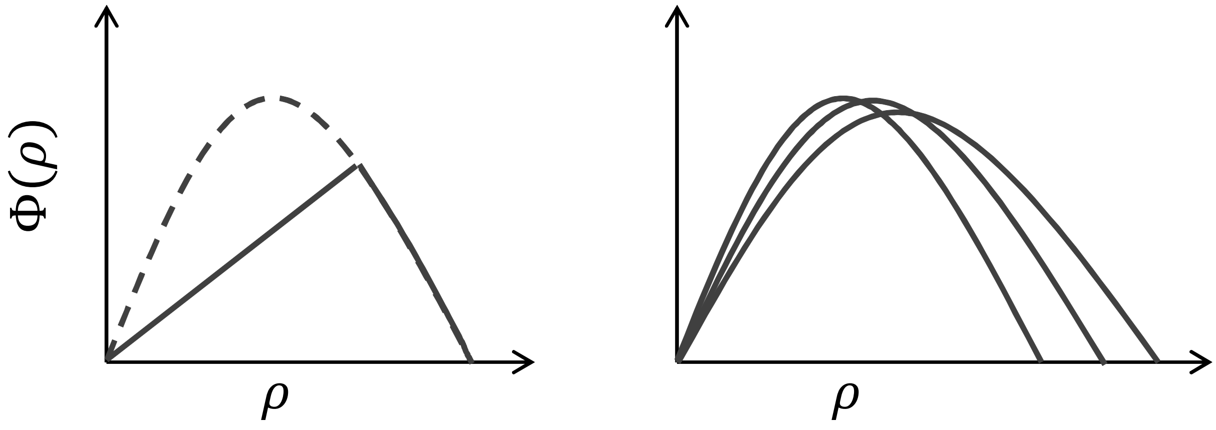

Figure 2.5: Effects of variable speed limits on fundamental diagrams. Left : Heygi et al. [25], Right : Carlson et al. [4]

are used. In the case that there is a off-ramp in between the on-ramp and the bottleneck location, the effectiveness of ramp metering may be limited, especially when the exact portion of vehicles leaving the off-ramp is not known. Finally, the presence of queue regulators and queue overrides, along with the limited storage capacity of on-ramps limit the effectiveness of ramp metering in dealing with the capacity drop. In contrast, the VSL controller directly uses the freeway mainline to store the additional vehicles present due to excessive demands.

Model based predictive controllers

Model based predictive controllers use predicted demands along with a model of the freeway network to specify ramp metering rates and/or variable speed limits for freeway traffic control. These strategies typically employ an optimal control/optimization framework to design strategies to minimize a chosen performance objective function. Eq. (2.7) shows some of the commonly used objective functions used in these formulations.

Wattleworth [72] was the first to use an optimization approach to specify ramp metering rates using a simple steady state model. Blinkin [2] and Papageorgiou [50] present some of the other early efforts in the development of model based optimal control strategies. While many of the early efforts used simple models in the formulations, with the advent of powerful numerical tools, various optimal control strategies based on more accurate macroscopic models have been investigated over the last couple of decades. For freeway networks, first order models (Cell Transmission model, CTM [11]) and second order models (METANET [52]) are commonly used to describe the traffic dynamics within these controllers.

The following approaches employ METANET as the underlying traffic model. Kostalis et al. [31], present an freeway control approach using on-ramp metering in conjunction with routing suggestions through variable messaging signs (VMS). Kostalis et al. have also developed an model based coordinated ramp metering strategy which is presented in [29]. Hegyi et al. [25], presented an optimal control strategy using the METANET model, employing ramp metering as well as variable speed limits. Some of the other recent efforts using METANET models can be found in [53, 5]. Second order models have an advantage over the first order models in incorporating the capacity drop. However, the optimization problems based on the second order models are non-linear, computationally intensive and the solutions obtained are usually only locally optimal. The former proves to be a drawback when the controller is embedded as a part of a model predictive framework, since this requires fast optimizations to be executed repeatedly[29].

While many optimal control efforts have focussed on using the CTM as the underlying model, two particular efforts are most relevant in terms of the computational efficiency of solutions proposed. Ziliaskopoulos[77] presents the problem of complete control of a road network based on the CTM model. The model assumes that all users are headed to a single destination, and the system operator can specify flow controllers as well as routes taken by various users. The flow controllers used in this formulation include ramp metering and variable speed limits. Ziliaskopoulos showed that under complete control, the optimal control problem can be solved using a linear program, even though the CTM model is inherently non-linear. Gomes and Horowitz [20] present an optimal coordinated ramp metering strategy based on the Asymmetric Cell Transmission Model (ACTM) [19, 20], which is a simplified model of the CTM developed to simulate traffic dynamics in freeways. In this effort, the authors demonstrate that under certain restrictions, the ramp metering problem can be solved using a relaxed linear program.

Chapter 3

Modeling and calibration of freeway networks

In this chapter, we review and discuss the application of the cell transmission model for simulation of traffic dynamics in freeways. We present in detail, how a freeway is represented in the macroscopic modeling framework, and discuss the calibration process used to create a simulation model of a chosen freeway stretch. The cell transmission model presented here will form the basis of different algorithms discussed in the rest of this dissertation. This chapter will also highlight the usage of the imputation algorithm, which is presented in detail in the next chapter. The process of model creation is work done along with other members of the TOPL group, notably Gunes Dervisoglu.

3.1 Link Node Cell transmission model for freeway traffic flow simulation

The cell transmission model [11] was developed as a versatile model to describe traffic dynamics in networks. In Section 2.2, we reviewed the basic cell transmission model which was used to describe the traffic dynamics in a stretch of roadway without any junctions. Daganzo presented an extension of the cell transmission model to general network topologies [13], including traffic merges and diverges. The Link-Node Cell transmission model, also referred as the LN-CTM, was presented in [32] as a modification of Daganzo's network model, and this will be used here. The LN-CTM is implemented in Aurora, a simulation platform used as a part of TOPL. Aurora uses the LN-CTM to simulate traffic flows in freeway networks as well as arterial streets.

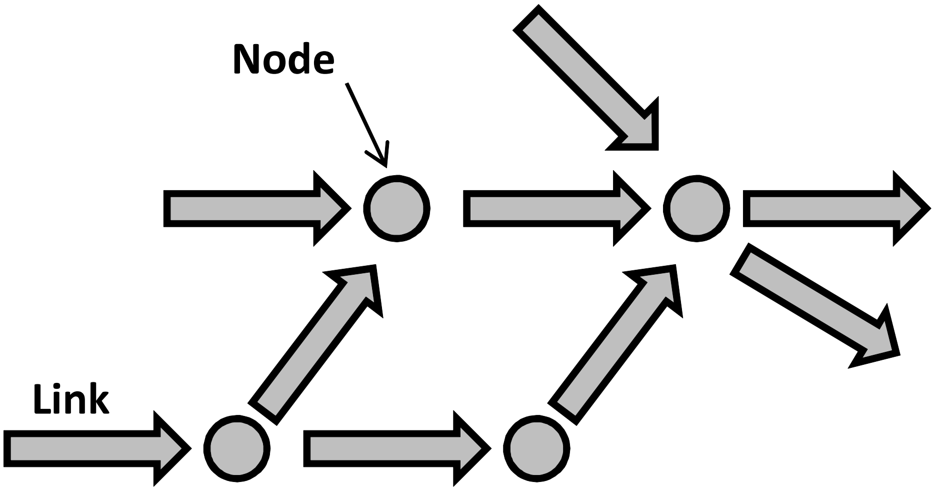

Figure 3.1: A directed graph representation of an arbitrary traffic network.

In LN-CTM, the traffic network that is to be modeled is represented as a directed graph of links, joined together at Nodes. Figure 3.1 shows an example of a network represented as a directed graph in the model. Each link represents a finite road segment, with uniform geometric properties. A network has normal links which represent road segments connecting an upstream node to a downstream node. Links that do not have an upstream node represent a source, and these links introduce vehicular demand into the network. Links without a downstream node serve as sinks, and they discharge the traffic out from the traffic network. For each link, traffic flow properties are defined through a fundamental diagram. Additionally, source links are also provided with input demand profiles, which specify the number of vehicles entering the link at any particular time. Sinks are sometimes specified with a flow capacity profile, which denotes a time varying restriction on the flow out of this link. Nodes are located at road junctions, each node transfers traffic from its input links to the output links. These nodes can be made to represent simple road junctions like the merge (with 2 input links and 1 output links) and the diverge (with 1 output link and 2 input links), which are the junctions modeled in Daganzo's network model. However, the node model in the LN-CTM is more general and it can be used to represent any general traffic junction with multiple inputs and outputs. Each node is specified with a time-varying routing matrix, known as the split ratio matrix. The split ratio matrix entries represent the portion of traffic moving from one particular input link to any given output link. Nodes cannot store any vehicles, and any flow that enters the node through an input link is completely transferred to the output links according to the given split ratios. Hence the sum of the split ratios for any particular input link is 1. In limited cases, some of the split ratios can be undefined and calculated in real-time in response to traffic conditions [33].

The LN-CTM algorithm is explained in [32]. The model update is executed in two steps: (a) Density updates and (b) Flow updates. Density updates employ a conservation equation involving the flows entering and exiting each link. These flows are calculated using the flow update equations. This density update can be simultaneously executed for each link in the network. Flow updates, executed in parallel at each node, involve a series of sub-steps. First, the demand and supply are calculated for each input/output link respectively, as explained in Section 2.2. Using the split ratio matrix at each node, the portion of this calculated demand destined to a particular output link is determined. The total demand for each output link can then be calculated as the sum of all demands destined to enter the output link. If the total demand for any output link exceeds the supply, then all input demands contributing to the traffic demand are reduced, in order to reduce the total outflow to equal the supply. For each input link, the demands are scaled corresponding to the supply of the most restrictive output link, to determine the realized flows. In case none of the output links have supply restrictions, the realized flows equal the demand. The flows out of the input links, along with the split ratios, provide the flows entering into each output link. This procedure is applicable for general networks and is explained in [32] (Pg. 128).

In the case of a freeway network, the update equations can be simplified. Figure 3.2 shows the actual geometry of a freeway. There are two types of junctions typically encountered in freeways - a merge (when an on-ramp joins the freeway) and a diverge (when an off-ramp breaks diverge from the freeway). In freeways in the U.S, off-ramp junctions are located upstream of on-ramps and there is only a short segment of freeway located in between. In cases when the segment is quite short, we can choose to disregard the corresponding link in the network model, and represent both junctions with the same node. In this case, we indicate that the split ratio for flows from the on-ramp onto the off-ramp is 0, since traffic entering the on-ramp cannot exit through the off-ramp.

On-ramps are modeled as sources which feed traffic into the network, and the off-ramps are modeled as sinks through which flows exit the network. The first freeway link is also a source, while the last freeway link is modeled as a sink. All the boundaries, including the off-ramps as well as the last freeway link are assumed to be congestion free. Off-ramp boundaries are generally observed to be congestion free. In many cases, the freeway downstream boundary can also be chosen to be in free-flow. This condition is not necessary for modeling the base scenario, since it is possible to introduce flow capacity restrictions on sinks. However, when the modeled freeway is used for freeway control simulation studies, the resulting simulations are usually inaccurate, since levels of congestion in boundaries change as traffic flow within the network changes. In this dissertation, we will assume that all the boundaries, including the off-ramps are in free-flow.

Table 3.1 defines different symbols and variables used in throughout this dissertation. The variables can be interpreted in conjunction to the freeway described in Figure 3.2. The freeway is assumed to have N links, which form the freeway mainline. A link indexed i, $i = 1 \cdots N-1$ connects upstream node $i-1$ to downstream node i. The first link (Link 0) and the last link (Link N) are a source and sink respectively. Similarly, each of the on-ramps and off-ramps are represented as a source and sink respectively. Ramps are indexed by the nodes to which they connect, as seen in Figure 3.2.

All the freeway links have a fundamental diagram associated with them. We adopt a triangular fundamental diagram as shown in Figure 3.3. The triangular fundamental diagram is characterized

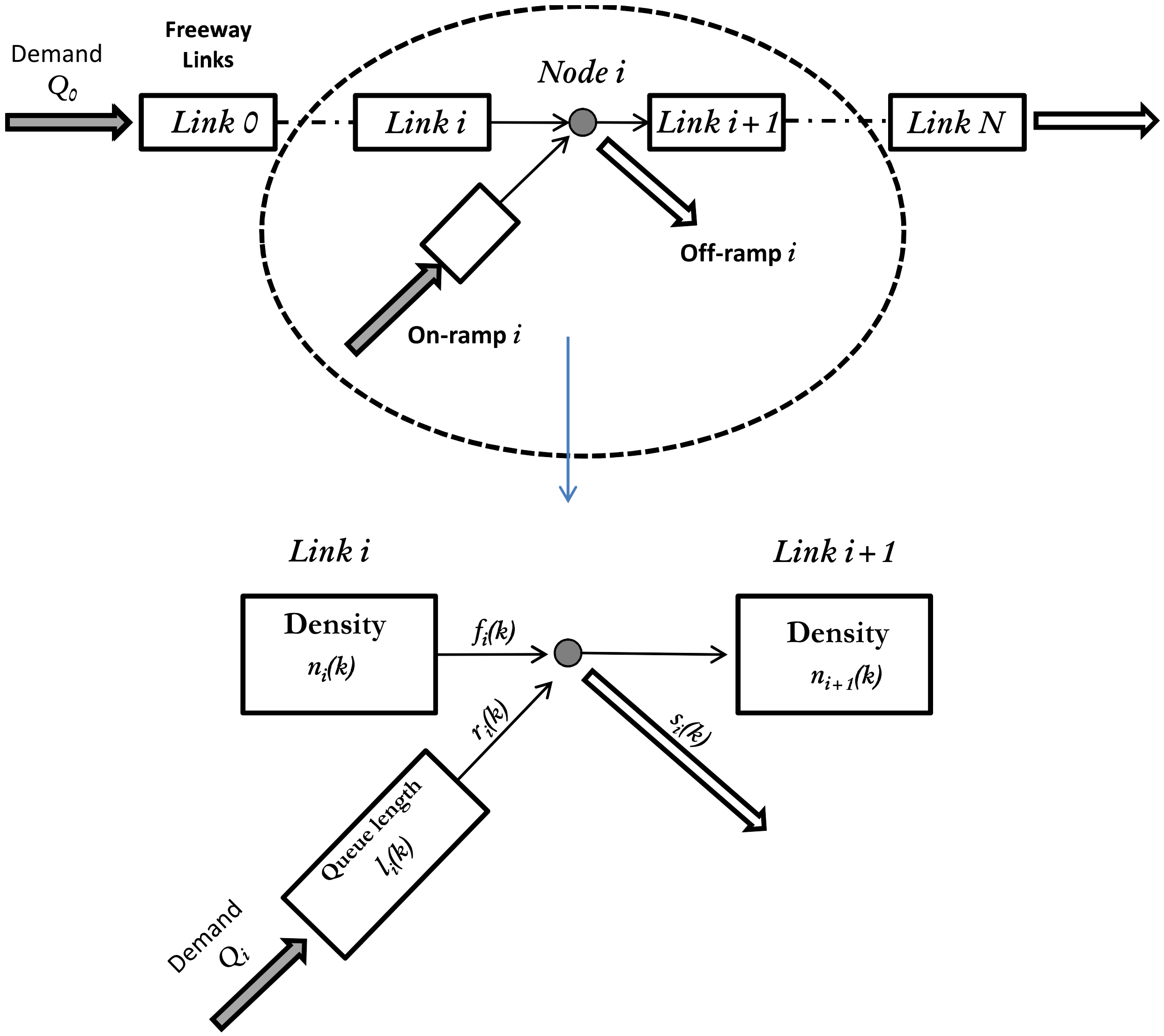

Figure 3.2: Freeway with N links. Each Node contains a maximum of one on- and one off-ramp. Note that Node *i* is upstream of Link *i*

by the free-flow speed (V_i), congestion wave speed (W_i), Capacity (F_i) and the jam density (n_i^J). With each freeway link, we associate two variables, its current density - $n_i(k)$, represented in units of number of vehicles per section and the flow exiting the link at the current time step $f_i(k)$, represented as number of vehicles per period. For each on-ramp, we keep track of the on-ramp queue ($l_i(k)$), on-ramp demands ($d_i(k)$) and the flow out of the on-ramp ($r_i(k)$). The split ratio $\beta_i(k)$ represents the portion of the demand from Link *i* that is intended to exit through the off-ramp *i*. Since the CTM model uses a First In First Out (FIFO) principle, $\beta_i(k)$ also represents the portion of the realized flow from Link *i* that exits through the off-ramp *i*. The units adopted in this

Symbol	Name	Unit
F_i	flow capacity of Link i	veh/period
V_i	free flow speed of Link i	section/period
W_i	congestion wave speed of Link i	section/period
n_i^J	jam density of Link i	veh/section
k	period number	dimensionless
$\beta_i(k)$	split ratio at node *i*	dimensionless
$f_i(k)$	flow out of Link *i*	veh/period
$n_i(k)$	number of vehicles (vehicle density) in Link *i*	veh/period
$s_i(k), r_i(k)$	off-ramp, on-ramp flow in node *i*	veh/period
$d_i(k)$	on-ramp *i* demand	veh/period
$l_i(k)$	queue length on on-ramp *i*	veh/period
$r_i^c(k)$	ramp metering rate for on-ramp *i*	veh/period
C_i	flow capacity for on-ramp *i*	veh/period
L_i	queue capacity for on-ramp *i*	veh/period
$Q_i(k)$	input flow for on-ramp *i*	veh/period

Table 3.1: Model variables and parameters.

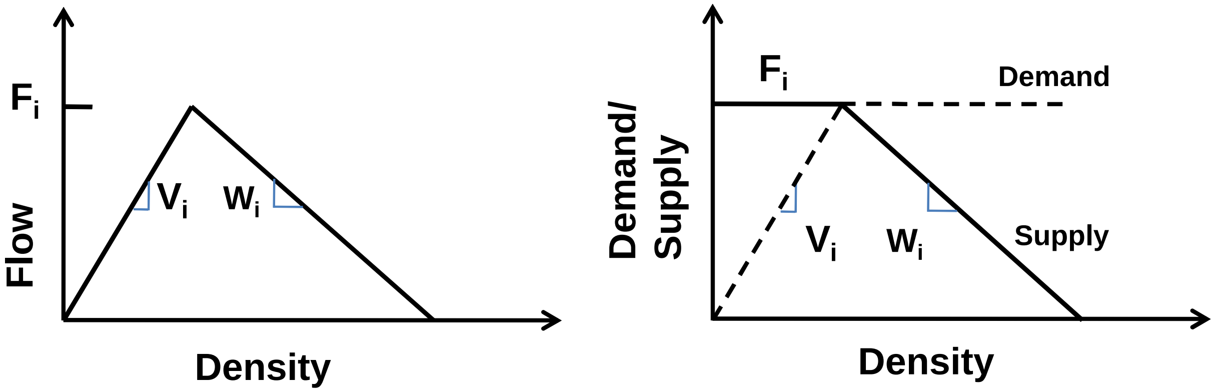

Figure 3.3: A triangular fundamental diagram, with the demand and supply functions

dissertation, along with the conversion factor from the commonly used units is given in Table 3.2. The simulation time step is chosen to ensure that $0 < W_i < V_i < 1$. If a chosen simulation time step does not satisfy this we can reduce the simulation time step until the condition is satisfied. For freeway networks with minimum link length of $1500 ft$ (excluding sources and sinks), a simulation time step of $T = 10s$ is appropriate under maximum free-flow speeds up to $90mph$.

Variable	**Commonly reported units**	**X Conversion factor**	**New units**
Flows	veh/hr	T	veh/period
Density	veh/mile	L_i	veh/section
Speeds	miles/hr	$\frac{T}{L_i}$	veh/period

Table 3.2: Conversion factors for units. Simulation time step T is given in units of hr, while L_i is given in miles.

The LN-CTM update equations for the freeway networks described above can be simplified as follows :

Density Update Equations : Mainline/Queue Conservation Equation

$$n_0(k+1) = n_0(k) + Q_0(k) - f_0(k)$$

$$n_i(k+1) = n_i(k) + f_{i-1}(k)(1-\beta_{i-1}(k)) + r_{i-1}(k) - f_i(k) \qquad i = 1,\cdots,N$$

$$l_i(k+1) = l_i(k) + Q_i(k) - r_i(k) \qquad i = 1,\cdots,N \qquad (3.1)$$

Flow Update Equations

$$f_N(k) = D_n(k)$$

$$f_i(k) = D_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \qquad i = 0,\cdots,N-1$$

$$r_i(k) = d_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \qquad i = 1,\cdots,N$$

$$s_i(k) = f_i(k)(1-\beta_i(k)) \qquad i = 1,\cdots,N$$

where

$$D_i(k) = \min(n_i(k)V_i, F_i),$$

$$R_i(k) = D_i(k)(1-\beta_i(k)) + d_i(k),$$

$$S_{i+1}(k) = \min(W_{i+1}(n^J_{i+1} - n_{i+1}(k)), F_{i+1})$$

$$d_i(k) = \min\left(r^c_i(k), l_i(k), C_i\right) \tag{3.2}$$

The density update equations include the conservation equations for the on-ramps and the mainline. The first link (Link 0) and the on-ramps are source links and they implement a simple queuing model. The closed form expressions for flows are obtained by comparing the total demand functions destined for each link i ($R_{i-1}(k)$) with its supply ($S_i(k)$). The total demand function for each link i ($R_{i-1}(k)$) is composed of two terms : **(a)** On-ramp demand ($d_i(k)$) **(b)** Demand from previous link, minus the portion that exits through the off-ramp, which equals $D_i(k)(1-\beta_i(k))$. In case this total demand is lesser than the supply for any link, the flows that are realized are equal to the demands, and the node corresponding to this traffic exchange is said to be in **free-flow**. Alternatively, the total demand can also be greater than the supply, and in this case the flow conditions in the corresponding node are **congested**. In congested conditions, the flows can be determined by scaling the demand function such that the total flow into link i equals the supply. There are many ways to scale the demands to meet the supply constraints, and this is done by assigning priorities to the input links of the node. The LN-CTM assigns priorities for each input link according to the total demand presented by each link. Thus the available supply is shared proportionally to the demands (i.e. $f_i(k)/r_i(k) = D_i(k)/d_i(k)$).

In the equations above, we do not include separate conservation equations for the off-ramps, detailing their dynamics. Off-ramps are assumed to be (sinks) without any capacity restrictions due to congestion in their boundary. As a result, they do not exhibit any influence on the discharge flow out of the network. This is consistent with the flow conditions encountered in many of the off-ramps in a majority of freeways. However, one situation where this assumption can be violated is in large freeway to freeway interconnects, as congestion can sometimes spill back into the modeled freeway due to flow restrictions at the exit of the interconnect. For modeling purposes, it is possible to include a time-varying capacity for the off-ramp ($\bar{s_i}(k)$), to capture the flow restriction out of the off-ramp. In this case, the equations for flow out of link i is given by

$$f_i(k) = \min\left(D_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)}, \frac{\bar{s_i}(k)}{\beta_i(k)}\right) \tag{3.3}$$

The first and the second terms corresponds to the scaling of demands to comply with the supply restrictions in link $i+1$ and the off-ramp i respectively. It can be seen that off-ramp boundary flow restrictions indirectly introduce a time-varying capacity for flows out of a link i, which is also dictated by the off-ramp split ratios. We again caution that when the model is used for simulation under modified conditions (for example, in the presence of a controller), these boundary conditions can change as the realized flows within the network impact congested boundaries. In this case, it is better to incorporate a complete model of the interacting network in order to get consistent results. This is beyond the scope of this dissertation, and we will assume that flow conditions in all boundaries are not congested for the reminder of this thesis.

The model presented here can be used to simulate traffic dynamics in a freeway when ramp metering is active. Ramp metering rates can be specified as a time varying profile through the variable $r_i^c(k)$. Ramp metering operation can be interpreted as a restriction to the flow capacity of an on-ramp. Another emerging form of traffic control is variable speed limits. The original version of the LN-CTM does not model the traffic dynamics under the effect of variable speed limits. We will address the problem of modeling variable speed limits when we introduce the optimal ramp metering and variable speed limit based congestion controller.

3.2 Building freeway models

In the previous section, we described the difference equations that can be used to model the traffic dynamics for a freeway network. In this section, we will discuss the process of creating a model of a given stretch of highway from observed data. Along the process, we will use the example of a stretch of I-80E freeway in the Bay area, CA.

Freeway Representation

The first part of the process is to represent the given freeway in the directed graph framework, based on the geometrical characteristics of the site. The freeway is divided into successive links, and nodes are created in the junction of on-ramps or off-ramps and at location of lane changes (lane drops or lane increases), along the freeway. As described in the previous section, the node corresponding to an off-ramp junction and the next on-ramp junction can be merged if there is only a short segment of freeway located in between. Some links created with this process can be quite long, and they can are further divided into smaller links. This helps to ensure that good quality simulations can be obtained as the congestion can be accurately modeled. The length of the freeway links dictate the sampling time used in simulation. When the minimum link size is around $1500ft$, a sampling time of $10s$ is usually sufficient.

Figure 3.4 shows an example of the directed graph representation of a short stretch of the I-80E. TOPL network editor [68], a **Google Maps** based tool developed by other members of the TOPL group, was used to generate this network. In this figure, for illustration, nodes are represented by round markers. Normal links can be seen to connect two round markers and they represent the freeway mainline. Sinks are highlighted in red, and their destination is represented by a square marker. Source links start at a square marker and end at a normal node. Freeway geometry can also be specified manually by listing the links and nodes, though this may be cumbersome.

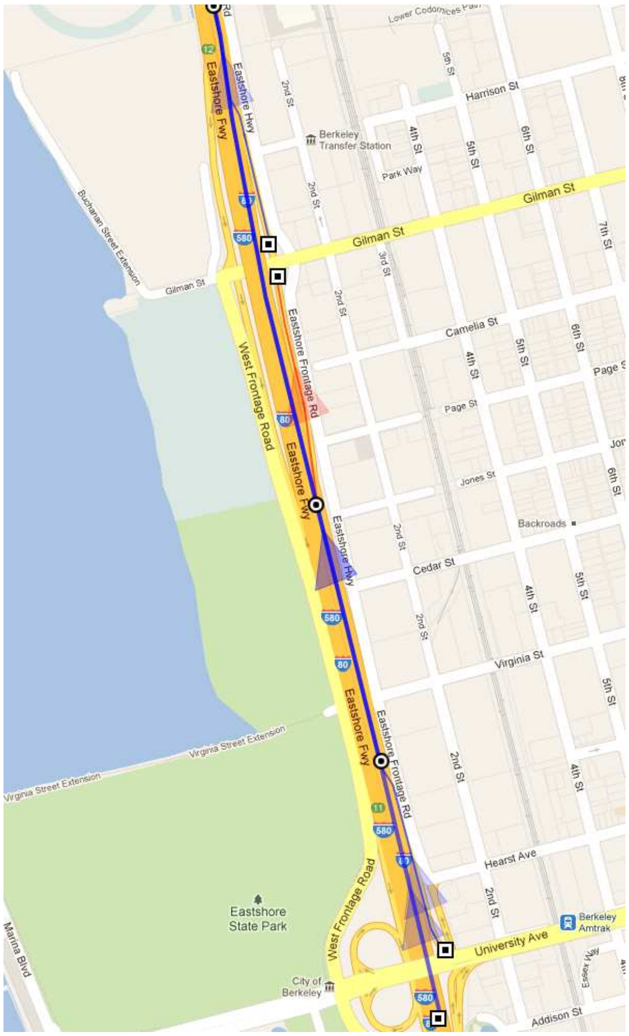

Figure 3.4: A short stretch of the I-80E freeway. The directed graph representation is overlaid.

Data acquisition and selection

Vehicle detector stations (VDS) containing loop detectors are located along the freeway to provide flow and occupancy data. PeMS processes and archives these data in form of time series over different days of operation. This archived data can be obtained from their website [57]. The data are aggregated over an interval of 5 minutes such that each day contains 288 data points for each reported quantity - density, flow and speed. However, detector and data health is a major concern. PeMS also reports detector performance for each day of operation. For the purpose of model calibration, we choose days in which PeMS reported over 75% functionality for all detectors in the freeway stretch, to ensure that the models generated are reliable. Figure 3.5 shows the detector health report, as well as the number of samples reported on different weekdays, over 3 months from July 2008 to September 2008. We typically choose multiple days of data with good detector health. Multiple days of detector data will be useful for fundamental diagram calibration, and also the specification of multiple sets of on-ramp demands and off-ramp split ratios to represent traffic dynamics of different representative days. We highlight that PeMS imputes missing mainline data using data from adjacent detectors, and the final data obtained does not contain any gaps. Days in which traffic congestion patterns were not representative of nominal freeway operations are generally discarded. This includes days with major traffic incidents, weekends, holidays and days with special events. It is also advisable to disregard days with adverse weather conditions, unless the intent is to specifically capture its effect on the realized traffic dynamics.

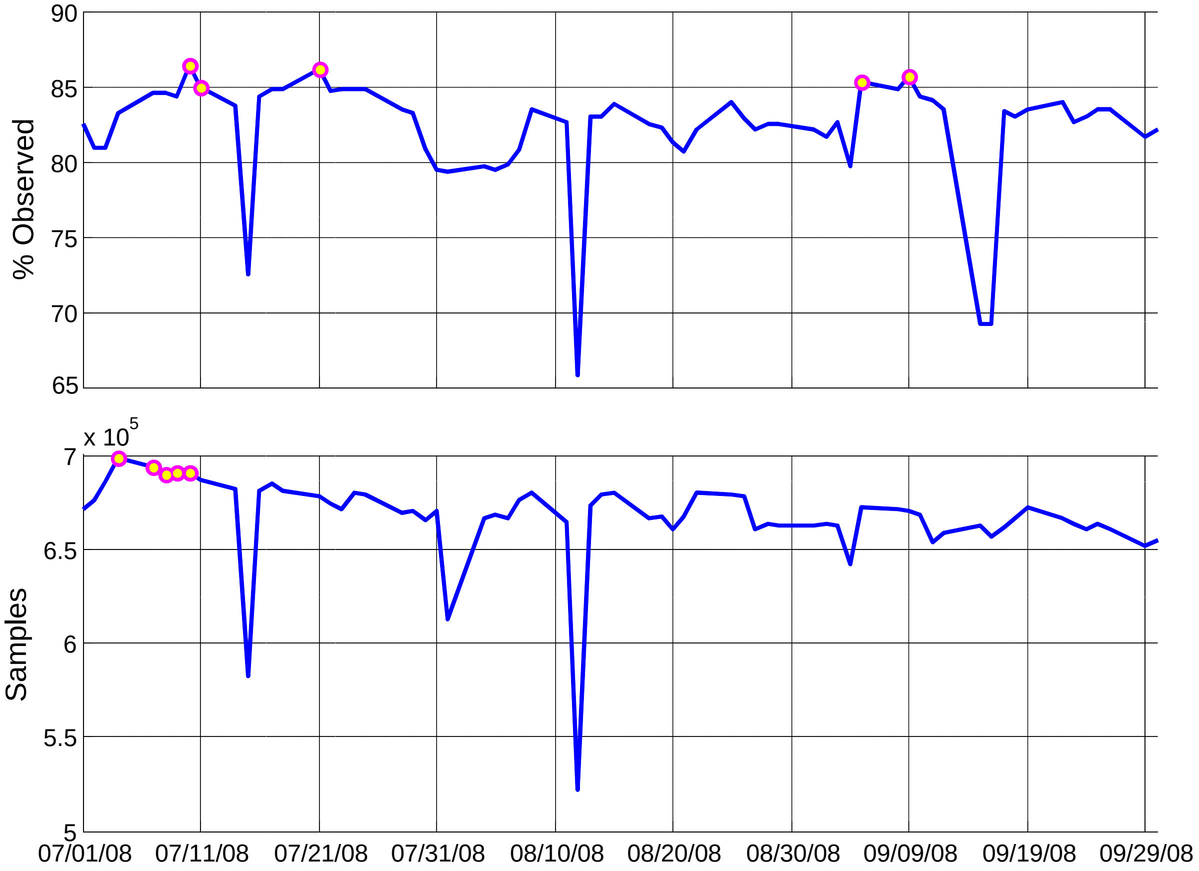

Figure 3.5: Detector health and number of samples reported during weekdays.

Throughout the model creation process, we choose a 5 min granularity of data so that the chosen profiles are sufficiently smooth. In general, observed traffic data is non-smooth, as they represent observation of discrete vehicles, whereas, the CTM models the vehicular traffic using a continuous fluid approximation. Also, the goal of the modeling process is to reproduce macroscopic features of traffic, and for this purpose 5 min averaged profiles are usually suitable.

Fundamental Diagram Calibration

Fundamental diagram calibration is the process of estimating the parameters of the fundamental diagrams using empirical data. For the triangular fundamental diagram, we need to estimate four parameters - (1) freeflow speed (V) (2) congestion wave speed (W), (3) flow capacity (F) and (4) jam density (n^J) (out of which three are independent). We calibrate a fundamental diagram for data obtained from each detector. For this purpose, we use multiple days of data obtained from the data selection process described before. We briefly describe the process of calibration below. For a more thorough reference, the reader is referred to [17, 39].

For the purpose of calibration, we use flow-density scatter plots. The free-flow speed, V, is estimated by performing a least-squares fit on the flow-density data at the time instants where the speed was reported to be above 55 mph (this threshold is valid for locations with speed limits around 65mph). This portion of data is assumed to correspond to free-flow conditions. The capacity estimate for model calibration is chosen deterministically to be the highest observed flow throughout all investigated days. This maximum value of flow across the section is then projected horizontally to the free-flow line, to establish the tip of the triangular fundamental diagram (Figure 3.6). The intersection point (apex point) is the critical density for the section, above which the flow is congested. The last parameter to be calibrated is the congestion speed parameter, W, which also defines the jam density for the section. A constrained quantile regression of data points whose speeds are lesser than 55mph is used to obtain the congestion wave parameter (W). The point where the regression line crosses zero flow is the jam density of the section. In certain detectors, congestion (i.e data points with speed $<$ 55mph) is not observed due to the nature of prevalent demands. In this case, the capacity of the section, congestion wave speed and the jam density cannot be estimated. In this case, nominal values are usually used, as long as they are consistent with the data observed (particularly the flow capacity).

A different fundamental diagram can be calibrated for each detector located along the freeway. The fundamental diagram of a link is usually obtained from the parameters calibrated using the detector located in the link. In many cases, links might not have functioning detectors associated with them, and detectors from nearby links, which have similar geometric characteristics are used.

On-ramp input demands and off-ramp split ratio specification

The final step in the model creation process is to specify the on-ramp input demands and the off-ramp split ratio profiles. It is necessary that the ramp demands and split ratios specified for the model are derived from a consistent set of on-ramp flows and off-ramps flows, which require that

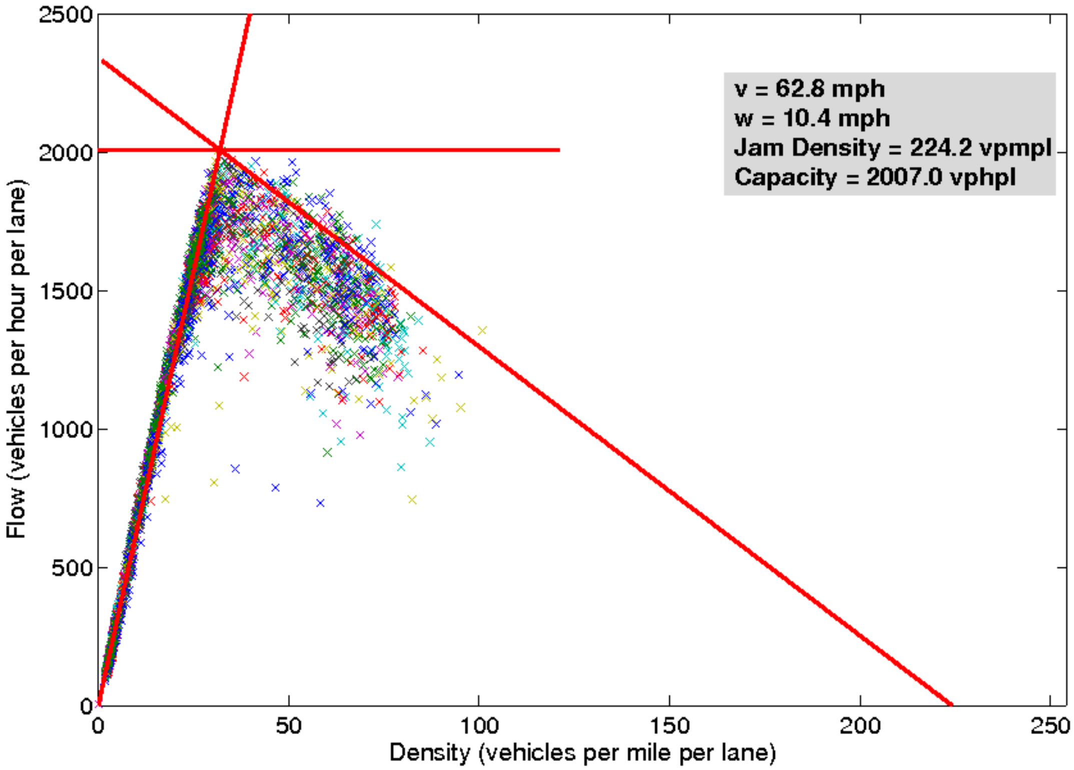

Figure 3.6: The end result of the fundamental diagram calibration

the flows are recorded during the same time period on the same day. In the case all of the data are available, on-ramp flows (measured at the start of the on-ramp) can be used as the on-ramp demand into the freeway. Split ratios are obtained by dividing the off-ramp flows by the flows measured at the previous mainline section. Both the on-ramp demand and off-ramp split ratios are provided as time varying profiles with a time step of 5 mins.

In freeways in California, mainline detector data is usually archived and available, while on-ramp and off-ramp flow data is missing. This is sometimes due to lack of detectors on ramps (particularly off-ramps) or lack of data feeds from the ramp detectors into the PeMS archival center. From the point of view of performance monitoring, mainline data is usually sufficient to obtain measures for characterizing freeway operations, and therefore, there is a general lack of investment (or interest) in installing detectors on ramps and setting up data feeds for the existing detectors. However, for the purposes of freeway model creation, ramp flow data is a critical component. In the next two chapters we discuss imputation algorithms which can be used to estimate the ramp flows using detector measurements from the freeway mainline. The imputation algorithm provides the missing ramp flows and split ratios, and is therefore an essential component of the model creation process in the freeways.

Base case simulation and model validation

The final model consists of the following components : (a) A directed graph representing the freeway geometry (b) fundamental diagram parameters for each link (c) demand profiles for on-ramps (d) split ratio profiles for nodes with off-ramp diverges (e) Ramp metering rates for active controllers. Once these details are provided, the LN-CTM model can be used to simulate the traffic dynamics. It produces density, flow and vehicle speed profiles for each link.

A base case simulation refers to a simulation which replicates the conditions observed on a particular day. In our case, this corresponds to the simulation with no additional active controllers. Many of the freeways we have modeled in California do not contain an active ramp meter. The purpose of the base case simulation is to validate the model performance and compare it against the detector data measured along the freeway. This simulation is expected to reproduce the flow, density and speed profiles observed along the freeway mainline reasonably.

There are various metrics that can be used to validate the model. We compare the model by calculating the mean absolute density/flow error, evaluated as a percentage of the mean observed density/flow. This is referred to as the density/flow error in this dissertation. We also calculate hourly VMT, VHT and VCD for the freeway and compare it against the measured values to obtain the VMT/VHT/VCD errors. The formulaes for the error calculations are listed below.

$$
\begin{aligned}
\text{Density error} &= \frac{\sum_i \sum_k |n_i(k) - n_i^{meas}(k)|}{\sum_i \sum_k |n_i^{meas}(k)|} \\
\text{Flow error} &= \frac{\sum_i \sum_k |f_i(k) - f_i^{meas}(k)|}{\sum_i \sum_k |f_i^{meas}(k)|} \\
\text{VMT error} &= \frac{\sum_\kappa |VMT(\kappa) - VMT^{meas}(\kappa)|}{\sum_\kappa VMT(\kappa)} \\
\text{VHT error} &= \frac{\sum_\kappa |VHT(\kappa) - VHT^{meas}(\kappa)|}{\sum_\kappa VHT(\kappa)} \\
\text{VCD error} &= \frac{\sum_\kappa |VCD(\kappa) - VCD^{meas}(\kappa)|}{\sum_\kappa VCD(\kappa)} \qquad (3.4)
\end{aligned}
$$

where VMT/VHT/VCD are calculated hourly from the density and flow data using the following equations.

$$
\begin{aligned}
VHT(\kappa) &= \sum_{i} \sum_{k=\kappa*3600/T_s}^{(\kappa+1)*3600/T_s} n_i(k) \\
VMT(\kappa) &= \sum_{i} \sum_{k=\kappa*3600/T_s}^{(\kappa+1)*3600/T_s} f_i(k) \\
VCD(\kappa) &= \sum_{i} \sum_{k=\kappa*3600/T_s}^{(\kappa+1)*3600/T_s} (n_i(k) - f_i(k)/V_i)\,\mathbb{I}(V_s(k) < 55mph)
\end{aligned}
\tag{3.5}
$$

Apart from the metrics listed above, one of the major source of validation is the visual inspection of contour plots of flow, density and speeds. For these plots, x-axis represents consecutive links along the freeway and the y-axis represents the time during the day. For the contour plots used in this dissertation, direction of traffic flows from left to right along the x axis. A visual inspection is used to confirm that the system bottlenecks, regions of congestion and the extent of congestion are replicated as close as possible in the simulation.

Simulation studies

The main purpose of creating simulation models is to use them to evaluate various operational management strategies and to assess their benefits before they can be deployed in the field. The simulation model created above can be used to simulate the effect of various control strategies include ramp metering and variable speed limits. For example, any of the control strategies explained in Chapter 2 can be simulated using the simulation model. In this dissertation, we will use these calibrated models in Chapter 6 to demonstrate the performance of our optimal controllers.

Example

We present the base case scenario of an calibrated model of the I-80E freeway in California. The fundamental diagram parameters were estimated from over 45 days of data obtained over the 3 month period shown in Figure 3.5. We chose Aug 21st, 2008 to obtain on-ramp demands and off-ramp split ratios to create the model. For the I80 freeway in the bay area, no ramps reported data, and thus we used the imputation procedure described in Chapter 5 to obtain the on-ramp demands and off-ramp split ratios.

Figure 3.8 shows the density and flow contour plots obtained from the simulations and compares it to the observed measurements. Figure 3.7 shows the velocity contours obtained using simulated and measured data. In all the contour plots, x-axis denotes the distance along the freeway, denoted by Post Miles, and the y-axis represents the time of day. The final density and flow errors in this simulation were 3.1 % and 6.8 %. In this case, we can see that the simulations replicate the observed freeway dynamics with good accuracy, except for a section between PostMile 10-15. This section corresponded to the Berkeley Highway Lab facility, and the detector data was verified to be faulty/unreliable. We can see that the simulation is able to capture the range and temporal extent of the congestion in other locations. Additionally, we are also able to match the location of the bottlenecks in the simulations. Figure 3.9 compares the simulated and measured performance measures. We see that VMT and VHT agree very closely with each other, while the total delay error is 10%. In general, we observe larger errors in delays as compared to other performance measures (we have observed delay errors of 25-30 % in some cases). Based on our conversations with other researchers/engineers who model freeway networks, delay errors of less than 40% indicate a well calibrated model, as long as the delay profiles are visually similar. One reason can be attributed to the errors in flow measurements that are usually seen in mainline detectors.

