Title
Discontinuous Galerkin Methods for Hyperbolic Systems of PDEs and the Bathtub Model for Traffic Flow

Permalink
https://escholarship.org/uc/item/6hm488fd

Author
Buli, Joshua

Publication Date
2019

Peer reviewed|Thesis/dissertation
Discontinuous Galerkin Methods for Hyperbolic Systems of PDEs and the Bathtub Model for Traffic Flow

A Dissertation submitted in partial satisfaction of the requirements for the degree of

Doctor of Philosophy

in

Mathematics

by

Joshua Buli

June 2019

Dissertation Committee:

Dr. Yulong Xing, Chairperson
Dr. Richard Arnott
Dr. Mark Alber
The Dissertation of Joshua Buli is approved:

________________________________________

________________________________________

________________________________________

Committee Chairperson

University of California, Riverside
Acknowledgments

I would like to express my dearest thanks and gratitude to the many people who have been part of my time at the University of California, Riverside, and have provided uncountable amounts of support, advice, and laughs throughout the years.

First, I would like to thank my dissertation committee: my dissertation advisor Professor Yulong Xing (OSU), and Professors Richard Arnott (UCR) and Mark Alber (UCR) who are most certainly co-advisors to this dissertation. Some students are lucky to have a single advisor; I am grateful to have had the opportunity to have three during my time at UCR. Many productive conversations with Dr. Xing both during his time at UCR and through Skype at OSU have formed the groundwork for the first two chapters of this dissertation. I owe my knowledge of discontinuous Galerkin methods to your incredible guidance as my thesis advisor. Many thanks to Dr. Arnott for the multitude of thought-provoking conversations about a variety of topics, and stories about your past experiences. Your guidance and advice in both research and life has greatly influenced me during my time at UCR. Much thanks to Dr. Alber for your exceptional insights into mathematics and computational biology to advance our projects and Applied Math group, and the generous support you have provided, both through GSRs and valuable career advice. Finally, I would like to thank the whole committee for the financial support through GSRs that allowed for the significant research that was able to be accomplished in the past few years.

Second, I would like to thank all of my friends and office mates: Bryan Carrillo, Mikahl Banwarth-Kuhn, Kevin Tsai, Adam Yassine, Eddie Voskanian, Ryan Moruzzi, Kenny Courser, James Ogaja, Jolene Britton, and Sam Britton. You guys are great and
I could not have made it through without all of you! Words cannot portray how much the time we have spent together at UCR has impacted my life. I would especially like to thank UCR faculty David Weisbart for his valuable assistance in preparation for job interviews and advice for giving great talks. I would also like to thank UCR faculty and post-docs Weitao Chen, Ali Nematbakhsh, and Francesco Pancaldi for great help and advice in research projects, and navigating the graduate school. Thank you to department chairs during my time at UCR, Dr. Gerhard Gierz and Dr. Yat Sun Poon for generous department support for travel. Many, many thanks to the staff, Melissa Gomez, Margarita Roman, James Marberry, Rob Lam, and Deidra Kornfeld for all of their help over the years, and reassurance through the grad program.

I would also like to thank my early mentors and teachers from my AP Physics and AP Calculus courses, Dr. Dan Yox and Mr. Chris Watson, respectively, for giving me an amazing introduction into the field which inspired me to pursue mathematics both as an undergraduate and graduate student. In the same vein, I would like to thank Dr. Jae-Hun Jung (University at Buffalo, SUNY) for his guidance on my senior thesis and help in applying to graduate school.

Of course, I would not have made it this far without the generous and constant support from my family: Brian and Michele Buli, and my brothers Zach and Jake Buli. Thank you for supporting me on this great and long journey.

I would also like to thank the National Science Foundation which partially supported this research through the NSF Grants DMS-1621111, DMS-1753581, CMMI-1629195, as well as the Office of Naval Research which partially supported this research through the
ONR Grant N00014-16-1-2714. Additionally, I would like to thank the National Center for Sustainable Transportation (NCST) for providing fellowship support during the 2016-2017 academic year to support this research.

For those who seek knowledge and truth through reason.

Vi Veri Vniversum Vivus Vici
Macroscopic models for flows strive to depict the physical world by considering quantities of interest at the aggregate level versus focusing on discrete particles. This thesis contains three major sections which construct numerical methods for macroscopic flow models.

In the first part, we develop local discontinuous Galerkin (LDG) methods for the Boussinesq coupled BBM system. The system contains nonlinear dispersive wave equations, containing nonlinearities and dispersive effects. We provide two choices of numerical fluxes, namely the upwind and alternating fluxes, and establish their stability analysis. The error estimate for the linearized BBM system is carried out with the alternating flux. A time discretization conserving the Hamiltonian numerically using the midpoint rule with a nontrivial nonlinear term is proposed. Numerical examples demonstrate the accuracy, long-time simulation, and Hamiltonian conservation properties of the proposed method.
In the second part, we consider the second-order Aw-Rascle (AR) traffic flow model on a network, and propose a discontinuous Galerkin (DG) method for solving the AR system of hyperbolic PDEs with appropriate coupling conditions at the junctions. On each road, the standard DG method with Lax-Friedrichs flux is employed, and at the junction, we solve an optimization problem to evaluate the numerical flux of the DG method. As the choice of well-posed coupling conditions for the AR model is not unique, we test different coupling conditions at the junctions. Numerical examples are provided to demonstrate the high-order accuracy, and comparison of results between the first-order Lighthill-Whitham-Richards model and the second-order AR model. The ability of the model to capture the capacity drop phenomenon is also explored.

In the third part, we consider the basic bathtub model of traffic congestion. The model has an unfamiliar mathematical structure, with a delay differential equation with an endogenous delay at its core. Early papers on the model circumvented this complication by making approximating assumptions, but without solution of the proper model it is unclear how accurate the results are. We develop a customized method for computational solution of equilibrium in the basic bathtub model with smooth preferences that exploits the mathematical structure of the problem.
# Contents

1 Introduction 1  
1.1 Overview 1  
1.2 Background for Models 4  
  1.2.1 *abcd*-Boussinesq System and Coupled BBM system 4  
  1.2.2 First and Second Order Models for Network Traffic 8  
  1.2.3 Basic Bathtub Model 14  
1.3 Discontinuous Galerkin Methods 17  

2 Local Discontinuous Galerkin Methods for the Coupled BBM System 21  
2.1 Local Discontinuous Galerkin Method 22  
  2.1.1 Alternating Flux 24  
  2.1.2 Upwind Flux 27  
2.2 Error Estimate for the Linearized Systems 31  
2.3 Time Discretization 36  
  2.3.1 SSPRK Methods 36  
  2.3.2 Midpoint Rule 37  
2.4 Numerical Experiments 39  
  2.4.1 Accuracy Test 40
2.4.2 Comparisons of Numerical Flux and Time Discretization . . . . . . . 42
2.4.3 Solitary Wave Generation . . . . . . . . . . . . . . . . . . . . . . . . 44
2.4.4 Solitary Wave Collision . . . . . . . . . . . . . . . . . . . . . . . . . 46

3 Discontinuous Galerkin Method for the Aw-Rascle Traffic Model on Networks 54

3.1 Notation and Conventions . . . . . . . . . . . . . . . . . . . . . . . . . . . 55
3.2 Coupling Conditions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 58
3.3 Discontinuous Galerkin Method . . . . . . . . . . . . . . . . . . . . . . . . . 61
  3.3.1 DG method for the AR Model . . . . . . . . . . . . . . . . . . . . . . . 61
  3.3.2 Implementation of the Coupling Conditions . . . . . . . . . . . . . . 63
  3.3.3 Determining the Admissible Regions . . . . . . . . . . . . . . . . . . . 67
  3.3.4 Maximization of flux of the density . . . . . . . . . . . . . . . . . . . 70
  3.3.5 1 incoming road, 1 outgoing road . . . . . . . . . . . . . . . . . . . . 73
  3.3.6 1 incoming road, 2 outgoing road . . . . . . . . . . . . . . . . . . . . 73
  3.3.7 2 incoming road, 1 outgoing road . . . . . . . . . . . . . . . . . . . . 75
  3.3.8 2 incoming road, 2 outgoing road . . . . . . . . . . . . . . . . . . . . 78
  3.3.9 Optimization Problems . . . . . . . . . . . . . . . . . . . . . . . . . 79
3.4 Numerical Experiments for the AR Model . . . . . . . . . . . . . . . . . . . 82
  3.4.1 Accuracy Test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 82
  3.4.2 Numerical Experiments Comparing LWR and AR Models . . . . . . . 83
    3.4.2.1 One Incoming Road and Two Outgoing Roads . . . . . . . . . 86
    3.4.2.2 Two Incoming Roads and One Outgoing Road . . . . . . . . . 87
List of Figures