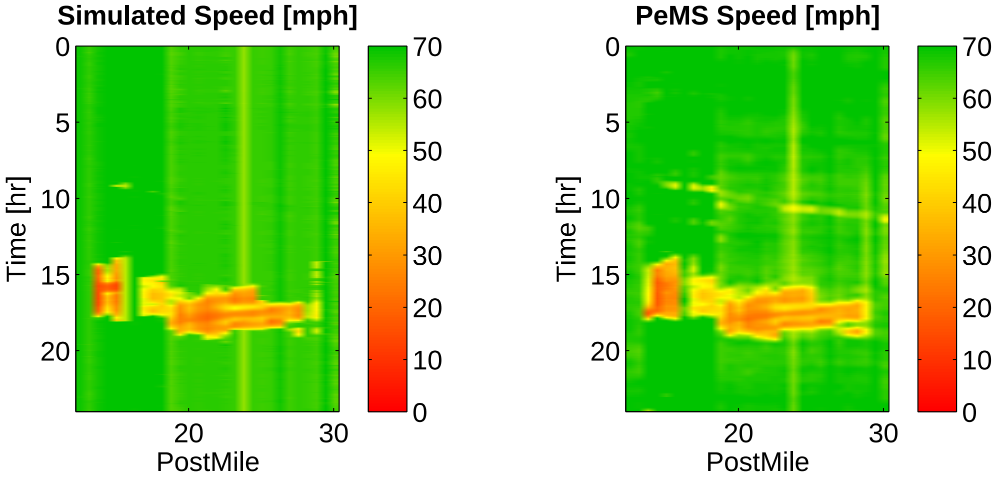

Figure 3.7: Contour plots of simulated and measured velocity contours

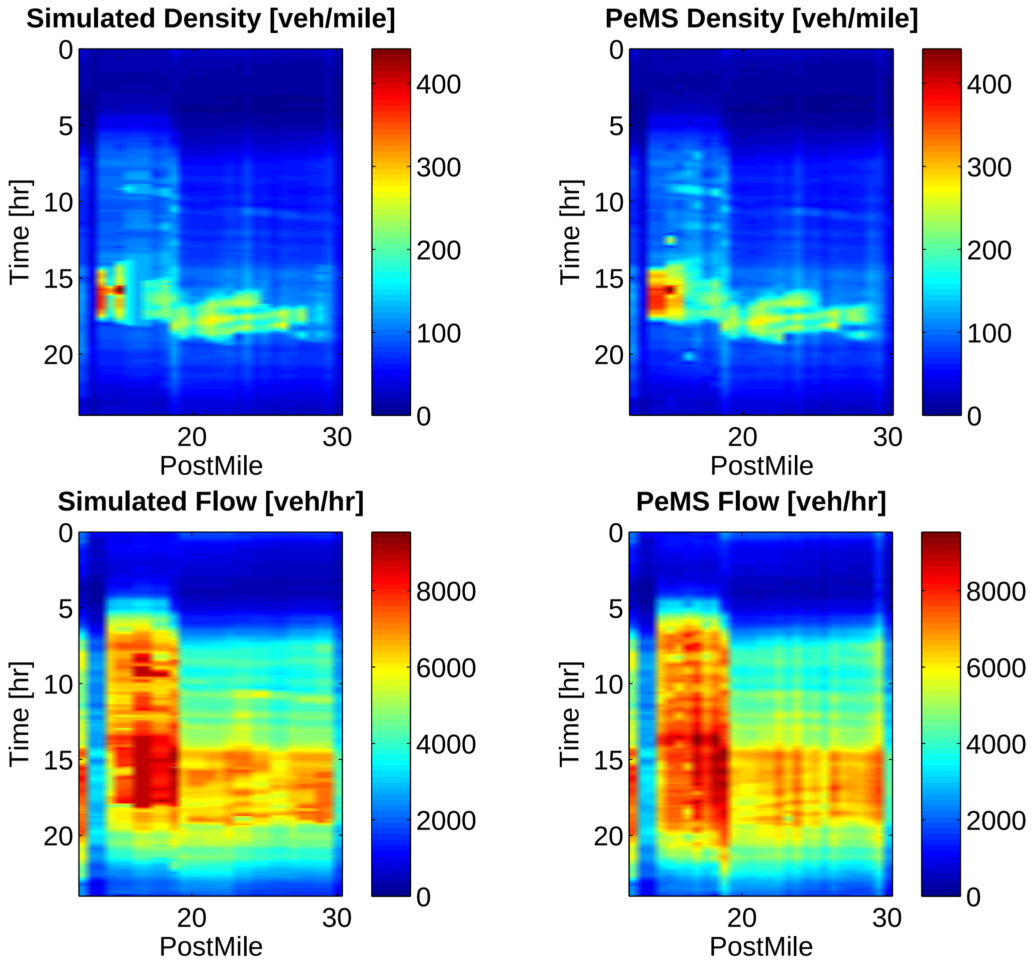

Figure 3.8: Simulated and measured contour plots. Top : Simulated, Measured densities, Bottom : Simulated, Measured flows

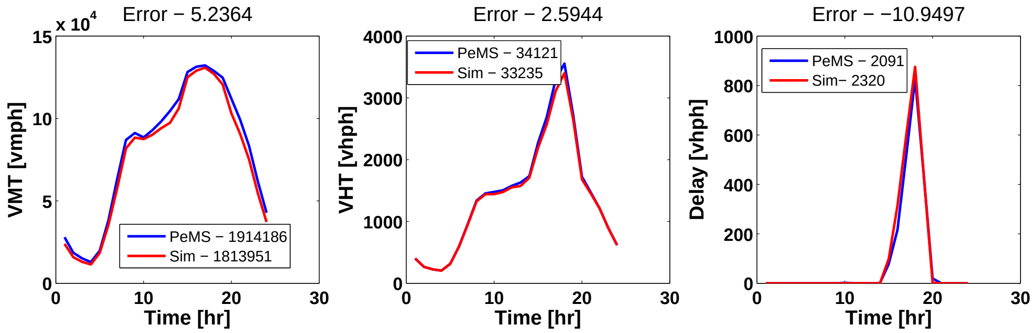

Figure 3.9: Comparison of performance measures

Chapter 4

Imputation of ramp flow data using the asymmetric cell transmission model

On-ramp demands and off-ramp split ratios are critical inputs for simulation of freeway traffic dynamics. On-ramp demands are obtained from detectors located at the on-ramp entrance while off-ramp splits can be estimated by using measured flows along the mainline and the off-ramps. For freeways in California, mainline detector data is usually available, while ramp flow data is missing - either due to lack of detectors or lack of reliable data feeds into the archival system. Traditional imputation algorithms [14, 9] based on statistical models, have been successfully applied for imputing missing data for detectors along the freeway. They exploit statistical dependence in traffic measurements between detectors from adjacent stations or the detectors in the nearby lanes. However, these algorithms are not suitable for ramp flow imputation, as good quality historical data is usually not available. Also the measurements from adjacent ramps usually exhibit weak statistical dependence between each other. In some cases, long stretches of freeways can completely lack any ramp detection data.

In this chapter, we present an imputation algorithm for on-ramp/off-ramp flows based on the Asymmetric Cell Transmission Model (ACTM). This algorithm can be described as a “model-based” imputation method, as it estimates on-ramp flows and off-ramp flows that match the available measurements in the freeway when fed into a simulation model describing the traffic dynamics. We use the ACTM as the underlying model for the imputation algorithm in this chapter. ACTM is a simplified version of the CTM for freeway traffic simulations. The model dynamics can be represented by a set of piecewise affine differential/difference equations as compared to the LN-CTM model. The simplified model lends to the development of the first provably convergent algorithm for ramp flow imputation in freeways. In the following sections, we first describe the ACTM model and the ramp flow imputation algorithms, and then present the stability and convergence results for this algorithm. Finally we present some examples of its application in a practical scenario.

4.1 Asymmetric cell transmission model

The ACTM [19, 20] was developed as a simplification to Daganzo's CTM [12] for simulating freeway dynamics. The main difference is the treatment of merges of on-ramps into the freeway. The CTM model treats the merge to be symmetric, such that switching the order of consideration of merging flows will not result in different flow realizations. In comparison, the ACTM introduces an asymmetry in the treatment of joining flows, and makes a distinction in the consideration of flows from on-ramps. As we will see below (comparing to the model presented in the previous chapter) this simplifies the model equations. The original motivation for the development of the ACTM was to use it as an approximation of the freeway dynamics in controller analysis and optimal controller synthesis [19]. CTMSIM, a matlab based simulation tool implements the ACTM [32].

We present a short summary of the ACTM (The reader can refer to [19, 20] for a detailed presentation). The freeway is specified as a sequence of segments, each with (at most) an on-ramp near the beginning of the section and an off-ramp near the end of the section. This is slightly different from the way the freeway geometry is represented in the LN-CTM model, which uses nodes to represent flow exchanges. Figure 4.1 shows the freeway divided into *N* sections or cells, where vehicles move from left to right. Boundary conditions can be specified in different ways in the ACTM. Vehicles can be fed into the freeway through a queue, while the downstream is in free-flow (BC-1). Alternatively, density of the cells upstream of the first section and downstream of the last section can also be specified as the boundary conditions for simulation (BC-2). As we had noted before, BC-2 is appropriate to simulate the base scenario, but BC-1 is preferred for use in simulation model. This is because under different operational strategies like ramp metering, the control strategy usually modifies the densities at the boundaries. It must be noted that BC-1 places restrictions on the freeway sections chosen for simulation, since the beginning and end of the freeway section simulated should always be in free-flow. However, for our imputation algorithm, any one of the stated boundary conditions can be used, depending on the availability of detector data.

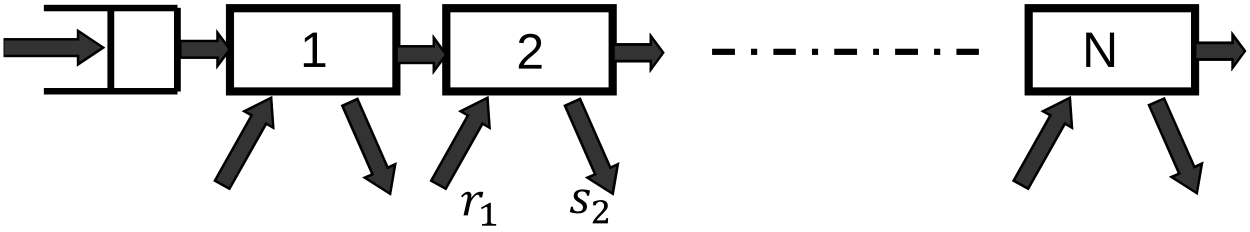

Figure 4.1: Freeway with *N* sections.

Table 4.1 lists the model variables and parameters. Many of the variable definitions are similar to the definitions presented before, except $f_i(t)$, which is defined as the flow entering into Link $i+1$ from Link i according to the ACTM. These changes will be adopted only for the algorithms and proofs presented in this chapter. We associate each freeway link with a triangular fundamental diagram, similar to the ones used in the LN-CTM (Figure 3.3). The section lengths are absorbed in the fundamental diagram parameters for convenience.

Symbol	Name	Units
F_i	maximum flow (capacity) of section i	veh/s
V_i	free flow speed of section i	section/s
W_i	congestion wave speed of section i	section/s
n_i^c	critical density of section i	veh/section
n_i^J	jam density of section i	veh/section
$f_i(t)$	flow from section i to $i+1$ at time t	veh/s
$r_i(t)$	on-ramp i flows at time t	veh/s
$s_i(t)$	off-ramp i flows at time t	veh/s
$n_i(t)$	number of vehicles in section i at time t	veh/section
$n_0(t)$	number of vehicles in the input queue to section 1 at time t	veh
$Q_0(t)$	input flow at upstream queue at time t	veh/s

Table 4.1: Model variables and parameters.

The Cell Transmission Models are a time and space discretization of the Lighthill-Whitham-Richards (LWR) equation. Thus, the ACTM can also be represented as a continuous time spatially discretized model, as presented here. This continuous time model is more amenable for development of a provably convergent imputation algorithm. As a result, the units for various variables listed in Table 4.1 are slightly different from the ones used in the other chapters. Also, the general model can be specified with off-ramp flows or off-ramp split ratios. We will consider the version with off-ramp flows, as these flows can also be converted easily to split ratios. We will estimate the off-ramp flows in our imputation algorithm. When BC-1 is used, the following equations describe the model.

$$\begin{aligned}
\dot{n}_i(t) &= f_{i-1}(t) - f_i(t) + r_{i-1}(t) - s_i(t), && 1 \le i \le N \\
f_i(t) &= \min(V_i n_i(t) - s_i(t), W_{i+1}[n_{i+1}^J - n_{i+1}(t)], F_i) && 1 \le i < N \\
f_N(t) &= \min(V_N n_N(t) - s_N(t), F_N) \\
f_0(t) &= \min(V_0 n_0(t), W_1[n_1^J - n_1(t)], F_0) \\
\dot{n}_0(t) &= Q_0(t) - f_0(t)
\end{aligned} \tag{4.1}$$

When density boundary conditions (n_0 and n_{N+1}) are specified, the model is specified as

$$\begin{aligned}
\dot{n}_i(t) &= f_{i-1}(t) - f_i(t) + r_{i-1}(t) - s_i(t) && 1 \le i \le N \\
f_i(t) &= \min(V_i n_i(t) - s_i(t), W_{i+1}[n_{i+1}^J - n_{i+1}(t)], F_i) && 0 \le i \le N
\end{aligned} \tag{4.2}$$

where w_{N+1} and n^J_{N+1} are the congestion wave speed and jam density of the cell directly following the boundary. The flow, denoted by $f_i(t)$ corresponds to free-flow when

$$V_i n_i(t) - s_i(t) < \min[W_{i+1}[n^J_{i+1} - n_{i+1}(t)], F_i] \tag{4.3}$$

Otherwise, the resulting flow corresponds to congested conditions. With respect to each section, the inflow (from upstream/previous link) can be either in free-flow or in congestion and the outflow (to downstream link) can also be either in congestion/freeflow. In each of the four cases, the density and the flow equations can be combined to a single update equation. Thus the model can also be represented using a four mode model. Finally, the off-ramp splits can be represented as $\beta_i(t) = \frac{s_i(t)}{f_i(t)+s_i(t)}$

There are some differences in the model presented here, as compared to the general model. In the original ACTM, there is a blending coefficient ($\gamma_i \in [0,1]$) associated with each ramp, signifying the location of the ramp. In our freeway geometry, the on-ramps are located at the beginning of each link, and in this case the blending coefficient equals zero, and the corresponding terms are not included in the model equations represented above. Another approximation is that ramp flows are directly allowed to merge into the freeway, which is usually the case when the ramp flows are not very large. However, there are some instances when this model will be inaccurate, such as in freeways that have large on-ramps due to freeway to freeway connectors. The original ACTM model includes additional parameters which provide a slightly better approximation for these situations, but the ACTM still lacks the model accuracy of the LN-CTM to represent freeway dynamics when large on-ramps are modeled.

The expression for the freeway dynamics represented above use flows exiting out of the on-ramps and flows entering into the off-ramps. Our imputation model will be designed to estimate these flows. The flows entering the off-ramp, along with the mainline flows, provide a direct estimate of the off-ramp split ratio profile. We assume that the flows exiting the on-ramps are a good estimate of on-ramp demands (i.e the flows into the on-ramps). This approximation is valid when ramps are in free-flow and they are not metered. Most of the ramps in California are usually not metered, and rarely exhibit queues. In the case when ramps are metered, this approximation is less accurate. This approximation is least accurate when ramps are metered, and mainline is congested, which leads to queues on the ramps. In all these cases, the errors decrease when aggregates over large time intervals are considered.

4.2 Imputation algorithm

The imputation algorithm presented in this section uses the ACTM model described above. We estimate on-ramp flows and off-ramp flows, which replicate the observed mainline measurements when used with the ACTM model. Formally, the ramp flow imputation problem can be stated as:

Problem. *Estimate on-ramp and off-ramp flows $\hat{r}_i(t), \hat{s}_i(t),\ t \in [0,T]$, such that the model evolution, described using Eq.* (4.1) *with the ramp flow estimates generate flow/density profiles ($\hat{f}_i(t), \hat{n}_i(t),\ t \in [0,T]$) that replicate the corresponding measurements ($f_i(t), n_i(t),\ t \in [0,T]$) obtained from detectors along the freeway.*

The imputation algorithm presented in this section is based on an adaptive repetitive learning technique described in [38, 26]. This is a control technique used to identify periodic input profiles for a dynamic system so that it can track a given periodic output profile. For our imputation algorithm, the input profiles are the on-ramp and the off-ramp flows, and the target output profiles are the measured densities and the flow profiles. In the learning algorithm, we assume initial estimates for the input profiles, and adaptively improve these estimates as the process is executed repeatedly. The final estimates learnt from this process are expected to track the actual output profiles, in case that a solution exists.

The adaptive repetitive learning algorithm requires the density and ramp flow profiles to be periodic. For example, the density profiles are expected to satisfy $n(0) = n(T)$, where T is the period. Under this condition, the algorithm can execute multiple runs using the observed profiles corresponding to a single day (even though the actual measured densities/flows can vary day to day, particularly during morning/evening commute times), thereby simulating a repetitive process. In our case, we assume that the profiles are measured starting from 12:00am and ending at 11:59pm. The measurements have a typical sample time of 5 mins and we use linear interpolations to specify the complete profiles which satisfy the continuity conditions. Traffic conditions are very light (as seen by the low density/flow values) in the early morning. Low density/flow values in the night ensure that we obtain sufficiently smooth profiles.

In our imputation algorithm, the on-ramp and off-ramp flows are represented as a convolution of a kernel on a constant periodic ramp parameter (influence) vector.

$$r(t) = \int_0^T K_r(\tau,t)c_r(\tau)d\tau, \quad s(t) = \int_0^T K_s(\tau,t)c_s(\tau)d\tau \tag{4.4}$$

where $K_r(\tau,t)$ and $K_s(\tau,t)$ represent periodic, time dependent kernel functions with period T, which is also the period of the process considered. Some typical kernel functions include an impulse or a gaussian window centered at time k. Kernel function width is chosen with respect to the degree of smoothness expected from the imputed profile. A short kernel window (eg. an impulse function) will lead to noisy estimations as compared to a kernel with a large window.

The structure of the ACTM allows us to decouple the estimation of ramp flows. The imputation is carried out section by section sequentially, starting from the most upstream section 1. For estimation of section i ramp flows, we consider the immediate upstream section $i-1$ and the immediate downstream section $i+1$. For convenience, the upstream (downstream) section for a section i is specified with the subscript i,up (i,dn). Figure 4.2 shows the parameters and measurement data used for imputation of ramp flows in section i. The upstream boundary conditions

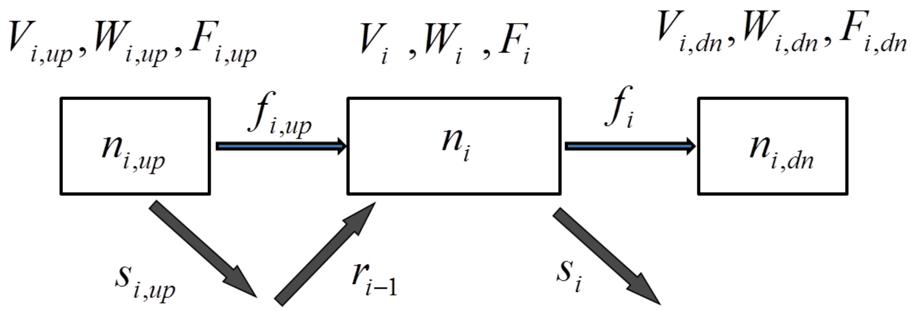

Figure 4.2: Imputation parameters and cell definitions

includes the upstream density, fundamental diagram parameters as well as the off-ramp flow $s_{i,up}$. The imputation proceeds sequentially from upstream to downstream and either $s_{i,up}$ or its estimate is available. Since all the parameters and variables carry the subscript i, for clarity, we drop it in the following equations.

In the imputation algorithm, unknown on-ramp and/or off-ramp flows are estimated indirectly by estimating their respective influence coefficients $\hat{c_r}(\tau,t)$ and $\hat{c_s}(\tau,t)$ using a repetitive adaptive learning algorithm, which runs continuously cycling through the 24 hour traffic data. For each section, the imputation procedure assumes initial estimates for the ramp parameter functions $\hat{c_r}(\tau,t)$ and $\hat{c_s}(\tau,t)$. These estimates are then adapted so that the model calculated densities match the density profile recorded in the vehicle detector station. Let P represent Plant (i.e. the actual system described using the ACTM) while M represents the Model, calculated using the estimates. The model variables and the estimates are represented with a 'hat' ($\hat{n}(t), \hat{r}(t)$ etc.) and the errors with a 'tilde' (e.g. $\tilde{n}(t) = n(t) - \hat{n}(t)$, where $n(t)$ is the section's measured number of vehicles and $\hat{n}(t)$ is the number of vehicles generated by the model at instant k). The actual variables (ie. the measured variables) are represented without any accent. Table 4.2 presents the various modes considered in the imputation. In this table, and throughout this section, we will introduce a new variable $\bar{w_{dn}}(t)$ to simplify the expressions.

$$\bar{w_{dn}}(t) = min\left(\frac{F}{n^J_{dn} - n_{dn}(t)}, W_{dn}\right) \tag{4.5}$$

The modes considered here only refer to the flow conditions downstream (i.e out of the section considered) and $P - F$ and $P - C$ correspond to free-flow and congested flow conditions downstream respectively.

For each section, we assume that we have the following measurements: upstream link density $n_{up}(t)$, upstream off-ramp flows $s_{up}(t)$ or its estimate $\hat{s}_{up}(t)$, current link density $n(t)$, link outflow $f(t)$ and downstream link density $n_{dn}(t)$. In addition, the fundamental diagram parameters are also available for all three links. The mode dependent adaptation laws for the parameters at each step

Symbol	Condition
$P-F$	$f(t) < \bar{w}_{dn}(t)[n^J_{dn} - n_{dn}(t)]$
$P-C$	$f(t) = \bar{w}_{dn}(t)[n^J_{dn} - n_{dn}(t)]$
$M-F$	$v\hat{n}(t) - \hat{s}(t) < \bar{w}_{dn}(t)[n^J_{dn} - n_{dn}(t)]$
$M-C$	$v\hat{n}(t) - \hat{s}(t) > \bar{w}_{dn}(t)[n^J_{dn} - n_{dn}(t)]$

Table 4.2: Plant and model modes.

are given by

(a) *P*-F , *M*-F (plant and model are both in free-flow downstream)

$$\begin{aligned}
\dot{\hat{c}}_r(\tau,t) &= G_1 K_r(\tau,t)\tilde{n}(t), && \forall\tau \in [0,T] \\
\dot{\hat{c}}_s(\tau,t) &= -G_2 K_s(\tau,t)\tilde{f}_d(t), && \forall\tau \in [0,T]
\end{aligned} \tag{4.6}$$

(b) *P*-C , *M*-C (plant and model are both in congestion downstream)

$$\begin{aligned}
\dot{\hat{c}}_r(\tau,t) &= G_1 K_r(\tau,t)\tilde{n}(t), && \forall\tau \in [0,T] \\
\dot{\hat{c}}_s(\tau,t) &= -G_1 K_s(\tau,t)\tilde{n}(t), && \forall\tau \in [0,T]
\end{aligned} \tag{4.7}$$

(c) *P*-C , *M*-F (plant is in congestion and model is in free flow downstream)

$$\begin{aligned}
&Case(i) \qquad \tilde{n}(t) > 0 \\
&\quad \dot{\hat{c}}_r(\tau,t) = G_1 K_r(\tau,t)\tilde{n}(t), && \forall\tau \in [0,T] \\
&\quad \dot{\hat{c}}_s(\tau,t) = -G_1 K_s(\tau,t)\tilde{n}(t) - G_2 K_s(\tau,t)\frac{\tilde{f}_d(t) + |\tilde{f}_d(t)|}{2}, && \forall\tau \in [0,T] \\
&Case(ii) \qquad \tilde{n}(t) \leq 0 \\
&\quad \dot{\hat{c}}_r(\tau,t) = G_1 K_r(\tau,t)\tilde{n}(t), && \forall\tau \in [0,T] \\
&\quad \dot{\hat{c}}_s(\tau,t) = -G_2 K_s(\tau,t)\tilde{f}_d(t), && \forall\tau \in [0,T]
\end{aligned} \tag{4.8}$$

(d) *P*-F , *M*-C (plant is in free flow and model is in congestion downstream)

$$\begin{aligned}
&Case(i) \qquad \tilde{n}(t) < 0 \\
&\quad \dot{\hat{c}}_r(\tau,t) = G_1 K_r(\tau,t)\tilde{n}(t), && \forall\tau \in [0,T] \\
&\quad \dot{\hat{c}}_s(\tau,t) = -G_1 K_s(\tau,t)\tilde{n}(t) - G_2 K_s(\tau,t)\tilde{f}_d(t), && \forall\tau \in [0,T] \\
&Case(ii) \qquad \tilde{n}(t) \geq 0 \\
&\quad \dot{\hat{c}}_r(\tau,t) = G_1 K_r(\tau,t)\tilde{n}(t), && \forall\tau \in [0,T] \\
&\quad \dot{\hat{c}}_s(\tau,t) = -G_2 K_s(\tau,t)\tilde{f}_d(t), && \forall\tau \in [0,T]
\end{aligned} \tag{4.9}$$

where G_1, G_2 are user defined positive gains.The model update equations at each step are given by

$$\begin{align}
\tilde{n}(t) &= n(t) - \hat{n}(t) \notag \\
\dot{\hat{n}}(t) &= \hat{f}_{up}(t) - \hat{f}(t) + \hat{r}(t) - \hat{s}(t) + a\tilde{n}(t) \tag{4.10} \\
\hat{f}_{up}(t) &= min(n_{up}(t)V_{up} - s_{up}(t), F_{up}, W(n^J - \hat{n}(t))) \notag \\
\hat{f}(t) &= min(\hat{n}(t)V - \hat{s}(t), \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t))) \notag \\
\hat{r}(t) &= \int_0^T K_r(\tau,t)\hat{c}_r(\tau,t)d\tau \notag \\
\hat{s}(t) &= \int_0^T K_s(\tau,t)\hat{c}_s(\tau,t)d\tau \notag \\
\tilde{f}_d(t) &= f(t) - (n(t)V - \hat{s}(t)) \tag{4.11}
\end{align}$$

The error term $\tilde{f}_d(t)$ is designed to capture the errors in off-ramp flows during free-flow. The parameter $a > 0$ in (4.10) is chosen so as to make the error equations asymptotically stable. In the update equations, on-ramp flows are always updated to decrease the density error, and hence the updates are proportional to the current density error. The off-ramp flows are adapted using the density error (terms with gain G_1) and/or flow error $\tilde{f}_d(t)$ (terms with gain G_2), depending on the mode. This allows the downstream flows to converge to the measured values, as we will see in the next section.

The imputation algorithm is initiated from the most upstream section. After convergence, the off-ramp flow estimates from the current section is used as boundary data for imputing the ramp flows in the next section. The imputation algorithm then is employed section-wise to the most downstream section.

While the parameter and model density update equations are given in continuous time, this procedure is implemented in discrete time with a small time step and small gains, so that the imputation procedure as well as the model are stable. Typically the time step Δt is chosen such that $V_{max}\Delta t < 1$, where $V_{max} = \max_i V_i$ and V_i is the free flow speed at section i. The adaptation is carried out for the entire density profile multiple times, so as to reduce the 24-hour 'errors' $\sum_k |\tilde{n}(k)|$ and $\sum_k |\tilde{f}(k)|$. This procedure is repeated until both the errors become insignificant, i.e.

$$\sum_k |\tilde{n}(k)| < 0.005 \times \sum_k n(k), \quad \text{and} \quad \sum_k |\tilde{f}(k)| < 0.005 \times \sum_k f(k)$$

or stop decreasing

$$\Delta\left(\sum_k |\tilde{n}(k)|\right) < 0.005 \times \sum_k n(k), \quad \text{and} \quad \Delta\left(\sum_k |\tilde{f}(k)|\right) < 0.005 \times \sum_k f(k)$$

In the expression above, $\Delta(.)$ is the change in errors across iterations.

The final ramp parameter functions $\hat{c}_r$ and $\hat{c}_s$ give us the imputed estimates of on-ramp and off-ramp flow profiles $(\hat{r}_i(t), \hat{s}_i(t))$. The off-ramp split ratios are computed from the off-ramp split profiles and the model calculated flows, as $\hat{\beta}_i(t) = \frac{\hat{s}_i(t)}{\hat{f}_i(t)+\hat{s}_i(t)}$.

4.3 Analysis of the algorithm

In this section, we will study the stability and convergence of the density errors under the adaptation laws given in Section 4.2. The error equations are given by

$$\dot{\tilde{n}}(t) = \tilde{f}_{up}(t) - \tilde{f}(t) + \tilde{r}(t) - \tilde{s}(t) - a\tilde{n}(t) \tag{4.12}$$

$$\tilde{r}(t) = \int_0^T K_r(\tau,t)\tilde{c}_r(\tau,t)d\tau$$

$$\tilde{s}(t) = \int_0^T K_s(\tau,t)\tilde{c}_s(\tau,t)d\tau \tag{4.13}$$

We will also show that the downstream flow converge with zero errors in all the modes. The condition stated below will be used in the following lemma and theorems.

Condition 4.3.1. *For the system described in Figure 4.2, the following conditions apply:*
(1) $s_{up}(t) = \hat{s}_{up}(t)$ when the plant upstream is in free-flow.
(2) $W(n^J - n(t)) < \min(F_{up}, n_{up}(t)V_{up} - \hat{s}_{up}(t)))$ when the plant upstream is in congestion.

Condition 4.3.1 guarantees that the upstream off ramp estimation error $\tilde{s}_{up}(t) = s_{up}(t) - \hat{s}_{up}(t)$ is either zero or it does not affect the upstream (input) flows in the current section. For the freeway described in Figure 4.1, this condition is easily achieved for the first cell (i=1) and as it will later be shown by induction in Theorem 4.3.2, it will apply to all cells.

Lemma 4.3.1. *For the system described by Figure 4.2 executing the imputation algorithm defined by Eq.* (4.6) - (4.11)*, given $n_{up}(t)$, $\hat{s}_{up}(t)$, $n_{dn(t)}$, $f(t)$, $n(t)$ and the fundamental diagram parameters for all the cells, under Condition 4.3.1, $\tilde{f}_{up}(t) = f_{up}(t) - \hat{f}_{up}(t)$ is given by $\tilde{f}_{up}(t) = -\zeta(t)W\tilde{n}(t)$ where $0 \leq \zeta(t) \leq 1$ at any time t.*

The flows entering the section can either be in free-flow or in congestion, and they could possibly be different for the Plant and the Model. This lemma provides a compact representation for the $\tilde{f}_{up}(t)$ for all the cases. In fact this term will be a stabilizing term in the proofs that follow. The proof of this lemma is presented in Section 4.5.

Theorem 4.3.1. *For the system described in Figure 4.2, given $n_{up}(t)$, $\hat{s}_{up}(t)$, $n_{dn}(t)$, $f(t)$, $n(t)$, and the fundamental diagram parameters, the parameter update laws in Section 4.2 stabilize the error equations* (4.12),(4.13) *when Condition 4.3.1 applies. The density errors ($\tilde{n}(t)$) and the flow errors ($\tilde{f}(t), \tilde{f}_{up}(t)$) converge to 0. Moreover, $s(t) = \hat{s}(t)$ when the plant downstream is in free-flow and $\bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) \leq \min(F, \hat{n}(t)V - \hat{s}(t)))$ otherwise.*

Proof. Consider the Lyapunov functional $V(t)$ (and its time derivative) given by

$$\begin{aligned}\mathbb{V}(t) &= \frac{1}{2}\tilde{n}(t)^2 + \frac{1}{2G_1}\int_0^T \tilde{c}_r(\tau,t)^2 d\tau + \frac{1}{2G_1}\int_0^T \tilde{c}_s(\tau,t)^2 d\tau \\ \dot{\mathbb{V}}(t) &= \tilde{n}(t)\dot{\tilde{n}}(t) + \int_0^T \tilde{c}_r(\tau,t)G_1^{-1}\dot{\tilde{c}}_r(\tau,t)d\tau + \int_0^T \tilde{c}_s(\tau,t)G_1^{-1}\dot{\tilde{c}}_s(\tau,t)d\tau \qquad (4.14)\end{aligned}$$

In deriving $\dot{\mathbb{V}}(t)$, we use the fact that ramp parameter functions $c_r(\tau), c_s(\tau)$ are not a function of time. We need to show that $\dot{\mathbb{V}}(t)$ is negative semi-definite, for the error equations to be stable.

From lemma 4.3.1, we see that $\tilde{f}_{up}(t) = -\zeta(t)W\tilde{n}(t)$ for some $0 \le \zeta(t) \le 1$, irrespective of the mode of the plant/model with respect to upstream flow. Hence, the error equations can be simplified into four cases corresponding to the downstream flow. The following equations show that $\dot{V}(t) \le 0$ in all the four cases.

(i) P-F, M-F
In this case, we have

$$\begin{aligned}\dot{\tilde{n}}(t) &= \dot{n}(t) - \dot{\hat{n}}(t) \\ &= (f_{up}(t) - (n(t)V - s(t)) + r(t) - s(t)) \\ &\qquad - \big(a\tilde{n}(t) + \hat{f}_{up}(t) - (\hat{n}(t)V - \hat{s}(t)) + \hat{r}(t) - \hat{s}(t)\big) \\ &= -a\tilde{n}(t) - \zeta(t)W\tilde{n}(t) - V\tilde{n}(t) + \tilde{r}(t) \\ \tilde{f}_d(t) &= (n(t)V - s(t)) - (n(t)V - \hat{s}(t)) = -\tilde{s}(t) \\ \dot{\tilde{c}}_r(\tau,t) &= -G_1K_r(\tau,t)\tilde{n}(t) \\ \dot{\tilde{c}}_s(\tau,t) &= G_2K_s(\tau,t)\tilde{f}_d(t) = -G_2K_s(\tau,t)\tilde{s}(t)\end{aligned}$$

Substituting these terms in Eq. (4.14), we get

$$\begin{aligned}\dot{V}(t) &= \tilde{n}(t)\left(-a\tilde{n}(t) - \zeta(t)W\tilde{n}(t) - V\tilde{n}(t) + \tilde{r}(t)\right) \\ &\qquad + \int_0^T \tilde{c}_r(\tau,t)G_1^{-1}(-G_1K_r(\tau,t)\tilde{n}(t))d\tau + \int_0^T \tilde{c}_s(\tau,t)G_1^{-1}(-G_2K_s(\tau,t)\tilde{s}(t))d\tau \\ &= \big(-(a+V+\zeta(t)W)\tilde{n}(t)^2 + \tilde{n}(t)\tilde{r}(t)\big) - \tilde{n}(t)\tilde{r}(t) - \frac{G_2}{G_1}\tilde{s}(t)^2 \\ &\le -a\tilde{n}(t)^2 - \frac{G_2}{G_1}\tilde{s}(t)^2 \le 0 \qquad (4.15)\end{aligned}$$

(ii) P-F, M-C

For this case,

$$\begin{align*}
\dot{\tilde{n}}(t) &= \dot{n}(t) - \dot{\hat{n}}(t) \\
&= (f_{up}(t) - (n(t)V - s(t)) + r(t) - s(t)) \\
&\qquad - \left(a\tilde{n}(t) + \hat{f}_{up}(t) - (\bar{w}_{dn}(t)(n_{dn}^J - n_{dn}(t))) + \hat{r}(t) - \hat{s}(t)\right) \\
&= -(\zeta(t)W + a)\tilde{n}(t) - n(t)V + s(t) + \bar{w}_{dn}(t)(n_{dn}^J - n_{dn}(t)) + \tilde{r}(t) - \tilde{s}(t) \tag{4.16} \\
&= -(\zeta(t)W + a + V)\tilde{n}(t) - \hat{n}(t)V + \hat{s}(t) + \bar{w}_{dn}(t)(n_{dn}^J - n_{dn}(t)) + \tilde{r}(t) \tag{4.17} \\
\tilde{f}_d(t) &= (n(t)V - s(t)) - (n(t)V - \hat{s}(t)) = -\tilde{s}(t)
\end{align*}$$

When plant is in free-flow and model is in congestion, we have

$$n(t)V - s(t) \leq \bar{w}_{dn}(t)(n_{dn}^J - n_{dn}(t)) \leq \hat{n}(t)V - \hat{s}(t) \tag{4.18}$$

We substitute for the individual terms in Eq. (4.14), and use this condition in the following expressions.

$(a) \quad \tilde{n}(t) < 0$

Under this condition, we have

$$\dot{\tilde{c}}_r(\tau,t) = -G_1 K_r(\tau,t)\tilde{n}(t)$$

$$\dot{\tilde{c}}_s(\tau,t) = G_1 K_s(\tau,t)\tilde{n}(t) + G_2 K_s(\tau,t)\tilde{f}_d(t)$$

Using these terms along with Eq. (4.17) in Eq. (4.14), we get

$$\begin{align*}
\dot{V}(t) &= \tilde{n}(t)\left(-(\zeta(t)W + a)\tilde{n}(t) - n(t)V + s(t) + \bar{w}_{dn}(t)(n_{dn}^J - n_{dn}(t)) + \tilde{r}(t) - \tilde{s}(t)\right) \\
&\qquad + \int_0^T \tilde{c}_r(\tau,t) G_1^{-1}(-G_1 K_r(\tau,t)\tilde{n}(t))d\tau \\
&\qquad + \int_0^T \tilde{c}_s(\tau,t) G_1^{-1}(G_1 K_s(\tau,t)\tilde{n}(t) - G_2 K_s(\tau,t)\tilde{s}(t))d\tau \\
&= -(a + \zeta(t)W)\tilde{n}(t)^2 + \tilde{n}(t)\left(-n(t)V + s(t) + \bar{w}_{dn}(t)(n_{dn}^J - n_{dn}(t))\right) \\
&\qquad + \tilde{n}(t)\tilde{r}(t) - \tilde{n}(t)\tilde{s}(t) - \tilde{n}(t)\tilde{r}(t) + \tilde{n}(t)\tilde{s}(t) - \frac{G_2}{G_1}\tilde{s}(t)^2 \\
&\leq -a\tilde{n}(t)^2 - \frac{G_2}{G_1}\tilde{s}(t)^2 \leq 0 \tag{4.19}
\end{align*}$$

where we utilize the fact that

$$n(t)V - s(t) \leq \bar{w}_{dn}(t)(n_{dn}^J - n_{dn}(t))$$

$(b) \quad \tilde{n}(t) \geq 0$

Here, we have

$\dot{\tilde{c}}_r(\tau,t) = -G_1 K_r(\tau,t)\tilde{n}(t)$

$\dot{\tilde{c}}_s(\tau,t) = G_2 K_s(\tau,t)\tilde{f}_d(t)$

Substituting these terms along with Eq. (4.17) in Eq. (4.14), we get

$$\begin{aligned}
\dot{V}(t) = \tilde{n}(t)\big(&-(\zeta(t)W + a + V)\tilde{n}(t) - \hat{n}(t)V + \hat{s}(t) + \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) + \tilde{r}(t)\big) \\
&+ \int_0^T \tilde{c}_r(\tau,t)G_1^{-1}(-G_1 K_r(\tau,t)\tilde{n}(t))d\tau + \int_0^T \tilde{c}_s(\tau,t)G_1^{-1}(-G_2 K_s(\tau,t)\tilde{s}(t))d\tau \\
&= -(a + \zeta(t)W + V)\tilde{n}(t)^2 - \tilde{n}(t)(\hat{n}(t)V - \hat{s}(t) - \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t))) \\
&\quad + \tilde{n}(t)\tilde{r}(t) - \tilde{n}(t)\tilde{r}(t) - \frac{G_2}{G_1}\tilde{s}(t)^2 \\
&\le -a\tilde{n}(t)^2 - \frac{G_2}{G_1}\tilde{s}(t)^2 \le 0 \qquad (4.20)
\end{aligned}$$

where we utilize the fact that

$$\hat{n}(t)V - \hat{s}(t) \ge \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t))$$

(iii) P-C, M-F

$$\begin{aligned}
\dot{\tilde{n}}(t) &= \big(f_{up}(t) - (\bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) + r(t) - s(t)\big) \\
&\qquad - \big(a\tilde{n}(t) + \hat{f}_{up}(t) - (\hat{n}(t)V - \hat{s}(t))) + \hat{r}(t) - \hat{s}(t)\big) \\
&= -(\zeta(t)W + a)\tilde{n}(t) - \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) + \hat{n}(t)V - \hat{s}(t) + \tilde{r}(t) - \tilde{s}(t) && (4.21) \\
&= -(\zeta(t)W + a + V)\tilde{n}(t) - \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) + n(t)V - s(t) + \tilde{r}(t) && (4.22) \\
\tilde{f}_d(t) &= \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) - (n(t)V - \hat{s}(t)) \\
&= \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) - (n(t)V - s(t)) - \tilde{s}(t) && (4.23) \\
&= \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) - (\hat{n}(t)V - \hat{s}(t)) - \tilde{n}(t)V && (4.24)
\end{aligned}$$

When plant is in congestion and model is in free-flow, we have

$$\hat{n}(t)V - \hat{s}(t) \le \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) \le n(t)V - s(t) \qquad (4.25)$$

We use this condition in the results below. Depending on the sign of $\tilde{n}(t)$, we have

$(a)\quad \tilde{n}(t) > 0$

Under this condition, we have

$$\dot{\tilde{c}}_r(\tau,t) = -G_1 K_r(\tau,t)\tilde{n}(t)$$

$$\dot{\tilde{c}}_s(\tau,t) = G_1 K_s(\tau,t)\tilde{n}(t) + G_2 K_s(\tau,t)\frac{\tilde{f}_d(t) + |\tilde{f}_d(t)|}{2}$$

From Eq. (4.23), whenever $\tilde{f}_d(t) \geq 0$, we see that

$$\begin{aligned} &\tilde{f}_d(t) = \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) - (n(t)V - s(t)) - \tilde{s}(t) \geq 0 \\ &\quad \implies \tilde{s}(t) \leq \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) - (n(t)V - s(t)) \leq 0 \end{aligned} \tag{4.26}$$

Substituting these terms along with Eq. (4.21) in Eq. (4.14), we get

$$\begin{aligned} \dot{V}(t) &= \tilde{n}(t)\left(-(\zeta(t)W + a)\tilde{n}(t) - \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) + \hat{n}(t)V - \hat{s}(t) + \tilde{r}(t) - \tilde{s}(t)\right) \\ &\quad + \int_0^T \tilde{c}_r(\tau,t)G_1^{-1}(-G_1 K_r(\tau,t)\tilde{n}(t))d\tau \\ &\quad + \int_0^T \tilde{c}_s(\tau,t)G_1^{-1}(G_1 K_s(\tau,t)\tilde{n}(t) + G_2 K_s(\tau,t)\frac{\tilde{f}_d(t) + |\tilde{f}_d(t)|}{2})d\tau \\ &\quad = -(a + \zeta(t)W)\tilde{n}(t)^2 - \tilde{n}(t)(\bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) - \hat{n}(t)V + \hat{s}(t)) + \tilde{n}(t)\tilde{r}(t) \\ &\quad - \tilde{n}(t)\tilde{s}(t) - \tilde{n}(t)\tilde{r}(t) + \tilde{n}(t)\tilde{s}(t) + \frac{G_2}{G_1}\tilde{s}(t)\frac{\tilde{f}_d(t) + |\tilde{f}_d(t)|}{2} \leq 0 \\ &\quad \leq -a\tilde{n}(t)^2 + \frac{G_2}{G_1}\tilde{s}(t)\frac{\tilde{f}_d(t) + |\tilde{f}_d(t)|}{2} \leq 0 \end{aligned} \tag{4.27}$$

where we utilize Eq. (4.25) and Eq. (4.26)

$(b)\quad \tilde{n}(t) \leq 0$

From Eq. (4.25), (4.24), we have

$$\tilde{n}(t)V - \tilde{s}(t) \geq 0 \implies 0 \geq \tilde{n}(t)V \geq \tilde{s}(t), \quad \text{since} \quad \tilde{n}(t) \leq 0$$

$$\text{and} \quad \tilde{f}_d(t) = \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) - (\hat{n}(t)V - \hat{s}(t)) - \tilde{n}(t)V \geq 0$$

$$\dot{\tilde{c}}_r(\tau,t) = -G_1 K_r(\tau,t)\tilde{n}(t)$$

$$\dot{\tilde{c}}_s(\tau,t) = G_2 K_s(\tau,t)\tilde{f}_d(t) \geq 0$$

Substituting these terms along with Eq. (4.22) in Eq. (4.14), we get

$$\begin{aligned} \dot{V}(t) &= \tilde{n}(t)\left(-(\zeta(t)W + a + V)\tilde{n}(t) - \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) + n(t)V - s(t) + \tilde{r}(t)\right) \\ &\quad + \int_0^T \tilde{c}_r(\tau,t)G_1^{-1}(-G_1 K_r(\tau,t)\tilde{n}(t))d\tau + \int_0^T \tilde{c}_s(\tau,t)G_1^{-1}(G_2 K_s(\tau,t)\tilde{f}_d(t))d\tau \end{aligned}$$

$$
\begin{aligned}
&= -(a+\zeta(t)W+V)\tilde{n}(t)^2 - \tilde{n}(t)(\bar{w}_{dn}(t)(n^J_{dn}-n_{dn}(t)) - n(t)V + s(t)) + \tilde{n}(t)\tilde{r}(t) \\
&-\tilde{n}(t)\tilde{r}(t) + \frac{G_2}{G_1}\tilde{s}(t)\tilde{f}_d(t) \\
&\leq -a\tilde{n}(t)^2 + \frac{G_2}{G_1}\tilde{s}(t)\tilde{f}_d(t) \leq 0 \qquad (4.28)
\end{aligned}
$$

(iv) P-C, M-C

$$
\begin{aligned}
\dot{\tilde{n}}(t) &= \big(f_{up}(t) - (\bar{w}_{dn}(t)(n^J_{dn}-n_{dn}(t)) + r(t) - s(t)\big) \\
&\quad - \big(a\tilde{n}(t) + \hat{f}_{up}(t) - (\bar{w}_{dn}(t)(n^J_{dn}-n_{dn}(t))) + \hat{r}(t) - \hat{s}(t)\big) \\
&= -(a+\zeta(t)W)\tilde{n}(t) + \tilde{r}(t) - \tilde{s}(t) \\
\dot{\tilde{c}}_r(\tau,t) &= -G_1K_r(\tau,t)\tilde{n}(t) \\
\dot{\tilde{c}}_s(\tau,t) &= G_1K_s(\tau,t)\tilde{n}(t)
\end{aligned}
$$

When we substitute these terms in Eq. (4.14), we get

$$
\begin{aligned}
\dot{V}(t) &= \tilde{n}(t)\left(-(a+\zeta(t)W)\tilde{n}(t) + \tilde{r}(t) - \tilde{s}(t)\right) + \int_0^T \tilde{c}_r(\tau,t)G_1^{-1}(-G_1K_r(\tau,t)\tilde{n}(t))d\tau \\
&\quad + \int_0^T \tilde{c}_s(\tau,t)G_1^{-1}(G_1K_s(\tau,t)\tilde{n}(t))d\tau \\
&= -(a+\zeta(t)W)\tilde{n}(t)^2 + \tilde{n}(t)\tilde{r}(t) - \tilde{n}(t)\tilde{s}(t) - \tilde{n}(t)\tilde{r}(t) + \tilde{n}(t)\tilde{s}(t) \\
&\leq -a\tilde{n}(t)^2 \leq 0 \qquad (4.29)
\end{aligned}
$$

Therefore the Lyapunov functional $\mathbb{V}(t)$ is bounded and non-increasing. By Lyapunov's theorem [64], we conclude that

$$
\begin{aligned}
|\tilde{n}(t)| &< \infty \quad \forall t \\
\int_0^T \tilde{c}_r(\tau,t)^2 d\tau &< \infty \quad \forall t \\
\int_0^T \tilde{c}_s(\tau,t)^2 d\tau &< \infty \quad \forall t
\end{aligned}
$$

$\tilde{r}(t)$ and $\tilde{s}(t)$ are also bounded, by Schwartz's inequality, as shown below

$$
\begin{aligned}
|\tilde{r}(t)| = |\int_0^T K(\tau,t)\tilde{c}_r(\tau,t)d\tau| &\leq \left(\int_0^T K(\tau,t)^2 d\tau\right)^{\frac{1}{2}} \left(\int_0^T \tilde{c}_r(\tau,t)^2 d\tau\right)^{\frac{1}{2}} < \infty \\
|\tilde{s}(t)| = |\int_0^T K(\tau,t)\tilde{c}_s(\tau,t)d\tau| &\leq \left(\int_0^T K(\tau,t)^2 d\tau\right)^{\frac{1}{2}} \left(\int_0^T \tilde{c}_s(\tau,t)^2 d\tau\right)^{\frac{1}{2}} < \infty
\end{aligned}
$$

Since our system is periodic, by LaSalle's invariance principle [64, 62], the error equations converge to the largest invariant set which satisfies $\dot{\mathbb{V}}(t) = 0$. Examining the above expressions for

$\dot{\mathbb{V}}(t)$ in all the four cases shown above, we can conclude that $\tilde{n}(t) \to 0$. Since $\tilde{f}_{up}(t) = \zeta(t)W\tilde{n}(t)$, $\tilde{f}_{up}(t) \to 0$ also.

We can also show that model converges to the correct congestion mode at equilibrium. When the plant is in free-flow, as seen in Case (i) and Case (ii), $\tilde{s}(t) \to 0$, since $\dot{\mathbb{V}}(t) = 0$ at equilibrium. In fact, when the plant is in free-flow, the model cannot converge in the congestion mode (except along its boundary, which can be interpreted as the free-flow mode). This is because the off-ramp flows satisfy $\tilde{s}(t) = 0$ after convergence.

We can also show that the model cannot converge in the free-flow mode when the plant is in congestion (Case(iii)). This is because, under this condition we have

$$\begin{aligned} n(t)V - \hat{s}(t) < \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) \leq n(t)V - s(t) \\ \implies \tilde{f}_d(t) = \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) - (n(t)V - \hat{s}(t)) > 0 \quad \text{and} \quad \tilde{s}(t) < 0 \end{aligned}$$

and these violate $\dot{\mathbb{V}}(t) = 0$. Thus, the model always converges in the congested mode during time instants where the plant is congested. When the plant/model is in congestion (Case (iv)), we see that the convergence conditions ($\dot{\mathbb{V}}(t) = 0$) do not dictate that $\tilde{s}(t) = 0$. In fact, in this mode the off-ramp flows need not converge to their actual values. We will demonstrate this in the example presented in the next section.

Utilizing the observations in the last two paragraphs, the following equations show that $\tilde{f}(t) \to 0$.

$$\tilde{f}(t) = \min(n(t)V - s(t), \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t))) - \min(\hat{n}(t)V - \hat{s}(t), \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)))$$

$$\begin{aligned} &\text{Plant- F}: \quad && n(t)V - s(t) = \hat{n}(t)V - \hat{s}(t) && \implies \tilde{f}(t) = 0 \\ &\text{Plant- C}: \quad && \hat{n}(t)v - \hat{s}(t) \geq \bar{w}_{dn}(t)(n^J_{dn} - n_{dn}(t)) && \implies \tilde{f}(t) = 0 \end{aligned}$$

This shows the the imputed off-ramp flows satisfy Condition 4.3.1. □

The results derived for the above system with density boundary conditions also apply with other boundary conditions. The following theorem states the applicability of the sequential imputation of ramp flows to a multi-section freeway.

Theorem 4.3.2. *For the freeway described by Figure 4.1, given the upstream & downstream boundary conditions, density, flow measurements and fundamental diagram parameters of all the sections, we can impute the ramp flows sequentially from upstream to downstream section-wise using the update laws described in Section 4.2. In all the sections,* $\tilde{f}_i = 0$ *and* $\tilde{n}_i = 0$.

Proof. For the first section, since upstream boundary conditions are given, the imputation ensures $\tilde{f}_1 = 0$ and $\tilde{n}_1 = 0$, and $\hat{s}_1$ satisfies the Condition 4.3.1. We can see that for any section i, given measurement data and upstream off-ramp flow estimate satisfying Condition 4.3.1, the imputation algorithm ensures $\tilde{f}_i = 0$, $\tilde{n}_i = 0$ after imputation. Moreover, the imputed off-ramp flow, which forms the upstream boundary condition for section $i+1$ satisfies Condition 4.3.1 by Theorem 4.3.1. This proves the theorem by induction. □

4.4 Examples

The imputation algorithm is demonstrated with three examples. In the first example, we demonstrate the performance of the imputation algorithm on a single section of a freeway network. We use artificial data, in the form of known boundary conditions and on-ramp/off-ramp flows, to generate density and flow profiles in freeway section. Then these ramp flows (assumed unknown) are imputed using the imputation algorithm. The initial ramp estimates were set to be identically zero before the start of the imputation algorithm. Figure 4.3 compares the imputed trajectories generated after convergence, with the actual densities and flows which are known. As seen in Figure 4.3, the resulting flow and the density errors are very small after convergence. However, in some time periods the on-ramp and the off-ramp values have not converged to their true values. These segments correspond to the P-C M-C mode, and in this case, the on-ramp and off-ramp flows cannot be determined individually, only the effective ramp flow $r(t) - s(t)$ can be determined. In fact, there are infinitely many combinations of on-ramp and off-ramp flows in the P-C M-C mode that can produce the observed density/flows. Only when a measurement of the flow is available at a location just before the off-ramp or just after the on-ramp (in addition to the already available measurements), we can estimate the ramp flows uniquely. Here, these measurements corresponds to $f_i(t) + s_i(t)$ and $f_{i-1}(t) + r_{i-1}(t)$ respectively. Since we have measurements of $f_i(t)$, these additional measurements allow us to uniquely determine the ramp flows in the P-C, M-C mode. However, in many freeways in the U.S, these quantities are not usually measured.

In the second example, a 6.3 mile highway segment from I-210E was chosen. This freeway segment was divided into 8 sections with 9 on-ramps and 7 off-ramps, out of which 3 on-ramps and 5 off-ramps were imputed. In any section, if one of the on-ramp flows/off-ramp split ratios are known, then we directly substitute the known ramp flows in the model. We only execute the adaptation equations for the ramps flows that need to be imputed. In locations where one of the ramp flows is known, we can impute the other ramp flows uniquely. After obtaining all the imputed ramp flows, we use these ramp flows to simulate the traffic dynamics in the entire freeway. For simulation, we use the ACTM model. The density, flow and velocity contours of the simulation are shown in Figure 4.4. The x-axis of the contours represent PostMiles, which measure distance along the freeway, and the Y-axis represents the time of the day. Table 4.3 lists the density/flows errors for this simulation.

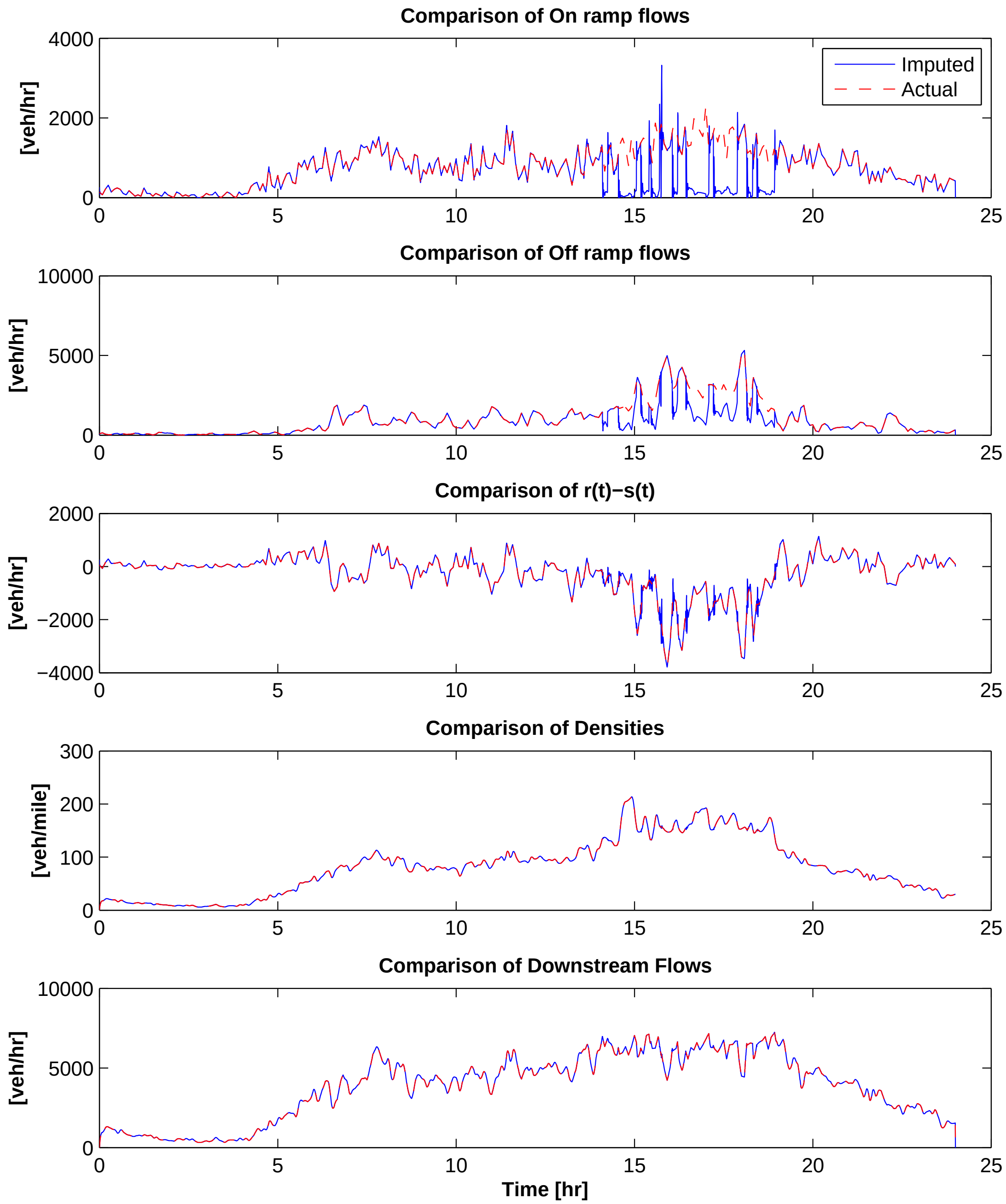

Figure 4.3: Results of the Imputation with artificial data.

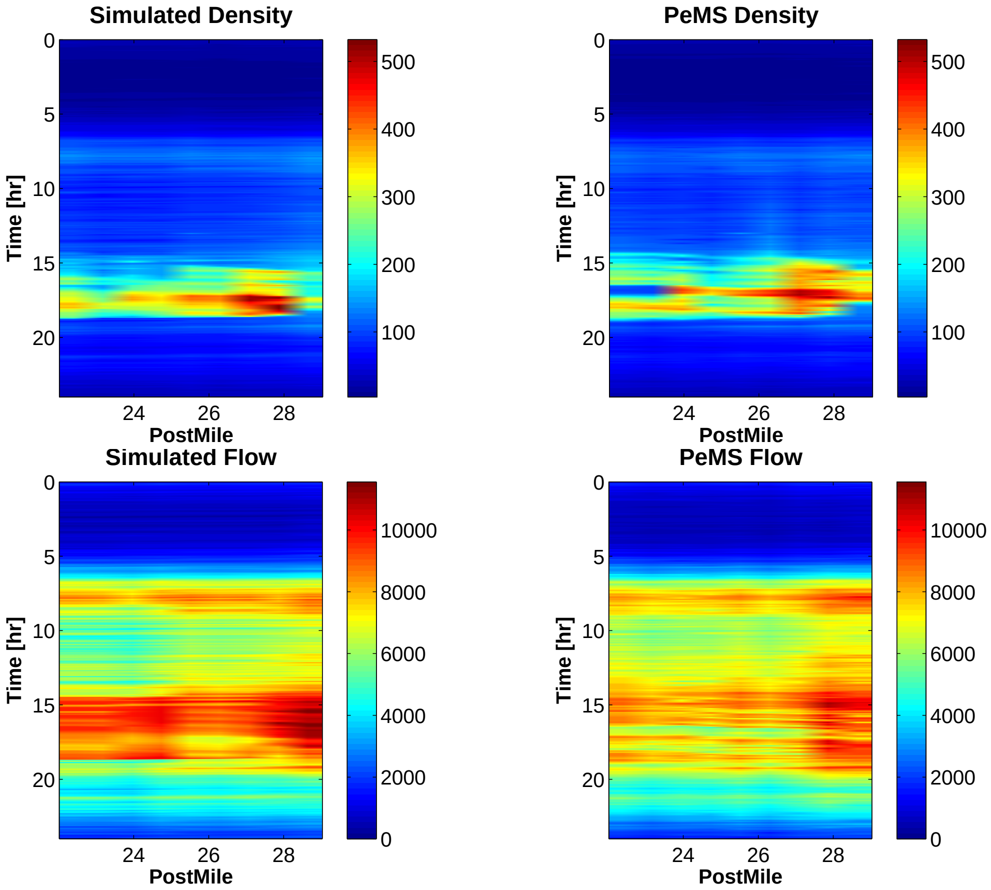

Figure 4.4: Simulation results for I-210E.

In the third example, a 8.8 mile highway segment from I-80W was chosen. This freeway segment was divided into 10 sections with 8 on-ramps and 9 off-ramps, all of which had no measurements. Ramp flows were imputed and then used to simulate the entire section. The density, flow and velocity contours of the simulation are shown in Figure 4.5. The density and flow errors for the simulation are given in Table 4.3.

Freeway	**Density error**	**Flow error**
I-210E	15%	11 %
I-80W	8.4%	8.75 %

Table 4.3: Final errors for simulations carried out with ramp flows imputed using the ACTM based imputation algorithm.

It was observed that in many sections (for I80W, I210E imputation examples), the imputation algorithm stopped converging and the solutions showed significant errors in flow and density. While small errors in the final density/flow are expected due to some model mismatch, larger errors

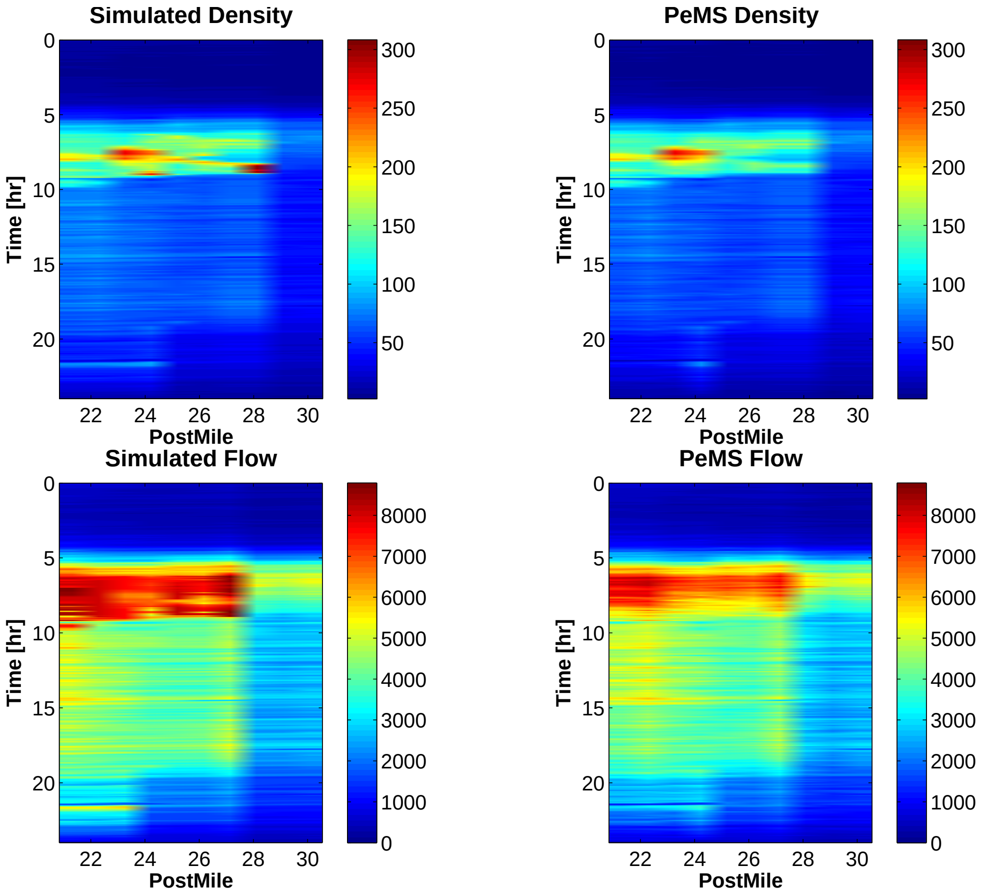

Figure 4.5: Simulation results for I-80W.

indicate that the some of the measurements may be faulty. This is because the imputation algorithm is expected to converge to zero density/flow error values in case there exists some plausible ramp flows that replicate the freeway dynamics. These faulty measurements may either correspond to the mainline density/flow measurements of the current link, or the density/flow measurements in one of the boundaries. In [18], Dervisoglu and Horowitz use this property to determine the freeway mainline detectors that report erroneous/faulty measurements, even under conditions when on-ramps and off-ramps are imputed using these mainline detector data. Extending the results presented in this chapter, the authors analyze cases where systematic faults can be detected. This algorithm is also successfully demonstrated on a 23 mile segment in the I210W freeway in California in their paper.

4.5 Proofs

Proof of Lemma 4.3.1

Proof. Depending on the upstream flow condition in the model and plant, the system falls into four modes.

Case (a) : Plant upstream (F), Model upstream (F)

$$\begin{aligned} &f_u(t) = \min(F_{up}, n_{up}(t)v_{up} - s_{up}(t))) \\ &\hat{f}_u(t) = \min(F_{up}, n_{up}(t)v_{up} - s_{up}(t))) \\ &\tilde{f}_u(t) = 0 \end{aligned} \tag{4.30}$$

Case (b) : Plant upstream (F), Model upstream (C)

$$\begin{aligned} &f_u(t) = \min(F_{up}, n_{up}(t)v_{up} - s_{up}(t))) \\ &\hat{f}_u(t) = w(n^J - \hat{n}(t)) \\ &\tilde{f}_u(t) = -\zeta(t)w\tilde{n}(t) \qquad 0 \leq \zeta(t) \leq 1 \\ &\text{since} \quad w(n^J - n(t)) > \min(F_{up}, n_{up}(t)v_{up} - s_{up}(t))) > w(n^J - \hat{n}(t)) \end{aligned} \tag{4.31}$$

Case (c) : Plant upstream (C), Model upstream (C)

$$\begin{aligned} &f_u(t) = w(n^J - n(t)) \\ &\hat{f}_u(t) = w(n^J - \hat{n}(t)) \\ &\tilde{f}_u(t) = -w\tilde{n}(t) \end{aligned} \tag{4.32}$$

Case (d) : Plant upstream (C), Model upstream (F)

$$\begin{aligned} &f_u(t) = w(n^J - n(t)) \\ &\hat{f}_u(t) = \min(F_{up}, n_{up}(t)v_{up} - \hat{s}_{up}(t))) \\ &\tilde{f}_u(t) = -\zeta(t)w\tilde{n}(t) \qquad 0 \leq \zeta(t) \leq 1 \\ &\text{since} \quad (n^J - n(t)) < \min(F_{up}, n_{up}(t)v_{up} - \hat{s}_{up}(t))) < w(n^J - \hat{n}(t)) \end{aligned} \tag{4.33}$$

Hence $\tilde{f}_u(t) = -\zeta(t)w\tilde{n}(t)$ where $0 \leq \zeta(t) \leq 1$, irrespective of the plant/model upstream flow conditions □

Chapter 5

Imputation of ramp flow data using the link-node cell transmission model

In this chapter, we present an imputation algorithm for on-ramp/off-ramp flows based on the Link-Node Cell Transmission Model (LN-CTM). The LN-CTM is well suited for modeling traffic in freeways with large on-ramps (for example, freeway-freeway interconnections) as compared to the ACTM. However, the design of an algorithm based on the LN-CTM poses additional challenges. The ACTM simplifies the non-linear model equations of the CTM and the resulting dynamic equations are piecewise affine. Also, the structure of the ACTM allows the estimation of the ramp flows section-wise, and leads to easy analysis of the convergence of the imputation algorithm. As we will see in this chapter, the LN-CTM presents additional complications due to the non-linearities of the model.

We first recap the LN-CTM, and rewrite the equations to motivate the development of the imputation algorithm based on the LN-CTM. The structure of the LN-CTM dictates that all of the ramp flows are simultaneously estimated for the entire freeway. The imputation procedure tackles the problem in two steps, first calculating a total/effective demand parameter capturing aggregate flows into and out of each link and then decomposing the effective demand into on-ramp demands and off-ramp split ratios, along with the corresponding flows. The first step of the algorithm uses an adaptive iterative learning algorithm, that matches the model calculated densities and the measured densities. The second step uses a linear program to minimize the error between the model calculated flows and the measurements.

5.1 Link Node Cell Transmission Model

The LN-CTM is used as the underlying model for the imputation algorithm presented in this chapter. Section 3.1 presents the LN-CTM for traffic flow simulations, along with the geometry specification, variable definitions and their units.

For the purposes of the imputation algorithm, we introduce a few additional terms,

Capacity adjusted free-flow speed ($\bar{v}(k)$) : $\bar{v}_i(k) = \bar{v}_i(n_i(k)) = \min\left(V_i, \frac{F_i}{n_i(k)}\right)$

Capacity adjusted congestion wave speed ($\bar{w}(k)$) : $\bar{w}_i(k) = \bar{w}_i(n_i(k)) = \min\left(W_i, \frac{F_i}{(n_i^J - n_i(k))}\right)$

Total/effective demand ($c_i(k)$) : $c_i(k) = n_i(k)\bar{v}_i(k)(1-\beta_i(k)) + d_i(k)$

Under these definitions, the demand, and supply functions can be represented as

$$\begin{aligned} D_i(k) &= \min(n_i(k)V_i, F_i) = n_i(k)\bar{v}_i(k) \\ S_i(k) &= \bar{w}_i(k)(n_i^J - n_i(k)) \end{aligned} \tag{5.1}$$

The total effective demand closely resembles the total demand vector $R_i(k)$, described in Section 3.1, with some additional assumptions on the on-ramp demand function (originally defined as $d_i(k) = \min(l_i(k), r^c(k)_i, C_i)$). We assume that ramp metering is inactive, and there are only short queues present in the ramps. The first assumption implies that $r_i^c(k) = C_i$, while the second assumption can be rigorously stated as $l_i(k) \leq C_i$. These assumptions are necessary to ensure observability of the on-ramp queue lengths through the on-ramp demand function (i.e $d_i(k) = l_i(k)$). This is necessary to determine the on-ramp input flows, which will be extracted from the queue lengths. These assumptions are reflected in the model equations given below,

$$n_i(k+1) = g(n_i(k), c_{i-1}(k), c_i(k), n_{i+1}(k)) = n_i(k) + f_i^{in}(k) - f_i^{out}(k) \tag{5.2}$$

$$\begin{aligned} f_i^{in}(k) &= f_i^{in}(c_{i-1}(k), n_i(k)) = \min\left(\bar{w}_i(k)(n_i^J - n_i(k)), c_{i-1}(k))\right) \\ f_i^{out}(k) &= f_i^{out}(c_i(k), n_i(k), n_{i+1}(k)) = n_i(k)\bar{v}_i(k) min\left(1, \frac{\bar{w}_{i+1}(k)(n_{i+1}^J - n_{i+1}(k))}{c_i(k)}\right) \\ c_i(k) &= n_i(k)\bar{v}_i(k)(1-\beta_i(k)) + d_i(k) \\ d_i(k+1) &= d_i(k) + Q_i(k) - r_i(k) \\ s_i(k) &= \beta_i(k) f_i^{out}(k) \\ r_i(k) &= \frac{min(c_i(k), \bar{w}_{i+1}(k)(n_{i+1}^J - n_{i+1}(k)))}{c_i(k)} d_i(k) \end{aligned} \tag{5.3}$$

The above equations represent the model evolution for an interior link. In the case of the first link (Link 0), $f_i^{in} = Q_0$ and for the last link, under free-flow conditions, $f_i^{out}(k) = n_i(k)\bar{v}_i(k)$.

The on-ramp and off-ramp flows/demands affect the dynamics of the freeway through the effective demands. Hence, from the point of view of the freeway mainline dynamics, given the effective demand, the on-ramp and off-ramp flows provide no additional information for the simulation process. We take advantage of this fact in the imputation algorithm, and first estimate the effective demands before obtaining the individual ramp flows.

The update equation for any particular link can be represented as a four mode piecewise non-linear model for density evolution. Each link on the freeway has a four mode update equation, and the modes are dependent on the flow conditions at the input and output node. Flow conditions at Node $i-1$ (i.e. the in-flow into Link i and flow out of Link $i-1$) is said to be in congestion if $c_{i-1}(k) > \bar{w_i}(k)(n_i^J - n_i(k))$, otherwise it is in free-flow. Considering the possibility of occurrence of one of the two modes in the input and output nodes of a link, the density evolution can be described by a four mode update model, where the modes will be referred as FF, FC, CC ,CF (C-congestion, F- free flow) with the first (second) letter specifying the input (output) node conditions. The model evolution in these four modes can be written as

$$\begin{aligned}
n_i(k+1) &= n_i(k) + c_{i-1}(k) - n_i(k)\bar{v_i}(k) && \textbf{-FF} \\
n_i(k+1) &= n_i(k) + c_{i-1}(k) - n_i(k)\bar{v_i}(k)\frac{\bar{w}_{i+1}(k)(n_{i+1}^J - n_{i+1}(k))}{c_i(k)} && \textbf{-FC} \\
n_i(k+1) &= n_i(k) + \bar{w_i}(k)(n_i^J - n_i(k)) - n_i(k)\bar{v_i}(k)\frac{\bar{w}_{i+1}(k)(n_{i+1}^J - n_{i+1}(k))}{c_i(k)} && \textbf{-CC} \\
n_i(k+1) &= n_i(k) + \bar{w_i}(k)(n_i^J - n_i(k)) - n_i(k)\bar{v_i}(k) && \textbf{-CF} \qquad (5.4)
\end{aligned}$$

From the above equations, we see that the effective demand corresponding to a Node i appears in the density evolution equation of Link i or Link $i+1$ (not both) depending on whether the node conditions are in free-flow or congestion. This means that the demands may be observable from the densities of the link before/after the ramp locations, depending on the traffic mode. Also, the mode is itself dependent on the effective demand, which means that an incorrect estimate of the effective demands can possibly lead to density evolution in the wrong mode. Finally, we also note that the density update equations are either a linear or a non-linear function of the effective demand depending on the congestion mode. All these factors dictate that any estimation procedure needs to simultaneously estimate these effective demands for the entire freeway, and the non-linearity prevents us from performing the imputation section-wise like the procedure adopted for the ACTM imputation.

5.2 The Imputation Algorithm

The imputation algorithm is based on the four-mode model presented in the previous section. The algorithm estimates ramp flows that replicate observed freeway behavior, captured by the freeway

density and flow measurements. Formally, the problem can be stated as :
Problem. *Estimate ramp demands, ramp input flows and split ratio profiles ($\hat{d}_i(k), \hat{Q}_i(k), \hat{\beta}_i(k)$, $k = 1 \cdots K$ and $i = 1 \cdots N$) such that the model evolution (Eq.* (5.3) *) using these estimates replicate the given flow and density measurements ($n_i(k), f_i^{meas}(k)$) obtained from detectors along the freeway mainline.*

In contrast to the imputation algorithm based on the ACTM , which simultaneously tries to match the density and flow profiles, we split the problem into two steps. From the density evolution equations, we can see that the on-ramp flows and off-ramp split ratios have a combined effect captured by the effective demand function. In the first step, we will estimate the effective flow demand function $c_i(k)$ with the function estimate $\hat{c}_i(k)$ and then extract the on-ramp flows and off-ramp split ratio functions.

Estimation of the effective demands

The density evolution (Eq. (5.3)) along the freeway can be completely specified using the effective demand profile. The first step of the imputation algorithm computes the unknown effective demands for the entire freeway section by allowing the model calculated density profiles to track the measured density profiles.

This contrasts the section-wise procedure adopted for the ACTM based imputation algorithm because the LN-CTM does not offer the same decoupling properties as the ACTM. In particular, the LN-CTM has both input and state non-linearity and it does not have the piecewise linear density update structure of the ACTM. This imputation algorithm also improves on some of the assumptions used in the ACTM algorithm. First, we directly present the algorithm in discrete time, and the convergence results are also presented in discrete time. Secondly, the ACTM imputation algorithm required the density evolution profiles to be 24-hour periodic, so that an adaptive repetitive (periodic) algorithm can be used to estimate the (periodic) ramp flow parameters. The algorithm presented here does not utilize that assumption, and is based on an adaptive learning algorithm where exact re-initialization of the initial conditions is used. This class of adaptation algorithms, which requires exact re-initialization of the initial conditions are generally known as adaptive iterative learning laws and have been widely used in robotics and other mechatronics applications [74, 61]. Given initial estimates for the effective demands ($\hat{c}_i^0(k), k = 1 \cdots K,\ i = 1, \cdots N-1$), we run the model for the given simulation period, and simultaneously adaptively estimate the parameters to decrease the density errors. This forms a single iteration of the algorithm. The procedure is iterated with exact re-initialization of the initial state of the link densities. Unlike many iterative control applications, exact re-initialization does not pose a problem here, since initial conditions are usually specified in the simulation model. In fact, in our case, the initial conditions are the density measurements at the start time. Finally, the current algorithm can be applied to situations where partial profiles are known.

We will distinguish between actual measured variables and their estimates by placing a hat on top of the estimates. Moreover, since the effective demand function estimates will be updated at each simulation iteration by an adaptive iterative learning algorithm, we will introduce the superscript j to denote the iteration index. For example, the density (number of vehicles) in Link i at period k estimated by the model at iteration j will be denoted by $\hat{n}_i^j(k)$, in order to differentiate it from the actual measured number of vehicles $n_i(k)^1$. According to our model, at each node there is at most one on-ramp and one off-ramp. If both the on-ramp demand and the off-ramp split ratio are measurable during some time instants, then the effective demand function estimate can be calculated by $\hat{c}_i^j(k) = \hat{n}_i^j(k)\hat{v}_i^j(k)(1-\beta_i(k)) + d_i(k)$. By contrast, if either the on-ramp or the off-ramp flow measurements are not available and need to be imputed at the time instant k, the effective demand function estimate $\hat{c}_i^j(k)$ will have to be imputed using the adaptive iterative learning algorithm. Let $I_i(k) = 1$ denote that the effective demand function c_i is imputed at time instant k, while $I_i(k) = 0$ denotes that the ramp measurements are available.

As discussed above, within each iteration we adaptively estimate the effective demand function estimates $\hat{c}_i^j(k)$ at each simulation time step. The demand function estimates are used in turn to update the density (number of vehicle) estimates and their corresponding estimation errors. Two different density estimates (and their corresponding errors) will be generated: the a-priori estimates and the a-posteriori estimates. A-priori density estimates are generated by applying the effective demand function estimates from the previous iteration (i.e. $\hat{c}_i^{j-1}(k)$, where j is the current iteration index) to the density update equation. In contrast, A-posteriori density estimates use the demand function estimates from the current iteration (i.e. $\hat{c}_i^j(k)$, where j is the current iteration index). In the formulae that follow, a-priori errors are represented by placing a tilde and a ‘^{o}’ (e.g. $\tilde{n}_i^{j,o}(k)$) while a-posteriori errors estimates and actual a-posteriori errors are represented with a bar (e.g. $\bar{n}_i^j(k)$) and tilde (e.g. $\tilde{n}_i^j(k)$).

It is assumed that effective demand estimates are bounded $0 < c_{min} < \hat{c}_i(k) < c_{max} < \infty$ with known bounds c_{min}, c_{max}. Any non-zero feasible initial estimates are allowed (e.g. $\hat{c}_i^j = c_{min}$ for $j = 0$). Also exact re-initialization implies $\hat{n}_i^j(0) = n_i(0) \quad \forall i = 1...N$.

$$M_i^j(k) = sign(\hat{c}_{i-1}^{j-1}(k) - \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)))$$
$$\bar{M}_i^j(k) = sign(\hat{c}_{i-1}^j - \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)))$$

determine the a-priori and a-posteriori mode (congested / free-flow conditions) corresponding to flows in Node i (from Link $i-1$ to Link i). It can be seen that $M_i^j(k) = 1$ under congested flow conditions, and $M_i^j(k) = -1$ under free-flow conditions. When $M_i^j(k) = 0$, the flow conditions can either be interpreted as congested or free-flow.

The adaptation equations at the simulation step (k) and iteration (j) involves the following sequence of steps, for each link:

(a) Compute the a-priori density errors for Link i, $\tilde{n}_i^{j,o}(k+1)$ and congestion modes $M_i^j(k)$

[1] The jam number of vehicles of Link i is denoted n_i^J, which is a constant parameter.

and $M_{i+1}^{j}(k)$ for the corresponding upstream and downstream node pairs (i, $i+1$):

$$\begin{aligned}
&\tilde{n}_i^{j,o}(k+1) = n_i(k+1) - g(\hat{n}_i^j(k), \hat{c}_{i-1}^{j-1}(k), \hat{c}_i^{j-1}(k), \hat{n}_{i+1}^j(k)) \\
&M_i^j(k) = sign(\hat{c}_{i-1}^{j-1}(k) - \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k))) \\
&M_{i+1}^j(k) = sign(\hat{c}_i^{j-1}(k) - \hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k)))
\end{aligned} \tag{5.5}$$

where the function $g(\cdot)$ is defined in Eq. (5.2).

(b) Compute a-posteriori density error estimate. This depends on the a-priori congestion mode (for example, in FC mode both $c_{i-1}(k)$ and $c_i(k)$ affect link density).

$$\begin{aligned}
&Set\ \Lambda_1^{i,j} = 0 \quad and \quad \Lambda_2^{i,j} = 0 \\
&if \quad M_i^j(k) < 0\ or\ \left(M_i^j(k) = 0\ and\ \tilde{n}_{i-1}^{j,o}(k+1) \le 0\right) \\
&\qquad\qquad \Lambda_1^{i,j} = G_1 \times I_{i-1}(k) \\
&if \quad M_{i+1}^j(k) > 0\ or\ \left(M_{i+1}^j(k) = 0\ and\ \tilde{n}_i^{j,o}(k+1) > 0\right) \\
&\qquad\qquad \Lambda_2^{i,j} = G_2 \times I_i(k)\hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k))) \times \hat{n}_i^j(k)\hat{v}_i^j(k)
\end{aligned}$$

$$\overline{n}_i^j(k+1) = \frac{\tilde{n}_i^{j,o}(k+1)}{1 + \Lambda_1^{i,j} + \Lambda_2^{i,j}} \tag{5.6}$$

If the effective demand function corresponding to a particular node is not imputed, we see that the corresponding term $\Lambda_1^{i,j}/\Lambda_2^{i,j}$ is set to zero. We are able to calculate the a-posteriori error estimate, without knowing the effective demand estimate at the current iteration. This is possible through mathematical manipulations that are commonly used in the adaptive control literature. This result will be derived in detail in the next section when we analyze the properties of this algorithm.

(c) Update effective demand estimates using the a-posteriori density estimate.

$$\begin{aligned}
&if \quad \Lambda_1^{i,j} > 0 \\
&\qquad \hat{c}_{i-1}^j(k) = max\left(c_{min}, min\left(\hat{w}_i(n_i^J - \hat{n}_i^j(k)), c_{t,1}\right)\right) \\
&\qquad where\, c_{t,1} = \hat{c}_{i-1}^{j-1}(k) + G_1\overline{n}_i^j(k+1) \\
&\qquad \acute{M}_i^j(k) = 0
\end{aligned}$$

$$
\begin{aligned}
&if \quad \Lambda_2^{i,j} > 0\\
&\quad \hat{c}_i^j(k) = \frac{1}{min\left(\frac{1}{\hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k))}, max\left(\frac{1}{c_{max}}, \frac{1}{c_{t,2}}\right)\right)}\\
&\quad where\, c_{t,2} = \frac{1}{1/\hat{c}_{i-1}^{j-1}(k) - G_2\overline{n}_i^j(k+1)}\\
&\quad \acute{M}_{i+1}^j(k) = 1\\
&\hat{n}_i^j(k+1) = g(\hat{n}_i^j(k), \hat{c}_{i-1}^j(k), \hat{c}_i^j(k))
\end{aligned}
\tag{5.7}
$$

From the previous step, we note that $\Lambda_1^{i+1,j} \times \Lambda_2^{i,j} = 0$. This implies that $\hat{c}_i$ is updated with density errors corresponding to either Link i or Link $i+1$, not both. As seen in Eq. (5.4), the current congestion mode, determined by the effective demand estimate $\hat{c}_i^{j-1}(k)$ from the previous iteration, determines whether changes in the effective demand estimate affect the density equations of Link i or Link $i+1$. The effective demand estimates are updated using the density errors of the Link directly affected by its changes. This guarantees that the a-posteriori errors are not larger than the a-priori errors. The bounds used for the update equations (Eq. (5.7)) ensure that the a-posteriori mode is the same as the a-priori mode (i.e. $\overline{M}_i^j(k) \times M_i^j(k) \geq 0$), so that the update equations can exploit the piecewise nonlinearity structure of the state equations. This ensures that any changes in the effective demand parameter estimates only affect the density errors of the link which is used in its update equations. When $M_i^j(k) = 0$ (the flow is both free-flow and congested), the update is chosen to ensure that parameter updates at-least decrease the error in the assigned link, as seen in Eq. (5.6). $\acute{M}_i^j(k)$ is used to capture the congestion mode corresponding to the updates used. It equals 0 or 1 depending on whether the updates are carried out under the first (free-flow)/second (congested) conditions in Eq. (5.7). When the effective demand estimate is not imputed at a particular time step, we set $\acute{M}_i^j(k) = 1$, if $\hat{c}_i^{j-1}(k) \leq \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)))$ and 0 otherwise.

(d) Calculate a-posteriori density estimate, which can be used in the next simulation step.

$$
\begin{aligned}
\hat{n}_i^j(k+1) &= g(\hat{n}_i^j(k), \hat{c}_{i-1}^j(k), \hat{c}_i^j(k))\\
\tilde{n}_i^j(k+1) &= n_i(k+1) - \hat{n}_i^j(k+1)
\end{aligned}
\tag{5.8}
$$

The a-posteriori density error estimate $\overline{n_i}(k)$ is different from the actual a-posteriori error ($\tilde{n}_i(k)$), since the updates adhere to the minimum and maximum bounds. If the bounds are not enforced (active) during the update procedure, then these will be the same.

At any iteration, we execute the four steps detailed above for each of the links at the current time period k. This provides the initial conditions to execute the update equations for the next time period $k+1$, where the process is repeated. When $k = K$, the current iteration is finished, and we move on to the next iteration. At the start of any iteration, we execute the exact re-initialization condition

$$\hat{n}_i^j(0) = n_i(0) \quad \forall i = 1...N$$

which gives us the initial condition for link density estimates. We execute multiple iterations until the density errors across all links ($\sum_k |n(k) - \hat{n}(k)|$) becomes insignificant, i.e.,

$$\sum_k |\tilde{n}(k)| < 0.005 \times \sum_k n(k)$$

or stop decreasing,

$$\Delta\left(\sum_k |\tilde{n}(k)|\right) < 0.005 \times \sum_k n(k)$$

where $\Delta(.)$ is the change in errors across iterations.

The four steps presented above can be elaborated into different update equations for each of the four modes (FF/FC/CF/CC). It is interesting to note that in the case of the CF mode, the update equations do not depend on either $\hat{c}_i$ or $\hat{c}_{i-1}$ and hence the a-posteriori error will equal the a-priori error when this congestion mode is in effect. In contrast, in the FC mode, both $\hat{c}_i$ and $\hat{c}_{i-1}$ affect the density estimate. The update equations are decentralized, local updates only require knowledge of local modes, not the actual mode of the entire system. This is a desirable feature as the total number of modes in the entire systems is 2^N, which grows as size of the freeway increases. Finally, its easy to see that density measurements are only needed in links when the ramps in the input node/output node need to be imputed.

The adaptive iterative algorithm presented above can incorrectly converge to some modes. The most common occurrence is the incorrect convergence in the CF mode, where none of the ramps are affected by the density errors, and incorrect convergence in this mode (at some links) cannot be thwarted by allowing more iterations of the algorithm. In these cases, it is possible to identify (after the algorithm converges), the "actual" mode of the link and reset the parameter updates to ensure correct convergence. There exists principled (though theoretically slow) methods to ensure perfect convergence when an input profile exists which can follow the given profiles. However, in most of the cases, due to noisy/incorrect measurements, no c_i exists, satisfying $c_{min} < c_i(k) < c_{max}$, that allows the system to track the measured density profiles. Hence, we need to resort to heuristic methods. Here we list a heuristic method that has worked well in practice.

Trigger Algorithm

$$\forall \quad i,k \quad \text{s.t.} \quad \tilde{n}_i^j(k+1) > \text{tolerance},\ I_{i-1}(k) = 1,\ I_i(k) = 1\ \acute{M}_i^j(k) = 1,\ \acute{M}_{i+1}^j(k) = 0$$

$$\mathit{set} \quad \hat{c}_i^j(k) = \Upsilon_1 \hat{w}_{i+1}^j(k)(n_{i+1}^J - n_{i+1}^j(k))$$

$$\forall \quad i,k \quad \text{s.t.} \quad \tilde{n}_i^j(k+1) < -\text{tolerance},\ I_{i-1}(k) = 1,\ I_i(k) = 1\ \acute{M}_i^j(k) = 1,\ \acute{M}_{i+1}^j(k) = 0$$

$$\mathit{set} \quad \hat{c}_{i-1}^j(k) = \Upsilon_2 \hat{w}_i^j(k)(n_i^J - n_i^j(k))$$

$$\begin{aligned}
\forall \quad i,k \quad & \text{s.t.} \quad \tilde{n}_i^j(k+1) > \text{tolerance},\ I_{i-1}(k)=0,\ I_i(k)=1,\ \acute{M}_{i+1}^j(k)=0 \\
& \textit{set} \quad \hat{c}_i^j(k) = \Upsilon_1 \hat{w}_{i+1}^j(k)(n_{i+1}^J - n_{i+1}^j(k)) \\
\forall \quad i,k \quad & \text{s.t.} \quad \tilde{n}_i^j(k+1) < -\text{tolerance},\ I_{i-1}(k)=1,\ I_i(k)=0\ \acute{M}_i^j(k)=1 \\
& \textit{set} \quad \hat{c}_{i-1}^j(k) = \Upsilon_2 \hat{w}_i^j(k)(n_i^J - n_i^j(k))
\end{aligned}$$

where $\Upsilon_1 > 1$ and $\Upsilon_2 < 1$ are positive reset factors. The first two resets presented above correspond to the condition when link *i* gets stuck in the CF mode. These resets are made according to the sign of the error, so as to ensure that the algorithm is able to decrease the errors in future iterations, by modifying the effective demand estimates which are triggered. A similar situation arises when one of the demand estimates is known and not imputed. These cases are captured by the last two reset conditions.