2.4.1 Wave profiles for time discretization comparison . . . . . . . . . . . . . . 48
2.4.2 Error and conserved quantity plots for time discretization comparison . . 49
2.4.3 Wave profiles for alternating and upwind flux comparison . . . . . . . . . . 50
2.4.4 Error and conserved quantity plots for alternating and upwind flux com-
parison . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51
2.4.5 Clean solitary wave generation for $\eta(x, t)$ . . . . . . . . . . . . . . . . . . 52
2.4.6 Solitary wave collision for $\eta(x, t)$ . . . . . . . . . . . . . . . . . . . . . . . 53
3.1.1 Notation at the junction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 56
3.4.2 One incoming and one outgoing road case with initial and boundary data
given in (3.4.39) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 85
3.4.3 One incoming and two outgoing roads - density plots . . . . . . . . . . . . 95
3.4.4 Two incoming and one outgoing roads - density plots . . . . . . . . . . . . 96
3.4.5 Two incoming and one outgoing roads - density plots . . . . . . . . . . . . 97
3.4.6 Two incoming and one outgoing roads - AR-LWR comparison, $T = 0.15$ . 98
3.4.7 Two incoming and one outgoing roads - AR-LWR comparison, $T = 0.30$ . 99
3.4.8 Comparison of coupling conditions between [42] and [45] - One incoming
and two outgoing roads . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 100
3.4.9 Comparison of coupling conditions between [42] and [45] - Two incoming and one outgoing roads ........................................ 101

3.4.10 Capacity drop test: density and velocity plots over time ............... 102

3.4.11 Capacity drop test: density flux and $w = q/u$ plots over time ........... 103

3.4.12 Capacity drop test: density profiles at $T = 0.655$ ....................... 104

4.2.1 Terminology and notation for time and cycles ......................... 118

4.2.2 Walters’ user cost diagram. ........................................ 137

4.2.3 Utility versus time plot ............................................. 141

4.2.4 Flowchart outlining the algorithm for a fixed number of cycles ........ 145

4.3.5 User cost versus $N$: $N$-trip-timing equilibrium ......................... 162

4.3.6 User cost versus $N$: full-trip-timing equilibrium ........................ 163

4.4.7 The case of one partial entry cycle .................................. 164

4.4.8 The case of three full cycles and one partial entry cycle ................ 165

4.4.9 The case of one full and one partial entry cycle: hypercongestion ....... 166

4.4.10 The user cost curve for the logarithmic utility function ................. 167

A.0.1 Discrete departure set diagram ...................................... 181

A.0.2 Entry mass diagram .................................................. 183
List of Tables

2.4.1 Convergence rate test for the alternating flux in space and SSPRK4 in time for the traveling wave solutions (2.4.48) ........................................ 41

2.4.2 Convergence rate test for the alternating flux in space and 2nd order midpoint rule in time for the traveling wave solutions (2.4.48) ........................................ 41

2.4.3 Convergence rate test for the upwind flux in space and SSPRK4 in time for the traveling wave solutions (2.4.48) ........................................ 41

3.4.1 Accuracy test for the system (1.2.13) ........................................ 84

4.2.1 Outline of theorems for the existence and uniqueness of equilibrium ........................................ 112

4.4.2 Logarithmic utility and travel duration functions ........................................ 152

B.2.1 Logarithmic utility and travel time functions ........................................ 200

B.2.2 Signs of important partial derivatives in the bathtub model ........................................ 201

B.3.3 Exponential utility and travel time functions ........................................ 202
Chapter 1

Introduction

1.1 Overview

In the first two chapters, we are interested in developing numerical methods (specifically discontinuous Galerkin methods, outlined in section 1.3) for systems of hyperbolic conservation laws. A one-dimensional scalar conservation law is a first-order partial differential equation of the form

\[ \partial_t u + \partial_x f(u, x) = g(u, x), \]  

(1.1.1)

where the function \( u(t, x) \) is called the conserved quantity, \( f(\cdot) \) is the flux, and \( g(\cdot) \) is a source term. For a system of conservation laws, we will use the same notational convention as (1.1.1), but with capital letters signifying the vector valued variables, ie. \( U \) replacing \( u \) and \( F \) replacing \( f \). Equations of this type generally describe some kind of transport phenomena. Conservation laws are central equations in physics, as they describe fundamental laws of nature such as the conservation law of energy, mass, electric charge, angular momentum,
etc. In physical terms, conservation laws state that the total quantity being described remains unchanged over time.