With the trigger algorithm in place, we summarize the complete algorithm below :

Step a : Assume initial estimates
Step b : Iterate until rate of change of errors is below tolerance
Stop if error is within tolerance, otherwise go to Step c
Step c : Trigger and go back to step b.

As we execute multiple triggers, there is a possibility that the errors can increase after a particular trigger is executed. This mostly happens due to the inherent noise present in the measurements. In this case, the best results across iterations are used.

Estimation of the on-ramp flows and off-ramp split ratios

Once the effective demands are estimated for all sections, we need to extract the on-ramp demand and off-ramp split ratios from the effective demand vector. In order to ensure that the on-ramp flow and off-ramp splits track the dynamics obtained using the effective demand parameter estimate, we require that that $\hat{\beta}_i(k)$ and $\hat{d}_i(k)$ satisfy, $\hat{c}_i(k) = \hat{n}_i(k)\hat{v}_i(k)(1-\hat{\beta}_i(k)) + \hat{d}_i(k)$ at all time instants, where $\hat{n}_i(k)$ is the density profile obtained using $\hat{c}_i(k)$ in the model equations. With this constraint it can be seen that ramp demand and split ratio estimates are non-unique, unless either (a) On-ramp flows/demands are known (b) Off-ramp flows/splits are measured (c) Mainline flow in between the on-ramp and the off-ramp, are available or (d) one of the ramps are absent. Figure 5.1 illustrates the position of the mainline detector, from which flow data is available. On-ramp detectors are usually placed at the entrance into the freeway, and hence they measure on-ramp exit flow, not the on-ramp input demand. Off-ramp detectors are placed near the off-ramp entry.

When any of the measurements listed above are available, we can frame the problem as a linear program. The structure of the linear program depends on whether the node is in free-flow/congestion. In both cases, depending on which measurements are available, the objective

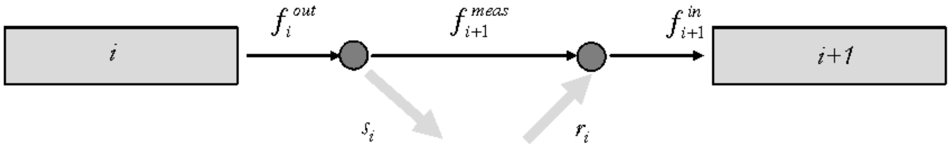

Figure 5.1: Decouple on-ramp and off-ramp flows.

of the linear program can be written as,

$$\begin{aligned}
&J^1 = |(\hat{f}^{in}_{i+1}(k) - f^{meas}_{i+1}(k)) - \hat{r}_{i+1}(k)| + |(\hat{f}^{out}_{i}(k) - f^{meas}_{i+1}(k)) - \hat{s}_{i+1}(k)| \\
&J^2 = |r_{i+1}(k) - \hat{r}_{i+1}(k)| \\
&J^3 = |s_{i+1}(k) - \hat{s}_{i+1}(k)| \\
&J^4 = |(\hat{f}^{in}_{i+1}(k) - f^{meas}_{i+1}(k)) - \hat{r}_{i+1}(k)| + |(\hat{f}^{out}_{i}(k) - f^{meas}_{i+1}(k)) - \hat{s}_{i+1}(k)| + \alpha|s_{i+1}(k) - \hat{s}_{i+1}(k)| \\
&J^5 = |(\hat{f}^{in}_{i+1}(k) - f^{meas}_{i+1}(k)) - \hat{r}_{i+1}(k)| + |(\hat{f}^{out}_{i}(k) - f^{meas}_{i+1}(k)) - \hat{s}_{i+1}(k)| + \alpha|r_{i+1}(k) - \hat{r}_{i+1}(k)|
\end{aligned} \tag{5.9}$$

$\hat{f}^{in}_{i+1}(k)$ and $\hat{f}^{out}_{i}(k)$ are obtained using $\hat{c}_i(k)$ in the model equations. $\alpha \geq 1$ is used to increase the weight of the errors in ramp flows in the last two objectives. we The optimization problem for decoupling the ramp flows is given by

Free-flow

$$\begin{aligned}
\min \quad & J^* \\
s.t \quad & \hat{c}_i(k) = \hat{n}_i(k)\hat{v}_i(k) - \hat{s}_i(k) + \hat{r}_i(k) \\
& \hat{r}_i(k) \geq d_{min}(k) \\
& \hat{s}_i(k), \hat{r}_i(k) \geq 0
\end{aligned}$$

In the case of free-flow, $\hat{r}_i(k) = \hat{d}_i(k)$ and $\hat{\beta}_i(k) = \frac{\hat{s}_i(k)}{\hat{n}_i(k)\hat{v}_i(k)}$.

Congestion

$$\begin{aligned}
\min \quad & J^* \\
s.t \quad & \hat{c}_i(k) = \hat{n}_i(k)\hat{v}_i(k) - \hat{f}r_i(k) + \hat{d}_i(k) \\
& \hat{s}_i(k) = \frac{\hat{w}^j_{i+1}(k)(n^J_{i+1} - \hat{n}^j_{i+1}(k))}{\hat{c}_i(k)} \hat{f}r_i(k) \\
& \hat{r}_i(k) = \frac{\hat{w}^j_{i+1}(k)(n^J_{i+1} - \hat{n}^j_{i+1}(k))}{\hat{c}_i(k)} \hat{d}_i(k) \\
& \hat{d}_i(k) \geq d_{min}(k) \\
& \hat{s}_i(k), \hat{r}_i(k) \geq 0
\end{aligned}$$

In the case of congested conditions, we can obtain the split ratio estimate as $\hat{\beta}_i(k) = \frac{\hat{fr}_i(k)}{\hat{n}_i(k)\hat{v}_i(k)}$.

In both the problems above $d_{min}(k) = \hat{d}_i(k-1) - \hat{r}_i(k-1) \quad k = 2..N$ with $d_{min}(0) = d_{init}$ where d_{init} denotes the initial condition of the system. This variable tracks the residual demand from the previous time instant. Finally, the on-ramp input flows can be calculated as $\hat{Q}_i(k) = \hat{d}_i(k+1) - (\hat{d}_i(k) - \hat{r}_i(k))$.

5.3 Convergence Analysis

The imputation algorithm presented here has been designed in view of obtaining favorable convergence properties. This is particularly beneficial to certify the performance of the algorithm in relatively unsupervised applications. We analyze the first step of the imputation algorithm, involving the adaptive iterative estimation of effective demands. The first important property we will explore is the boundedness and convergence of the density errors and the effective demand estimates. The change in density across iterations is given by

$$\begin{aligned}\hat{n}_i^j(k+1) - \hat{n}_i^{j-1}(k+1) = \hat{n}_i^j(k) - \hat{n}_i^{j-1}(k) + \Big(f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \Big) \\ - \Big(f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) \Big) \qquad (5.10)\end{aligned}$$

The following lemmas will be useful for analyzing the equations above.

Lemma 5.3.1. *The following two relations hold*

$$\hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) - \hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) = -\eta_1^j(k) W_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))$$

$$\hat{n}_i^j(k)\hat{v}_i^j(k) - \hat{n}_i^{j-1}(k)\hat{v}_i^{j-1}(k) = \eta_2^j(k) V_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))$$

where $0 \leq \eta_1^j(k), \eta_2^j(k) \leq 1$.

To analyze the error equation Eq. (5.10), we need to simplify the expressions for input flow difference and output flow difference across iterations. We analyze each term separately in the following lemmas.

Lemma 5.3.2. *For the imputation algorithm defined in the previous section, when effective demand parameter $c_{i-1}(k)$ is imputed (i.e. $I_{i-1}(k) = 1$),*

$$\begin{aligned} & f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\ & \quad = (1 - \acute{M}_i^j)(\hat{c}_{i-1}^j(k) - \hat{c}_{i-1}^{j-1}(k)) - \zeta_i^{j,1}(k) W_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k)) \end{aligned}$$

where $0 \leq \zeta_i^{j,1}(k) \leq 1$.

Lemma 5.3.3. *For the imputation algorithm defined in the previous section, when effective demand parameter $c_{i-1}(k)$ is not imputed,*

$$\begin{aligned}
&\exists 0 \leq \alpha_i^{j,1}(k), \zeta_i^{j,1}(k) \leq 1\, s.t \\
&\quad f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) = \\
&\qquad \alpha_i^{j,1}(k)(\hat{n}_{i-1}^j(k) - \hat{n}_{i-1}^{j-1}(k))V_i - \zeta_i^{j,1}(k)(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))W_i
\end{aligned}$$

Defining $\alpha_i^{j,1}(k) = 0$ when $I_{i-1}(k) = 1$, we can combine the results from the two lemmas above to get

$$\begin{aligned}
&f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) = h_i^1(k) + I_{i-1}(k)(1 - \acute{M}_i^j)(\hat{c}_{i-1}^j(k) - \hat{c}_{i-1}^{j-1}(k)) \\
&\text{where,} \quad h_i^1(k) = -\zeta_i^{j,1}(k)W_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k)) + \alpha_i^{j,1}(k)(\hat{n}_{i-1}^j(k) - \hat{n}_{i-1}^{j-1}(k))V_{i-1} \qquad (5.11)
\end{aligned}$$

Lemma 5.3.4. *For the imputation algorithm, when $I_i(k) = 1$,*

$$\begin{aligned}
&f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) \\
&\quad = \zeta_i^{j,2}(k)V_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k)) - \zeta_i^{j,3}(k)\frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)}\hat{w}_{i+1}(\hat{n}_{i+1}^j(k) - \hat{n}_{i+1}^{j-1}(k)) \\
&\quad + \acute{M}_{i+1}^j \hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k)))\hat{n}_i^j(k)\hat{v}_i^j(k)\left[\frac{1}{\hat{c}_i^j(k)} - \frac{1}{\hat{c}_i^{j-1}(k)}\right]
\end{aligned}$$

where $0 \leq \zeta_i^{j,2}(k), \zeta_i^{j,3}(k) \leq 1$.

Lemma 5.3.5. *For the imputation algorithm, when $I_i(k) = 0$,*

$$\begin{aligned}
&f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) \\
&\quad = \zeta_i^{j,2}(k)V_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k)) - \zeta_i^{j,3}(k)\frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)}w_{i+1}(\hat{n}_{i+1}^j(k) - \hat{n}_{i+1}^{j-1}(k)) \\
&\qquad - \alpha_i^{j,2}(k)\frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)}\left[(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))\right]
\end{aligned}$$

where $\hat{c}_i^j(k) = \hat{n}_i^j(k)\hat{v}_i^j(k)(1 - \beta_i(k)) + d_i(k)$ and $0 \leq \zeta_i^{j,2}(k), \zeta_i^{j,3}(k), \alpha_i^{j,2}(k) \leq 1$.

Again, we define $\alpha_i^{j,2}(k) = 0$ when $I_i(k) = 1$ and combine the lemmas above to get

$$\begin{aligned}
&f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) \\
&\qquad = h_i^2(k) + I_i(k)\acute{M}_{i+1}^j \hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k))\hat{n}_i^j(k)\hat{v}_i^j(k)\left[\frac{1}{\hat{c}_i^j(k)} - \frac{1}{\hat{c}_i^{j-1}(k)}\right] \\
&h_i^2(k) = \zeta_i^{j,2}(k)V_i(\hat{n}_i^j - \hat{n}_i^{j-1}) - \zeta_i^{j,3}(k)\frac{\hat{n}_i^j\hat{v}_i^j}{\hat{c}_i^{j-1}(k)}w_{i+1}(\hat{n}_{i+1}^j(k) - \hat{n}_{i+1}^{j-1}(k)) \\
&\qquad\qquad - \alpha_i^{j,2}(k)\frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)}\left[(\hat{n}_i^j - \hat{n}_i^{j-1})\right] \qquad (5.12)
\end{aligned}$$

We present the proofs to these lemmas in Section 5.6.

Lemma 5.3.6. *Consider the parameter updates corresponding to Link i in Eq.* (5.7). *There exists* $0 \leq \Gamma_1^{i,j}(k) \leq G_1$ *and* $0 \leq \Gamma_2^{i,j}(k) \leq G_2$ *such that*

$$\begin{aligned}
&if \quad \Lambda_1^{i,j} > 0 \\
&\quad \hat{c}_{i-1}^j(k) = \hat{c}_{i-1}^{j-1}(k) + \Gamma_1^{i,j}(k)\tilde{n}_i^j(k+1) \\
&if \quad \Lambda_2^{i,j} > 0 \\
&\quad \frac{1}{\hat{c}_i^j(k)} = \frac{1}{\hat{c}_{i-1}^{j-1}(k)} - \Gamma_2^{i,j}(k)\tilde{n}_i^j(k+1)
\end{aligned}$$

The derivation of this lemma is presented in Section 5.6. In the proof of this lemma, we show how the a-posteriori density error estimates are obtained only using the a-priori density error estimates. In the next theorem, we detail how the imputation algorithm has estimates with bounded errors during any iteration.

Theorem 5.3.1. *Given* $0 < n_i(k) < n_i^J \; \forall i \in 1, ..., N$*, the imputation algorithm ensures that the density errors are bounded. In particular,* $|\tilde{n}_i^j(k)| < n_i^J \forall i \in 1, ..., N$ *across all iterations.*

Proof. To prove this theorem we will first prove that $0 < \hat{n}_i^j(k) < n_i^J \; \forall \, i \in 1, ..., N$ holds during any iteration. For a given iteration j, we can prove the preceding claim using induction on k, the time index. Since $\hat{n}_i^j(0) = n_i(0) \, \forall i$, the claim holds for $k = 0$. Assume $0 < \hat{n}_i^j(k) < n_i^J$ for some k. We also note that $0 < c_{min} < \hat{c}_i^j < c_{max} < \infty$ using the update laws. Then for period $k+1$,

$$\begin{aligned}
&\hat{n}_i(k+1) = \hat{n}_i^j(k) + f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{out}(\hat{c}_i^j(k), \hat{n}_i^j(k), \hat{n}_{i+1}^j(k)) \\
&0 < f_i^{in,j}(k) < \hat{w}_i^j(n_i^J - \hat{n}_i^j(k)) \quad \text{and} \quad f_i^{out,j}(k) = \eta_1 \hat{n}_i^j(k)\hat{v}_i^j(k) \ge 0 \text{ where } 0 \le \eta_1 \le 1 \\
&\implies \hat{n}_i^j(k+1) \le \hat{n}_i(k) + f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) \le \hat{n}_i(k) + \hat{w}_i^j(n_i^J - \hat{n}_i^j(k)) \\
&\qquad\qquad\qquad < \hat{n}_i(k) + (n_i^J - \hat{n}_i^j(k)) = n_i^J \\
&\quad \text{and}, \hat{n}_i^j(k+1) \ge \hat{n}_i^j(k) - \eta_1 \hat{n}_i^j(k)\hat{v}_i^j(k) > 0
\end{aligned}$$

Hence $0 < \hat{n}_i^j(k) < n_i^J \, \forall i \in 1,\dots,N$ by induction. Since $0 < n_i(k) < n_i^J \, \forall i \in 1,\dots,N$, we see that $|\tilde{n}_i^j(k)| < n_i^J \, \forall i \in 1,\dots,N$ for any iteration. □

Substituting the results from Eq. (5.11) and Eq. (5.12) into Eq. (5.10), we see that

$$\begin{aligned}
\hat{n}_i^j(k+1) - \hat{n}_i^{j-1}(k+1) &= \hat{n}_i^j(k) - \hat{n}_i^{j-1}(k) + I_{i-1}(k)(1 - \acute{M}_i^j)(\hat{c}_{i-1}^j(k) - \hat{c}_{i-1}^{j-1}(k)) + h_i^1(k) \\
&+ h_i^2(k) - I_i(k)\acute{M}_{i+1}^j \hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k)))\hat{n}_i^j(k)\hat{v}_i^j(k)\left[\frac{1}{\hat{c}_i^j(k)} - \frac{1}{\hat{c}_i^{j-1}(k)}\right]
\end{aligned}$$

Collecting terms and noting that $\tilde{n}_i^j(k) - \tilde{n}_i^{j-1}(k) = \hat{n}_i^{j-1}(k) - \hat{n}_i^j(k)$, we get

$$\begin{aligned}
\tilde{n}_i^j(k+1) - \tilde{n}_i^{j-1}(k+1) &= (\tilde{n}_i^j(k) - \tilde{n}_i^{j-1}(k))\left(1 - \zeta_i^{j,1}(k)W_i - \zeta_i^{j,2}(k)V_i + \alpha_i^{j,2}(k)\frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)}\right) \\
&- \alpha_i^{j,1}(k)(\tilde{n}_{i-1}^j(k) - \tilde{n}_{i-1}^{j-1}(k))V_{i-1} + \zeta_i^{j,3}(k)\frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)} w_{i+1}(\tilde{n}_{i+1}^j(k) - \tilde{n}_{i+1}^{j-1}(k)) \\
&- I_{i-1}(k)(1 - \acute{M}_i^j)(\hat{c}_{i-1}^j(k) - \hat{c}_{i-1}^{j-1}(k)) \\
&+ I_i(k)\acute{M}_{i+1}^j \hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k)))\hat{n}_i^j(k)\hat{v}_i^j(k)\left[\frac{1}{\hat{c}_i^j(k)} - \frac{1}{\hat{c}_i^{j-1}(k)}\right] \qquad (5.13)
\end{aligned}$$

Finally, substituting Lemma 5.3.6 and re-arranging the terms,

$$\begin{aligned}
&\tilde{n}_i^j(k+1)(1 + I_{i-1}(k)\overline{\Gamma}_1^{i,j}(k) + I_i(k)\overline{\Gamma}_2^{i,j}(k)) - \tilde{n}_i^{j-1}(k+1) \\
&\quad = (\tilde{n}_i^j(k) - \tilde{n}_i^{j-1}(k))\left(1 - \zeta_i^{j,1}(k)W_i - \zeta_i^{j,2}(k)V_i + \alpha_i^{j,2}(k)\frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)}\right) \\
&\quad - \alpha_i^{j,1}(k)(\tilde{n}_{i-1}^j(k) - \tilde{n}_{i-1}^{j-1}(k))V_{i-1} + \zeta_i^{j,3}(k)\frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)} w_{i+1}(\tilde{n}_{i+1}^j(k) - \tilde{n}_{i+1}^{j-1}(k)) \qquad (5.14)
\end{aligned}$$

where

$$
\begin{aligned}
\bar{\Gamma}_1^{i,j}(k) &= (1-\acute{M}_i^j)\Gamma_1^{i,j}(k)\\
\bar{\Gamma}_2^{i,j}(k) &= \acute{M}_{i+1}^j\Gamma_2^{i,j}(k)\hat{w}_{i+1}^j(k)(n_{i+1}^J-\hat{n}_{i+1}^j(k))\hat{n}_i^j(k)\hat{v}_i^j(k)\\
\bar{\Gamma}^{i,j}(k) &= I_{i-1}(k)\bar{\Gamma}_1^{i,j}(k)+I_i(k)\bar{\Gamma}_2^{i,j}(k)
\end{aligned}
$$

Taking norm on both sides and using the triangular inequality we get,

$$
\begin{aligned}
&|\tilde{n}_i^j(k+1)(1+\bar{\Gamma}^{i,j}(k))-\tilde{n}_i^{j-1}(k+1)| \leq \varepsilon_i^j(k) \qquad (5.15)\\
&\varepsilon_i^j(k) = \left(1+\frac{F_i}{c_{min}}\right)|\tilde{n}_i^j(k)-\tilde{n}_i^{j-1}(k)|+\frac{F_i}{c_{min}}|\tilde{n}_{i+1}^j(k)-\tilde{n}_{i+1}^{j-1}(k)|+|\tilde{n}_{i-1}^j(k)-\tilde{n}_{i-1}^{j-1}(k)|\\
&\varepsilon_i^j(k) \geq 0\,\forall k
\end{aligned}
$$

From the equations above, we can also get

$$
\begin{aligned}
|\tilde{n}_i^j(k+1)| &\leq \frac{|\tilde{n}_i^{j-1}(k+1)|}{(1+\Gamma^{i,j}(k))}+\varepsilon_i^j(k)\\
\text{and}\,|\tilde{n}_i^j(k+1)| &\leq |\tilde{n}_i^{j-1}(k+1)|+\varepsilon_i^j(k) \qquad (5.16)
\end{aligned}
$$

Theorem 5.3.2. *For the imputation algorithm defined in* (5.8) *the error equations and the demand estimates are bounded and convergent.*

Proof. From Theorem 5.3.1 and the update equations, we can easily see that the error equations and demand estimates are bounded. We will prove the convergence of the error equations using induction on time index k and link index i. Clearly, $\tilde{n}_i^j(0) = 0\forall i, j$, and hence $\tilde{n}_i^j(0)$ converges along iteration axis. Suppose $\tilde{n}_i^j(p)$ converges $\forall p \leq k\, and\, \forall i$, we will prove that $\tilde{n}_i^j(k+1)$ converges.

Since $\tilde{n}_i^j(k)$ converges for all i, $\lim_{j\to\infty}\varepsilon_i^j(k) = 0$. From (5.16), we also get that $limsup_{j\to\infty}$ $|\tilde{n}_i^j(k+1)|-|\tilde{n}_i^{j-1}(k+1)| \leq 0$. In addition $|\tilde{n}_i^j(k+1)(1+\bar{\Gamma}^{i,j}(k))-\tilde{n}_i^{j-1}(k+1)| \to 0$ as $j\to\infty$, as seen from Eq. (5.15).

Given $\tilde{n}_i^0(k+1)$, we can see that eq. (5.14) generates bounded sequences a_j, g_j with $a_j = \tilde{n}_i^j(k+1)$ and $g_j = \bar{\Gamma}^{i,j}(k)$. Thus, there exists a convergent subsequence a_{j_p} such that $\lim_{p\to\infty}a_{j_p} = \bar{a}_1$, where j_p, $p \in \mathbb{N}$, $j_k < j_{k+1}\,\forall k \in \mathbb{N}$. Since the sequence is bounded, there exists another subsequence that converges to $\bar{a}_2 \neq \bar{a}_1$ if the sequence is not convergent. However, from (5.16) we get $limsup_{j\to\infty}|a_j| - |a_{j-1}| \leq 0$. This implies that $|\bar{a_2}| = |\bar{a_1}|$. In addition, we also have $|a_j(1+g_j) - a_{j-1}| \xrightarrow{j\to\infty} 0$, where $(1+g_j) \geq 1$ which implies that $\bar{a_2} = \bar{a_1} = \bar{a}$. This contradicts our assumption that $\bar{a_2} \neq \bar{a_1}$ and therefore the sequence a_j is convergent ($a_j \xrightarrow{j\to\infty} \bar{a}$). However $\bar{a}$ is not necessarily zero. If $\bar{a} \neq 0$, since $|a_j(1+g_j) - a_{j-1}| \xrightarrow{j\to\infty} 0$, we get that $g_j \xrightarrow{j\to\infty} 0$.

Hence $\tilde{n}_i^j(k+1)$ converges to a limit (which is not necessarily zero), whenever $\tilde{n}_i^j(k)$ converges. We also see that $c_i^j(k)$ converges, since either $\tilde{n}_i^j(k+1)$ or $\bar{\Gamma}^{i,j}(k)$ converges to $0 \quad \forall i$. Hence by induction, the above theorem is true. □

It is to be noted that no restrictions have been assumed with regards to the actual profile $n_i(k+1)$ (except $0 \leq n_i(k+1) \leq n_i^J$). In fact, it might happen that no feasible inputs exist for driving the system to follow the profile. In this case, the algorithm converges with non-zero profile tracking errors. However, one must also note that even if an input signal profile exists for tracking the given profile, the algorithm need not necessarily converge with zero errors. The following lemma provides some insight into this. Let us denote the converged estimates by $\hat{n}_i(k)$,$\hat{c}_i(k)$ (i.e. we drop the iteration index j).

Lemma 5.3.7. *Let c_i, $c_{min} < c_i(k) < c_{max}$ be the effective demand parameter that can exactly track the given density profile. Suppose at some time k, $\tilde{n}_p(k) = 0, p = i, i+1, i-1$ and $\tilde{n}_i(k+1) \neq 0$. Then the following statements are true :*

$$\begin{array}{ll} (a) & \text{Atleast} \quad I_{i-1} = 1 \quad \text{or} \quad I_i = 1 \\ (b) & I_{i-1} = 1 \text{ and } I_i = 1 \implies M_i(k) > 0 \text{ and } M_{i+1}(k) < 0 \\ (c) & I_{i-1} = 1 \text{ and } I_i = 0 \implies M_i(k) > 0 \text{ and } \tilde{n}_i(k+1) < 0 \\ (d) & I_{i-1} = 0 \text{ and } I_i = 1 \implies M_{i+1}(k) < 0 \text{ and } \tilde{n}_i(k+1) > 0 \end{array}$$

Also, in all the cases above, it is possible to modify the estimates and rerun the adaptation algorithm to ensure convergence. In addition, in case (c) and (d) the estimates can be modified to correspond to the "true" mode.

Proof. First we prove statement (a). Suppose $I_{i-1} = 0 \quad$ and $\quad I_i = 0$, we have $\hat{c}_p(k) = c_p(k), p = i, i-1$ since $\tilde{n}_p(k) = 0, p = i, i+1, i-1$. Therefore, we have

$$\begin{aligned} & g(\hat{n}_i(k), \hat{c}_{i-1}(k), \hat{c}_i(k), \hat{n}_{i+1}(k)) = g(n_i(k), c_{i-1}(k), c_i(k), n_{i+1}(k)) \\ \implies & \tilde{n}_i(k+1) = 0 \end{aligned}$$

which contradicts our assumption. Therefore $I_{i-1} = 1$ or $I_i = 1$.

To prove statement (b), we analyze all the four possible cases.

Case (i) $M_i(k) < 0$ and $M_{i+1}(k) > 0$

This case corresponds to the FC mode. In this mode both $\hat{c}_{i-1}(k)$ and $\hat{c}_i(k)$ affect the density update. Clearly, if $\tilde{n}_i(k+1) \neq 0$, then atleast one of $\Gamma_1^{i,j}, \Gamma_2^{i,j}$ is non-zero. At steady state, the algorithm will not converge in this mode with non-zero errors.

Case (ii) $M_i(k) < 0$ and $M_{i+1}(k) \leq 0$

This case corresponds to the FF mode, and $\hat{c}_{i-1}(k) < w(n_i^J - \hat{n}_i(k))$. If $\tilde{n}_i(k+1) > 0$, clearly $\Gamma_1^{i,j} \neq 0$ which violates steady state assumptions. If $\tilde{n}_i(k+1) < 0$ and $\Gamma_1^{i,j} = 0$, then $\hat{c}_{i-1}(k) = c_{min}$. But this is not possible since $\exists c_{i-1}(k)$ s.t. $c_{min} < c_{i-1}(k) < c_{max}$ which can track the density profile, and $n_i(k+1) \geq n_i(k) + c_{min} - n_i(k)\bar{v_i}(k) = \hat{n}_i(k+1)$, which conflicts with our assumption that $\tilde{n}_i(k+1) < 0$.

Case (iii) $M_i(k) \geq 0$ and $M_{i+1}(k) > 0$

This case is not possible by an argument similar to Case (ii).

Case (iv) $M_i(k) > 0$ and $M_{i+1}(k) < 0$

Clearly, this case corresponds to the CF mode, where both $\hat{c}_i(k)$ and $\hat{c}_{i-1}(k)$ do not affect the state equations. Hence non-zero errors can exist. If $\tilde{n}_i(k+1) > 0$, we can set $\hat{c}_i(k) > \hat{w}_{i+1}(k)(n_{i+1}^J - n_{i+1}(k))$ (this is the only possible way to increase the density estimate) and restart the adaptation algorithm. In fact, in this case the actual mode is either the FC or the FF mode, but we modify the estimate to the FF mode. If $\tilde{n}_i(k+1) < 0$, we set $\hat{c}_{i-1}(k) < \hat{w}_i(k)(n_i^J - n_i(k))$ before restarting. In this case, the actual mode is either the CC or the FC mode, but we perturb the estimate to the CC mode.

For statement (c), we can see that $\hat{c}_i(k) = c_i(k)$. Hence, for the system to converge in this case with nonzero errors, $M_i(k) > 0$ which implies that $\tilde{n}_i(k+1) < 0$. This can be shown by considering individual cases like the proof for statement (b). In this case we set $\hat{c}_{i-1}(k) < \hat{w}_i(k)(n_i^J - n_i(k))$. Statement (d) can be proven similarly. For this case, we reset the estimate $\hat{c}_i(k)$ such that it satisfies $\hat{c}_i(k) > \hat{w}_{i+1}(k)(n_{i+1}^J - n_{i+1}(k))$. In both these, the reset switches the link to the correct mode.

In all the cases discussed above, once the parameter resets are executed, we restart the iterations (corresponding to the imputation algorithm). Ff we execute these modifications in an orderly fashion (starting from the earliest time instant k, and executing from the most downstream link to the first upstream link for each iteration and allowing the estimates to converge before executing the next trigger), we can ensure that the estimates exactly track the measured profiles. This can be seen as the trigger algorithm never changes the node estimate to the wrong mode, and each trigger instance will at-least correct the mode in one of the nodes. Also, the subsequent iterations will never switch the perturbed nodes back to the wrong mode. Hence the algorithm will converge after at most $N * K$ triggers, where N is the number of links and K is the total number of time steps in a single simulation run. □

In the lemma above, we outlined the cases when the algorithm can converge with non-zero density errors during the parameter updates. We also specified a provably convergent algorithm to modify the estimates to ensure exact density tracking. In practice, the procedure is slow, since we need to execute triggers in sequence both in time and space. The measurements also tend to be noisy, and the assumptions of the algorithm are violated. The heuristic algorithm presented in the previous section extends on the ideas presented here, with the following differences (i) We do not enforce $\tilde{n}_p(k) = 0, p = i, i+1, i-1.$ before executing the trigger (ii) All triggers are simultaneously executed (iii)There is a tolerance parameter, to account for errors. The heuristic method has worked well in practice, and it usually leads to sufficient convergence within 5 instances of trigger updates.

Assuming noise-free measurements, suppose that we converge with zero density errors, the total demand vector need not converge to its actual values. For the total demand vectors to converge to the actual values, we first require that the mode of the link converges to the actual value. Table 5.1 lists the possible mode errors that occur even under conditions of zero density errors, when the effective demands corresponding to the input and the output node is imputed. The table is obtained by analyzing the reachable sets in each mode.

Actual Mode	Mode Estimate	Comments
CC	FF	Not possible, since this would lead to non-zero errors.
FF	CC	Not possible, since this would lead to non-zero errors.
CF	FF/CC	Not possible, since the parameter updates will switch the mode.
FF/CC	CF	Not possible, since this would lead to non-zero errors.
FF/CC/CF	FC	Possible.
FC	FF/CC/CF	Possible.

Table 5.1: Converged mode/ true mode misclassification.

If the converged mode of a particular link is incorrect, either the previous/next link also contains an incorrect mode. To understand in detail, we need to consider the combined mode of the entire freeway. At any time instant, the freeway can be divided into sections, with each section being in congestion or free-flow. Thus the true mode can be written as F..FC..CF..FC..CF..F (where F...F represents consecutive nodes with the free-flow mode), with the boundaries in free-flow . The modes in the imputation algorithm estimate can also be represented in a similar fashion. Assume that each congestion section spans multiple (> 1) nodes. From the rules explained in Table 5.1, we can see that convergence to the wrong mode might occur only at the tail of the congestion section, where the FC mode is prevalent. Moreover, the location of the incorrect “mode” convergence can be determined to be around the location where the estimate converges to the FC mode. In this case, the mode estimates disagree with the true modes in at-most two links. The first link is the link at which the mode estimate is FC. The other link can either be the one before/after the first link. Combining the rules in the table, we can see that only one of the two scenarios is possible

(1) Actual mode : *FFC*, Mode Estimate: *FCC* or (2) Actual mode : *FCC*, Mode Estimate : *FFC*.

In links where at least one of the affecting ramps measurements are available (i.e. one of the effective demand estimates is available), the mode estimate converges to the correct mode. A special case of this is when there are two adjacent detectors in each freeway stretch between ramps. In this case, the demand and split ratios are trivially known (since there are no ramps in between), and they help in the parameter convergence. Clearly, since incorrect modes occur in pairs of links, this means that incorrect convergence requires at least three contiguous effective demands to be imputed. Even at locations where there are three contiguous effective demands that are imputed which can lead to convergence in the wrong mode as indicated in the table, it must be noted that the density value might not be reachable using the dynamic equations of the wrong mode. As an extreme example, at low densities, or during heavy congestion, the imputation procedure converges in the correct mode.

Thus, for the imputation algorithm, the total demand vector need not converge to its true value as : (a) the adaptive learning procedure does not ensure exact density profile matching due to incorrect convergence in the CF mode (the application of the trigger algorithm helps avoid this) (b) Even in case the density profiles match, the mode estimate might be different from the actual congestion mode (c) It is not possible to uniquely determine the total demand vector in the FC mode, due to lack of observability. The FC mode is present at the upstream of the congestion region, and it is usually transient as the congestion tail passes through the section.

Theorem 5.3.3. *Assume that the measurements are noise-free. If the total demand vector converges to its actual value, then the solution of the linear program will correspond to zero errors between model calculated flows and measured flows. Moreover, the ramp flow estimates will correspond to the actual measurements.*

Proof. The total demand vector is combination of the on-ramp demand and the off-ramp flow. We can see that if any one of the objective functions given in Eq. (5.9) is chosen, we are provided with measurements that can uniquely separate the demand vector into the components. Moreover, the objective function is only minimized when all the flow (along the freeways and the on-ramps) estimates agree with the measurements. □

5.4 Examples

We present two examples of application of the imputation algorithm. Both these examples are based on a 23 mile section (with 32 on-ramps and 26 off-ramps) of the I-210W freeway in Pasadena, California. The geometry of the freeway gives rise to some constraints on the estimation procedure.

In particular, not all nodes have both an on-ramp and an off-ramp. This corresponds to additional constraints (bounds) on $\hat{c}_i^j(k)$ (e.g. $\hat{c}_i^j(k) \leq \hat{n}_i^j(k)\hat{v}_i^j(k)$ for sections without on-ramps). We run the actual imputation algorithm without these constraints for 10 iterations, to allow sufficient convergence, and then apply these bounds during the update equations.

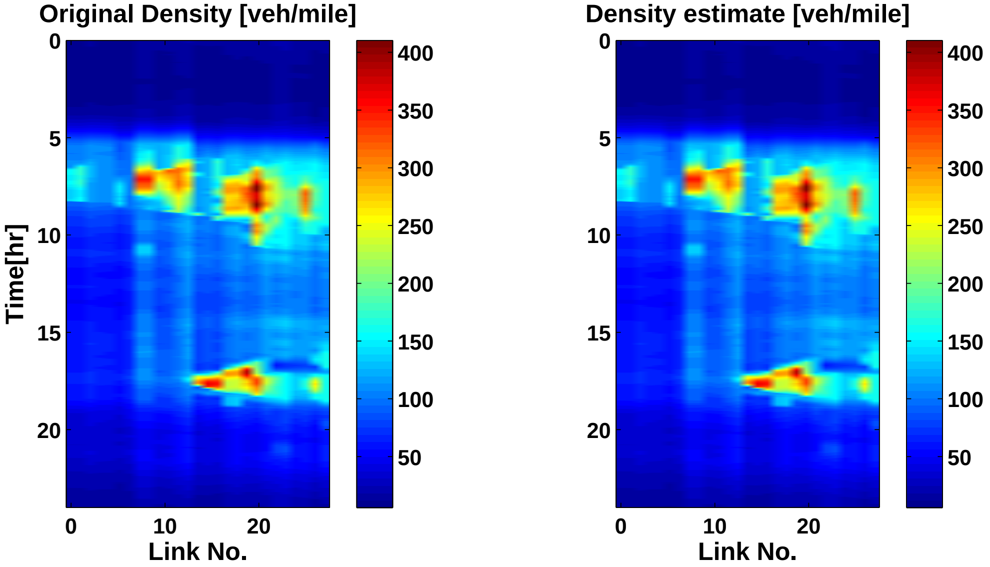

Figure 5.2: Final density contours obtained after imputation.

The first example corresponds to the application of the imputation algorithm on a simulated scenario. In this case, we know the exact on-ramp flows and off-ramp split ratios, which we use to generate density and flow profiles using the LN-CTM model. After this, we assume that some of the ramps (4 on-ramps and 11 off-ramps) need to be imputed, and estimate these using the imputation algorithm. This example will demonstrate the ideal performance of the algorithm, including the convergence of the estimates. Figure 5.2 shows the original simulated density (left) and the converged density estimate of the imputation algorithm. It can be seen that the density estimates have converged to their true values. This is clearly seen in Figure 5.3, which shows the decrease in error across algorithm iterations $\left(error = 100 \times \frac{\sum_{i,k} |\tilde{n}_i^j(k)|}{\sum_{i,k} n_i(k)} \right)$. We execute the trigger algorithm after iterations $5, 9, 12, 15$. If no triggers are executed, the error converges to 0.04%, while after the resets, the error decreases to 0.003%.

Figure 5.4 plots the contour map of the difference between the original effective demand parameters and the estimated effective demand parameter. Figure 5.5 shows the location of the FC mode. These figures demonstrate the theoretical analysis conducted in the previous section. We see that in most cases, the effective demand parameters converge to the actual mode and the lack of convergence occurs near the location of the FC mode. Moreover, it does not occur during all the instances where the FC mode is active, since there are some density ranges where parameters

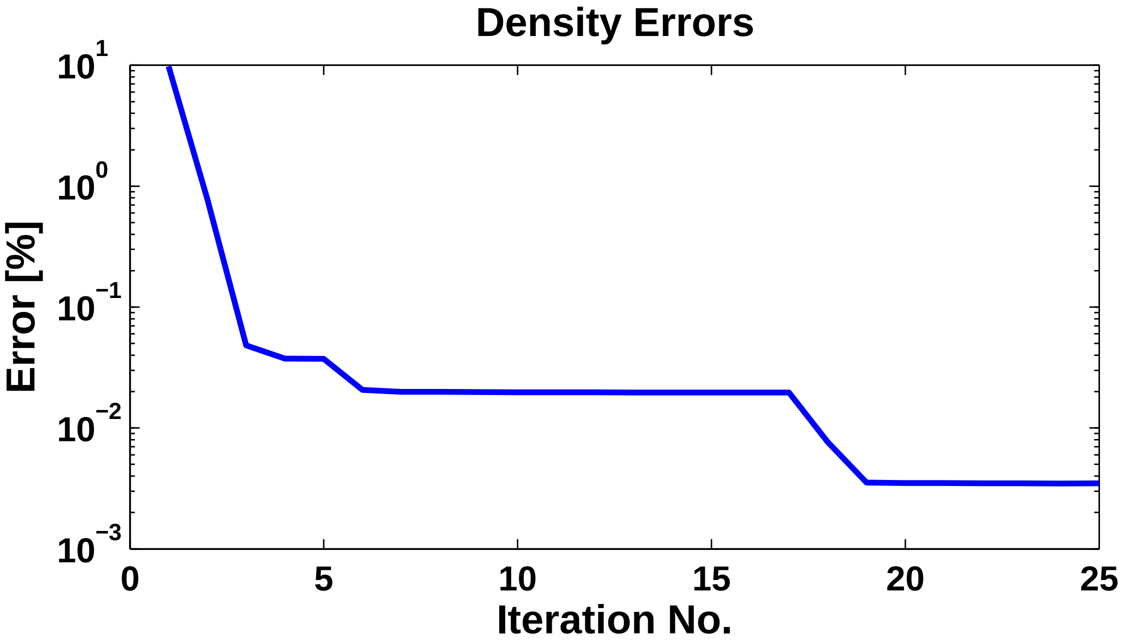

Figure 5.3: Density errors across iterations.

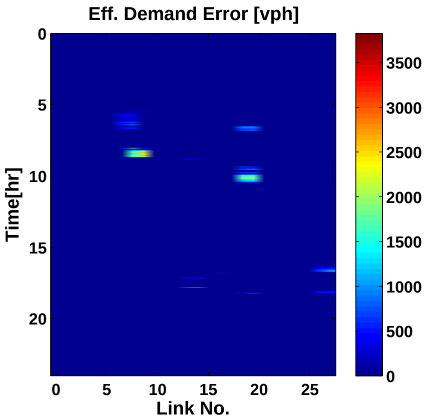

Figure 5.4: Effective demand errors.

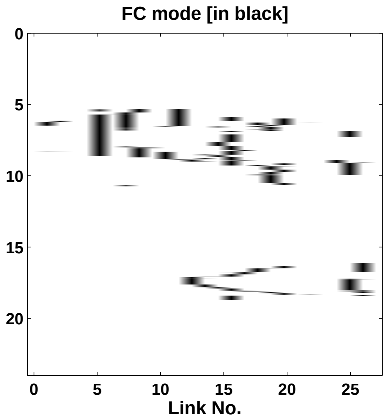

Figure 5.5: Location of the FC mode.

cannot converge in the wrong modes. Moreover, the presence of some measurements also helps in the exact convergence of the effective demand parameters.

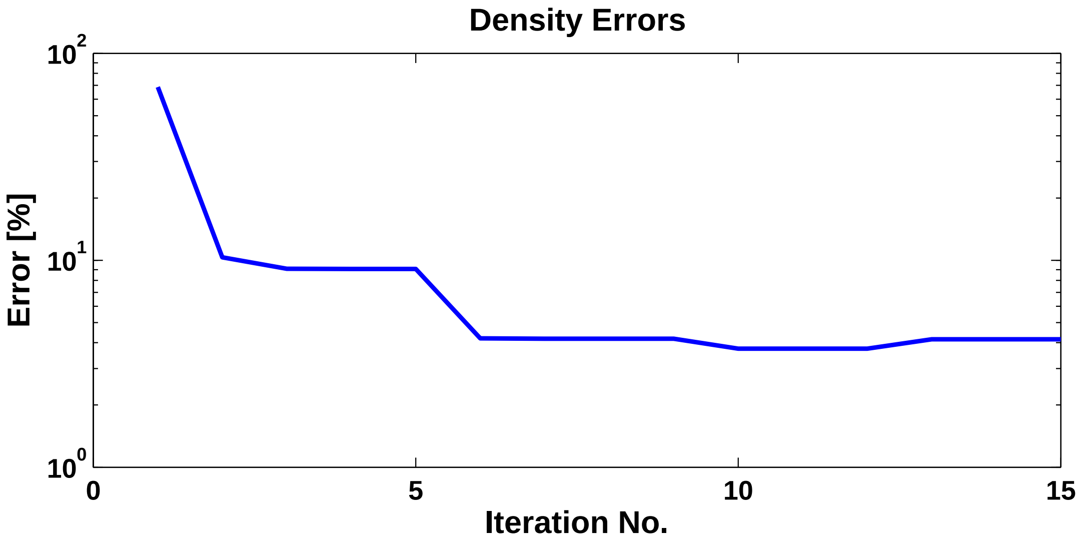

Figure 5.6: Density error across iterations.

In the second example, measured data from loop detectors obtained from PeMS was used. 5 on-ramps and 12 off-ramps needed to be imputed and the sections with both on-ramp/off-ramp measurements were not imputed. Figure 5.6 shows the decrease in errors across iterations. In this case, the final error with trigger resets is 4.1% as compared to 9.1% final error obtained without any heuristic resets. We also notice that after the trigger is executed in Iteration 12, there is an increase in the error. In this case, we use the estimates corresponding to the best errors (i.e. estimates from iteration 12, before the trigger algorithm is executed). Figure 5.7 and Figure 5.8 presents a comparison of simulation results (obtained using imputed ramp flows/split ratios) with the loop detector measurements. Heavy congestion regions (density greater than 300 veh/mile) are also well captured in the simulation. The final density and flow errors for this simulation were 4.2% and 9.37 % respectively.

5.5 Summary

In this chapter, we presented a model based imputation procedure to estimate the on-ramp flows and off-ramp split ratios in a freeway section. The problem is solved in two steps, with the first step employing an adaptive iterative learning procedure for estimation of the total demand vector from the density measurements across the freeway. We presented a detailed convergence analysis for this algorithm, and we derived an exact as well as a heuristic trigger algorithm to ensure good convergence properties. We presented situations where the estimated effective demand parameter

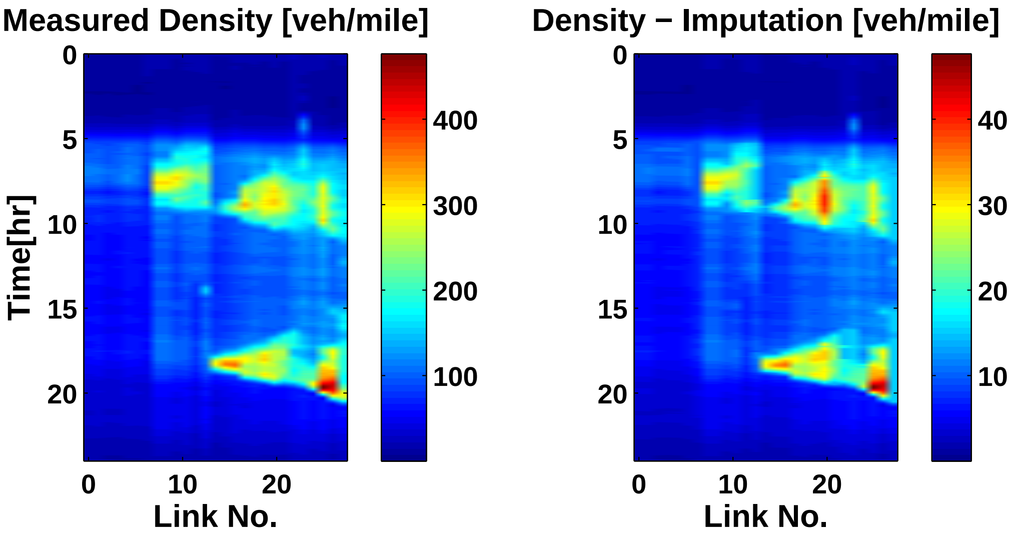

Figure 5.7: Final density contours obtained after imputation.

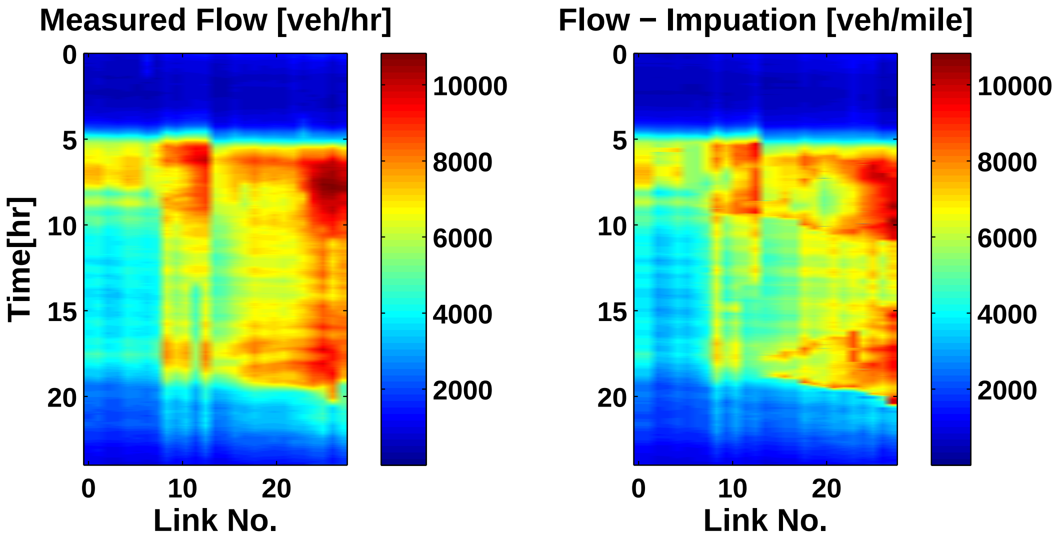

Figure 5.8: Flow contours comparison.

correctly tracks the actual effective demand parameter. This is also illustrated using a synthetic example in the previous section. Once the effective demand parameters were obtained, we described a linear program to decouple the on-ramp flows and off-ramp splits. We showed that by the appropriate design of the objective cost function, we can ensure exact identification of the unknown ramp flows if the effective demand parameters have converged to the true values.

The imputation algorithm developed here is computationally fast. For example, for the models shown in this dissertation, which involved imputing 24-hour ramp flow profiles for a freeway with up to 30 links, the algorithm, programmed in MATLAB was executed within 5 minutes. We have been able to explicitly solve the linear program, and derive closed form solutions for on-ramp flows and off-ramp splits. It is easy to derive these on paper, but a user can also use multi-parametric programs[3] to arrive at a solution. The use of these explicit solutions is necessary to achieve fast computations. Before using these explicit solutions, we encountered computation times which ran up to 0.5 hour for these cases.

In our experience, the imputation algorithm presented in this chapter is more ‘robust’ in comparison to the algorithm based on the ACTM. The imputation algorithm based on the ACTM is executed section-wise. We have noticed that the algorithm converges with non-zero errors, since measurement errors exist. Faulty detectors and faulty measurements also lead to significant errors (after convergence) in some sections. Since we use actual measurements as boundary conditions in the sectionwise imputation procedure, once the estimates obtained from each section is combined and fed into the final model of the complete freeway, the simulations might have large errors, especially if there was an interior section with faulty measurements. Even nominally encountered measurement errors might be amplified in the final simulation. We expect that the final results deteriorate as number of links in the freeway increase. In comparison, the LN-CTM imputation algorithm imputes all ramps simultaneously. Also, any iteration of the imputation algorithm is actually a simulation of the entire freeway. This ensures that the errors of the imputation procedure correspond to the actual errors of the final simulation. This is a very useful feature, particularly when the imputation algorithm is used as a part of an automated calibration/model building routine. As we have stressed before, another advantage of the LN-CTM algorithm is that it is based on a more reliable model of the the freeway traffic dynamics. The only drawback (as compared to the ACTM imputation algorithm) is that there might be some locations/time periods when the ramp flow estimates might not converge to the true value, even when measurement errors are non-existent.

5.6 Proofs

Proof of Lemma 5.3.1

Proof. The function $\min(F_i, W_i(n_i^J - \hat{n}_i^j(k)))$ is concave and non-increasing. We use it to prove the following.

$$\begin{aligned}
&\hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) - \hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) \\
&\qquad\qquad = \min(F_i, W_i(n_i^J - \hat{n}_i^j(k))) - \min(F_i, W_i(n_i^J - \hat{n}_i^{j-1} j(k))) \\
&\text{Without loss of generality, assuming,} W_i(n_i^J - \hat{n}_i^j(k)) \geq W_i(n_i^J - \hat{n}_i^{j-1}(k)) \\
&\implies 0 \leq \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) - \hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) \leq W_i(n_i^J - \hat{n}_i^j(k)) - W_i(n_i^J - \hat{n}_i^{j-1}(k)) \\
&\implies \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) - \hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) = -\eta_1^j(k) W_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k)) \\
&\text{Similarly,} \hat{n}_i^j(k)\hat{v}_i^j(k) - \hat{n}_i^{j-1}(k)\hat{v}_i^{j-1}(k) = \eta_2^j(k) V_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k)).
\end{aligned}$$

□

Proof of Lemma 5.3.2

Proof. We consider four cases
Case (a) $\acute{M}_i^j(k) = 1$ and $\acute{M}_i^{j-1}(k) = 1$

$$\begin{aligned}
f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) =& \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) - \hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) \\
=& -\zeta_i^{j,1}(k) W_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))
\end{aligned}$$

Case (b) $\acute{M}_i^j(k) = 1$ and $\acute{M}_i^{j-1}(k) = 0$

$$\begin{aligned}
&\hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) \geq \hat{c}_{i-1}^{j-1}(k) \geq \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) \\
&f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) = \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) - \hat{c}_{i-1}^{j-1}(k) \\
&\qquad\qquad = -\zeta_i^{j,1}(k) W_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))
\end{aligned}$$

Case (c) $\acute{M}_i^j(k) = 0$ and $\acute{M}_i^{j-1}(k) = 0$

$$f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) = \hat{c}_{i-1}^j(k) - \hat{c}_{i-1}^{j-1}(k)$$

Case (d) $\quad \acute{M}_i^j(k) = 0$ and $\acute{M}_i^{j-1}(k) = 1$

$$\begin{aligned}
&\hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) \leq \hat{c}_{i-1}^{j-1}(k) \leq \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) \\
&f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\qquad = \hat{c}_{i-1}^j(k) + \left(\hat{c}_{i-1}^{j-1}(k) - \hat{c}_{i-1}^{j-1}(k)\right) - \hat{w}_i^j(k)(n_i^J - \hat{n}_i^{j-1}(k)) \\
&\qquad = \hat{c}_{i-1}^j(k) - \hat{c}_{i-1}^{j-1}(k) - \zeta_i^{j,1}(k)W(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))
\end{aligned}$$

These cases generalize to the expression given in the lemma. □

Proof of Lemma 5.3.3

Proof. We consider four cases
Case (a) $\quad \acute{M}_i^j(k) = 1$ and $\acute{M}_i^{j-1}(k) = 1$

$$\begin{aligned}
f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) &= \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) - \hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) \\
&= -\zeta_i^{j,1}(k)W_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))
\end{aligned}$$

Case (b) $\quad \acute{M}_i^j(k) = 0$ and $\acute{M}_i^{j-1}(k) = 0$

$$\begin{aligned}
&f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\qquad = (\hat{n}_{i-1}^j(k)\hat{v}_{i-1}^j(k)(1 - \beta_{i-1}(k)) + d_{i-1}(k)) - (\hat{n}_{i-1}^{j-1}(k)\hat{v}_{i-1}^{j-1}(k)(1 - \beta_{i-1}(k)) + d_{i-1}(k)) \\
&\qquad = \alpha_i^{j,1}(k)(\hat{n}_{i-1}^j(k) - \hat{n}_{i-1}^{j-1}(k))V_i\,, \text{with } \alpha_i^{j,1}(k) \in [0,1]
\end{aligned}$$

Case (c) $\quad \acute{M}_i^j(k) = 1$ and $\acute{M}_i^{j-1}(k) = 0$

$$\begin{aligned}
&\hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) \geq \hat{n}_{i-1}^{j-1}(k)\hat{v}_{i-1}^{j-1}(k)(1 - \beta_{i-1}(k)) + d_{i-1}(k) \\
&\qquad \textit{and} \quad \hat{n}_{i-1}^j(k)\hat{v}_{i-1}^j(k)(1 - \beta_{i-1}(k)) + d_{i-1}(k) \geq \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) \\
&f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\qquad = \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) - (\hat{n}_{i-1}^{j-1}(k)\hat{v}_{i-1}^{j-1}(k)(1 - \beta_{i-1}(k)) + d_{i-1}(k)) \\
&\implies f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\qquad \leq (\hat{n}_{i-1}^j(k)\hat{v}_{i-1}^j(k)(1 - \beta_{i-1}(k)) + d_{i-1}(k)) - (\hat{n}_{i-1}^{j-1}(k)\hat{v}_{i-1}^{j-1}(k)(1 - \beta_{i-1}(k)) + d_{i-1}(k)) \\
&\text{and} \quad f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\qquad \geq (\hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k))) - (\hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k))) \\
&\implies f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\qquad = \alpha_i^{j,1}(k)(\hat{n}_{i-1}^j(k) - \hat{n}_{i-1}^{j-1}(k))V_i - \zeta_i^{j,1}(k)(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))W_i
\end{aligned}$$

The last expression can be obtained by noting that the input flow difference has the same sign as either its upper bound or its lower bound. Depending upon the sign, the flow difference can be written as a scaled version of the expressions given in the above two cases.
Case (d) $\acute{M}_i^j(k) = 0$ and $\acute{M}_i^{j-1}(k) = 1$

$$\begin{aligned}
&\hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k)) \leq \hat{n}_{i-1}^{j-1}(k)\hat{v}_{i-1}^{j-1}(k)(1-\beta_{i-1}(k)) + d_{i-1}(k) \\
&\quad \textit{and} \quad \hat{n}_{i-1}^{j}(k)\hat{v}_{i-1}^{j}(k)(1-\beta_{i-1}(k)) + d_{i-1}(k) \leq \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) \\
&f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\quad = \hat{n}_{i-1}^{j}(k)\hat{v}_{i-1}^{j}(k)(1-\beta_{i-1}(k)) + d_{i-1}(k) - \hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k)) \\
\Longrightarrow\ &f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\quad \geq (\hat{n}_{i-1}^{j}(k)\hat{v}_{i-1}^{j}(k)(1-\beta_{i-1}(k)) + d_{i-1}(k)) - (\hat{n}_{i-1}^{j-1}(k)\hat{v}_{i-1}^{j-1}(k)(1-\beta_{i-1}(k)) + d_{i-1}(k)) \\
&\text{and} \quad f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\quad \leq (\hat{w}_i^j(k)(n_i^J - \hat{n}_i^j(k))) - (\hat{w}_i^{j-1}(k)(n_i^J - \hat{n}_i^{j-1}(k))) \\
\Longrightarrow\ &f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{in}(\hat{c}_{i-1}^{j-1}(k), \hat{n}_i^{j-1}(k)) \\
&\qquad = \alpha_i^{j,1}(k)(\hat{n}_{i-1}^j(k) - \hat{n}_{i-1}^{j-1}(k))V_i - \zeta_i^{j,1}(k)(\hat{n}_{i-1}^j(k) - \hat{n}_{i-1}^{j-1}(k))W_i
\end{aligned}$$

These cases generalize to the expression given in the lemma. □

Proof of Lemma 5.3.4

Proof. We consider four cases. In all the cases we assume $0 \leq \zeta_i^{j,2}(k), \zeta_i^{j,3}(k) \leq 1$
Case (a) $\acute{M}_{i+1}^j(k) = 1$ and $\acute{M}_{i+1}^{j-1}(k) = 1$

$$\begin{aligned}
&f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) \\
&\quad = \frac{\hat{w}_{i+1}^j(n_{i+1}^J - \hat{n}_{i+1}^j(k))}{\hat{c}_i^j(k)}\hat{n}_i^j(k)\hat{v}_i^j(k) - \frac{\hat{w}_{i+1}^{j-1}(n_{i+1}^J - \hat{n}_{i+1}^{j-1}(k))}{\hat{c}_i^{j-1}(k)}\hat{n}_i^{j-1}(k)\hat{v}_i^{j-1}(k) \\
&\quad = \Big[\frac{\hat{w}_{i+1}^j(n_{i+1}^J - \hat{n}_{i+1}^j(k))}{\hat{c}_i^j(k)}\hat{n}_i^j(k)\hat{v}_i^j(k) - \frac{\hat{w}_{i+1}^j(n_{i+1}^J - \hat{n}_{i+1}^j(k))}{\hat{c}_i^{j-1}(k)}\hat{n}_i^j(k)\hat{v}_i^j(k)\Big]
\end{aligned}$$

$$
\begin{aligned}
&+\Big[\frac{\hat{w}^{j}_{i+1}(n^{J}_{i+1}-\hat{n}^{j}_{i+1}(k))}{\hat{c}^{j-1}_{i}(k)}\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)-\frac{\hat{w}^{j-1}_{i+1}(n^{J}_{i+1}-\hat{n}^{j-1}_{i+1}(k))}{\hat{c}^{j-1}_{i}(k)}\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)\Big]\\
&+\Big[\frac{\hat{w}^{j-1}_{i+1}(n^{J}_{i+1}-\hat{n}^{j-1}_{i+1}(k))}{\hat{c}^{j-1}_{i}(k)}\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)-\frac{\hat{w}^{j-1}_{i+1}(n^{J}_{i+1}-\hat{n}^{j-1}_{i+1}(k))}{\hat{c}^{j-1}_{i}(k)}\hat{n}^{j-1}_{i}(k)\hat{v}^{j-1}_{i}(k)\Big]\\
&=\acute{M}^{j}_{i+1}\hat{w}^{j}_{i+1}(k)(n^{J}_{i+1}-\hat{n}^{j}_{i+1}(k)))\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)\left[\frac{1}{\hat{c}^{j}_{i}(k)}-\frac{1}{\hat{c}^{j-1}_{i}(k)}\right]\\
&\quad-\zeta^{j,3}_{i}(k)\frac{\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)}{\hat{c}^{j-1}_{i}(k)}w_{i+1}(\hat{n}^{j}_{i+1}(k)-\hat{n}^{j-1}_{i+1}(k))+\zeta^{j,2}_{i}(k)V_{i}(\hat{n}^{j}_{i}-\hat{n}^{j-1}_{i})
\end{aligned}
$$

Since, $\quad 0\leq\hat{w}^{j}_{i+1}(n^{J}_{i+1}-n^{j-1}_{i+1}(k))\leq\hat{c}^{j-1}_{i}(k)$

Case (b) $\quad \acute{M}^{j}_{i+1}(k)=0$ and $\acute{M}^{j-1}_{i+1}(k)=0$

$$
\begin{aligned}
f^{out}_{i}(\hat{c}^{j}_{i}(k),\hat{n}^{j}_{i+1}(k))-f^{out}_{i}(\hat{c}^{j-1}_{i}(k),\hat{n}^{j-1}_{i+1}(k))&=\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)-\hat{n}^{j-1}_{i}(k)\hat{v}^{j-1}_{i}(k)\\
&=\zeta^{j,2}_{i}(k)V_{i}(\hat{n}^{j}_{i}(k)-\hat{n}^{j-1}_{i}(k))
\end{aligned}
$$

Case (c) $\quad \acute{M}^{j}_{i+1}(k)=1$ and $\acute{M}^{j-1}_{i+1}(k)=0$

$$
\begin{aligned}
&f^{out}_{i}(\hat{c}^{j}_{i}(k),\hat{n}^{j}_{i+1}(k))-f^{out}_{i}(\hat{c}^{j-1}_{i}(k),\hat{n}^{j-1}_{i+1}(k))\\
&\quad=\frac{\hat{w}^{j}_{i+1}(n^{J}_{i+1}-\hat{n}^{j}_{i+1}(k))}{\hat{c}^{j}_{i}(k)}\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)-\hat{n}^{j-1}_{i}(k)\hat{v}^{j-1}_{i}(k)\\
&\quad=\Big[\frac{\hat{w}^{j}_{i+1}(n^{J}_{i+1}-\hat{n}^{j}_{i+1}(k))}{\hat{c}^{j}_{i}(k)}\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)-\frac{\hat{w}^{j}_{i+1}(n^{J}_{i+1}-\hat{n}^{j}_{i+1}(k))}{\hat{c}^{j-1}_{i}(k)}\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)\Big]\\
&\quad+\Big[\frac{\hat{w}^{j}_{i+1}(n^{J}_{i+1}-\hat{n}^{j}_{i+1}(k))}{\hat{c}^{j-1}_{i}(k)}\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)-\hat{n}^{j-1}_{i}(k)\hat{v}^{j-1}_{i}(k)\Big]\\
&\quad=\acute{M}^{j}_{i+1}\hat{w}^{j}_{i+1}(k)(n^{J}_{i+1}-\hat{n}^{j}_{i+1}(k)))\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)\left[\frac{1}{\hat{c}^{j}_{i}(k)}-\frac{1}{\hat{c}^{j-1}_{i}(k)}\right]\\
&\quad-\zeta^{j,3}_{i}(k)\frac{\hat{n}^{j}_{i}(k)\hat{v}^{j}_{i}(k)}{\hat{c}^{j-1}_{i}(k)}w_{i+1}(\hat{n}^{j}_{i+1}(k)-\hat{n}^{j-1}_{i+1}(k))
\end{aligned}
$$

$$\begin{aligned}
&\text{Since,} \quad \hat{w}^j_{i+1}(n^J_{i+1} - \hat{n}^j_{i+1}(k)) \le \hat{c}^{j-1}_i(k) \le \hat{w}^{j-1}_{i+1}(n^J_{i+1} - n^{j-1}_{i+1}(k)) \\
&\Longrightarrow \quad \hat{w}^j_{i+1}(n^J_{i+1} - \hat{n}^j_{i+1}(k)) - \hat{c}^{j-1}_i(k) \\
&\qquad = \bar{\zeta}^j_{i+1}(k)(\hat{w}^j_{i+1}(n^J_{i+1} - \hat{n}^j_{i+1}(k)) - \hat{w}^{j-1}_{i+1}(n^J_{i+1} - \hat{n}^{j-1}_{i+1}(k))) \\
&\qquad = -\zeta^{j,3}_i(k) w_{i+1}(\hat{n}^j_{i+1}(k) - \hat{n}^{j-1}_{i+1}(k)) \\
&\textit{where} \quad 0 \le \bar{\zeta}^j_{i+1}(k) \le 1
\end{aligned}$$

Case (d) $\acute{M}^j_{i+1}(k) = 0$ and $\acute{M}^{j-1}_{i+1}(k) = 1$

$$\begin{aligned}
& f^{out}_i(\hat{c}^j_i(k), \hat{n}^j_{i+1}(k)) - f^{out}_i(\hat{c}^{j-1}_i(k), \hat{n}^{j-1}_{i+1}(k)) \\
&\quad = \hat{n}^j_i(k)\hat{v}^j_i(k) - \frac{\hat{w}^{j-1}_{i+1}(n^J_{i+1} - \hat{n}^{j-1}_{i+1}(k))}{\hat{c}^{j-1}_i(k)} \hat{n}^{j-1}_i(k)\hat{v}^{j-1}_i(k) \\
&\quad = \left[\hat{n}^j_i(k)\hat{v}^j_i(k) - \hat{n}^{j-1}_i(k)\hat{v}^{j-1}_i(k)\right] \\
&\qquad + \left[\hat{n}^{j-1}_i(k)\hat{v}^{j-1}_i(k) - \frac{\hat{w}^{j-1}_{i+1}(n^J_{i+1} - \hat{n}^{j-1}_{i+1}(k))}{\hat{c}^{j-1}_i(k)} \hat{n}^{j-1}_i(k)\hat{v}^{j-1}_i(k)\right] \\
&\quad = \zeta^{j,2}_i(k) V_i(\hat{n}^j_i - \hat{n}^{j-1}_i) - \zeta^{j,3}_i(k) \frac{\hat{n}^j_i(k)\hat{v}^j_i(k)}{\hat{c}^{j-1}_i(k)} w_{i+1}(\hat{n}^j_{i+1}(k) - \hat{n}^{j-1}_{i+1}(k)) \\
&\text{Since,} \quad \hat{w}^j_{i+1}(n^J_{i+1} - \hat{n}^j_{i+1}(k)) \ge \hat{c}^{j-1}_i(k) \ge \hat{w}^{j-1}_{i+1}(n^J_{i+1} - \hat{n}^{j-1}_{i+1}(k)) \\
&\Longrightarrow \quad \hat{c}^{j-1}_i(k) - \hat{w}^{j-1}_{i+1}(n^J_{i+1} - \hat{n}^{j-1}_{i+1}(k)) \\
&\qquad = \bar{\zeta}^j_{i+1}(k)(\hat{w}^j_{i+1}(n^J_{i+1} - \hat{n}^j_{i+1}(k)) - \hat{w}^{j-1}_{i+1}(n^J_{i+1} - \hat{n}^{j-1}_{i+1}(k))) \\
&\qquad = -\zeta^{j,3}_i(k) w_{i+1}(\hat{n}^j_{i+1}(k) - \hat{n}^{j-1}_{i+1}(k)) \\
&\qquad \textit{where} \quad 0 \le \bar{\zeta}^j_{i+1}(k) \le 1
\end{aligned}$$

These cases generalize to the expression given in the lemma. □

Proof of Lemma 5.3.5

Proof. We consider four cases. In all the cases we assume $0 \le \zeta^{j,2}_i(k), \zeta^{j,3}_i(k), \alpha^{j,2}_i(k) \le 1$
Case (a) $\acute{M}^j_{i+1}(k) = 1$ and $\acute{M}^{j-1}_{i+1}(k) = 1$

Following along the lines of Case (a) in 5.3.4, we get

$$
\begin{aligned}
& f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) \\
& \quad = \zeta_i^{j,2}(k) V_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k)) - \zeta_i^{j,3}(k) \frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)} w_{i+1}(\hat{n}_{i+1}^j(k) - \hat{n}_{i+1}^{j-1}(k)) \\
& \quad + \hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k)))\hat{n}_i^j(k)\hat{v}_i^j(k) \left[\frac{1}{\hat{c}_i^j(k)} - \frac{1}{\hat{c}_i^{j-1}(k)}\right] \\
& \quad = \zeta_i^{j,2}(k) V_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k)) - \zeta_i^{j,3}(k) \frac{\hat{n}_i^j \hat{v}_i^j}{\hat{c}_i^{j-1}(k)} w_{i+1}(\hat{n}_{i+1}^j(k) - \hat{n}_{i+1}^{j-1}(k)) \\
& \qquad\qquad - \alpha_i^{j,2}(k) \frac{\hat{n}_i^j(k)\hat{v}_i^j(k)}{\hat{c}_i^{j-1}(k)} \left[(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))\right]
\end{aligned}
$$

Since, $\hat{c}_i^j(k) - \hat{c}_i^{j-1}(k) = \eta_i^j(k)(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))V_i(1-\beta_i(k)) = \alpha_i^{j,2}(k)(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))$
and $\hat{w}_{i+1}^j(k)(n_{i+1}^J - \hat{n}_{i+1}^j(k))) \leq \hat{c}_i^j(k)$

Case (b) $\acute{M}_{i+1}^j(k) = 0$ and $\acute{M}_{i+1}^{j-1}(k) = 0$

$$
\begin{aligned}
f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) &= \hat{n}_i^j(k)\hat{v}_i^j(k) - \hat{n}_i^{j-1}(k)\hat{v}_i^{j-1}(k) \\
&= \zeta_i^{j,2}(k) V_i(\hat{n}_i^j(k) - \hat{n}_i^{j-1}(k))
\end{aligned}
$$

Case (c) $\acute{M}_{i+1}^j(k) = 1$ and $\acute{M}_{i+1}^{j-1}(k) = 0$

$$
\begin{aligned}
& f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) \\
& \quad = \frac{\hat{w}_{i+1}^j(n_{i+1}^J - n_{i+1}^j(k))}{\hat{c}_i^j(k)} \hat{n}_i^j(k)\hat{v}_i^j(k) - \hat{n}_i^{j-1}(k)\hat{v}_i^{j-1}(k), \quad \text{and} \\
& \frac{\hat{w}_{i+1}^j(n_{i+1}^J - n_{i+1}^j(k))}{\hat{c}_i^j(k)} \hat{n}_i^j(k)\hat{v}_i^j(k) - \frac{\hat{w}_{i+1}^{j-1}(n_{i+1}^J - n_{i+1}^{j-1}(k))}{\hat{c}_i^{j-1}(k)} \hat{n}_i^{j-1}(k)\hat{v}_i^{j-1}(k) \leq \\
& \qquad f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) - f_i^{out}(\hat{c}_i^{j-1}(k), \hat{n}_{i+1}^{j-1}(k)) \leq \hat{n}_i^j(k)\hat{v}_i^j(k) - \hat{n}_i^{j-1}(k)\hat{v}_i^{j-1}(k)
\end{aligned}
$$

The output flow difference in Case(c) is bounded by the expressions given in Case (a) and Case (b). Depending upon the sign of the flow difference, the flow difference can be written as a scaled version of the expressions given in the above two cases. In both these case, the resulting expression satisfies the lemma.

Case (d) $\acute{M}^{j}_{i+1}(k) = 0$ and $\acute{M}^{j-1}_{i+1}(k) = 1$

$$\begin{aligned}
& f^{out}_i(\hat{c}^j_i(k), \hat{n}^j_{i+1}(k)) - f^{out}_i(\hat{c}^{j-1}_i(k), \hat{n}^{j-1}_{i+1}(k)) \\
&\quad = \hat{n}^j_i(k)\hat{v}^j_i(k) - \frac{\hat{w}^{j-1}_{i+1}(n^J_{i+1} - n^{j-1}_{i+1}(k))}{\hat{c}^{j-1}_i(k)}\hat{n}^{j-1}_i(k)\hat{v}^{j-1}_i(k) \\
& \frac{\hat{w}^{j}_{i+1}(n^J_{i+1} - n^{j}_{i+1}(k))}{\hat{c}^{j}_i(k)}\hat{n}^{j}_i(k)\hat{v}^{j}_i(k) - \frac{\hat{w}^{j-1}_{i+1}(n^J_{i+1} - n^{j-1}_{i+1}(k))}{\hat{c}^{j-1}_i(k)}\hat{n}^{j-1}_i(k)\hat{v}^{j-1}_i(k) \geq \\
&\quad f^{out}_i(\hat{c}^j_i(k), \hat{n}^j_{i+1}(k)) - f^{out}_i(\hat{c}^{j-1}_i(k), \hat{n}^{j-1}_{i+1}(k)) \geq \hat{n}^j_i(k)\hat{v}^j_i(k) - \hat{n}^{j-1}_i(k)\hat{v}^{j-1}_i(k)
\end{aligned}$$

The result for this case is similar to the one given above. □

Proof of Lemma 5.3.6

Proof. For any node i,

$$\Lambda^{i,j}_1 > 0 \implies \hat{c}^j_{i-1}(k) = max\left(c_{min}, min\left(\hat{w}_i(n^J_i - \hat{n}^j_i(k)), \hat{c}^{j-1}_{i-1}(k) + G_1\overline{n}^j_i(k+1)\right)\right)$$

$$\Lambda^{i,j}_2 > 0 \implies \frac{1}{\hat{c}^j_i(k)} = min\left(\frac{1}{\hat{w}^j_{i+1}(k)(n^J_{i+1} - \hat{n}^j_{i+1}(k))}, max\left(\frac{1}{c_{max}}, \frac{1}{\hat{c}^{j-1}_{i-1}(k)} - G_2\overline{n}^j_i(k+1)\right)\right)$$

Remember that, by construction, the a-priori mode is the same as the a-posteriori mode.

$$\begin{aligned}
\Lambda^{i,j}_1 > 0 \implies & \hat{c}^j_{i-1}(k) = \hat{c}^{j-1}_{i-1}(k) + G_1\delta^{j,1}_i(k)\overline{n}^j_i(k+1) \\
& \text{since} \quad \hat{c}^{j-1}_i(k) \in [c_{min}, \hat{w}_i(n^J_i - \hat{n}^j_i(k))] \\
\Lambda^{i,j}_2 > 0 \implies & \frac{1}{\hat{c}^j_i(k)} = \frac{1}{\hat{c}^{j-1}_{i-1}(k)} - G_2\delta^{j,2}_i(k)\overline{n}^j_i(k+1) \\
& ,\text{since} \quad \hat{c}^{j-1}_{i-1}(k) \in [\hat{w}^j_{i+1}(k)(n^J_{i+1} - \hat{n}^j_{i+1}(k)), c_{max}]
\end{aligned}$$

where, $\overline{n}^j_i(k+1) = \dfrac{\tilde{n}^{j,o}_i(k+1)}{1+\Lambda^{i,j}_1+\Lambda^{i,j}_2}$ $\quad$ *and* $\quad 0 \leq \delta^{j,1}_i(k), \delta^{j,2}_i(k) \leq 1$

Also,

$$\hat{n}^{j,o}_i(k+1) = \hat{n}^j_i(k) + f^{in}_i(\hat{c}^{j-1}_{i-1}(k), \hat{n}^j_i(k)) - f^{out}_i(\hat{c}^{j-1}_i(k), \hat{n}^j_{i+1}(k))$$

$$
\begin{aligned}
&\hat{n}_i^j(k+1) = \hat{n}_i^j(k) + f_i^{in}(\hat{c}_{i-1}^j(k), \hat{n}_i^j(k)) - f_i^{out}(\hat{c}_i^j(k), \hat{n}_{i+1}^j(k)) \\
&\implies \hat{n}_i^j(k+1) - \hat{n}_i^{j,o}(k+1) = I_{i-1}(k)(1 - \acute{M}_i^j(k))(\hat{c}_{i-1}^j(k) - \hat{c}_{i-1}^{j-1}(k)) \\
&\qquad + I_i(k)\acute{M}_{i+1}^j(k)\hat{w}_{i+1}^j(n_{i+1}^J - n_{i+1}^j(k))\hat{n}_i^j(k)\hat{v}_i^j(k)\Big(\frac{1}{\hat{c}_i^j(k)} - \frac{1}{\hat{c}_i^{j-1}(k)}\Big) \\
&\implies \tilde{n}_i^{j,o}(k+1) - \tilde{n}_i^j(k+1) = \hat{n}_i^j(k+1) - \hat{n}_i^{j,o}(k+1) \\
&\qquad\quad = (\delta_i^{j,1}(k)\Lambda_1^{i,j} + \delta_i^{j,2}(k)\Lambda_2^{i,j})\frac{\tilde{n}_i^{j,o}(k+1)}{1 + \Lambda_1^{i,j} + \Lambda_2^{i,j}}
\end{aligned}
$$

$$
\begin{aligned}
&\implies \tilde{n}_i^j(k+1) = \tilde{n}_i^{j,o}(k+1)\frac{1 + (1 - \delta_i^{j,1}(k))\Lambda_1^{i,j} + (1 - \delta_i^{j,2}(k))\Lambda_2^{i,j}}{1 + \Lambda_1^{i,j} + \Lambda_2^{i,j}} \\
&\qquad = \overline{n}_i^j(k+1)\Big(1 + (1 - \delta_i^{j,1}(k))\Lambda_1^{i,j} + (1 - \delta_i^{j,2}(k))\Lambda_2^{i,j}\Big) \\
&\implies \overline{n}_i^j(k+1) = \tilde{n}_i^j(k+1)\overline{\delta}_i^j(k), \\
&\qquad \text{where,} \quad 0 \leq \overline{\delta}_i^j(k) = \frac{1}{1 + (1 - \delta_i^{j,1}(k))\Lambda_1^{i,j} + (1 - \delta_i^{j,2}(k))\Lambda_2^{i,j}} \leq 1
\end{aligned}
$$

Substituting for $\overline{n}_i^j(k+1)$ in the update equations, we prove the lemma. Additionally, we note that when the updates do not hit the boundary, i.e. $\delta_i^{j,1}(k) = \delta_i^{j,2}(k) = 0$, we have $\overline{n}_i^j(k+1) = \tilde{n}_i^j(k+1)$. We can see that we are able to estimate the a-posteriori errors without using the a-posteriori estimate $\hat{c}_i^j(k)$, directly from the a-priori errors and the update gains. □

Chapter 6

Optimal control of freeway networks

Traffic control is an important operational management strategy that can be used to relieve traffic congestion in freeways. Ramp metering and variable speed limits are two commonly used control strategies to regulate traffic flow and delay the onset of congestion. The performance of any controller is primarily judged by their ability to decrease the traffic congestion, and this can be usually captured by performance metrics like Total Travel Time (TTT), or the Total Congestion Delay(TCD) of all the vehicles using the freeway system. Given one of these performance functions, optimal control theory allows us to compute the state trajectories as well as the control inputs which minimize the performance objective. Optimal controllers require a model of the freeway to compute these control laws. In this chapter, we present an optimal controller utilizing Link Node Cell Transmission Model (LN-CTM) as its underlying model.

Macroscopic models, including first order models (eg. Cell Transmission Models) as well as second order models (METANET - [29]) have been used in freeway optimal control formulations reported in literature[29, 20, 25, 4]. While the formulation of these optimal control problems is typically easy, the challenge remains in specifying a solution technique which can calculate good quality solutions without being computationally intensive. This is because the optimization problems that arise in these optimal control formulations are large-scale in nature (typically involving thousands of variables, at the least, for even a small freeway section), apart from being non-linear and non-convex. Applying commonly available solution techniques lead to large computation times [29] with no guarantees of global optimality of the solution. Optimal controller formulations based on second order models like METANET suffer from these disadvantages.

In contrast, optimal controller formulations based on the Cell Transmission Model show more promise in terms of computational efficiency and global optimality of the generated solution. Gomes and Horowitz [20] present an optimal ramp metering controller based on the Asymmetric Cell Transmission Model (ACTM) along with an efficient solution strategy. The underlying freeway dynamics in the controller formulations is the ACTM, which is presented as a simplification to the CTM. The motivation for this simplification is to provide a higher quality and efficiently computable solution as compared to the original optimal control problem. The authors presented a relaxed version of this optimal ramp metering problem, and proved that the problems are equivalent in terms of the optimal solution trajectory. The relaxed problem is a linear optimization problem, which can be solved efficiently for large freeway networks with long time horizons.

Compared to the ACTM, the LN-CTM uses a more accurate model of link merges which makes it suitable for simulating on-ramp merges even when on-ramp inflows are appreciable (eg. freeway-freeway interconnections). However, this comes at an additional cost of added non-linearity, and therefore the results and techniques presented in [20] cannot be translated for this case. As we will see in this chapter and the next, the use of the LN-CTM to describe underlying dynamics results in the optimal controller utilizing both variable speed limits and ramp metering. This is different from the results and observations of Gomes and Horowitz that show the optimal controller using only ramp metering as its control mechanism.

6.1 Problem formulation

Traffic model

The underlying model for the optimal control problem is the LN-CTM model, which was presented in Section 3.1. This model captures the effect of ramp metering on the traffic dynamics on the freeway. Additionally, we also need to model the effect of variable speed limits on the traffic state evolution. Figure 6.1 shows the fundamental diagram (with free-flow speed V_i, congestion wave speed w_i and Flow capacity F_i) and the nominal demand and supply functions in solid lines. The nominal demand and supply functions, without any application of variable speed limits (VSL) are given by

$$\begin{aligned}\bar{D}_i(n_i(k)) &= \min(n_i(k)V_i, F_i)\\ \bar{S}_i(n_i(k)) &= \min(W_i(n_i^J - n_i(k)), F_i).\end{aligned}$$

Variable speed limit controllers specify speed limits $v_i(k)$ which are less than the nominal speed limits posted in the freeway. We assume that these variable speed limits are followed with full compliance. When variable speed limits are applied, the demand functions are modified while the supply functions are left unchanged, as seen below

$$\begin{aligned}D_i(n_i(k)) &= \min(n_i(k)v_i(k), F_i)\\ S_i(n_i(k)) &= \min(W_i(n_i^J - n_i(k)), F_i).\end{aligned}$$

The dashed line in Figure 6.1 show the demand function when a non-maximum speed limit is used.

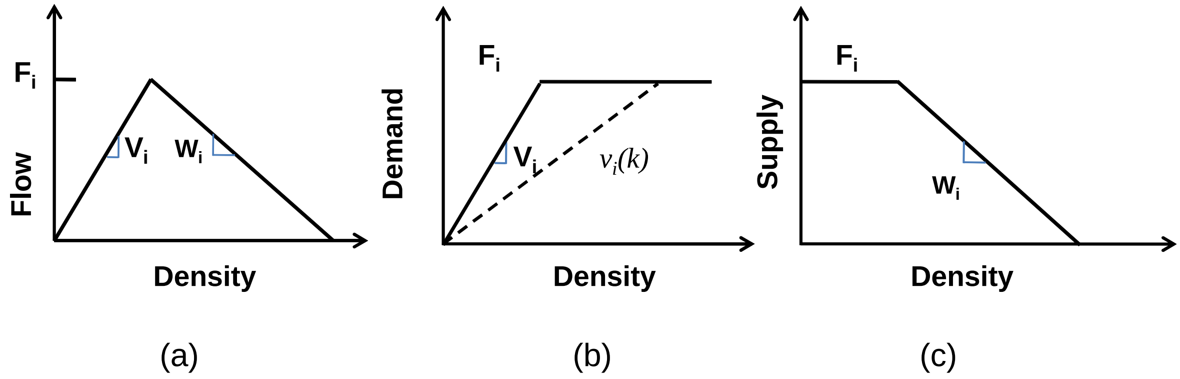

Figure 6.1: (a)The nominal triangular fundamental diagram. (b) The nominal demand function (solid line), and the demand function when a speed limit of $v_i(k)$ is imposed (dashed line) (c) The supply function, which does not depend on the speed limit.

The complete model, which incorporates the variable speed limits, described through a time varying speed limit profile $v_i(k)$ is described below. This will form the model used inside the optimal controller.

Density Update Equations : Mainline/Queue Conservation Equation

$$n_0(k+1) = n_0(k) + Q_0(k) - f_0(k)$$

$$n_i(k+1) = n_i(k) + f_{i-1}(k)(1-\beta_{i-1}(k)) + r_{i-1}(k) - f_i(k) \qquad i = 1, \cdots, N$$

$$l_i(k+1) = l_i(k) + Q_i(k) - r_i(k) \qquad i = 1, \cdots, N \qquad (6.1)$$

Flow Update Equations

$$f_N(k) = D_n(k)$$

$$f_i(k) = D_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \qquad i = 0, \cdots, N-1$$

$$r_i(k) = d_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \qquad i = 1, \cdots, N-1$$

$$s_i(k) = f_i(k)(1 - \beta_i(k)) \qquad i = 1, \cdots, N-1$$

where

$$D_i(k) = \min(n_i(k)v_i(k), F_i) \qquad i = 0, \cdots, N$$

$$R_i(k) = D_i(k)(1 - \beta_i(k)) + d_i(k) \qquad i = 0, \cdots, N-1$$

$$S_{i+1}(k) = \min(W_{i+1}(n^J_{i+1} - n_{i+1}(k)), F_{i+1}) \qquad i = 0, \cdots, N-1$$

$$d_i(k) = \min\left(r^c_i(k), l_i(k)\right) \qquad i = 1, \cdots, N \qquad (6.2)$$

In the model, we can see that the flow out of link i $(f_i(k))$ is a non-decreasing function of the speed limits. The reduction of speed, at any link, while keeping the downstream ramp metering rates constant, leads to a decrease in flow out of the link. Finally, changes in speed limits do not lead to increases in capacity of the freeway section.

Objective function

The objective function for the controller needs to directly reflect the level of congestion in the freeway. Total Travel Time (TTT) and Total Congestion Delay (TCD), captured in units of vehicle hours, are good candidate objective functions that capture the aggregate effect of traffic congestion on all users in the freeway. For our optimal controller, we define the following generalized objective function, based on the macroscopic variables defined in our model.

$$J = \sum_{k=1}^{K} \sum_{i=0}^{N} (n_i(k) + l_i(k) - \alpha_i(k) f_i(k) - \bar{\alpha_i}(k) r_i(k)) \qquad (6.3)$$

where $k = 1 \cdots K$ denotes the time period and $i = 0 \cdots N$ denotes the link $(n_i(k))$ or ramp $(l_i(k))$ index. By choosing values for the parameters $\alpha_i(k) \geq 0, \bar{\alpha_i}(k) \geq 0$, we can represent the following commonly used objective functions.

$$
\begin{aligned}
J_a &= TTT \\
J_b &= TTT - \kappa TTD \\
J_c &= TCD \\
TTT &= \sum_{k,i} (n_i(k) + l_i(k)) && \text{Total Travel Time} \\
TTD &= \sum_{k,i} (f_i(k) + r_i(k)) && \text{Total Travel Distance} \\
TCD &= \sum_{k,i} \left(n_i(k) + l_i(k) - \frac{1}{V_i} f_i(k) \right) && \text{Total Congestion Delay}
\end{aligned}
\tag{6.4}
$$

Control mechanisms and additional constraints

The optimal controller regulates the traffic using a speed limit profile $v_i(k)$ and a time varying ramp metering rate $r_i^c(k)$. The speed limit profile serves as an indirect control mechanism for regulating flows that exit any particular link of the freeway to enter into the next downstream section. Ramp metering rate serves to regulate the flow entering into the freeway through any particular ramp by storing additional vehicles in the ramps.

We impose the following constraints on the control actions.

$$
\begin{aligned}
0 &\leq v_i(k) \leq V_i \\
0 &\leq r_i^c(k) \leq C_i
\end{aligned}
\tag{6.5}
$$

As seen above, the variable speed limit controller is allowed to impose time varying speed limits up to the maximum speed limit of the freeway section. The ramp metering controller specifies any realizable flow rate up to the maximum flow capacity of the ramp C_i. Note that the ramp metering rate $r_i^c(k)$ can be zero, according to our constraints. This assumption will be useful to ensure the validity of the solutions proposed in the next section. In practice, many ramp meters require a minimum ramp metering rate to ensure that vehicles waiting in ramp queues get serviced without excessive delay. Towards the end of this chapter, we will discuss ways to indirectly implement a minimum ramp metering rate for all the ramps in the freeway.