The hyperbolic nature of the conservation law is determined by the Jacobian of the flux terms. In the scalar case this refers to $\partial_u f(u, x)$, and in the system case is the matrix $\nabla_u F(U, x)$. A conservation law is said to be hyperbolic if all the eigenvalues of the Jacobian are real, and there exists a complete set of eigenvectors. The flux in the conservation law is general, that is to say that $f$ (or $F$) can be nonlinear function of $u$ (or $U$), in which case numerical methods may not be able to capture the complex dynamics associated with the nonlinearity. The purpose of the first two chapters is to construct accurate and efficient high-order numerical methods to solve equations of the type (1.1.1) which contain the nonlinear flux terms. Our equations will model two different types of applications, (i) the coupled BBM system which models Boussinesq type water waves, and (ii) the second-order Aw-Rascle traffic model on a network of roads. The systems will be described in the proceeding section, section 1.2.

In the last chapter, we consider the basic bathtub model. In the model, traffic congestion over an isotropic space is described by a macroscopic fundamental diagram (MFD). Over the course of the rush hour, a fixed number of commuters with the same tastes each travel an exogenous distance from home to work, with traffic density evolving endogenously such that trip-timing equilibrium conditions are satisfied. The basic bathtub model may be the simplest model of rush-hour traffic dynamics which incorporates trip length and also admits hypercongestion, the phenomenon of increased traffic density associated with a decrease in traffic flow, ie. traffic jams. Research on hypercongestion has accelerated since
the seminal publication of Geroliminis and Daganzo [43], which empirically documented hypercongestion as a stable feature of rush-hour traffic dynamics at the level of a downtown neighborhood during the peak of the rush hour. The paper documented the existence of an MFD relating flow to density at the level of a neighborhood in multiple cities, and found that is was stable over the course of the day and over days. Subsequent empirical studies have also documented the importance of hypercongestion at the level of a downtown neighborhood.

Unfortunately, the bathtub model is analytically intractable, giving rise to a delay differential equation (DDE) with an endogenous delay, whose solution are at the forefront in applied mathematics research. The current transportation theory literature has responded to this intractability by making approximating assumptions that considerably simplify the mathematics so that analytical approaches can be studied. Our approach is to move directly to numerical solution of the basic bathtub model by using the equilibrium conditions to reformulate the DDE at the core of the model, which takes its natural cyclical structure into account. As the algorithm we develop constructs the equilibrium, it provides a constructive method of proof of existence of equilibrium. Restrictions on the functional forms of the utility and congestion functions that guarantee a unique solution not only provide sufficient conditions for the uniqueness of equilibrium but also imply several unambiguous comparative static properties of equilibrium when it is unique. Section 1.2 provides the detailed description of the bathtub model, and chapter 4 gives the construction of the numerical algorithm, side by side with the necessary theoretical properties of the solution which are required to prove uniqueness of equilibrium.
1.2 Background for Models

1.2.1 \textit{abcd-Boussinesq System and Coupled BBM system}

Nonlinear dispersive water wave models have numerous applications in a variety of engineering disciplines. One specific model is the Boussinesq approximation for water waves, which was developed by Joseph Boussinesq in 1871 [20] to take into account the vertical structure of the horizontal and vertical flow velocity. Derived from the Euler equations, this approximation results in nonlinear partial differential equations called Boussinesq-type equations, which incorporate frequency dispersion (as opposed to the nonlinear shallow water equations, where no frequency dispersion is present). In coastal engineering applications, Boussinesq-type equations are frequently used in computer models for the simulation of water waves in propagation of long-crested waves on the ocean, shallow seas, large lakes, coastlines, and harbors. These equations can also model water waves moving through a channel which have small amplitude and long wavelengths.

In practical applications, certain physical assumptions are made about the water waves, which lead to simplified mathematical models. A special case is the the \textit{abcd-Boussinesq systems} which describe the propagation of small amplitude and long wavelength surface waves that are irrotational, incompressible, inviscid, and are influenced by gravity. In [17], the \textit{abcd-Boussinesq systems} used to model such water waves are derived and given by

\begin{equation}
\begin{align*}
\eta_t + u_x + (\eta u)_x + au_{xxx} - b\eta_{xxt} &= 0 \\
u_t + \eta_x + uu_x + c\eta_{xxx} - du_{xxt} &= 0,
\end{align*}
\end{equation}
where $\eta(x, t)$ represents proportional deviation of the free surface from its rest position, and $u(x, t)$ represents proportional horizontal velocity. The constant parameters $a, b, c, d$ are not taken arbitrarily, and are chosen via the following relationships:

$$a + b = \frac{1}{2} \left( \theta^2 - \frac{1}{3} \right), \quad c + d = \frac{1}{2} (1 - \theta^2) \geq 0,$$

with $\theta \in [0, 1]$, representing the scaled height (where $\theta = 0$ is the bottom of the channel and $\theta = 1$ is the free surface). In [17], seven different choices of the parameters are given, which lead to multiple types of Boussinesq systems including the classical Boussinesq system, Kaup system and Bona-Smith system etc. When there is no dispersive term, i.e., $a, b, c, d = 0$, the model (1.2.2) reduces to

$$\begin{cases}
\eta_t + ((1 + \eta) u)_x = 0 \\
 u_t + (\eta + \frac{1}{2} u^2)_x = 0,
\end{cases}$$

which is the shallow water equation if we replace $1 + \eta$ by water height $h$.

In Chapter 2, we consider the coupled BBM system (taking $a = c = 0$ and $b = d = \frac{1}{6}$ or equivalently $\theta^2 = \frac{2}{3}$ in (1.2.3)), which are given by

$$\begin{cases}
\eta_t + u_x + (\eta u)_x - \frac{1}{6} \eta_{ext} = 0, \\
u_t + \eta_x + uu_x - \frac{1}{6} u_{ext} = 0, \quad (x, t) \in [a_0, a_1] \times [0, T],
\end{cases}$$

subject to the initial conditions

$$u(x, 0) = u_0(x), \quad \eta(x, 0) = \eta_0(x),$$

and periodic boundary conditions

$$u(a_0, t) = u(a_1, t).$$
The systems in (1.2.5) are coupled nonlinear partial differential equations with weak dispersion effects resulting from the $u_{xxt}$ and $\eta_{xxt}$ terms. In the derivation of (1.2.2) (and therefore (1.2.5)), dissipative effects are not considered, and the system given in (1.2.5) admit the following Hamiltonian functionals

$$\int_{\mathbb{R}} \eta \, dx, \quad \int_{\mathbb{R}} u \, dx, \quad \int_{\mathbb{R}} (\eta u + \eta_x u_x) \, dx, \quad \frac{1}{2} \int_{\mathbb{R}} [\eta^2 + (1 + \eta)u^2] \, dx.$$  (1.2.8)

The coupled BBM system is related to the single BBM equation, which was derived by Benjamin, Bona, and Mahoney in 1972 in [13]. The single BBM equation was developed as an alternative to the Korteweg-de Vries (KdV) equation. Recall the KdV equation given by

$$u_t + u_x + uu_x + u_{xxx} = 0,$$  (1.2.9)

and the single BBM equation can be written as

$$u_t + u_x + uu_x - u_{xxt} = 0,$$  (1.2.10)

where constants in front of the nonlinear and high-order terms have been suppressed. In [13], the assumptions involved in the derivation of the KdV equation are equally valid for the single BBM equation (1.2.10). The single BBM equation has some attractive features that the KdV equation lacks, one specific example is better dispersion properties resulting in stability in high wavenumber components (see [13] for details).