Apart from the control constraints, we also introduce a maximum queue limit constraint. This is necessary to ensure that queue lengths do not exceed the available storage space in the ramps. In practice, presence of queue limit constraints prevent the ramp meters from affecting traffic at the arterial streets which connect to the ramps. The queue constraints for our controller are.

$$
l_i(k) \leq L_i \tag{6.6}
$$

Initial conditions and model parameters

The following parameters and initial conditions must be specified for each link and on-ramp:

- Link i fundamental diagram parameters : Capacity F_i, Free-flow speed V_i and Congestion wave speed W_i.
- On-ramp i parameters (Flow capacity and maximum queue length): C_i, L_i
- Off-ramp i parameters (Split ratios): $\beta_i(k)\, k = 1, \cdots, K$
- Initial Conditions : $n_i(0), l_i(0) \quad i = 0, \cdots, N$
- Flow Demands : $Q_i(k) \quad i = 0, \cdots, N\,,\, k = 0, \cdots, K$

Optimal Control formulation

Combining the objective functions, the freeway dynamic model and the constraints, the final problem can be written as

$$
\begin{aligned}
\min : \quad & J, \text{ given by Eq. (6.3)} \\
S.t. \quad : \quad & For \quad k = 1, \cdots, K \\
& \underline{\text{Conservation equations}} \\
& \text{Equations (6.1)} \\
& \underline{\text{Flow equations}} \\
& \text{Equations (6.2)} \\
& \underline{\text{Constraint equations}} \\
& \text{Equations (6.6), (6.6)} \\
& n_i(k), l_i(k), f_i(k), r_i(k) \geq 0, \quad \forall i \\
& \text{with given initial conditions/fundamental diagram parameters.}
\end{aligned} \tag{6.7}
$$

We will present an efficient solution methodology for this problem in the next section.

6.2 Efficient solutions

The goal of our optimal control problem is to specify ramp metering rates $r_i^c(k)$ and variable speed limit profiles $v_i(k)$ for all the links in the freeway such that our chosen objective is minimized. We see that the original optimal control problem has constraints which are non-linear and non-convex. We will present two optimization problems, whose solutions can be used to derive the optimal state and control trajectory for the original problem.

We now define two optimal control problems. The first, which we denote **Problem A**, is very similar to the optimization problem corresponding to the optimal controller formulated above. Its solution involves nonlinear optimization. The second problem, which we denote **Problem B** constitutes a relaxed optimization problem since its solution only involves linear programming. Subsequently we prove that a solution of **Problem B** can be extended to provide a solution of **Problem A**.

For the first problem, which we will denote **Problem A**, we absorb the ramp metering profile variables $r_i^c(k)$ in the constraints in the optimal controller formulation(Eq. (6.7)).
Problem A Original Problem

$$\begin{aligned}
\min: &\quad J\text{, given by Eq. (6.3)} \\
S.t. \quad : &\quad For \quad k = 1, \cdots, K \\
&\quad \underline{\text{Conservation equations}} \\
&\quad \text{Equations (6.1)} \\
&\quad \underline{\text{Flow equations}} \\
&\quad \text{Equations (6.2)} \\
&\quad \underline{\text{Constraint equations}} \\
&\quad 0 \leq v_i(k) \leq V_i \\
&\quad 0 \leq d_i(k) \leq \min(C_i, l_i(k)) \\
&\quad l_i(k) \leq L_i \\
&\quad n_i(k), l_i(k), f_i(k), r_i(k) \geq 0 \\
&\quad \text{with given initial conditions/parameters.} \qquad (6.8)
\end{aligned}$$

Problem A is equivalent to the optimization problem corresponding to the original optimal control formulation. The optimal controller defined in **Problem A** provides a ramp demand profile ($d_i(k)$) for all ramps and a speed limit ($v_i(k)$) profile for all links in the network. This ramp demand profile can be used to extract the ramp metering rate profiles ($r_i^c(k)$). Since $d_i(k) = \min\left(r_i^c(k), l_i(k)\right)$ (Eq. (6.2)), by choosing $r_i^c(k) = d_i(k)$ we can get a ramp metering rate profile that will be optimal according to the original formulation. The optimal profiles from **Problem A** and the ramp metering rate determined by choosing $r_i^c(k) = d_i(k)$ will be the optimal solution for the original optimal control problem. Note that we are able to eliminate the variables corresponding to the ramp flow constraints without introducing additional non-linear constraints since the lower bound of the ramp metering rate is set to 0.

We now pose an alternate relaxed optimization problem with a solution that only involves a linear program.

Problem B Relaxed Problem

$$\begin{aligned}
\min: \quad & J\text{, given by Eq. (6.3)} \\
S.t. \quad : \quad & For \quad k = 1, \cdots, K \\
& \underline{\text{Conservation equations}} \\
& \text{Equations (6.1)} \\
& \underline{\text{Relaxed Flow equations}} \\
& \bar{f}_i(k) \leq \bar{n}_i(k) V_i & i = 0, \cdots, N \\
& \bar{f}_i(k) \leq F_i & i = 0, \cdots, N \\
& \bar{f}_i(k)(1 - \beta_i(k)) + \bar{r}_i(k) \leq F_{i+1} & i = 0, \cdots, N-1 \\
& \bar{f}_i(k)(1 - \beta_i(k)) + \bar{r}_i(k) \leq W_{i+1}(n^J_{i+1} - \bar{n}_{i+1}(k)) & i = 0, \cdots, N-1 \\
& \underline{\text{Constraint equations}} \\
& 0 \leq \bar{r}_i(k) \leq \min(C_i, \bar{l}_i(k)) & i = 1, \cdots, N \\
& \bar{l}_i(k) \leq L_i & i = 1, \cdots, N \\
& n_i(k), l_i(k), f_i(k), r_i(k) \geq 0 & \forall i \\
& \text{with the same initial conditions/parameters.} & (6.9)
\end{aligned}$$

Notice that we have chosen to use an upper bar to denote the optimization variables in **Problem**

B (e.g. $\overline{n_i}(k)$, $\overline{f_i}(k)$, $\overline{r_i}(k)$) in order to distinguish them from their counterparts in **Problem A**. The main differences between the two problems is that we do not explicitly consider the link velocity variables (e.g $\overline{v_i}(k)$) and the on-ramp demands (e.g. $\overline{d_i}(k)$) in **Problem B**. Next, we will outline the methodology adopted to convert a solution of **Problem B** to a solution of **Problem A**.

Conversion algorithm

Let $\overline{n}_i^*(k), \overline{f}_i^*(k), \overline{l}_i^*(k), \overline{r}_i^*(k)$ denote the optimal (or a feasible) solution of **Problem B**. **Algorithm A** given below generates outputs $n_i^*(k), f_i^*(k), l_i^*(k), r_i^*(k), v_i^*(k), d_i^*(k)$.

Algorithm A

```
For each time period $k$ and link $0 \leq i \leq N$,
$n_i^*(k) = \overline{n}_i^*(k)$
$f_i^*(k) = \overline{f}_i^*(k)$
$l_i^*(k) = \overline{l}_i^*(k)$
$r_i^*(k) = \overline{r}_i^*(k)$
For each time period $k$ and link $0 \leq i < N-1$,
$if \quad f_i^*(k) = \min(n_i^*(k)V_i, F_i)$
  $v_i^*(k) = V_i$
  $d_i^*(k) = r_i^*(k)$
$else\,if \quad f_i^*(k)(1-\beta_i(k)) + r_i^*(k) < S_{i+1}^*(k)$
    $d_i^*(k) = r_i^*(k)$
    $v_i^*(k) = f_i^*(k)/n_i^*(k)$
```

$$else\,if \quad \frac{r_i^*(k)}{S_{i+1}^*(k)} \leq \frac{\min\left(C_i, l_i^*(k)\right)}{\min(n_i^*(k)V_i, F_i)(1-\beta_i(k)) + \min\left(C_i, l_i^*(k)\right)}$$

$$v_i^*(k) = V_i$$

$$d_i^*(k) = r_i^*(k) \times \frac{\min(n_i^*(k)V_i, F_i)(1-\beta_i(k))}{S_{i+1}^*(k) - r_i^*(k)}$$

$else$

$$v_i^*(k) = \frac{\min\left(C_i, l_i(k)\right)}{n_i^*(k)(1-\beta_i(k))} \times \left(\frac{S_{i+1}^*(k)}{r_i^*(k)} - 1\right)$$

$$d_i^*(k) = \min\left(C_i, l_i(k)\right)$$

$$where \quad S_i^*(k) = \min\left(W_i(n_i^J - \overline{n}_i^*(k)), F_i(k)\right)$$

and for each time period k

$$if\ f_N^*(k) = \min(\overline{n}_N(k)V_N, F_N)$$

$$v_N^*(k) = V_N$$

$else$

$$v_N^*(k) = f_N^*(k)/n_N^*(k)$$

The conversion algorithm provides an optimal solution to the **Problem A**. The speed limit variables can be directly applied as the control input, while the on-ramp demands are used to obtain the ramp metering rates $r_i^c(k) = d_i(k)$.

Proofs

The following results will help prove that the variables $n_i^*(k), f_i^*(k), l_i^*(k), r_i^*(k), v_i^*(k), d_i^*(k)$ are feasible and optimal for **Problem A**.

Lemma_6.2.1. *Let $A := \{n_i(k), f_i(k), l_i(k), r_i(k), v_i(k), d_i(k)\}$ be the solution derived from $B := \{\overline{n}_i(k), \overline{f}_i(k), \overline{l}_i(k), \overline{r}_i(k)\}$ using* ***Algorithm A****. Then A is a feasible solution for* ***Problem A*** *if B is a feasible solution of* ***Problem B****. A and B evaluate to identical costs for the respective optimization problems.*

Proof. The constraints corresponding to the conservation equations and the queue limits are identical for both problems. Thus, A satisfies the conservation equations and the queue constraints since B satisfies the conservation equations/queue constraints. We need to prove that A satisfies the flow equations and other constraints of **Problem A**, which are

$$\begin{aligned}
&f_N(k) = D_n(k)\\
&f_i(k) = D_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)}\\
&r_i(k) = d_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \qquad i = 1, \cdots, N\\
&0 \le v_i(k) \le V_i\\
&0 \le d_i(k) \le \min(C_i, l_i(k))
\end{aligned} \tag{6.10}$$

As before, we define

$$\begin{aligned}
&S_{i+1} = \min\left(F_{i+1}(k), W_{i+1}(n^J_{i+1} - n_{i+1}(k))\right)\\
&R_i(k) = \min(n_i(k)v_i(k), F_i)(1-\beta_i(k)) + d_i(k)\\
&D_i(k) = \min(n_i(k)v_i(k), F_i)
\end{aligned}$$

The link densities and flows as well as the ramp flows and queue lengths are identical for both the problems. Therefore, from the constraints of **Problem B** we get,

$$\begin{aligned}
&f_i(k) \le \min(n_i(k)V_i, F_i) \quad\; i = 0..N\\
&f_i(k)(1-\beta_i(k)) + r_i(k) \le S_{i+1} \quad i = 0..N-1\\
&r_i(k) \le \min(C_i, l_i(k)) \quad i = 1..N
\end{aligned} \tag{6.11}$$

At each time instant k and for any link $i = 0 \cdots N-1$, the ramp demands and speed limits are obtained from one of the four different branches of the conditional algorithm. We analyze all the four cases.

Case (a) $f_i(k) = \min(n_i(k)V_i, F_i)$ **:**

In this case, we have

$v_i(k) = V_i$,

$r_i(k) = d_i(k) \le \min(C_i, l_i(k))$

Using $r_i(k) = d_i(k)$ and $f_i(k) = \min(n_i(k)V_i, F_i)$ along with Eq. (6.11),

$$R_i(k) = \min(n_i(k)v_i(k), F_i)(1-\beta_i(k)) + d_i(k) = f_i(k)(1-\beta_i(k)) + r_i(k) \leq S_{i+1}$$

$$\Rightarrow \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} = \frac{R_i(k)}{R_i(k)} = 1$$

$$\text{Therefore,} \quad \min(n_i(k)v_i(k), F_i) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} = f_i(k)$$

$$\textit{and} \quad d_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} = d_i(k) = r_i(k)$$

This shows that the generated variables satisfy the constraints given in Eq. (6.10).

Case (b) $f_i(k) < \min(n_i(k)V_i, F_i)$ and $f_i(k)(1-\beta_i(k)) + r_i(k) < S_{i+1}(k)$ **:**

In this case, we have

$d_i(k) = r_i(k) \leq \min(C_i, l_i(k))$,

$v_i(k) = f_i(k)/n_i(k) \leq (n_i(k)V_i)/n_i(k) = V_i$,

Using the expressions given above along with Eq. (6.11),

$$\min(n_i(k)v_i(k), F_i) = \min\left(n_i(k)\frac{f_i(k)}{n_i(k)}, F_i\right) = \min(f_i(k), F_i) = f_i(k) \quad \text{and}$$

$$R_i(k) = \min(n_i(k)v_i(k), F_i)(1-\beta_i(k)) + d_i(k) = f_i(k)(1-\beta_i(k)) + r_i(k) \leq S_{i+1}$$

$$\Rightarrow \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} = \frac{R_i(k)}{R_i(k)} = 1$$

$$\text{Therefore,} \quad \min(n_i(k)v_i(k), F_i) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} = f_i(k)$$

$$\textit{and} \quad d_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} = d_i(k) = r_i(k)$$

Hence the new variables generated in this case satisfy the constraints given in Eq. (6.10).

Case (c) $f_i(k) < \min(n_i(k)V_i, F_i)$, $f_i(k)(1-\beta_i(k)) + r_i(k) = S_{i+1}(k)$ and $\frac{r_i(k)}{S_{i+1}(k)} \leq \frac{\min(C_i, l_i(k))}{\min(n_i(k)V_i, F_i)(1-\beta_i(k)) + \min(C_i, l_i(k))}$ **:**

In this case, we have

$v_i(k) = V_i$

$$d_i(k) = r_i(k)\frac{\min(n_i(k)V_i, F_i)(1-\beta_i(k))}{S_{i+1}(k) - r_i(k)}$$

Therefore,

$$\frac{d_i(k)}{r_i(k)} = \frac{\min(n_i(k)V_i,F_i)(1-\beta_i(k))}{S_{i+1}(k)-r_i(k)} = \frac{[\min(n_i(k)V_i,F_i)(1-\beta_i(k))]+[d_i(k)]}{[S_{i+1}(k)-r_i(k)]+[r_i(k)]} = \frac{R_i(k)}{S_{i+1}(k)}$$

$$\implies r_i(k) = d_i(k)\frac{S_{i+1}(k)}{R_i(k)}$$

Since $f_i(k)(1-\beta_i(k))+r_i(k) = S_{i+1}(k)$ and $f_i(k) < \min(n_i(k)V_i,F_i)$,

$$\frac{R_i(k)}{S_{i+1}(k)} = \frac{\min(n_i(k)V_i,F_i)(1-\beta_i(k))}{S_{i+1}(k)-r_i(k)} = \frac{\min(n_i(k)V_i,F_i)}{f_i(k)} > 1$$

$$\implies S_{i+1}(k) < R_i(k) \quad \text{and} \quad \frac{\min(R_i(k),S_{i+1}(k))}{R_i(k)} = \frac{S_{i+1}(k)}{R_i(k)}$$

Combining the results stated above,

$$r_i(k) = d_i(k)\frac{\min(R_i(k),S_{i+1}(k))}{R_i(k)}, \quad f_i(k) = \min(n_i(k)v_i(k),F_i) \times \frac{\min(R_i(k),S_{i+1}(k))}{R_i(k)}$$

Also $\dfrac{r_i(k)}{S_{i+1}(k)} = \dfrac{d_i(k)}{\min(n_i(k)V_i,F_i)(1-\beta_i(k))+d_i(k)}$

$$\leq \frac{\min(C_i,l_i(k))}{\min(n_i(k)V_i,F_i)(1-\beta_i(k))+\min(C_i,l_i(k))}$$

$$\implies d_i(k) \leq \min(C_i,l_i(k))$$

In this case, we see that the new variables generated by the algorithm satisfy the constraints given in Eq. (6.10).

Case (d) $f_i(k) < \min(n_i(k)V_i,F_i)$, $f_i(k)(1-\beta_i(k))+r_i(k) = S_{i+1}(k)$ and $\frac{r_i(k)}{S_{i+1}(k)} > \frac{\min(C_i,l_i(k))}{\min(n_i(k)V_i,F_i)(1-\beta_i(k))+\min(C_i,l_i(k))}$ **:**

In this case, we have

$$d_i(k) = \min(C_i,l_i(k)) \geq r_i(k),$$

$$v_i(k) = \frac{\min(C_i,l_i(k))}{n_i(k)(1-\beta_i(k))} \times \left(\frac{S_{i+1}(k)}{r_i(k)} - 1\right)$$

Using $\dfrac{r_i(k)}{S_{i+1}(k)} > \dfrac{\min(C_i,l_i(k))}{\min(n_i(k)V_i,F_i)(1-\beta_i(k))+\min(C_i,l_i(k))}$,

$$v_i(k) \leq \frac{\min(C_i,l_i(k))}{n_i(k)(1-\beta_i(k))} \times \left(\frac{\min(n_i(k)V_i,F_i)(1-\beta_i(k))+\min(C_i,l_i(k))}{\min(C_i,l_i(k))} - 1\right)$$

$$\leq \frac{\min(n_i(k)V_i,F_i)(1-\beta_i(k))}{n_i(k)(1-\beta_i(k))} = \frac{\min(n_i(k)V_i,F_i)}{n_i(k)} \leq V_i$$

and $\quad n_i(k)v_i(k) = \min(n_i(k)v_i(k),F_i) \leq \min(n_i(k)V_i,F_i)$

$$\text{Finally,} \quad \frac{\min(n_i(k)v_i(k), F_i)(1-\beta_i(k))}{\min(C_i, l_i(k))} = \left(\frac{S_{i+1}(k)}{r_i(k)} - 1\right) = \frac{f_i(k)(1-\beta_i(k))}{r_i(k)}$$
$$\text{and} \quad f_i(k)(1-\beta_i(k)) + r_i(k) = S_{i+1}(k)$$
$$\implies f_i(k) = n_i(k)v_i(k) \times \frac{S_{i+1}(k)}{R_i(k)} \quad \textit{and} \quad r_i(k) = d_i(k) \times \frac{S_{i+1}(k)}{R_i(k)}$$
$$\text{Moreover,} \quad r_i(k) \le d_i(k) \implies S_{i+1}(k) \le R_i(k) \implies \frac{S_{i+1}(k)}{R_i(k)} = \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)}$$

The variables generated from this conditional branch also satisfy Eq. (6.10).

From the analysis of all the four cases, we see that the generated variables satisfy the flow conditions of **Problem A**. By construction, we see that A and B evaluate to identical costs for the respective optimization problems. □

Lemma 6.2.2. *Let $A = \{n_i(k), f_i(k), l_i(k), r_i(k), d_i(k), v_i(k)\}$ be a feasible solution of **Problem A**, then $B = \{n_i(k), f_i(k), l_i(k), r_i(k)\}$ is a feasible solution for **Problem B**. Moreover, A and B evaluate to identical costs for the respective optimization problems.*

Proof. Clearly, B satisfies the constraints corresponding to the conservation equations and the queue limits of **Problem B**. We show below that B satisfies the relaxed flow constraints of **Problem B**.

$$\text{Noticing that} \quad v_i(k) \le V_i \quad i = 0 \cdots N$$
$$f_N(k) = D_n(k) = \min(n_N(k)v_N(k), F_i) \le \min(n_N(k)V_N, F_i)$$
$$\implies \bar{f}_N(k) \le \bar{n}_N(k)V_N, \quad \bar{f}_N(k) \le F_N$$

For any link $i = 0 \cdots N-1$

$$f_i(k) = D_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \le D_i(k) \le \min(n_i(k)V_i, F_i)$$
$$\implies \bar{f}_i(k) \le \bar{n}_i(k)V_i, \quad \bar{f}_i(k) \le F_i$$

$$
\begin{aligned}
\bar{f}_i(k)(1-\beta_i(k)) + \bar{r}_i(k) &= (D_i(k)(1-\beta_i(k)) + d_i(k)) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \\
&= \min(R_i(k), S_{i+1}(k)) \leq S_{i+1}(k) \\
\implies \bar{f}_i(k)(1-\beta_i(k)) + \bar{r}_i(k) \leq F_{i+1} \quad &\textit{and} \quad \bar{f}_i(k)(1-\beta_i(k)) + \bar{r}_i(k) \leq W_{i+1}(n^J_{i+1} - \bar{n}_{i+1}(k)) \\
\text{Also, for any on-ramp } i \quad \bar{r}_i(k) &= d_i(k)\frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \leq d_i(k) \leq \min(C_i, \bar{l}_i(k))
\end{aligned}
$$

Hence, B is a feasible solution for **Problem B**. It is also easy to see that A and B have identical costs for the respective optimization problems. □

Theorem 6.2.1. *Let $B = \{\bar{n}^*_i(k), \bar{f}^*_i(k), \bar{l}^*_i(k), \bar{r}^*_i(k)\}$ be an optimal solution of **Problem B** and $A = \{n^*_i(k), f^*_i(k), l^*_i(k), r^*_i(k), v^*_i(k), d^*_i(k)\}$ be the solution derived using **Algorithm A**. Then A is an optimal solution for **Problem A**.*

Proof. Lemma 6.2.1 shows that A is a feasible solution for **Problem A**. Suppose A is not optimal for **Problem A**, i.e. there is another solution A' which evaluates to a lower cost. Lemma 6.2.2 allows us to compute a new solution B' which is feasible for **Problem B**, and also has a lower cost than B (which has a cost equivalent to A). This contradicts the fact that B is optimal for **Problem B**. Thus A, derived from B is an optimal solution of **Problem A**. □

Extensions

As we will demonstrate in the next section, the optimal controller and the solution technique presented above can be used inside a model predictive controller. In this situation, it is beneficial to modify the formulation presented above, by converting the hard constraints on queues to soft constraints. In practice, when the on-ramp demands cannot be predicted accurately, queue constraints are expected to be frequently violated when an MPC incorporating hard constraints is executed. Soft constraints on queues can help in maintaining feasibility even when queue constraints are violated. Let $\zeta_i(k)$ be the new variable that captures the queue violation. Then, we modify the cost function as $\bar{J} = J + C\sum_{i,k}\zeta_i(k)$, and add $\bar{l}_i(k) - L_i \leq \zeta_i(k) \quad i = 1..N, k = 1..K$, and $0 \leq \zeta_i(k) \quad i = 1..N, k = 1..K$ to the constraints. The presence of these new soft constraints eliminate problems related to infeasibility.

The optimal controller presented here may specify very low ramp metering rates during certain time periods, as we do not include constraints on minimum ramp metering rates. In many ramps, minimum ramp metering rates (typically around 180 vphpl) are usually specified, to ensure that vehicles in some ramps are not subjected to long wait times when ramp metering is active. As specified in [19, 20], we could obtain metering rates from the optimal controller, and replace metering rates below the accepted minimum by the minimum ramp metering rate. Another method we could use to permit a fair service time is to ensure that ramp flows exceed a given proportion of the current queue length, using the constraint $r_i(k) \geq l_i(k)p$, with $0 \leq p \leq \frac{C_i}{L_i}$, so as to ensure that ramp flows do not exceed capacity. Suppose we wish to target a ramp flow rate of C_i^{min} when ramp queue reaches its limit L_i, we can choose $p = \frac{C_i^{min}}{L_i}$. We typically convert this into a soft constraint, using the same technique presented above.

Finally, we can also add constraints to limit the average wait time in different queues. The average wait time accumulated for all vehicles between time periods $k = k_1$ to $k = k_2$ is given by

$$\frac{\sum_{k=k1}^{k=k2} l_i(k)}{l_i(k_1) + \sum_{k=k1}^{k=k2} Q_i(k)}$$

Notice that the above definition does not include wait times accumulated by vehicles before the start time period k_1 (i.e. it does not include the initial wait times of vehicles which were part of the initial queue). Similarly, it does not include the complete wait times of vehicles which may be still waiting at the end of time period k_2. Due to these limitations, this approximation may not be suitable when short time periods are considered. Nonetheless, we can add linear constraints to limit the average wait times during different periods. For example, let T_i^{max}, be the max wait time specified for the controller, then the following linear inequality constraints are added to the optimal controller.

$$\sum_{k=k1}^{k=k2} l_i(k) \leq T_i^{max} \left(l_i(k_1) + \sum_{k=k1}^{k=k2} Q_i(k) \right)$$

To ensure that adding these constraints do not lead to infeasibility, we can also convert them into soft constraints, and add a penalty term to the cost function.

In all of the three extensions presented above, we only add linear constraints to the optimal controller formulation, and the solution techniques and the theoretical results presented above still apply.

Remarks

In the solution methodology presented here, we have used a relaxation technique to map the non-linear optimization problem to a linear optimization problem. The relaxation technique works only when variable speed limits are applied to all links, and all ramps are metered. In the problem formulation, we have also allowed the split ratio, $\beta_i(k)$ to be time-varying. However, one needs to be careful while searching for an optimal speed control profile in case of time-varying split ratios. For example, consider the case where the split ratios for the first cell increase with time. In this case, an optimal speed control law might initially hold back vehicles (by decreasing the speed limit), so that the vehicles catch a higher split ratio, and exit the freeway. This does not reflect reality, since the vehicles are routed to the wrong destination. This effect is exacerbated when J_a

is considered as the objective, since vehicles that exit do not contribute to the Total Travel Time in the downstream links. In contrast, augmenting the objective with the flow terms $(-f_i(k))$ serves to alleviate this effect here, as vehicles exiting the freeway do not contribute to the flow downstream. In the case of decreasing split ratios, the roles of these terms are reversed. Hence we argue that J_b or J_c is a better objective function to consider for the problem. In the case of constant split ratios, this problem does not arise. This problem is not unique to a optimal control formulation using the LN-CTM, but arises due to the use of a split-ratio based routing scheme adopted by this model. This observation was hinted by the authors in [20].

6.3 Model Predictive ramp metering and speed control

We present a model predictive controller (MPC) based on the optimal control formulation presented in the previous section. The model predictive controller solves an open loop optimal control problem online based on a plant model at each sampling time, using the state information measured at the current sampling time. The controller implements the control steps of the obtained optimal control profile till the next sampling time, and then the process is repeated [3].

Let T and N_p denote the model time step and prediction horizon used in the optimization problem respectively. We execute the MPC every $T_c = N_c \times T$ time instants (here we assume that N_p, N_c are natural numbers). In the model predictive controller, the split ratio is assumed to be constant, equal to the split ratio observed at the instant the controller is initiated. This averts the problem related to time varying split ratios detailed in the previous section, and does not usually lead to any appreciable decrease in the controller performance within the MPC framework. We also adopt soft constraints for queues, as presented in the previous section. We choose total congestion delay, as the controller optimization cost.

For the simulation experiments presented here, we use a calibrated model of the I-80E freeway in the Bay area between the Bay Bridge and the Carquinez Bridge. The model was calibrated to replicate the congestion patterns observed on September 2nd, 2008. Figure 6.2 (Top) shows the speed contours produced by the model without any control measures activated. This freeway experiences congestion during the evening commute periods, and we limit the temporal axis to cover the evening congestion. We apply a model predictive controller with $T = 10s$, $N_p = 100$ and $N_c = 9$ to specify the ramp metering rates and variable speed profile for this freeway. A queue limit of $L_i = 50\,\forall i$ was imposed for this simulation. The actual demand profiles were assumed to be known and used to specify the demands in the controller. Constant split ratios, equal to the split ratios at the time period of controller actuation, were used. Figure 6.2 (middle) represents the speed contour observed when the MPC is used, and Figure 6.2 (bottom) shows the speed limit profile generated by the MPC. Given the limited queue size constraint imposed on the controller, the controller did not completely eliminate the congestion present in the freeway. However, the MPC succeeds in delaying the onset of congestion on the freeway. In this scenario, the controller resulted in a delay reduction of 17.85%. In Figure 6.3, we show the resulting queues on all the

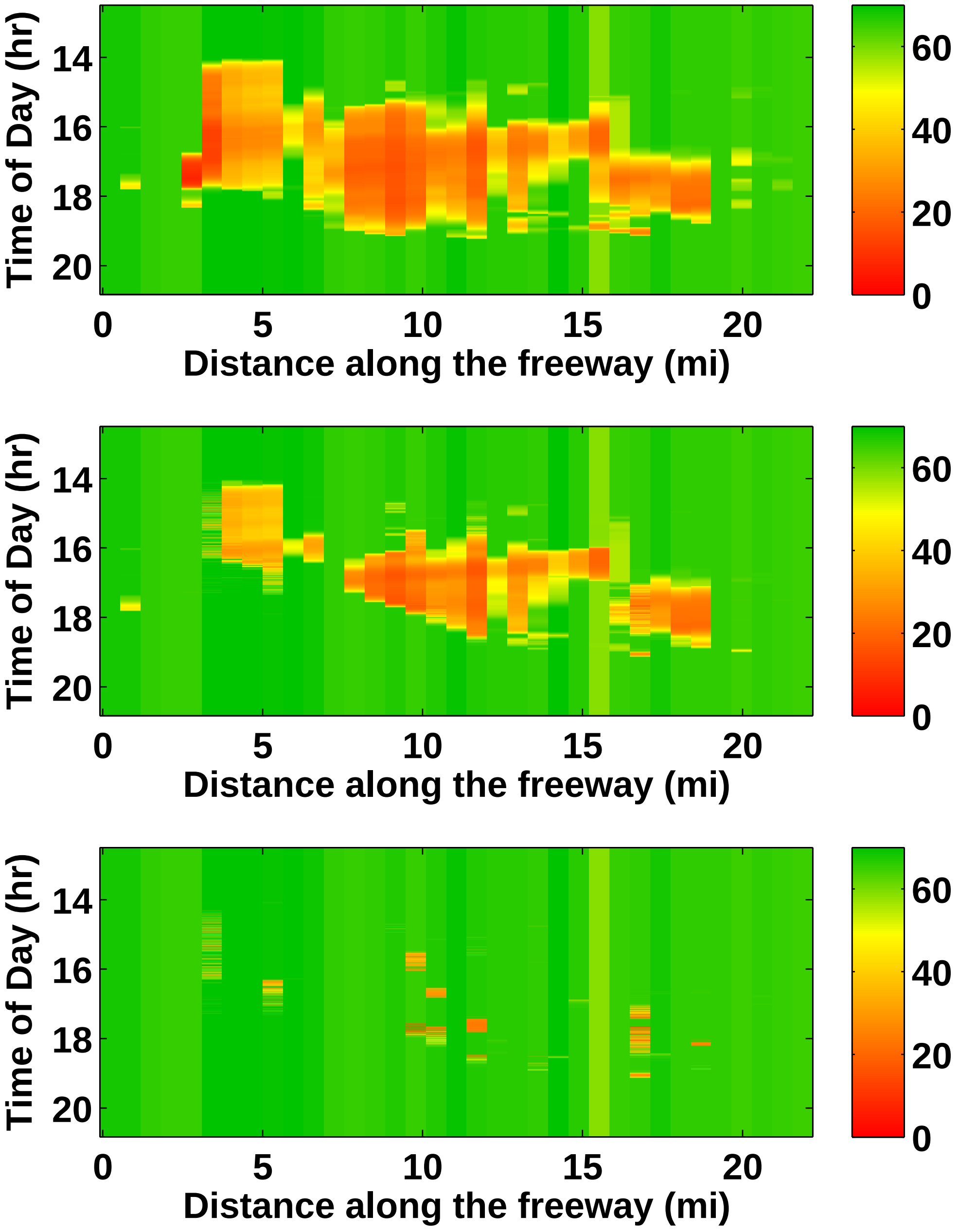

Figure 6.2: Top : Speed contours in the uncontrolled case. Middle : Speed contours with ramp metering and VSL. Bottom : VSL specified by the MPC.

onramps. We see that the queue constraints are not adversely violated in this case (when $C = 5$ was chosen), as the queues lengths are within the 50 veh/ramp limit.

Table 6.1: Role of demand and split ratio information

Demands	Split ratio	Delay reduction
Exact	Exact	18.85%
Constant	Exact	18.28%
Exact	Constant	17.85%
Constant	Constant	17.42%

Next, we explore the role of perfect demand and split ratio information on the performance gains obtained using the MPC, as seen in Table 6.1. We choose the demand/split ratio information to either be exact (ie. equal to the actual realized profiles in the simulation model), or constant (equal to the realized value at the instant the controller is initiated). We caution that the delay reduction with exact split ratios (with/without exact demands) are just shown for comparison. In this case, the optimal controller decreased the speeds to very low values during some periods, so that vehicles can exit the freeway at a later time, when the split ratio values are higher, and this might not reflect reality. Nonetheless, this study shows that we can expect a marginal decrease in delay reduction when operating with constant splits. The effect of not knowing the exact demand information also leads to a small decrease in the performance gains. We have observed that the controller performance is more sensitive to the inaccuracy of demand information around the start time of the prediction horizon. For example, if incorrect demand information is provided for the first N_c steps, we observed a marked decrease in the controller performance. In contrast, decrease in accuracy of the demands along the prediction horizon does not affect the controller performance, as long as the demand information around the current time periods is accurate. In this case, when the MPC is executed during the next time period, we get more accurate demands to base the future control actions on. In fact, when constant demands (equal to the realized value at the instant the controller is initiated) are chosen, the demand information around the controller actuation period is quite accurate, since our demand profiles are sufficiently smooth.

We also explore the effect of various parameters on the performance of the model predictive controller. In these parametric studies, we use exact demands and constant split ratios. Table 6.2 lists the performance of the MPC when the control horizon, prediction horizon and the maximum queue limit, are varied. We have generally observed that the control horizon is more critical than the prediction horizon. In particular, we note that prediction horizons can be as short as 10 minutes ($N_p = 60$) in this case. It is expected that the prediction horizon can be decreased for shorter freeway sections. In contrast, we see that the control horizon needs to be sufficiently small i.e 2 or 3 mins (N_c $12 - 18$). Longer control horizons lead to a decrease in controller performance (this was found to be the case irrespective of the prediction horizon chosen). The main reason for the need for shorter controller horizons is the use of constant split ratios in our models. In the case of imperfect demand information, control horizons further determine the performance of

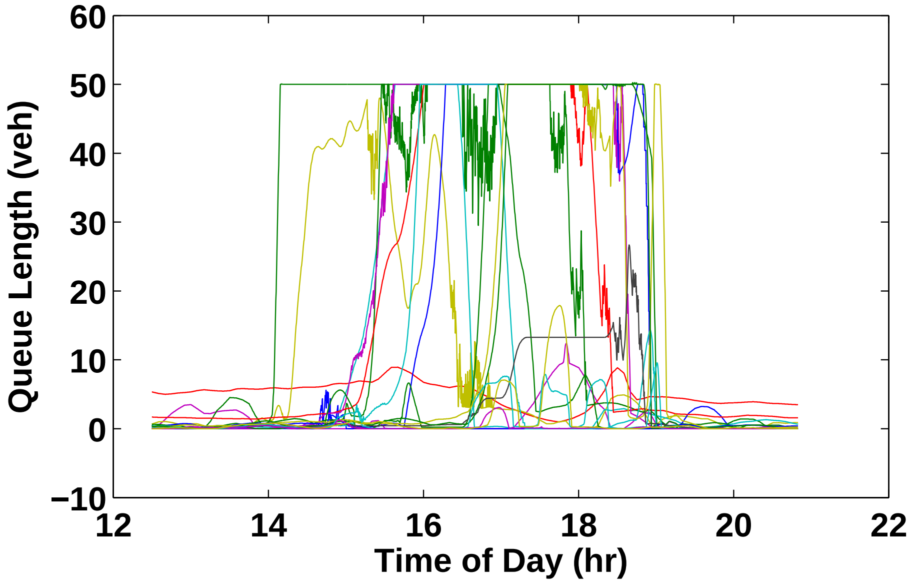

Figure 6.3: Queue lengths in the controlled case.

the controller, as short control horizons allow the controller to correct the demand estimates used inside the MPC, as well as measure the queues and indirectly account for the faulty ramp demand estimates. We observed a performance decrease of 2% as we changed the N_c from 6 to 18, when constant demands and constant split ratios were used. Short control horizons necessitate the use of a fast optimization routine in the MPC. Finally, we also see that ramp queues limits have a major effect on the efficiency gains that can be expected out of the controlled system.

Table 6.2: MPC Parameter study

N_c $(N_p = 120, L_i = 50)$	6	12	18	24	30
Delay reduction	17.91%	17.76%	17.55%	17.2%	15.96%
N_p $(N_c = 9, L_i = 50)$	30	60	90	120	150
Delay reduction	17.82%	17.85%	17.85%	17.85%	17.85%
L_i $(N_p = 120, N_c = 9)$	10	20	50	100	∞
Delay reduction	7.6%	11.7%	17.86%	23.8%	25.65%

Finally, we explore the role of variable speed limits in the optimal control formulation. In the scenario demonstrated in the first experiment, variable speed limits are important to ensure that ramp queue limits are not violated. Generally, when the link downstream of a ramp starts getting congested, the controller meters the ramp flows entering into the section, and the on-ramp queues build up. However, once the queue reaches its limit, the optimal controller needs to maintain the ramp flow to be equal to the demand entering the ramp, so that queues do not exceed the given limits. When the link downstream gets congested, and the demand from the upstream freeway

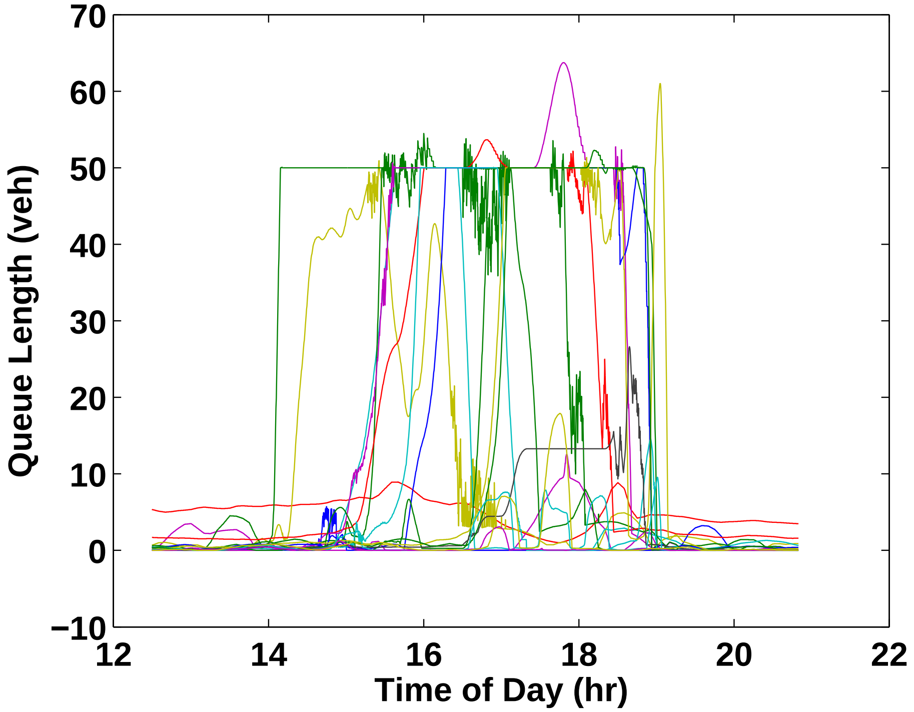

Figure 6.4: Queue lengths, when only the ramp metering portion of the optimal controller is used.

mainline is also high, these ramp flow rates cannot be realized only by specifying high ramp metering rates (this corresponds to the fourth conditional statement in **Algorithm A**). Variable speed limits help maintain the queue limits in this case. We carried out a simulation experiment (with the same parameters as the first experiment explained above), where we only applied the ramp metering portion of the control actions specified by the MPC, while discarding the variable speed limits. This resulted in a delay reduction of 17.78 %, which is very similar to the performance gains in the first simulation. However, the queues in some of the on-ramps were violated, as seen in Figure 6.4. When ramp queue limits were not used, discarding the VSL and applying the ramp metering portion of the control actions lead to a performance gain of 25.26 %, as compared to the delay reduction of 25.34 %, when the complete controller was used. It is our conjecture that variable speed limits do not contribute to significant performance improvements for the freeways, if maintaining exact queue limits are not a priority. However, speed limits play a central role, when capacity drop is present, as we will see in the next chapter.

6.4 Summary

In this chapter, we presented a framework for optimal congestion control for freeway networks, using ramp metering and variable speed limits. The model based predictive controller used the LN-CTM as the underlying model to describe the traffic dynamics in the freeway network. The optimization problem based on the LN-CTM had non-linear and non-convex constraints. We proposed a relaxed optimization problem, with only linear equalities and inequalities, and provided a procedure to map the solution of the relaxed problem to a ramp metering and speed limit profile for the original problem. We proved that the resulting control is also optimal for the original problem.

Given the large-scale nature of a typical centralized freeway optimal control problem, this methodology enables us to solve the optimization problem in real time to incorporate it in a model predictive controller. Typically, a problem with $N_p = 100$, $T = 10s$ and $n = 33$ had around 35000 and 7000 inequality and equality constraints of around 17000 variables. This problem can be efficiently solved within 5 seconds using the MOSEK Linear program solver, which is a fraction of the controller time horizon. Thus the controller presented here has a potential to be adopted for real time traffic control. The approach taken in this paper has advantages over the approaches presented in [29, 25] that use second order models; particularly with respect to global optimality and computation speed. The global optimality guaranteed by this approach ensures that optimal controller (executed using a MPC) can be used to compare and evaluate other control methodologies applied to the same setup. We can also use the MPC setup to perform parametric studies. The controller has been previously used within TOPL to study the effects of queue limits, as well as queue expansions, on the controller performance gains.

Chapter 7

Predictive control of freeway networks under weaving and capacity drop

In the last chapter, we presented an optimal controller based on the Link-Node Cell Transmission Model (LN-CTM), along with an efficient methodology for solving the actual non-linear optimization program presented by the optimal control problem. In this chapter, we extend the results when capacity drop and weaving are present.

Capacity drop denotes the reduction in the (maximum) flow throughput of a freeway section when traffic density at the section increases beyond a known threshold. Capacity drop is sometimes observed in locations of geometric discontinuities like lane drops along the freeways. First order models, like the Cell Transmission Models, do not model capacity drop. On the other hand, second order models (for example, METANET [52]), are shown to exhibit a drop in capacity in bottleneck locations. Even though capacity drop is not a universal phenomenon, the ability to include the capacity drop in the model used within an optimal controller formulation is expected to be useful for capturing additional performance improvements in relevant situations. While optimal control formulations based on second order models [29, 20, 25, 4] are useful in this regard, the lack of efficient solutions are definitely a drawback. The optimal control formulation presented in this chapter will use a modified LN-CTM model, which includes the capacity drop, as the underlying traffic model.

Another interesting feature that is not captured in the original LN-CTM model is traffic weaving. Weaving is usually observed when two traffic streams cross each other, leading to frequent lane change maneuvers. Weaving is usually accompanied by a reduction of the operational capacity of the freeway section where it is observed. In this chapter, we introduce a simple model to capture weaving/lane changing effects near the entrance of on-ramps as well as in links preceding the off-ramps. This model will be used within the predictive control formulation presented in this chapter.

7.1 Modeling capacity drop and weaving

The Link-Node Cell Transmission Model presented in this section contains two additions to the original model : (a) Ramp weaving and (b) Discontinuous capacity drop models. We will reuse all the notation followed in Chapters 3,6. Additional terms specific to this chapter are given in Table 7.1

Symbol	Name	Unit
η_i^r	Weaving coefficient for flows entering from on-ramp *i*	dimensionless
η_i^s	Weaving coefficient for flows exiting through off-ramp *i*	dimensionless
$\overline{F_i}$	Reduced flow capacity of link *i*	veh/period
n_i^{cd}	Density beyond which capacity drop is observed in link *i*	veh/section

Table 7.1: Model variables and parameters.

Ramp weaving

Weaving in freeways can occur at on-ramp merge locations as well as off-ramp diverge locations. Weaving at on-ramp merges occur near the on-ramp junctions when vehicles entering the freeway from the ramp execute lane change maneuvers to merge with the freeway traffic. In contrast, weaving near off-ramp diverges actually occur in the link preceding the off-ramp, as vehicles change lanes to leave the freeway. More complicated weaving behaviors can be seen in locations where large freeways merge/diverge. For example, the MacArthur Maze [73], experiences intense weaving during the commute periods. We present a simple model to capture on-ramp/off-ramp weaving behavior, and this might not be applicable to complicated situations like the MacArthur Maze.

During the lane changing operations, vehicles occupy multiple lanes and impact the operational capacity of a freeway section. This can be captured by modifying the demand function to reflect additional space occupied by the vehicles changing lanes. We define $\eta_i^r \geq 1$ and $\eta_i^s \geq 1$ to denote the weaving factor for on-ramp *i* and off-ramp *i* respectively. These variables capture the intensity of lane change behavior exhibited by the vehicles as they enter/exit the freeway. Under nominal conditions (i.e. under the absence of weaving), these factors are equal to 1.