There have been a wide range of theoretical work and various numerical methods available for the coupled BBM system. For the single BBM equation, Benjamin, Bona, and Mahoney [13] provided the comparison of the KdV and single BBM equations and justification for the single BBM equation to be a valid alternative model to the KdV equation.
Existence and uniqueness results are also provided for the single BBM equation. Bona and Chen provided existence, uniqueness, and regularity results of the coupled BBM system in [16]. The derivation of the system (1.2.2), and well-posedness results were given for multiple cases of (1.2.2), including (1.2.5) in [17] and [18]. Previous work established that solutions to (1.2.5) may blow-up in finite time if $1 + \eta < 0$. Chen and Liu [28] extended the result relating to the Hamiltonian (1.2.11), that solutions to (1.2.5) would not blow-up in finite time if $\eta$ was bounded from below. In [16], an unconditionally stable fourth-order accurate finite difference numerical scheme is developed for (1.2.5). In [2], the same numerical method from [16] was used to conduct an in depth analysis to compare the solutions to the single BBM equation and the coupled BBM system. Finite element methods have been designed for the two dimensional coupled BBM system in [40].

Also in Chapter 2, we discuss the derivation and development of LDG methods to solve the coupled BBM system (1.2.5), which conserve or dissipate the Hamiltonian functional

$$E(\eta, u) = \frac{1}{2} \int_I \left[ \eta^2 + (1 + \eta)u^2 \right] dx. \quad (1.2.11)$$

To that effect, we develop two different choices of numerical flux, where one choice (the alternating flux, to be defined in Section 2.1.1) conserves the Hamiltonian for long time simulations, and the second flux (upwind flux) adds numerical dissipation and has a Hamiltonian decreasing property. Optimal error estimate is derived for the linearized coupled BBM system. We have also experimented with two different types of time discretizations, the implicit second order midpoint rule which conserves the Hamiltonian, and the standard Runge-Kutta methods. Proof is provided to verify that the LDG method with the alter-
nating flux, coupled with the midpoint rule temporal discretization, and a special nonlinear term discretization, conserves the discrete Hamiltonian exactly. Note that the Hamiltonian (1.2.11) is only guaranteed to be non-negative when $1 + \eta \geq 0$. Under this condition, the stability of the LDG method follows from the Hamiltonian conservation (or dissipation). When such condition does not hold, we observe numerically that both Hamiltonian conserving and dissipative methods perform well. Numerical evidence is also provided to show that the Hamiltonian conserving methods have a smaller phase and shape error for the long time simulations.

### 1.2.2 First and Second Order Models for Network Traffic

Macroscopic continuum models for hyperbolic network flows have been applied in a diverse array of applications in a variety of scientific fields. Such models utilize partial differential equations (PDEs) to simulate the evolution of the macroscopic quantities across a network. The practicality of the macroscopic PDE methods is realized through the simulation of large networks with many junctions and edges. A large network poses computational time issues if microscopic models are used, as the behavior of each entity across the network must be taken into account. The computational cost can also increase dramatically when increased dimension size is considered in a model extension. Applications of hyperbolic network flows include blood circulation \[53, 59\], supply chains \[3\], data packet flow and telecommunications \[39\], air traffic management \[65\], vehicular traffic flow \[26\], and many more. A comprehensive overview of hyperbolic flows on networks is given in \[21\], which provides theoretical results, models for various types of applications of network flows, coupling conditions, and some numerical results.
The spatial component of the model is given by a network, or topological graph, consisting of a collection edges (also called links) and a set of junctions (also called nodes or vertices). Edges are connected at junctions, along with additional requirements so that the network can be described mathematically as a graph. On each edge, the dynamics are described by solutions to either a scalar PDE or systems of PDEs. As this paper is concerned with the application of traffic flow, terminology from traffic flow literature will be employed. A first-order model is a model where the evolution of vehicle density on the network are given by a single PDE on each edge. A second-order model is a model in which the dynamics are governed by a system of two PDEs on each edge, one for the density and one for the velocity of vehicles. This notion can be extended to larger systems as well. The use of the word “high-order” should not be confused with the terminology of the numerical analysis community, where a “high-order method” is a numerical scheme that can achieve a high-order of accuracy for computational implementation.