For on-ramp weaving, the modified demand function and the ramp flow are given by

$$\begin{aligned} d_i(k) &= \eta_i^r \min(l_i(k), r_i^c(k)) \\ r_i(k) &= \frac{d_i(k)}{\eta_i^r} \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \end{aligned} \tag{7.1}$$

$R_i(k), S_{i+1}(k)$ are defined as before as

$$\begin{aligned} R_i(k) &= D_i(k)(1-\beta_i(k)) + d_i(k), \\ S_{i+1}(k) &= \min(W_{i+1}(n_{i+1}^J - n_{i+1}(k)), F_{i+1}) \end{aligned} \tag{7.2}$$

We see that the modified demand function is magnified by a factor of η_i compared to the nominal model.The supply $S_{i+1}(k)$ (i.e. the amount of space available in the downstream link into which the ramp flows are destined) is the same as compared to the nominal model. On the other hand, the demand function $R_i(k)$ increases by an amount $d_i(k)(\eta_i^r - 1)$, which is proportional to the ramp demand. The total flow entering Link $i+1$ in the presence of weaving is given by

$$\begin{aligned} f_i(k)(1-\beta_i(k)) + r_i(k) &= (D_i(k)(1-\beta_i(k)) + \min(l_i(k), r_i^c(k))) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \\ &= \begin{cases} (D_i(k)(1-\beta_i(k)) + \min(l_i(k), r_i^c(k))) & \text{if } R_i(k) \le S_{i+1}(k) \\ S_{i+1}(k)\frac{D_i(k)(1-\beta_i(k))+\min(l_i(k), r_i^c(k))}{R_i(k)} = S_{i+1}(k) - (\eta_i^r - 1)r_i(k) & otherwise \end{cases} \end{aligned}$$

These equations show the impact of on-ramp weaving behavior on flow entering link $i+1$. We now show that for the same available supply, the total flow entering the downstream link $i+1$ can be lower when weaving is present. Defining the nominal demand (when weaving is absent) as $\bar{R}_i(k) = D_i(k)(1-\beta_i(k)) + \min(l_i(k), r_i^c(k))$, and noting that $R_i(k) > \bar{R}_i(k)$

$$\begin{aligned} f_i(k)(1-\beta_i(k)) + r_i(k) &= (D_i(k)(1-\beta_i(k)) + \min(l_i(k), r_i^c(k))) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \\ &= \begin{cases} \bar{R}_i(k)\frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} = \bar{R}_i(k)\frac{\min(\bar{R}_i(k), S_{i+1}(k))}{\bar{R}_i(k)} & \text{if } R_i(k) \le S_{i+1}(k) \\ \bar{R}_i(k)\frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} < \bar{R}_i(k)\frac{\min(\bar{R}_i(k), S_{i+1}(k))}{\bar{R}_i(k)} & otherwise \end{cases} \end{aligned}$$

where the total flow entering link $i+1$ when weaving is absent is given by $\bar{R}_i(k)\frac{\min(\bar{R}_i(k), S_{i+1}(k))}{\bar{R}_i(k)}$.

We can also show that weaving decreases the operational capacity of the section. When the downstream densities are under-critical, which leads to $W_{i+1}(n_{i+1}^J - n_{i+1}(k)) > F_{i+1}$, and under sufficiently high demands, $(D_i(k)(1-\beta_i(k)) + \min(l_i(k), r_i^c(k))) > S_{i+1}(k) = F_{i+1}$, the total flow entering link $i+1$ is given by

$$F_{i+1} - (\eta_i^r - 1)r_i(k) < F_{i+1} \qquad when \quad \eta_i^r > 1,\ r_i(k) > 0$$

When ramp weaving factor $\eta_i^r = 1$, the maximum downstream flow that can be sustained is F_{i+1}, while weaving decreases the flow of the downstream section. The flow reduction also increases as higher flows merge onto the freeway.

Finally, we also see that $f_i(k)(1-\beta_i(k)) + \eta_i^r r_i(k) = S_{i+1}(k)$, when the node exhibits congested conditions (i.e. $R_i(k) > S_{i+1}(k)$). In this case, when we interpret $S_{i+1}(k)$ as the total space available downstream, we can see that space occupied by the traffic merging from the on-ramp is inflated by a factor η_i^r. This reflects the increased space occupied by vehicles as they change lanes to merge into the freeway traffic.

Traffic weaving can also be observed as vehicles change lanes to exit an off-ramp. We assume that off-ramp weaving occurs in the freeway link that precedes the off-ramp diverge location. $\eta_i^s \geq 1$, captures the additional space occupied by the weaving traffic exiting the off-ramp. The effect of off-ramp weaving corresponding to off-ramp i can be captured by modifying the demand function of link i. Under weaving, the new capacity of link i is given by

$$\frac{F_i}{1+(\eta_i^s - 1)\beta_i(k)}$$

and the new demand function is given by

$$D_i(k) = \min\left(n_i(k)v_i(k), \frac{F_i}{1+(\eta_i^s - 1)\beta_i(k)} \right) \tag{7.3}$$

To understand the effect of off-ramp weaving, we note that the total flow exiting the link $f_i(k)$ is composed of the flow exiting through the off-ramp ($f_i(k)\beta_i(k)$) and the flow continuing onto the next freeway link ($f_i(k)(1-\beta_i(k))$). Weaving near off-ramp diverges leads to a decrease in the effective capacity of the input link, as the traffic exiting the off-ramp change lanes and occupy additional space. Interpreting η_i^s as the inflation factor that captures the space occupied by the weaving traffic, the capacity of the section imposes a restriction of total flow that can exit link i, given by $f_i(k)(1-\beta_i(k)) + f_i(k)\beta_i(k)\eta_i^s \leq F_i$. This is equivalent to replacing the capacity of the section by an effective weaving capacity $\frac{F_i}{1+(\eta_i^s-1)\beta_i(k)}$.

The simple model presented above captures the main features expected when traffic weaving is observed, i.e. the reduction in operational capacity of a section. Moreover, the reduction in operational capacity increases with the increase of flow in the traffic streams that contribute to the weaving. This model differs from the model presented by Jin [28], which assumes a space dependent weaving coefficient. The author considers a small stretch of road with a space and time dependent weaving factor. The weaving coefficient contributes to an increase in perceived density, and the author also modifies the demand and supply functions, defined through a nominal fundamental diagram, using a weaving factor. A simple model is provided to calculate the weaving coefficients as a function of ramp and mainline densities, when constant ramp flows are assumed. The model cannot be easily extended to situations when ramp flows are varying, due to presence of ramp controllers. TOPL simulator also incorporates a density based weaving model. For on-ramp weaving, this model keeps track of downstream densities contributed by the different traffic streams, and modifies the demand and supply function depending on the amount of weaving traffic (a reference for this model is not currently available, but it is expected to be posted in [67] when ready). While this model can keep track of the effect of merging traffic, the presence of additional state variables pose additional difficulties when used within an optimal controller. In contrast, the model presented here is simple, and can be easily integrated into the solution scheme presented in the previous chapter, as we will show here.

Capacity drop model

In any freeway section, congestion originates at bottlenecks and propagates upstream. A passive bottleneck exists when the capacity of the link upstream exceeds the capacity of a link downstream. Natural bottlenecks can occur due to lane drops, ramp merges and also less typically in graded locations and turns. Bottlenecks are said to be activated when the demand feeding into the bottleneck section (link) exceeds the capacity downstream. As a result, vehicle buildup occurs in the link before the bottleneck and congestion propagates upstream.

The flow exiting a bottleneck in the presence of a vehicle queue upstream is equal to the flow capacity. In normal sections, this flow capacity is given by the flow corresponding to the apex of the fundamental diagram. In some locations, bottlenecks are characterized by a drop in capacity, as vehicle queues build up, leading to higher densities in the bottleneck locations. Various researchers [24, 6], have observed capacity drop at bottleneck locations. The empirical relationship between capacity drop and the vehicle density at a bottleneck location has been investigated in [10].

The LN-CTM model is modified to simulate the capacity drop by incorporating a discontinuous link demand function, defined as

$$\bar{D}_i(k) = \begin{cases} \min(n_i(k)V_i, F_i) & \text{if } n_i(k) \leq n_i^{cd}, \\ \bar{F_i} & \text{if } n_i(k) > n_i^{cd}. \end{cases}$$

where n_i^{cd} is the density above which capacity drop occurs and $\bar{F_i} < F_i$. The flow out of any link is the minimum of its demand, and the supply imposed by the downstream link. Hence, to derive the effective capacity drop, one needs to consider the capacity imposed by the downstream supply, since the effective capacity of any junction is given by $\min(F_i, F_{i+1})$. In case the current link (link i) and the next downstream link (link $i+1$) do not have ramps in between, the effective capacity drop (for link i) is defined as $\min(F_i, F_{i+1}) - \min(\bar{F_i}, F_{i+1})$, which is different from $F_i - \bar{F_i}$. Clearly, even with a discontinuous demand function, unless $F_{i+1} > \bar{F_i}$, the link will not experience capacity drop. Figure 7.1 shows an example of a section with a discontinuous demand function. In this figure, we illustrate that capacity drop occurs at the density corresponding to the apex of the fundamental diagram, i.e. the critical density n_i^c. In general, the density used for the capacity drop can be located beyond this value, and the definition naturally extends in the case of a trapezoidal fundamental diagram. Finally, in case speed control is applied, the demand function is given by $D_i(n_i(k)) = \min(n_i(k)v_i(k), \bar{D}_i(n_i(k)))$.

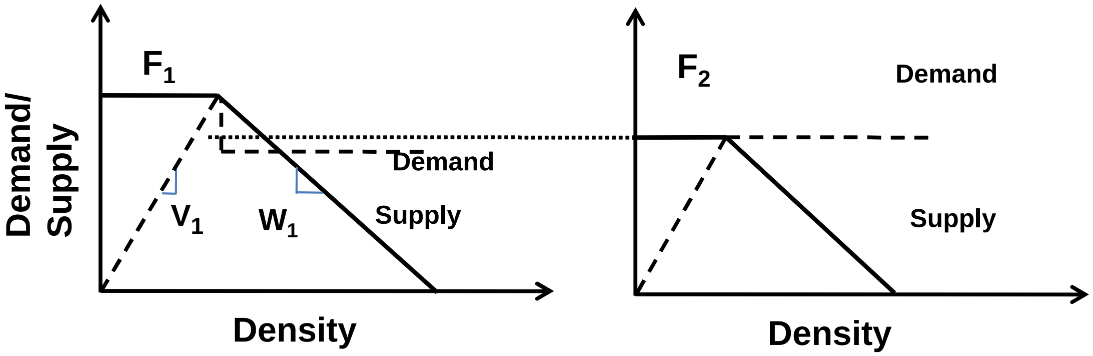

Figure 7.1: Demand (dashed line) and Supply (solid line) functions of two consecutive sections. The first section (left) experiences a capacity drop

Complete model

Let I denote the indices of all sections (links) considered, while I_d denote the freeway sections where discontinuous capacity model is used. Capacity drop is usually observed in a subset of locations corresponding to the locations of recurrent bottlenecks of the traffic system. The density and flow update equations are given by

Mainline/Queue Conservation Equation

$$n_0(k+1) = n_0(k) + Q_0(k) - f_0(k)$$

$$n_i(k+1) = n_i(k) + f_{i-1}(k)(1-\beta_{i-1}(k)) + r_{i-1}(k) - f_i(k)$$

$$l_i(k+1) = l_i(k) + Q_i(k) - r_i(k) \quad i = 1, \cdots, N \tag{7.4}$$

Flow Equations

$$f_N(k) = D_n(k)$$

$$f_i(k) = D_i(k) \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)}$$

$$r_i(k) = \frac{d_i(k)}{\eta_i^r} \times \frac{\min(R_i(k), S_{i+1}(k))}{R_i(k)} \quad i = 1, \cdots, N$$

where

$$D_i(k) = \min(n_i(k)v_i(k), \tilde{F}_i(k)), \forall i \in I \setminus I_d$$

$$
\begin{aligned}
D_i(k) &= \begin{cases} \min(n_i(k)v_i(k), \tilde{F}_i(k)) & \text{if } n_i(k) \leq \max\left(n_i^c, \frac{\bar{F_i}}{v_i(k)(1+(\eta_i^s-1)\beta_i(k))}\right) \\ \frac{\bar{F_i}}{1+(\eta_i^s-1)\beta_i(k)} & otherwise \end{cases}, \\
& \qquad \forall i \in I_d \\
\tilde{F_i}(k) &= \frac{F_i}{1+(\eta_i^s-1)\beta_i(k)} \\
R_i(k) &= D_i(k)(1-\beta_i(k)) + d_i(k), \\
S_{i+1}(k) &= \min(W_{i+1}(n_{i+1}^J - n_{i+1}(k)), F_{i+1}) \\
d_i(k) &= \eta_i^r \min\left(c_i(k), l_i(k)\right) \quad i = 0, \cdots, N-1
\end{aligned}
\tag{7.5}
$$

The complete model combines the capacity drop and weaving models presented above. This model will be used in our predictive controller formulation detailed in the next section.

7.2 Optimal controller formulation

We first present an optimal controller formulation along the lines of the problem defined in Section 6.1. The only difference is the traffic model used in the controller formulation. The control constraints, initial conditions and the performance objective remains the same. The optimization problem corresponding to the optimal control formulation is given below.

$$
\begin{aligned}
\min: \quad & J, \text{ given by Eq. (6.3)} \\
S.t. \quad : \quad & For \quad k = 1, \cdots, K \\
& \underline{\text{Conservation equations}} \\
& \text{Equations (7.4)} \\
& \underline{\text{Flow equations}} \\
& \text{Equations (7.5)} \\
& \underline{\text{Constraint equations}} \\
& \text{Equations (6.5), (6.6)}
\end{aligned}
$$

$$n_i(k), l_i(k), f_i(k), r_i(k) \geq 0 \quad \forall i$$

$$\text{initial conditions/fundamental diagram parameters given in List 6.1.} \tag{7.6}$$

This optimal controller with the modified model poses new challenges to the development of an efficient solution methodology. This is due to the presence of a discontinuous capacity drop function in the optimal controller formulation. The relaxation technique presented in the previous chapter cannot be directly adopted to reduce this optimization problem to a linear program. In the next section, we present some good heuristics which help us formulate an efficient predictive controller based on the optimal controller formulation.

7.3 Efficient predictive controller

The formulation in the previous chapter relied on absorbing the speed limit/ramp demand variables, thereby relaxing the flow constraints. Under the new modified demand function, when the same techniques are applied, the constraints corresponding to the flows in the links with the discontinuous capacity drop are no longer linear or even convex. In contrast, the weaving model can be directly integrated without additional difficulties into the control specification presented in the previous chapter.

To develop a computationally efficient controller, we employ a divide and conquer approach. As noted in [21], given a set of (stationary) ramp demands, the freeway can be divided into regions, with each region consisting of multiple sections/links. In this setup, the first link of each region is in free-flow, while the most downstream section of a region acts as a bottleneck. These bottleneck regions are accompanied by congested conditions upstream while the downstream is in free-flow. Therefore, the bottleneck discharge flows at its maximum flow capacity. Under time-varying demands, we can expect that these bottleneck regions could possibly change as bottleneck regions merge and new bottleneck regions are created. While in theory, every latent bottleneck can be triggered by available demands, a few of these bottlenecks are recurrent. If we observe traffic contours over multiple days, we usually find that a small number of these bottleneck locations are triggered frequently (for example, [8] presents an automatic bottleneck identification algorithm), even under the presence of time varying demands. The presence of capacity drop generally creates a recurrent bottleneck. For example, locations with lane drops with sufficient demand acts as a natural recurrent bottleneck.

To employ our approach, we separate the freeway into regions, with each region consisting of only one bottleneck with a modified demand function at its most downstream link. Note that inside each region, we may have multiple latent/active bottlenecks as long as they do not experience capacity drops. A controller based on an optimal control framework will be described for each region, where the controller will prescribe ramp metering rates and speed limits for all links belonging to the section. The complete control of the entire freeway will be managed by independent controllers that act on each region. In this section, we will describe the process of solving the optimal control problem for each of these sections. Without loss of generality, we state the following assumption.

Assumption 7.3.1. *The freeway section considered has only one bottleneck described using the modified demand function. This bottleneck with the capacity drop will be located in the most downstream section, ie. link N.*

It is expected that the bottleneck locations usually experience free-flow conditions downstream. However, congestion from another downstream bottleneck location can also impact the downstream boundary. First, we will develop our predictive controller under the assumption that the location downstream of the bottleneck is in free-flow. Later, the controller will be modified to account for congestion downstream.

We now define three optimal control problems. **Problem P** states the original non-linear optimal control problem for our individual region, and **Problem Q** poses additional restrictions on the optimal trajectory and reduces the problem to a mixed integer program. Finally, **Problem R** solves **Problem Q** through a sequence of relaxed linear programs.

Problem P Original Problem

$$\begin{aligned}
\min: &\quad J, \text{ given by Eq. (6.3)} \\
S.t. \quad : &\quad For \quad k = 1, \cdots, K \\
&\quad \underline{\text{Conservation Equations}} \\
&\quad \text{Equations (7.4)} \\
&\quad \underline{\text{Flow equations}} \\
&\quad \text{Equations (7.5)} \\
&\quad \underline{\text{Constraint equations}} \\
&\quad 0 \le v_i(k) \le V_i
\end{aligned}$$

$$
\begin{aligned}
&0 \leq d_i(k) \leq \min(C_i, l_i(k)) \\
&l_i(k) \leq L_i \\
&n_i(k), l_i(k), f_i(k), r_i(k) \geq 0 \quad i = 1 \cdots N \\
&\text{with given initial conditions/parameters.}
\end{aligned} \tag{7.7}
$$

As in the previous controller formulations, the optimal controller regulates freeway traffic using the ramp metering rates $r_i^c(k)$ for each ramp and the speed limit profiles $v_i(k)$ for each link. In **Problem P**, we absorb the ramp metering variables ($r_i^c(k)$) and retrieve it after solving the problem, as $r_i^c(k) = d_i(k)/\eta_i^r$. In addition, we can also introduce a new variable $\mu(k) \in \{0,1\}$, to capture the "mode" of the final link. This "mode" can either correspond to free-flow or capacity drop, depending on whether link density is less/greater than n_i^{cd}. We can add the new variable to **Problem P** to convert it into a mixed integer program. The constraints that replace the modified demand function are

$$
\begin{aligned}
&D_N(k) = \min(n_i(k)v_i(k), F_i + (\overline{F_i} - F_i)\mu(k)) \\
&n_i(k) \leq n_i^{cd}(1 - \mu(k)) + n_i^J \mu(k) \\
&n_i(k) \geq n_i^{cd} \mu(k) \\
&\mu(k) \in \{0,1\}
\end{aligned} \tag{7.8}
$$

We can see that $\mu(k) = 0 \Leftrightarrow n_i(k) \leq n_N^{cd}$, and $\mu(k) = 1 \Leftrightarrow n_i(k) \geq n_N^{cd}$. Under these new constraints, the demand function can either take on the value F_i or F_i when the density exactly equals n_{cd}^i. However, in our definition of the demand function with the capacity drop, we assumed a discontinuity at this operation point, and defined the demand function to take on values F_i at this density. Even though the constraints are not an exact representation of this discontinuous demand function, the solution of the optimal control problem will force the demand function to take the value F_i at $n_i(k) = n_i^{cd}$. An intuitive explanation for this fact is to realize that we can decrease the performance objective by maximizing the output flow at the final link.

The addition of these new constraints instead of the discontinuous demand function does not provide any computational advantage in solving the optimal control problem. In **Problem Q**, we replace the above mixed integer constraint for the modified demand function, and also make the following assumption.

Assumption 7.3.2. *For the freeway section considered, we restrict the system evolution such that once the downstream link switches to the "free-flow" mode, it remains in the free-flow mode.*

This heuristic restriction is expected to produce an optimal cost almost similar to **Problem P**, since the free-flow mode is more efficient as it allows vehicles to exit the region at a much faster rate. Hence, once the system switches into the free-flow mode, it is usually optimal for the controller to maintain this mode for maximum throughput. Using this assumption we get,

Problem Q Modified Problem

$$\begin{aligned}
\min: \quad & J, \text{ given by Eq. (6.3)} \\
S.t. \quad : \quad & For \quad k = 1, \cdots, K \\
& \underline{\text{Conservation equations}} \\
& \text{Equations (7.4)} \\
& \underline{\text{Flow equations}} \\
& \text{Equations (7.5) with (7.8) replacing the modified demand function} \\
& \underline{\text{Constraint equations}} \\
& 0 \leq v_i(k) \leq V_i \\
& 0 \leq d_i(k) \leq \eta_i^r \min(C_i, l_i(k)) \\
& l_i(k) \leq L_i \\
& n_i(k), l_i(k), f_i(k), r_i(k) \geq 0 \\
& \mu(k) \geq \mu(k+1) \; k = 1, \cdots, K-1 \\
& \mu(k) \in \{0, 1\} \; k = 1, \cdots, K \\
& \text{with given initial conditions/parameters.} \qquad (7.9)
\end{aligned}$$

In the above formulation, the constraints $\mu(k) \geq \mu(k+1)$ $k = 1, \cdots, K-1$ is equivalent to $\exists j \in \{1 \cdots K\}$ s.t. $\mu(k) = 1$, $k = 1, \cdots, j$ and $\mu(k) = 0$, $k = j+1, \cdots, K$. This interpretation is used to formulate **Problem R**. For a given j, we can formulate an equivalent linear program by relaxing the flow constraints, as presented in the previous chapter. We convert the non-linear equality constraints in the flow equations to a set of linear inequality constraints, by removing the variables $v_i(k)$ and $d_i(k)$ from the formulation. The final optimal control problem solves a linear program for each j and computes the minimum cost and corresponding control action.

Problem R Final Problem

$$\min_{j=0..K} : \quad J_j^*,$$

where

$$J_j^* \;=\; \min \quad J, \text{ given by Eq. (6.3)}$$

$$S.t. \quad For \quad k = 1, \cdots, K$$

<u>Conservation equations</u>

Equations (6.1)

<u>Relaxed flow equations</u>

$$\bar{f}_i(k) \le \bar{n}_i(k) V_i \quad i = 1, \cdots, N$$

$$\bar{f}_i(k)(1 + (\eta_i^s - 1) * \beta_i(k)) \le F_i \quad i = 1, \cdots, N$$

$$\bar{f}_i(k)(1 - \beta_i(k)) + \eta_i^r \bar{r}_i(k) \le F_{i+1} \quad i = 1, \cdots, N-1$$

$$\bar{f}_i(k)(1 - \beta_i(k)) + \eta_i^r \bar{r}_i(k) \le W_{i+1}(n_{i+1}^J - \bar{n}_{i+1}(k)) \quad i = 1, \cdots, N-1$$

<u>Constraint equations</u>

$$0 \le \bar{r}_i(k) \le \min(C_i, \bar{l}_i(k)) \quad i = 1, \cdots, N$$

$$\bar{l}_i(k) \le L_i$$

$$For \quad k = 1, \cdots, j$$

$$\bar{n}_N(k) \ge n_N^{cd}$$

$$\bar{f}_N(k)(1 + (\eta_N^s - 1) * \beta_N(k)) \le \bar{F}_i$$

$$For \quad k = j+1, \cdots, K$$

$$\bar{n}_N(k) \le n_N^{cd}$$

$$n_i(k), l_i(k), f_i(k), r_i(k) \ge 0 \quad i = 1 \cdots N$$

with the same initial conditions/parameters. (7.10)

Consistent with the notation used in the previous chapter, we have chosen to use an upper bar to denote the optimization variables in each subproblem of **Problem R** (e.g. $\bar{n}_i(k)$, $\bar{f}_i(k)$, $\bar{r}_i(k)$). Each subproblem of **Problem R** is a linear program. The j^{th} subproblem captures the situation when the system is in the capacity drop mode for the first j time instants and thereafter switches to the free-flow mode. Let $j^* = \underset{j=0..K}{\operatorname{argmin}} \quad J_j^*$, denote the subproblem that produces the optimal cost. We denote the corresponding optimal trajectory as $\bar{n}_i^*(k), \bar{f}_i^*(k), \bar{l}_i^*(k), \bar{r}_i^*(k)$. Along the lines of **Algorithm A**, we outline the methodology to extract ramp metering rates and speed limit profiles, along with the equivalent system trajectory corresponding to **Problem Q** using **Algorithm B**. Let $n_i^*(k), f_i^*(k), l_i^*(k), r_i^*(k), v_i^*(k), d_i^*(k)$ represent the trajectory corresponding to **Problem Q** from **Algorithm B** given below.

Algorithm B

```
For each time period $k$ and link $0 \le i \le N$,
$n_i^*(k) = \bar{n}_i^*(k)$
$f_i^*(k) = \bar{f}_i^*(k)$
$l_i^*(k) = \bar{l}_i^*(k)$
$r_i^*(k) = \bar{r}_i^*(k)$
$\tilde{F}_i^*(k) = \frac{F_i}{1 + (\eta_i^s - 1)\beta_i(k)}$
For each time period $k$ and link $0 \le i < N - 1$,
$if \quad f_i^*(k) = \min(n_i^*(k)V_i, \tilde{F}_i^*(k))$
  $v_i^*(k) = V_i$
  $d_i^*(k) = \eta_i^r r_i^*(k)$
$else\,if \quad f_i^*(k)(1 - \beta_i(k)) + \eta_i^r r_i^*(k) < S_{i+1}^*(k)$
    $d_i^*(k) = r_i^*(k)\eta_i^r$
    $v_i^*(k) = f_i^*(k)/n_i^*(k)$
```

$$\begin{aligned}
&else\,if \quad \frac{r_i^*(k)\eta_i^r}{S_{i+1}(k)} \leq \frac{\min\left(C_i, l_i(k)\right)\eta_i^r}{\min(n_i^*(k)V_i, \tilde{F}_i^*(k))(1-\beta_i(k)) + \eta_i^r \min\left(C_i, l_i(k)\right)} \\
&\qquad v_i^*(k) = V_i \\
&\qquad d_i^*(k) = \eta_i^r r_i^*(k) \times \frac{\min(n_i^*(k)V_i, \tilde{F}_i^*(k))(1-\beta_i(k))}{S_{i+1}^*(k) - \eta_i^r r_i^*(k)} \\
&else \\
&\qquad v_i^*(k) = \frac{\eta_i^r \min\left(C_i, l_i(k)\right)}{n_i^*(k)(1-\beta_i(k))} \times \left(\frac{S_{i+1}^*(k)}{\eta_i^r r_i^*(k)} - 1\right) \\
&\qquad d_i^*(k) = \eta_i^r \min\left(C_i, l_i(k)\right)
\end{aligned} \tag{7.11}$$

$$where \quad S_i^*(k) = \min\left(W_i(n_i^J - \overline{n}_i^*(k)), F_i(k)\right)$$

and for each time period k

$$\tilde{F}_N(k) = \begin{cases} \frac{F_N}{(1+(\eta_N^s - \underline{1})*\beta_N(k))} & \text{if } n_N(k) \leq n_N^{cd} \\ \frac{\bar{F}_N}{(1+(\eta_N^s - 1)*\beta_N(k))} & \text{otherwise} \end{cases},$$

$$\begin{aligned}
&i\,f\;\; f_N^*(k) = \min(n_N^*(k)V_N, \tilde{F}_N) \\
&\qquad v_N^*(k) = V_N \\
&else \\
&\qquad v_N^*(k) = f_N^*(k)/n_N^*(k)
\end{aligned} \tag{7.12}$$

Algorithm B is very similar to **Algorithm A**, with some additional modifications added to account for ramp weaving factors and the capacity drop in the final link.

Theorem 7.3.1. *Let $R = \{\overline{n}_i^*(k), \overline{f}_i^*(k), \overline{l}_i^*(k), \overline{r}_i^*(k)\}$ be an optimal solution of **Problem R** and $Q = \{n_i^*(k), f_i^*(k), l_i^*(k), r_i^*(k), v_i^*(k), d_i^*(k)\}$ be the solution derived using **Algorithm B**. Then Q is an optimal solution for **Problem Q**.*

Proof. The feasible sets of **Problem Q** and **Problem R** are equivalent, as each subproblem of **Problem Q** is mapped on to a feasible realization of $\mu(k)$ $k = 1, \cdots, K$, by construction and any feasible $\mu(k)$ corresponds to one of the subproblems of the linear program. We can easily show that **Algorithm B** maps a feasible solution of **Problem R** to a feasible solution of **Problem Q**, along the lines of the results shown in Chapter 6. Since the feasible sets, and the objective functions are identical, we can show that Q, obtained from R is optimal for **Problem Q**, with an argument similar to the one given for Theorem 6.2.1. □

Now we consider the effect of boundary conditions downstream of the bottleneck. We make the following assumption.

Assumption 7.3.3. *For the freeway section considered, the downstream boundary condition can be represented using a constant boundary flow restriction F^d.*

The boundary flow restriction means that flows from the most downstream location cannot exceed F^d. When the boundary is in free-flow, we have $F^d = F_N$. However, as the link downstream of our downstream boundary begins to get congested, it restricts the flow that can exit the region and $F^d < F_N$. We assume that the downstream flow restriction is constant, even though the downstream flow restriction will be usually time varying. In fact, the downstream boundary condition is a function of the upstream flows, and is usually indeterminate. However, when the optimal controller is used as a part of a model predictive control strategy, we can use the current downstream flow measurement to provide an estimate of F^d, which we can assume to be constant. This is updated to a better estimate during the next controller update step.

When $F_N \geq F^d > \bar{F}$, we can replace F_N by F^d, and solve the optimal control problem as before. In this case, even when $F_N > F^d$, we recognize that maintaining the first link in the free-flow mode would increase the current section throughput. However, when $\bar{F_N} > F^d$ this is no longer true, since both modes are equally efficient with respect to maximizing discharge from the particular section. In fact, in this case, it is efficient to switch to the congested mode, since this allows the freeway section to store more vehicles in the freeway (due to the increased density prevalent in the capacity drop mode). This will release some of the congestion upstream, leading to large exit flows in the blocked off-ramps. We replace F_N by F^d and solve the following linear program to obtain the optimal solution in this case.

$$\min \quad J, \text{ given by Eq. (6.3)}$$

$$S.t. \quad For \quad k = 1, \cdots, K$$

$$\underline{\text{Conservation equations}}$$

Equations (6.1)

Relaxed flow equations

$$\bar{f}_i(k) \leq \bar{n}_i(k)V_i \quad i = 1, \cdots, N$$

$$\bar{f}_i(k)(1 + (\eta_N^s - 1) * \beta_N(k)) \leq F_i \quad i = 1, \cdots, N$$

$$\bar{f}_i(k)(1 - \beta_i(k)) + \eta_i^r \bar{r}_i(k) \leq F_{i+1} \quad i = 1, \cdots, N-1$$

$$\bar{f}_i(k)(1 - \beta_i(k)) + \eta_i^r \bar{r}_i(k) \leq W_{i+1}(n_{i+1}^J - \bar{n}_{i+1}(k)) \quad i = 1, \cdots, N-1$$

Constraint equations

$$0 \leq \bar{r}_i(k) \leq \min(C_i, \bar{l}_i(k)) \quad i = 1, \cdots, N$$

$$\bar{l}_i(k) \leq L_i$$

$$n_i(k), l_i(k), f_i(k), r_i(k) \geq 0 \quad i = 1 \cdots N$$

with the same initial conditions/parameters. (7.13)

The optimal solution of the above linear program can be used to obtain speed limit profiles and ramp metering rates using **Algorithm B**, with a slight modification. We need to replace the calculations for determining the speed limit profile in the last section by

$$\begin{aligned} & if\ f_N^*(k) = \min\left(n_N^*(k)V_N, \frac{F_N}{(1 + (\eta_N^s - 1) * \beta_N(k))}\right) \\ & \qquad v_N^*(k) = V_N \\ & else \\ & \qquad v_N^*(k) = f_N^*(k)/n_N^*(k) \end{aligned} \tag{7.14}$$

Generally, the optimal controller does not specify a restrictive speed limit profile for the last link in any of the cases mentioned above, as these would increase the delay of vehicles in the freeway region considered.

In our final optimal control problem, we do not explicitly consider traffic speed variables. At any section, we can calculate the speed of the traffic as $v_i^s(k) = f_i(k)/n_i(k)$. One concern is to ensure that the variable speed limits do not lead to sudden changes in speed at a particular section. We adopt an indirect method to limit speed variations at any particular section. Let ΔV_i denote the nominally allowed speed variation within which we would like to operate. Then we add the following constraint

$$-\bar{\zeta}_i - \frac{\Delta V_i}{2}(n_i(k) + n_i(k+1)) \leq f_i(k+1) - f_i(k) \leq \bar{\zeta}_i + \frac{\Delta V_i}{2}(n_i(k) + n_i(k+1))$$

We will also add a penalty term $\bar{C}\sum_{i,k}\bar{\zeta}_i(k)$ to our cost function. The above constraint indirectly limits speed variations at a particular section by limiting flow variations across different time steps.

Characteristics of the solution

In this section, we study the characteristics of the solution of the optimal control problem, in the case of constant split ratios. The presence of capacity drop in the model necessitates the use of variable speed limits. In this section, we will show that these speed limits serve two purposes (a) Throttling of the flow into the link with the capacity drop (b) Facilitating the optimal merging, as well as limiting the queue on the on-ramps.

In order to investigate the properties of the solution, we consider the KKT conditions of the optimal subproblem of **Problem R**. Note that we do not consider any constraints corresponding to speed limit variations described at the end of the previous section. Here we assume that there is at least one subproblem which is feasible. Since we include hard queue constraints, this is not always guaranteed since the demand might be exceedingly high such that no feasible solution exists. One way to relax on this is to include soft queue constraints, and add a corresponding function to the objective penalizing excess queues. Note that the main results derived below do not change even for the problem with soft queue constraints. For clarity we write the constraints and the corresponding dual variables.

$$
\begin{aligned}
\min \quad & J, \text{ given by Eq. (6.3)} \\
& n_i(k+1) = n_i(k) + f_{i-1}(k)(1-\beta_{i-1}(k)) \\
& \qquad + r_{i-1}(k) - f_i(k) && : \eta_i(k+1) \\
& l_i(k+1) = l_i(k) + Q_i(k) - r_i(k) && : \bar{\eta}_i(k+1) \quad i = 1, \cdots, N \\
& n_0(k+1) = n_0(k) + Q_0(k) - f_0(k) && : \ \eta_0(k+1) \\
& \bar{f}_i(k) \le \bar{n}_i(k) V_i && : \nu_i^1(k) \quad i = 1, \cdots, N \\
& \bar{f}_i(k) \le \frac{F_i}{(1 + (\eta_i^s - 1) * \beta_i(k))} && : \nu_i^2(k) \quad i = 1, \cdots, N \\
& \bar{f}_i(k)(1-\beta_i(k)) + \eta_i^r \bar{r}_i(k) \le F_{i+1} && : \nu_i^3(k) \quad i = 1, \cdots, N-1
\end{aligned}
$$

$$\begin{aligned}
&\bar{f}_i(k)(1-\beta_i(k)) + \eta_i^r \bar{r}_i(k) \\
&\qquad \leq W_{i+1}(n_{i+1}^J - \bar{n}_{i+1}(k)) && : \nu_i^4(k) \quad i = 1, \cdots, N-1 \\
&\bar{f}_i(k) \geq 0 && : \nu_i^5(k) \quad i = 1, \cdots, N
\end{aligned}$$

Constraint equations

$$\begin{aligned}
&\bar{r}_i(k) \leq \bar{l}_i(k) && : \bar{\nu}_i^1(k) \quad i = 1, \cdots, N-1 \\
&\bar{r}_i(k) \leq C_i && : \bar{\nu}_i^2(k) \quad i = 1, \cdots, N-1 \\
&\bar{r}_i(k) \geq 0 && : \bar{\nu}_i^3(k) \quad i = 1, \cdots, N-1 \\
&\bar{l}_i(k) \leq L_i && : \bar{\zeta}_i(k) \quad i = 1, \cdots, N-1 \\
&For \quad k = 1, \cdots, j \\
&\bar{n}_N(k) \geq n_N^{cd} && : \zeta_N(k) \\
&\bar{f}_N(k) \leq \frac{\bar{F}_N}{(1 + (\eta_N^s - 1) * \beta_N(k))} && : \nu_N^6(k) \\
&For \quad k = j+1, \cdots, K \\
&\bar{n}_N(k) \leq n_N^{cd} && : \zeta_N(k)
\end{aligned}$$

with the same initial conditions/parameters. (7.15)

The optimal solution satisfies the KKT conditions for the above linear program (assuming that a feasible solution exists). We will present the stationarity condition for the LP given above. The lagrangian is given by

$$
\begin{aligned}
&\sum_{k=1}^{K}\Big[\sum_{i=0}^{N}\Big(n_i(k)+l_i(k)\Big)-\sum_{i=1}^{N}\Big(\alpha_i(k)f_i(k)+\bar{\alpha}_i(k)r_i(k)\Big)\\
&+\sum_{i=0}^{N}\Big((\bar{f}_i(k)-\bar{n}_i(k)V_i)\nu_i^1(k)+(\bar{f}_i(k)-F_i/(1+(\eta_i^s-1)*\beta_i(k)))\nu_i^2(k)+(-\bar{f}_i(k))\nu_i^5(k)\Big)+\\
&\sum_{i=0}^{N-1}\Big((\bar{f}_i(k)(1-\beta_i(k))+\eta_i^r\bar{r}_i(k)-F_{i+1})\nu_i^3(k)+(\bar{f}_i(k)(1-\beta_i(k))+\eta_i^r\bar{r}_i(k)\\
&-W_{i+1}(n_{i+1}^J-\bar{n}_{i+1}(k)))\nu_i^4(k)\Big)+\\
&\sum_{i=1}^{N}\Big((\bar{r}_i(k)-\bar{l}_i(k))\bar{\nu}_i^1(k)+(\bar{r}_i(k)-C_i)\bar{\nu}_i^2(k)+(-\bar{r}_i(k))\bar{\nu}_i^3(k)+(\bar{l}_i(k)-L_i)\bar{\zeta}_i(k)\Big)\Big]+\\
&\sum_{k=0}^{j}\Big((-\bar{n}_N(k)+n_N^{cd})\zeta_N(k)+(\bar{f}_N(k)-\bar{F}_i/(1+(\eta_N^s-1)*\beta_N(k)))\nu_N^6(k)\Big)\\
&+\sum_{k=j+1}^{K}\Big((\bar{n}_N(k)-n_N^{cd})\zeta_N(k)\Big)+\sum_{k=1}^{K}\Big[\sum_{i=0}^{N}\Big((l_i(k+1)-l_i(k)-Q_i(k)+r_i(k))\bar{\eta}_i(k+1)\Big)\Big]\\
&\sum_{k=1}^{K}\Big[\sum_{i=0}^{N}\Big((n_i(k+1)-n_i(k)-f_{i-1}(k)(1-\beta_{i-1}(k))-r_{i-1}(k)+f_i(k))\eta_i(k+1)\Big)\Big]\\
&+\sum_{i=0}^{N}\Big((n_i(1)-n_i^0)\eta_i(1)\Big)+\sum_{i=1}^{N}\Big((l_i(1)-l_i^0)\bar{\eta}_i(1)\Big)
\end{aligned}
$$

From the lagrangian, we get the following stationarity conditions.

$$
\begin{aligned}
&\mathit{For}\ i\ =\ 1\cdots N-1\ \mathit{and}\ k\ =\ 1\cdots K\\
&\quad \bar{\eta}_i(k)=-1+\bar{\eta}_i(k+1)+\bar{\nu}_i^1(k)-\bar{\zeta}_i(k)\\
&\quad \eta_i(k)=-1+\eta_i(k+1)+\nu_i^1(k)V_i-\nu_{i-1}^4(k)W_i\\
&\qquad\quad \mathit{with}\ \bar{\eta}_i(K+1)=0\ \mathit{and}\ \eta_i(K+1)=0\\
&\mathit{For}\ i\ =\ 0\cdots N-1\ \mathit{and}\ k\ =\ 1\cdots K\\
&\quad \alpha-\eta_i(k+1)+\eta_{i+1}(k+1)(1-\beta_i)\\
&\qquad =\nu_i^1(k)+\nu_i^2(k)+\nu_i^3(k)(1-\beta_i)+\nu_i^4(k)(1-\beta_i)-\nu_i^5(k)\\
&\mathit{For}\ i\ =\ 1\cdots N\ \mathit{and}\ k\ =\ 1\cdots K\\
&\quad \bar{\alpha}-\bar{\eta}_i(k+1)+\eta_{i+1}(k+1)=\bar{\nu}_i^1(k)+\bar{\nu}_i^2(k)+\nu_i^3(k)\eta_i^r+\nu_i^4(k)\eta_i^r-\bar{\nu}_i^3(k)
\end{aligned}
$$

$$
\begin{aligned}
&For\ k\ =\ 1\cdots j\\
&\quad \eta_N(k) = -1+\eta_N(k+1)+v_N^1(k)V_N+\zeta_N(k)\\
&\quad \alpha-\eta_N(k+1) = v_N^1(k)+v_N^2(k)-v_N^5(k)+v_N^6(k)\\
&For\ k\ =\ j+1\cdots K-1\\
&\quad \eta_N(k) = -1+\eta_N(k+1)+v_N^1(k)V_N-\zeta_N(k)\\
&\quad \alpha-\eta_N(k+1) = v_i^1(k)+v_i^2(k)-v_i^5(k)
\end{aligned}
$$

The optimal trajectory also satisfies the primary and dual constraints. The dual feasibility constraints are given by

$$
\begin{aligned}
&v_i^1(k),\ldots v_i^6(k) \geq 0\\
&\overline{v}_i^1(k),\ldots \overline{v}_i^3(k) \geq 0\\
&\overline{\zeta}_i(k),\zeta_N(k) \geq 0
\end{aligned}
$$

The usual complementary slackness conditions apply, and they are not explicitly stated here.

Lemma 7.3.1. *When the optimal trajectory satisfies $\eta_i^r\overline{r}_i(k) < \min\left(F_{i+1},W_{i+1}(n_{i+1}^J-\overline{n}_{i+1}(k))\right)$, $\forall$ k and $i = 1\ldots N-2$, we have $\alpha-\eta_i(k)+\eta_i(k+1)(1-\beta_i)\geq 0\ \forall$ k and $i = 1\ldots N-2$ and at least one of $v_i^1(k),v_i^2(k),v_i^3(k),v_i^4(k)$ is strictly positive. Moreover, $v_i^5(k)=0\ \forall\ k$*

Proof. For any $i = 1,\cdots,N-2$, we prove $\alpha-\eta_i(k)+\eta_i(k+1)(1-\beta_i)\geq 0$ by backward induction. Clearly, for $k = K+1$, we have $\alpha-\eta_i(K+1)+\eta_{i+1}(K+1)(1-\beta_i) = \alpha > 0$. Assume that the statement is true for $k+1$, then

$$
\begin{aligned}
&\alpha-\eta_i(k+1)+\eta_{i+1}(k+1)(1-\beta_i)\\
&\quad = v_i^1(k)+v_i^2(k)+v_i^3(k)(1-\beta_i)+v_i^4(k)(1-\beta_i)-v_i^5(k) > 0
\end{aligned}
$$

It can be easily shown that the optimal trajectories satisfy $\overline{n}_i(k) > 0$, given $\overline{n}_i(0) > 0$. When we consider the complimentary slackness conditions, we see that $v_i^1(k) > 0$ or $v_i^2(k) > 0$ implies that $v_i^5(k) = 0$. Also, we are given that $\eta_i^r\overline{r}_i(k) < \min\left(F_{i+1},W_{i+1}(n_{i+1}^J-\overline{n}_{i+1}(k))\right)$ for the optimal trajectory. Using this fact along with the complementary slackness conditions, we also get that $v_i^3(k) > 0$ or $v_i^4(k) > 0$ implies that $v_i^5(k) = 0$. Since $\alpha-\eta_i(k+1)+\eta_{i+1}(k+1)(1-\beta_i)$ is positive, at least one of $v_i^1(k), v_i^2(k), v_i^3(k)$, or $v_i^4(k)$ is non-zero. Thus $v_i^5(k) = 0$. Now,

$$
\begin{aligned}
&\alpha - \eta_i(k) + \eta_{i+1}(k)(1-\beta_i) \\
&\quad = \alpha - \Big[-1 + \eta_i(k+1) + v_i^1(k)V_i - v_{i-1}^4(k)W_i\Big] \\
&\qquad + (1-\beta_i)\Big[-1 + \eta_{i+1}(k+1) + v_{i+1}^1(k)V_{i+1} - v_i^4(k)W_{i+1}\Big] \\
&\quad = [\alpha - \eta_i(k+1) + \eta_{i+1}(k+1)(1-\beta_i)] + 1 - (1-\beta_i) \\
&\quad + v_{i+1}^1(k)V_{i+1}(1-\beta_i) + v_{i-1}^4(k)W_i - v_i^4(k)W_{i+1}(1-\beta_i) - v_i^1(k)V_i \\
&\quad > [\alpha - \eta_i(k+1) + \eta_{i+1}(k+1)(1-\beta_i)] - v_i^4(k)W_{i+1}(1-\beta_i) - v_i^1(k)V_i \\
&\quad > 0 \\
&\text{Since} \quad v_i^4(k)(1-\beta_i) + v_i^1(k) \leq \alpha - \eta_i(k+1) + \eta_{i+1}(k+1)(1-\beta_i)
\end{aligned}
$$

Hence, by induction, we prove the lemma stated above. □

The lemma presented here applies to all links which are not upstream of a link with a modified capacity drop function. The optimal controller specifies a speed limit and a ramp metering rate. For any link discharging its output to a "normal" link (i.e. without a capacity drop), the optimal controller does not throttle the flow by means of a speed limit, i.e.

$$f_i(k) = \min\left(\bar{n}_i(k)V_i, F_i, \frac{\min(F_{i+1}, W_{i+1}(n_{i+1}^J - \bar{n}_{i+1}(k))) - \eta_i^r \bar{r}_i(k)}{1-\beta_i}\right).$$

Particularly, when the next link has no ramp flows, we see that outflow follows the LN-CTM equations with nominal speed limits. However, when the ramp has non-zero flow and the downstream link is congested (i.e. the third term in the equation stated before is active), a speed limit may still be applied to ensure optimal merging. Even in this case, the total inflow into the next link will be the same as the flow in the no speed limit case, but in the case of the optimal trajectory the total outflow may be arbitrarily divided between the ramp and the previous link. In some cases, this will correspond to the application of a speed limit. For example, when ramp flows satisfy the third conditional statements in **Algorithm A/B**, speed limits are not necessary, while speed limits need to be specified when the fourth conditional statement is active. The most common reason for this is that in cases when queue limits are specified, in order to maintain the queue within its limit, the controller will try to assign more preference to the ramp flow, by means of reduced speed limits to the link. Even when queue limits are not specified, the optimal controller may still specify a variable speed limit, as we saw in the previous chapter. In cases of extreme congestion/excessive ramp demands, $\bar{r}_i(k) < W_{i+1}(n_{i+1}^J - \bar{n}_{i+1}(k))$ may not apply and the optimal controller might lead to additional throttling to ensure that sufficient space is available for ramp demand. In the case of on-ramps which are not freeway interconnects, this condition is not violated when $\eta_i^r C_i < W_{i+1}(n_{i+1}^J - \bar{n}_{i+1}(k))$ for the optimal trajectories. During nominal operation, freeways do not generally get very congested as to violate this condition. We also expect the same when the freeway is under the effect of the optimal controller.

The results of this lemma do not apply for the link feeding into a section which experiences capacity drop. Mathematically, this is due of the term $\zeta_N(k)$ in the recursive equation for $\alpha - \eta_{N-1}(k) + \eta_N(k)(1-\beta_{N-1})$. We can see that depending on the sign of this term, the flows feeding into the link with the capacity drop can be zero (i.e. $v_i^5(k) = 0$). We will see an example of this in the next section. The results for this lemma also apply when the optimal solution of the original optimal control problem is considered.

7.4 Simulation examples

We demonstrate the application of a model predictive controller based on the optimal control formulation presented in the previous section. Given the model time step T, the prediction horizon N_p (in units of number of periods), and the control horizon N_c, we can create a model predictive controller that is executed every $T_c = N_c \times T$ time instants. We demonstrate the application of our controller in the presence of weaving and capacity drop.

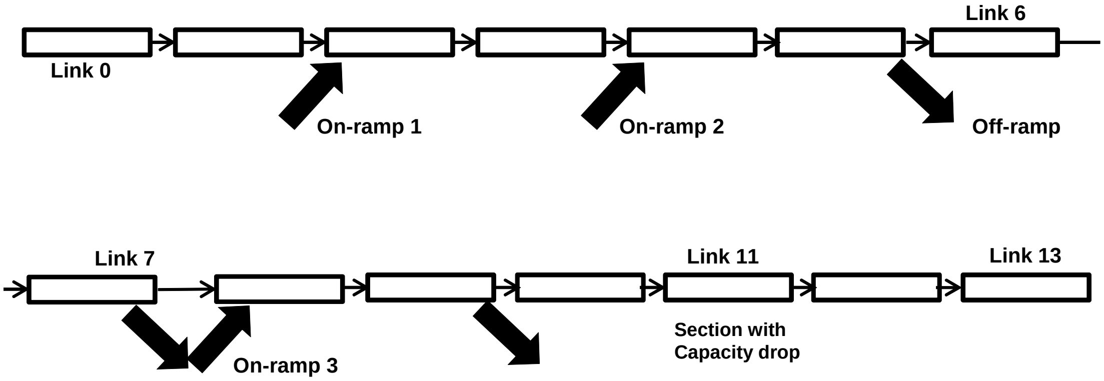

Figure 7.2: Freeway geometry with location of on-ramps and off-ramps.

Figure 7.2 represents the geometry of the freeway section (3.2 mile length) which is considered for our simulation studies. The geometry is artificially constructed to demonstrate the application of our controllers. In this portion of the freeway, link 11 is the only link which experiences a capacity drop. The fundamental diagram parameters are listed in Table 7.2. We can see that the maximum throughput of link 11 is 7600vph in free-flow conditions. This decreases to 7300vph once the density in link 11 exceeds the critical density of 121vpm. This represents a capacity drop of around 4% which is representative of the nominal capacity drop generally reported in literature. We assign our optimal controller to operate on links 0 to 11. The optimal controller can specify variable speed limits for these links, in addition to the ramp metering rates for the on-ramps 1-3. A constant split factor $\beta = 0.15$ is chosen for the three off-ramps during the entire time period considered. For all the on-ramps, we assume a weaving ratio $\eta_r = 1.3$. We do not consider any off-ramp weaving in the examples shown here. Figure 7.3 shows the on-ramp demands we use

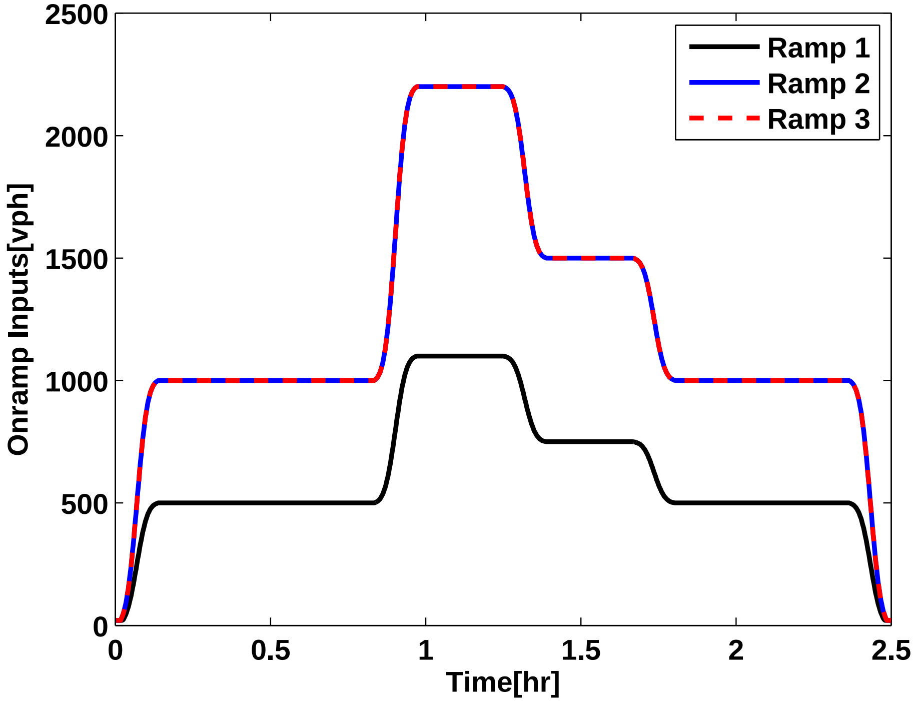

Figure 7.3: On-ramp input flows for all ramps.

for the simulation. The on-ramp demands are chosen such that the freeway is congested between T=1hr to T=2hr. We use a constant flow of 7000vph as the input flow into the first link on the freeway (link 0).

Table 7.2: Link parameters

Links	V	W	F	$\bar{F}$
0,1	65mph	20mph	8500vph	n.a
2,3	65mph	20mph	8900vph	n.a
4-10	65mph	20mph	10500vph	n.a
11	65mph	20mph	7900vph	7300vph
12,13	65mph	20mph	7600vph	n.a

We use the LN-CTM with the capacity drop/weaving model to perform our simulations. In the first simulation, we assume that the boundary downstream of link 13 is in free-flow. Figure 7.4 (top) shows the velocity contours that result when no control action is applied. In this simulation, the freeway starts to get congested around T=1hr. We can see two bottlenecks in the simulation, at link 3 and link 11 respectively. The bottleneck in link 11 is attributed to the capacity drop, while the bottleneck in link 3 is due to the on-ramp merge and weaving (this bottleneck disappears

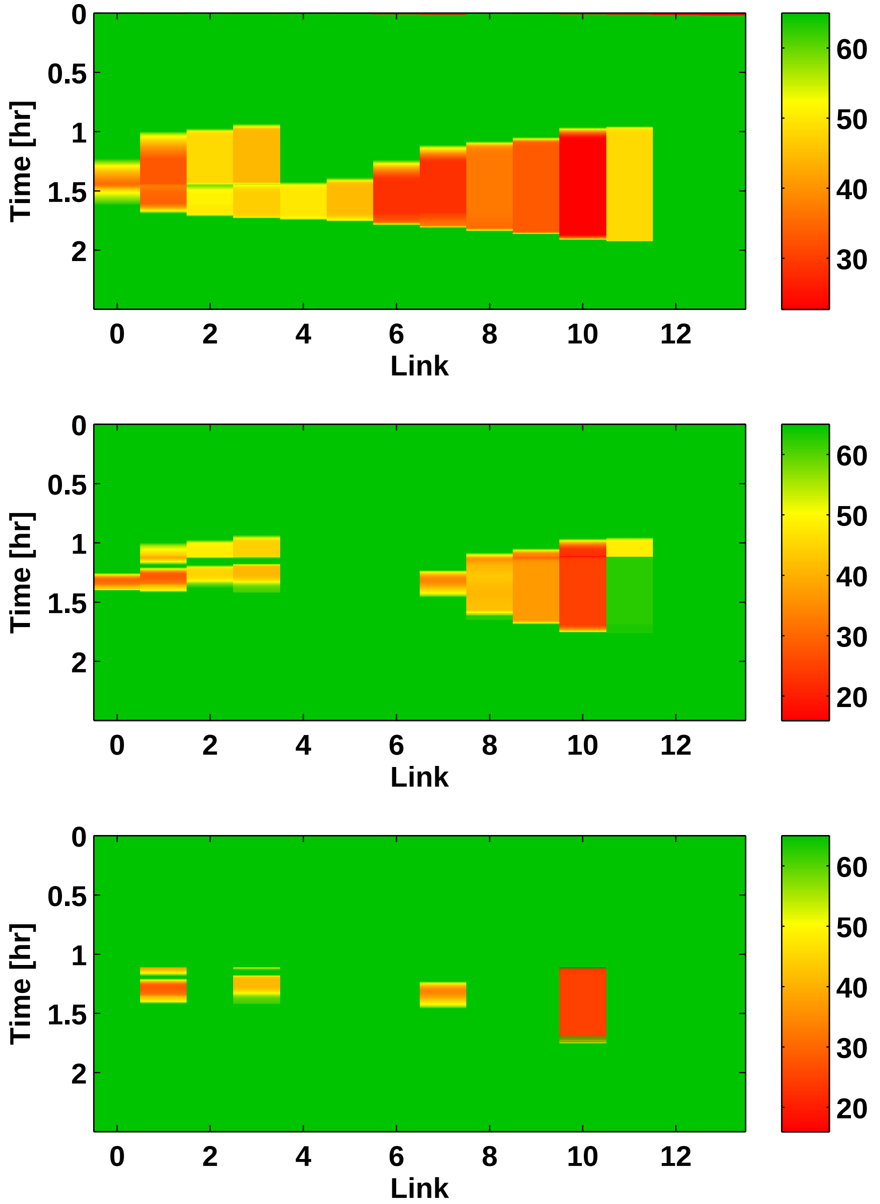

Figure 7.4: Top : Simulated Velocity contours [mph] - no control scenario. Middle : Simulated Velocity contours [mph] - optimal controller. Bottom : Optimal speed limit profile [mph].

when weaving factor equals 1 for the demands in the simulation example). Next, we simulate the freeway, with the model predictive controller specifying the metering rates and speed limits. The controller is initially inactive, and we start applying the controller at T=1.11hr, when link 11 already experiences capacity drop. We choose $N_p = 30$, $T = 10s$, $N_c = 6$ for our controller. The queue limits in both the ramps are chosen to be 75 vehicles. We choose ΔV_i to correspond to 5mph to limit the variations in speed in the link preceding the bottleneck section. In this case, the controller has to solve at most 30 (and in most cases, less) linear programs. Figure 7.4 (middle), shows results of the simulation in which our model predictive controller is used. In this case, the severity and the extent of congestion is reduced. Figure 7.4 (bottom) shows the variable speed limit profiles specified by our optimal controller. The optimal controller specifies a speed limit profile for link 10, and this helps decrease the density of link 11 to below critical density. Thereafter, it still maintains the speed limit which enables link 11 to stay in free-flow. Thus, the optimal controller creates a new bottleneck, through controlled congestion, at link 10 to prevent link 11 from experiencing a drop in capacity. This controlled congestion helps in increasing the throughput of link 11, which limits the extent of congestion, even though it is not completely eliminated. Finally, we see from Figure. 7.5 that the optimal controller maintains the ramp queue limits. The most downstream ramp meter corresponding to on-ramp 3 is used to control the congestion arising out of the bottleneck at link 13. In contrast, the flows in on-ramp 2 are controlled to alleviate the effects of weaving. Decreasing the on-ramp flows (until queue constraints are violated) increases the operational capacity of the section. For every additional vehicle stored in the on-ramp, $\eta_r = 1.3$ vehicles are discharged from the previous section. In cases when the congestion spills back to block other off-ramps, this increase in operational capacity can further help delay congestion. The total travel time and the delay experienced by all users in the no-control scenario are 1358vh (vehicle-hours) and 245vh respectively. In contrast, these reduce to 1256vh and 143vh respectively when the controller is used. This leads to a substantial delay reduction of 41.5%.

In the second simulation, we assume that the boundary downstream of link 13 is initially in free-flow. At 1.4hr, the boundary (link 13) begins to get congested and this congestion propagates back onto link 11 soon after. At 1.6hr, the boundary becomes free-flowing. All other parameters and demands are same as the first simulation. Figure 7.6 (top) shows the velocity contours under the no-control scenario. The congestion is more widespread as compared to the simulation with the optimal controller (Figure 7.6 (middle)). From Figure 7.6 (bottom), we note the speed limit profiles specified by the controller. In this case, we see that the controller specifies a speed limit profile until the boundary congestion reaches the location downstream of link 11. Thereafter, the predictive controller only resumes the speed limit control when the congestion due to the downstream section has dissipated. The total travel time and the delay experienced by all users in the no-control scenario are 1364 (vehicle-hours) and 252vh respectively. In contrast, these reduce to 1284vh and 172vh respectively when the controller is applied. This leads to a delay reduction of 31.8%.

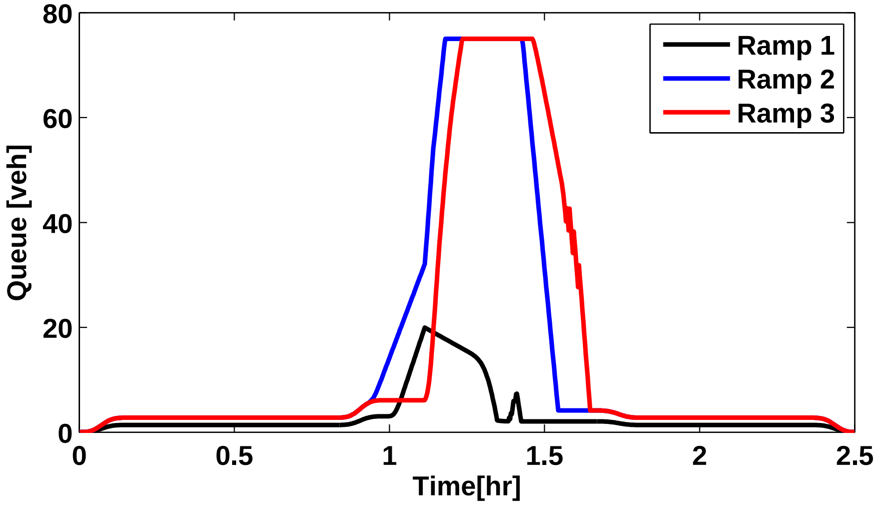

Figure 7.5: On-ramp queues for simulation 1 with optimal controller. Each ramp has a maximum queue size of 75veh.

7.5 Summary

In this chapter, we extended the predictive congestion controller presented in the previous chapter, by augmenting the traffic model used within the controller to include weaving and capacity drop phenomena. The modified model adds additional complexity to the optimization problem corresponding to the actual optimal control problem, and we present various assumptions that are needed to allow us to solve the optimal control problem efficiently. First, we divide the freeway into regions and assume that each region is controlled using an independent controller which controls all sections within that region. One drawback to this approach is that we cannot completely co-ordinate all controller actions, and this might limit the total performance improvements in some cases. When the downstream boundary of each region always remains in free-flow, lack of co-ordination is not expected to limit the controller performance. However, during periods when the region boundary transitions into congestion, better co-ordination can help manage and delay it. Next, we also assume that the optimal trajectory does not switch back from the free-flow mode to the capacity drop mode. When the downstream boundary is always in free-flow, this assumption is generally valid, since the free-flow mode is generally more efficient in discharging traffic out of the region. Finally, we assume that congestion in the downstream boundary can be represented by a constant flow capacity restriction for flows exiting the final link. This provides us a tractable method to calculate an predictive control strategy when the downstream boundary is congested.

All of these assumptions allows us to specify an efficient predictive controller used when sections of the freeway exhibit capacity drop. We highlight that when only weaving is present, these assumptions are not necessary, and we can solve the final optimal control problem exactly. With our approach, the optimization problem corresponding to each predictive controller step can be

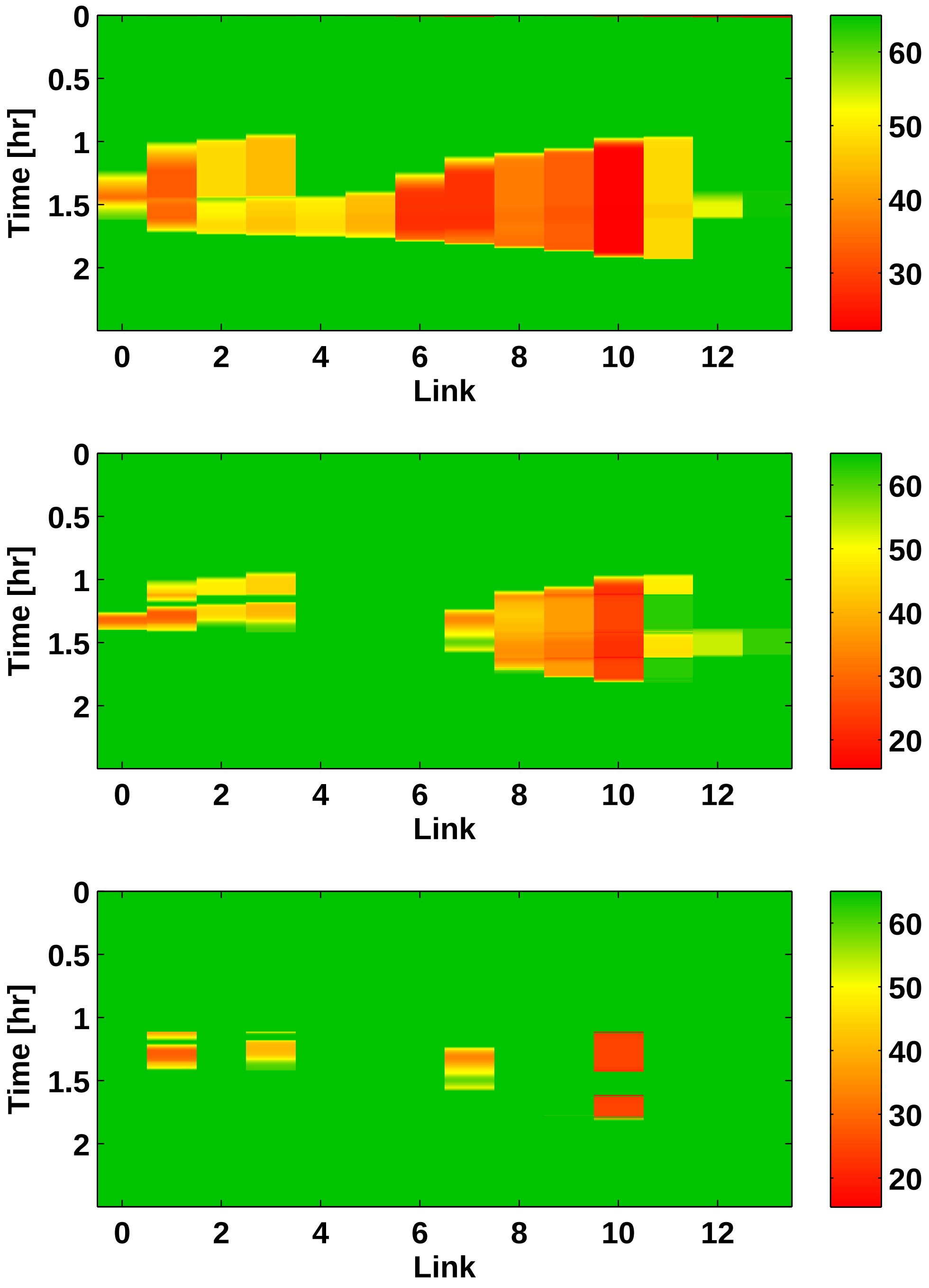

Figure 7.6: Top : Simulated Velocity contours [mph] - no control scenario. Middle : Simulated Velocity contours [mph] - optimal controller. Bottom : Optimal speed limit profile [mph].

solved within 5 seconds for our scenario, when we use the MOSEK linear program solver. This is a fraction of the controller time horizon. We also see that our sequence of linear programs can be solved completely in parallel. One single iteration of the controller can be executed within 10-20s for a realistic sized freeway, even for longer time horizons (though that may not be necessary) like the one shown in previous chapter when we exploit the inherent parallelism.

In this chapter, we also presented some details of the characteristics of the solution. Particularly, we investigated the role of variable speed limits within our controller. Variable speed limits are useful in two cases : (a) to ensure optimal merging in on-ramp junctions and (b) to limit the feeding flows into the section that experiences capacity drop. VSL application corresponding to (a) is generally useful in maintaining queue limits, as we had demonstrated with simulation results in Chapter 6. We expect that the performance gains with the application of VSL in this case may be limited, as shown by our simulations. In contrast, VSL specified in case (b) is necessary to increase the efficiency of the capacity drop section. We expect that an optimal controller that only uses ramp metering to prevent capacity drop might be more inefficient and less robust.

Finally, we presented simulation results where we compared the freeway characteristics with and without the application of our model predictive controller. In both scenarios, our controller leads to a substantial reduction of delay experienced by all travelers in the freeway, even though we only had one section with a modest capacity drop. From our experience of simulating freeway sections with and without capacity drop, we determine that capacity drop is usually the single most important factor that contributes to delay in the freeway, if present. Accounting for capacity drop in ramp metering and variable speed limit controllers, whenever they are present, can help us significantly improve traveler experience on the controlled freeways.

Chapter 8

Conclusions

In this dissertation, we have investigated specific aspects of traffic flow modeling and control of freeway networks. We presented an imputation algorithm, used as a part of a data driven model calibration process to build a model of a chosen freeway section. We also presented an efficient optimal controller, that can be used for congestion control of the given freeway system.

In Chapter 2, we reviewed concepts and previous work related to the development of this dissertation. We discussed the current state of traffic detection, the usually observed detection problems and commonly adopted solutions to impute missing data in freeway mainlines. This was followed by a review of the popular models commonly used to model freeway traffic dynamics. We also presented a detailed account on the control methodologies commonly adopted to combat congestion in freeways, along with a review of previous efforts in the area of model based optimal congestion control. We presented the Link-Node Cell Transmission Model (LN-CTM), a first order model used for simulating traffic dynamics in traffic networks, in Chapter 3. This chapter also discussed the steps taken by a user to create the freeway geometry and automatically calibrate a model. We identified the necessity of imputation of ramp flow data to complete the model creation process. In fact, to completely calibrate the model, we use the imputation algorithm presented in Chapter 5. Finally, we present a model created using the model creation and calibration process.

Chapters 4 and 5 described two imputation algorithms that can be used to estimate the missing ramp flow data. First, we developed a link-wise imputation algorithm based on the Asymmetric Cell Transmission Model (ACTM), along with the proof of convergence of the algorithm. The ACTM, being a simplified piecewise affine model, lent to easy analysis and design of the first model based ramp imputation algorithm. We were able to prove that ramp estimates converged to the actual values (assuming that the freeway dynamics is well approximated by the ACTM) in most cases. Even though in some cases, we might not be able to uniquely identify the on-ramp and off-ramp flows, the errors do not propagate and affect the imputed estimates of the ramp flows in the downstream links. We showed that the imputation algorithm results in zero density and flow errors in the steady state. Thus, a lack of convergence of the density or the flow errors usually imply that some of the measurements supplied may be erroneous/faulty. This forms the basis of fault detection algorithms, that have been developed by other members of the TOPL group for detecting bad detectors [18]. In Chapter 5, we presented an imputation algorithm based on the LN-CTM. The LN-CTM was more accurate for modeling on-ramp merge dynamics especially when the ramp flows are large. This algorithm simultaneously imputed all the ramp flows in two steps, first matching the observed mainline densities before matching the available mainline flows. We also demonstrated the convergence of the algorithm, and discussed various properties of the algorithm. We have observed that the LN-CTM algorithm provides better imputed estimates for simulation, since it simultaneously imputes all the ramp flows together. The presence of noisy/faulty measurements at any interior section does not impact the LN-CTM algorithm like in the case of the ACTM imputation algorithm. Both these algorithms are computationally fast, being able to impute the 24 hour ramp flow profiles for most freeways within 5 minutes.

The imputation algorithm forms an essential piece of the model creation process. With the imputation algorithm in place, under the presence of no adverse/faulty detection, a mildly experienced user can build freeway models for an entire freeway within 1 day. In some cases, faulty detectors present extend the effort to almost a week, if the faults are to be identified and the data discarded manually. A recently developed fault detection algorithm, based on the ACTM, can automate the process to reduce the model creation effort to half the week[18]. Once the initial model geometry along with the faulty detectors are identified, the imputation algorithms can be run autonomously to estimate daily ramp demands and split ratios for multiple days. This allows the user to build models for different days of the week, so that various operational strategies can be tested across multiple days before deployment. In contrast, transportation planners currently use microsimulation models, and their calibration is known to take around 3-6 months of user efforts. Moreover, micro simulation model based studies are usually limited to a single “nominal” day of operation, due to limited time and project budgets.

Once a validated freeway model is available from our calibration procedure, we can apply and test various operational management strategies. One of the popular strategies is the congestion control using ramp metering and variable speed limits. Chapter 6 presents an optimal controller based on the LN-CTM. The controller optimizes a parametric performance objective, which can be chosen to represent two commonly used congestion indicators - the total travel time or total congestion delay of all vehicles using the freeway network. This controller prescribes time varying ramp metering rates for each on-ramp link as well as time varying speed limit profiles for the freeway mainline. We demonstrate that, though the actual optimization problem corresponding to the optimal controller is non-linear and non-convex, we are able to absorb some variables and relax some constraints to present a linear program. We presented an algorithm to convert the optimal control/state trajectory prescribed by the relaxed linear program into an optimal solution of the actual optimal control problem. We also demonstrated a Model Predictive Controller, utilizing this optimal control formulation and established its efficiency in obtaining a solution. The optimal controller, used on a calibrated model, allows us to get an estimate of the best performance benefits that can be obtained by implementing ramp metering and variable speed limits in the field, since we obtain a globally optimal solution to the optimal control problem. Within TOPL, the optimal controller can be used to compare and certify (and possibly tune) other commonly deployed ramp metering controllers. Being computationally efficient, this controller can also be possibly deployed in the field in the future.

In Chapter 7, we extended the predictive control formulation, by modifying the underlying traffic model. The original LN-CTM model does not model the effect of on-ramp/ off-ramp weaving and capacity drop, when they are present. When these effects are significant, inclusion of these models can allow a predictive controller to obtain increased performance gains. We present an augmented LN-CTM model, with additional modifications to capture on-ramp/off-ramp weaving as well as the capacity drop. When this model is used within the optimal controller, it can no longer be efficiently solved using the techniques given in Chapter 6, due to the presence of a discontinuous demand function modeling the capacity drop. Hence, we divide the freeway into regions, with each region containing a capacity drop location at its most downstream section, and assign an independent controller to co-ordinate the ramp metering and variable speed limits in each section. With additional heuristic assumptions, which are not expected to significantly degrade the quality of the solutions, we demonstrate the optimal control problem can be solved using a sequence of linear programs, by using a relaxation technique similar to the one presented in Chapter 6. In this case, the optimal solution can be mapped back using an algorithm similar to the one presented before.

Future Work

There are multiple avenues of future work related to the material developed in this dissertation. First, our model creation procedure and the imputation algorithm should be extended to include weaving and capacity drop. When all ramp measurements are available, it is possible to directly extend the calibration process to include capacity drop and weaving. We also need to validate the weaving model, and ascertain its ability to replicate the traffic flow characteristics in ramp junctions. Multiple sites, with working mainline and ramp detectors, should be used to validate the model, and compare it with the other models discussed before. The extension of the imputation algorithm in the presence of capacity drop might not be an easy task, since the current imputation algorithm takes advantage of the monotonicity of the flow entering and exiting the link. Though the update equations for the ramps adjoining the capacity drop locations might need to be modified, it is expected that the imputation algorithm might remain the same for other ramps.

The LN-CTM imputation algorithm could also be extended to include statistical models of ramp inputs, as a part of the imputation process. As we argued before, even with ramp detector data is available, a purely statistical imputation algorithm is not expected to produce good quality imputed data for our simulation models. However, a hybrid approach, utilizing statistical models of ramp flow data along with the model based imputation scheme would be useful, particularly when some of the mainline measurements are noisy/missing. While the current imputation algorithm could be extended to impute ramp demand estimates in real-time, a hybrid approach holds more promise in providing better quality estimates.

The predictive control methods, which are specified in Chapter 7, define separate controllers for each region. This does not lead to any appreciable drop in performance, as long as the bottleneck downstream remains un-congested. However, in case the bottleneck downstream gets congested during the controller operation, the lack of coordination between the controllers acting in different regions limit the performance improvements that can be obtained from the controllers. One possible extension is to embed these controllers within a hierarchical control scheme, where a meta controller provides commands to coordinate the control actions of these controllers. In this case, we need to identify the set of control actions that can be specified by the meta controller, while also extending the individual region controllers to incorporate these control commands.

Bibliography

[1] Mobile millenium. `http://traffic.berkeley.edu/`, accessed 9/15/2011.

[2] M. Blinkin. Problem of optimal control of traffic flow on highways. *Automat. Control*, 37(3):662–667, 1976.

[3] F. Borrelli, A. Bemporad, and M. Morari. Predictive control. `http://www.mpc.berkeley.edu/mpc-course-material`, accessed 6/1/2012.

[4] Rodrigo C. Carlson, Ioannis Papamichail, Markos Papageorgiou, and Albert Messmer. Optimal mainstream traffic flow control of large-scale motorway networks. *Transportation Research Part C: Emerging Technologies*, 18(2):193 – 212, 2010.

[5] Rodrigo C. Carlson, Ioannis Papamichail, Markos Papageorgiou, and Albert Messmer. Optimal motorway traffic flow control involving variable speed limits and ramp metering. *Transportation Science*, 44(2):238–253, 2010.

[6] M. Cassidy and R. Bertini. Some traffic features at freeway bottlenecks. *Transportation Research. Part B*, 33(1), 1999.

[7] C. Chen, K. Petty, A. Skabardonis, and P. Varaiya. Freeway performance measurement: Mining loop detector data. In *Transportation Research Record 1748*, pages 96–102, Washington DC, 2001. TRB, National Research Council.

[8] C. Chen, A. Skabardonis, and P. Varaiya. Systematic identification of freeway bottlenecks. *Transportation Research Record*, 1867:46–52, 2004.

[9] Chen, C., J. Kwon, J. Rice, A. Skabardonis, and P. Varaiya. Detecting errors and imputing missing data for single loop surveillance systems. In *Transportation Research Record*, volume 1855, pages 160–167, Washington DC, 2003. TRB, National Research Council.

[10] Koohong Chung, Jittichai Rudjanakanoknad, and Michael J. Cassidy. Relation between traffic density and capacity drop at three freeway bottlenecks. *Transportation Research Part B: Methodological*, 41(1):82 – 95, 2007.

[11] C. Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research, Part B*, 28(4):269–287, 1994.

[12] C. Daganzo. The cell transmission model, Part II: Network traffic. *Transportation Research, Part B*, 29(2):79–93, 1995.

[13] C. Daganzo. Requiem for second-order fluid approximations of traffic flow. *Transportation Research Part B*, 29:277–286, 1995.

[14] D. Dailey. Improved error detection for inductive loop sensors. Technical Report WA-RD 3001, Washington State DOT, May 1993.

[15] Bill Eisele David Schrank, Tim Lomax. 2011 Annual Urban Mobility Report. Technical report, Texas Transportation Institute, 2006.

[16] Arthur P. Dempster, Nan M. Laird, and Donald B. Rdin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the royal statistical society, series B*, 39(1):1–38, 1977.

[17] G. Dervisoglu, G. Gomes, J. Kwon, R. Horowitz, and P. Varaiya. Automatic calibration of the fundamental diagram and empirical observations on capacity. Presented at the TRB 88th Annual Meeting, 2009.

[18] Gunes Dervisoglu and Roberto Horowitz. Model based fault detection of freeway traffic sensors. *ASME Conference Proceedings*, 2011(54754):425–432, 2011.

[19] G. Gomes. *Optimization and Microsimulation of On-ramp Metering for Congested Freeways*. PhD thesis, University of California, Berkeley, 2004.

[20] G. Gomes and R. Horowitz. Optimal freeway ramp metering using the asymmetric cell transmission model. *Transportation Research, Part C*, 14(4):244–262, 2006.

[21] Gabriel Gomes, Roberto Horowitz, Alex A. Kurzhanskiy, Pravin Varaiya, and Jaimyoung Kwon. Behavior of the cell transmission model and effectiveness of ramp metering. *Transportation Research Part C: Emerging Technologies*, 16(4):485 – 513, 2008.

[22] B. Greenshields. A study of traffic capacity. In *Proceedings of the 14th annual meeting of the Highway Research Board*, 1934.

[23] H. Haj-Salem and J.P. Lebacque. Reconstruction of false and missing data with first-order traffic volume model. pages 155–165, 2002. Transportation Research Record No 1802, Journal of the Transportation Research Board.

[24] Hall, F.L. and K. Agyemang-Duah. Freeway capacity drop and the definition of capacity. In *Transportation Research Record 1320*, pages 91–98, Washington, DC, 1991. TRB, National Research Council.

[25] Andreas Hegyi, Bart De Schutter, and Hans Hellendoorn. Model predictive control for optimal coordination of ramp metering and variable speed limits. *Transportation Research Part C: Emerging Technologies*, 13(3):185 – 209, 2005.

[26] R. Horowitz, W. Messner, and J. B. Moore. Exponential convergence of a learning controller for robot manipulators. *IEEE Transactions on Automatic Control*, 36-7:890–892, 1991.

[27] Z. Jia, B. Coifman, C. Chen, and P. Varaiya. The PeMS algorithm for accurate, real-time estimates of *g*-factors and speeds from single loop detectors. In *2001 IEEE Intelligent Transportation Systems Proceedings*, pages 536–541, Oakland, CA, 2001.

[28] Wen-Long Jin. A kinematic wave theory of lane-changing traffic flow. *Transportation Research Part B: Methodological*, 44(89):1001 – 1021, 2010.

[29] A. Kotsialos and M. Papageorgiou. Nonlinear optimal control applied to coordinated ramp metering. *Control Systems Technology, IEEE Transactions on*, 12(6):920 – 933, nov. 2004.

[30] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, and F. Middelham. Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool metanet. *Intelligent Transportation Systems, IEEE Transactions on*, 3(4):282 – 292, dec 2002.

[31] Apostolos Kotsialos, Markos Papageorgiou, Morgan Mangeas, and Habib Haj-Salem. Coordinated and integrated control of motorway networks via non-linear optimal control. *Transportation Research Part C: Emerging Technologies*, 10(1):65 – 84, 2002.

[32] A. Kurzhanskiy. *Modeling and Software Tools for Freeway Operational Planning*. PhD thesis, University of California, Berkeley, 2007.

[33] Alex Kurzhanskiy and Ajith Muralidharan. Macroscopic modeling of multiple vehicle types and freeway with hov lanes. `http://gateway.path.berkeley.edu/topl/papers/`, accessed 6/11/2012.

[34] Karric Kwong, Robert Kavaler, Ram Rajagopal, and Pravin Varaiya. Arterial travel time estimation based on vehicle re-identification using wireless magnetic sensors. *Transportation Research Part C: Emerging Technologies*, 17(6):586 – 606, 2009.

[35] David R.P. Gibson Lawrence A. Klein, Milton K. Mills. Traffic detector handbook: Third edition. Technical report, Federal Highway Administration, 2006.

[36] M. Lighthill and G. Whitham. On kinematic waves I: Flow movement in long rivers. II: A theory of traffic flow on long crowded roads. *Proc. Royal Society of London, Part A*, 229(1178):281–345, 1955.

[37] G. Liu, A. S. Lyrintzis, and P. G. Michalopoulos. Improved high-order model for freeway traffic flow. page 3746, 1998. Transportation Research Record No 1644, Journal of the Transportation Research Board.

[38] W. Messner, R. Horowitz, W-W. Kao, and M. Boals. A new adaptive learning rule. *IEEE Transactions on Automatic Control*, 36-2:188–197, 1991.

[39] Ajith Muralidharan, Gunes Dervisoglu, and Roberto Horowitz. Freeway traffic flow simulation using the cell transmission model. Proceedings of the American Control Conference, St. Louis, 2009.

[40] Ajith Muralidharan and Roberto Horowitz. Imputation of ramp flow data for freeway traffic simulation. *Transportation Research Record: Journal of the Transportation Research Board*, 2099(1):58–64, 2009.

[41] Ajith Muralidharan and Roberto Horowitz. Imputation of ramp flow data using the asymmetric cell transmission traffic flow model. *ASME Conference Proceedings*, 2009(48920):749–756, 2009.

[42] Ajith Muralidharan and Roberto Horowitz. Analysis of an adaptive iterative learning algorithm for freeway ramp flow imputation. *ASME Conference Proceedings*, 2011(54754):683–690, 2011.

[43] Ajith Muralidharan and Roberto Horowitz. Optimal control of freeway networks based on the link node ctm. *ACC Conference Proceedings*, 2012.

[44] Ajith Muralidharan, Roberto Horowitz, and Pravin Varaiya. Model predictive control of a freeway network with capacity drops. *To be presented at DSCC 2012*, 2012.

[45] D. Ni, J. D. Leonard, A. Guin, and C. Feng. Multiple imputation scheme for overcoming the missing values and variability issues in its data. *Intelligent Transportation Systems, IEEE Transactions on*, 131(12):931–938, Dec. 2005.

[46] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Yibing Wang. Review of road traffic control strategies. *Proceedings of the IEEE*, 91(12):2043 – 2067, dec 2003.

[47] M. Papageorgiou, H. Hadj-Salem, and J. Blosseville. ALINEA: a local feedback control law for on-ramp metering. *Transportation Research Record*, 1320:58–64, 1991.

[48] M. Papageorgiou, H. Hadj-Salem, and F. Middleham. ALINEA local ramp metering: Summary of field results. *Transportation Research Record*, 1603:90–98, 1997.

[49] M. Papageorgiou, E. Kosmatopoulos, and I. Papamichail. Effects of variable speed limits on motorway traffic flow. page 3748, 2008. Transportation Research Record No 2047, Journal of the Transportation Research Board.

[50] M. Papageorgiou and R. Mayr. Optimal decomposition methods applied to motorway traffic control. *International Journal of Control*, 35:269–280, 1982.

[51] Markos Papageorgiou, Jean-Marc Blosseville, and Habib Hadj-Salem. Modelling and real-time control of traffic flow on the southern part of boulevard peripherique in paris: Part I: Modelling. *Transportation Research Part A: General*, 24(5):345 – 359, 1990.

[52] Markos Papageorgiou, Ioannis Papamichail, Albert Messmer, and Yibing Wang. Traffic simulation with metanet. In Jaume Barcelo, editor, *Fundamentals of Traffic Simulation*, volume 145 of *International Series in Operations Research and Management Science*, pages 399–430. Springer New York, 2010.

[53] Ioannis Papamichail, Apostolos Kotsialos, Ioannis Margonis, and Markos Papageorgiou.

[54] Ioannis Papamichail, Markos Papageorgiou, Vincent Vong, and John Gaffney. Heuristic ramp-metering coordination strategy implemented at monash freeway, australia. *Transportation Research Record*, 2178:10–20, 2010.

[55] H. Payne. Models of freeway traffic and control. *Mathematical Models of Public Systems*, 28(1):51–61, 1971.

[56] H. Payne. Freflo: A macroscopic simulation model of freeway traffic. 1979. Transportation Research Record No 722, Journal of the Transportation Research Board.

[57] PeMS. PeMS website. `http://pems.dot.ca.gov`, accessed 6/11/2012.

[58] Li Qu, Jianming Hu, Li Li, and Yi Zhang. PPCA-based missing data imputation for traffic flow volume: A systematical approach. *Intelligent Transportation Systems, IEEE Transactions on*, 10(3):512 –522, sept. 2009.

[59] R. Rajagopal and P. Varaiya. Health of Californias loop detector system. 88th Annual Meeting of the Transportation Research Board, Washington, DC, 2009.

[60] P. Richards. Shock waves on the highway. *Operations Research*, 4(1):42–51, 1956.

[61] S.Arimoto, S. Kawamura, and F. Miyazaki. Bettering operation of robots by learning. 1984. Journal of Robotic Systems.

[62] Shankar Sastry. *Nonlinear Systems: Analysis, Stability and Control*. Springer, 1999.

[63] J. L. Schafer. *Analysis of incomplete multivariate data*. Monographs on statistics and applied probability ; 72. Chapman & Hall, London ; New York, 1997.

[64] J. J. Slotine and W. Li. *Applied Nonlinear Control*. Prentice-Hall International, 1991.

[65] E. Smaragdis and M. Papageorgiou. A series of new local ramp metering strategies. *Transportation Research Record*, 1856:7486, 2003.

[66] X. Sun and R. Horowitz. A localized switching ramp-metering controller with a queue length regulator for congested freeways. In *Proceedings American Control Conference*, June 2005. Portland, OR.

[67] TOPL. Tools for operations planning website. `http://path.berkeley.edu/topl/`, accessed 9/15/2011.

[68] TOPL Network Editor. Topl network editor. `http://vii.path.berkeley.edu:8097/NetworkEditor`, accessed 6/11/2012.

[69] Transportation Research Board. *Highway Capacity Manual 2000*, December 2000.

[70] E. van den Hoogen and S. Smulders. Control by variable speed signs: results of the dutch experiment. In *Road Traffic Monitoring and Control, 1994., Seventh International Conference on*, pages 145 –149, apr 1994.

[71] J. G. Wardrop. Some theoretical aspects of road trac research. In *Proceedings, Institution of Civil Engineers*, 1952.

[72] J. Wattleworth. Peak period analysis and control of a freeeway system. *Highway Research Record*, 157, 1967.

[73] Wikipedia. Macarthur maze. `http://en.wikipedia.org/wiki/MacArthur_Maze`, accessed 6/11/2012.

[74] JianXin Xu and Ying Tan. *Linear and Nonlinear Iterative Learning Control*. Springer-Verlag, 2003.

[75] H.M. Zhang. A non-equilibrium traffic model devoid of gas-like behavior. *Transportation Research Part B: Methodological*, 36(3):275 – 290, 2002.

[76] Michael Zhang, Taewan Kim, Xiaojian Nie, Wenlong Jin, Lianyu Chu, and Will Recker. Evaluation of on-ramp control algorithms. Technical Report UCB-ITS-PRR-2001-36, California PATH, 2001.

[77] Athanasios Ziliaskopoulos. A linear programming model for the single destination system optimum dynamic traffic assignment problem. *Transportation Science*, 34-1:37–49, 2000.