Much theoretical work has been done studying first and second order models of traffic flow. The seminal work done by Lighthill and Whitham [49] and Richards [61] developed the first order model for traffic flow, now known as the Lighthill-Whitham-Richards (LWR) model of traffic flow, describing evolution of traffic density over time. The LWR model can be written in conservative form as in (1.1.1), where \( f(u) = uv(u) \) represents the flow of vehicles, and \( u \) and \( v \) are the vehicle density and velocity, respectively. Eq. (1.1.1) is also known as the continuity equation, and states that the number of vehicles is to be conserved. The average velocity function \( v = v(u) \) is a decreasing function of the density \( u \), and the flow function \( f(u) = uv(u) \) defines a unique relationship between flow and density.
and is termed the “macroscopic fundamental diagram” (MFD) of traffic flow. Much work has been done on developing different MFDs, as well as showing that MFDs fit experimental data [38]. For our paper we are primarily interested in the numerical aspects of the junction problem, so when the LWR model is considered we simply take the normalized Greenshields’ relation $v(u) = 1 - u$, which states that velocity is a negative linear function of density, so that the flux becomes $f(u) = u(1 - u)$. In the 1990s, Daganzo [35] developed the cell-transmission model (CTM) for the first order traffic flow problem, which looks at discrete vehicles in a discretized road, and proved that in the limit, the CTM approaches the first order LWR model. In [37], the method was extended to the network case. Also in 1995, Holden and Risebro [47] developed theory for the first order model on a network.

Second-order models were developed in attempt to improve upon the first-order LWR model. There is a vast collection of literature which studies high-order macroscopic traffic flow models. The literature review paper [66] provides an excellent introduction to macroscopic traffic flow models, as well as micro and mesoscopic traffic models, spanning from the 1950s to present. Here, we remark on the most relevant developments relating to second-order macroscopic models. In 1971, Payne [58] derived a second-order model, using fluid flow as a basis for the extension. The model included a second PDE which made the velocity dynamic. This is due to certain limitations of using a single conservation law as in (1.1.1) to model the dynamics of traffic congestion, especially traffic jam situations, where decreased flow is observed on a traffic network when density rises above a critical value. The analogy of treating traffic as a fluid was advanced to use high-order models from fluid dynamics to add more details to traffic models, but the analogy between traffic flow and fluid
flow is imperfect. In [36], Daganzo provided strong criticism against second-order models of traffic flow up to 1995, as the motion of fluid particles are determined by information both in front of and behind the particle, while traffic flow is anisotropic in nature, in that the driver is only influenced by the conditions upstream from its current position.

Aw and Rascle [7] developed a “proper” second-order model of traffic flow which addressed the criticisms of Daganzo [36] that the existing second order models were non-physical. The model is a nonlinear hyperbolic system of PDEs, which we will refer to as the Aw-Rascle (AR) model, and is given by

\[
\begin{aligned}
\partial_t u + \partial_x(uv) &= 0 \\
\partial_t(u(v + p(u))) + \partial_x(uv(v + p(u))) &= 0,
\end{aligned}
\]

where \(u(x,t)\) and \(v(x,t)\) represent the density and velocity of the vehicles, respectively. The function \(p(u)\) is the “pressure” term, which is taken as an increasing function of the density \(u\). The pressure is a well defined quantity in fluid flow, but may seem inappropriate for traffic flow, as there is not an immediate physical intuition of what pressure would represent. In fact, in [7], the authors state that the pressure term describes how drivers react to a change in the concentration of vehicles in front of them. Thus the pressure function is often referred to as an “anticipation” factor. For our numerical work, we take the pressure function to have the form \(p(u) = u^\gamma\) for a constant \(\gamma > 0\). We can rewrite (1.2.12) as

\[
\begin{aligned}
\partial_t u + \partial_x(q - up(u)) &= 0 \\
\partial_t q + \partial_x \left( \frac{q^2}{u} - qp(u) \right) &= 0,
\end{aligned}
\]
where \( q = u(v + p(u)) \) represents a “pseudo-momentum”. For the numerical work in this paper, we will utilize the conservation form\(^1\) of the AR model given by (1.2.13). We are interested in the network problem corresponding to the AR model, which will be described in detail in section 3.1. A traffic network consists of edges, in this case roads, that meet at a single node (junction), where the dynamics on each road are given by (1.2.13), and some prescribed conditions are defined at the junction, which are termed coupling conditions. We only consider single junction cases, with a particular number of incoming and outgoing roads with one node. As the speed of propagation of the waves is finite due to the hyperbolic nature of the PDEs, our method can be extended to multiple junctions by combining the junction types together to form a complex network.

There has also been a wide variety of work done on the numerical analysis of hyperbolic network flows across different fields. Applications where numerical methods have been employed to solve hyperbolic flows on networks include shallow water equations in channels [22, 11], gas dynamics [9], blood flow circulation [53], and first order traffic flow models [26]. In particular, the discontinuous Galerkin (DG) method is applied to the first order LWR model on a network in [26], where the Riemann problem is solved exactly to determine the coupling conditions, and an equivalent form of the Godunov flux is applied to solve the junction problem. Other numerical methods using different approaches have been developed for traffic flow models including the cell transmission model [35], finite difference methods [52], WENO schemes for the multi-class LWR model [72], and finite volume methods for the Aw-Rascle-Zhang model [62].

---

\(^1\)We could also write (1.2.13) in system form using \( U = (u, q)^T \), which will be done in (3.3.4). The use of \( U \) will be used when defining the coupling conditions in a concise way.
In the present paper, we discuss the derivation and development of DG methods to solve the Aw-Rascle model (1.2.13) on a network with various junction types and different coupling conditions. In [26], DG methods with Godunov flux are presented to solve the LWR model on networks. The Godunov flux, based on the exact Riemann solution, is computed at the junction, which relies on multiple cases for each type of junction. The number of cases also increases with the number of roads, thus making the evaluation of the flux at the junction complicated and time consuming if there are many roads entering and leaving the junction. Also, when the second order AR model is considered in this paper, it becomes too complicated to solve the Riemann problem exactly at the junction to determine the numerical flux. In [19], a first order numerical algorithm is developed to solve the coupling conditions at a junction via an optimization problem, and the accuracy and effectiveness of this algorithm is validated for the shallow water equations on networks. In this paper, we combine this technique with the DG methods and introduce a high order algorithm which can efficiently solve the AR model on networks. At the junction, only a single optimization problem, based on the DG solutions of directly neighboring cells on each road, needs to be solved with the \textit{a priori} coupling conditions which can be handled by built-in solvers. One attractiveness of this formulation is that we can use fluxes other than the Godunov flux, say the Lax-Friedrichs flux or FORCE scheme, which do not require the exact solution of the Riemann problem. The novel contribution of this work is to introduce a high order DG framework for the AR traffic flow system which allows the usage of any numerical flux. The DG method allows for easy handling at the junction, as the method only relies on information from directly neighboring cells, making it an ideal choice for the
junction problem. It was emphasized in [26] that “the DG method perhaps is the only realistic and efficient high order method for network problems”.

1.2.3 Basic Bathtub Model

Prior to a decade ago, essentially no data had been collected on downtown traffic congestion at the level of an entire downtown neighborhood or over an entire downtown area. That changed with the publication of [43] (referenced as GD hereafter). The paper contained two central findings. The first was that there was a stable relationship between traffic flow and traffic density at the level of the neighborhood over the course of the day, and from day to day, which the authors referred to as the neighborhood’s macroscopic fundamental diagram (MFD). The second was that this relationship has an inverted-U shape, with traffic flow rising with traffic density up to a critical density and then falling with density. Economists refer to the phenomenon in which traffic flow falls as traffic density rises under heavily congested conditions as hypercongestion. Thus, for the first time GD documented hypercongested traffic flow at the level of a downtown neighborhood. Subsequent studies have confirmed the empirical regularities identified by GD, though the form of the MFD, and its degree of stability, vary over cities.

In both transportation science and transportation economics, Vickrey’s bottleneck model [67], as adapted by Arnott, de Palma, and Lindsey [5], has been the workhorse model of equilibrium rush-hour traffic dynamics for a quarter century. Traffic congestion takes the form of queues behind bottlenecks of fixed flow capacity, which rules out hypercongestion. Urban transportation economists have long recognized the potential importance of hypercongestion, particularly in downtown areas, and have been exploring alternative ways of
adapting the bottleneck model to accommodate hypercongestion. Two approaches have been taken. The first is to make the capacity of a bottleneck a function of the length of the queue behind it. The second is to develop isotropic models of downtown rush-hour traffic dynamics that incorporate MFD congestion (which assumes a stable relationship between traffic flow at a point in time and traffic density at that point in time), which have come to be referred to generically as *bathtub* models. The publication of GD has catalyzed their development. In the *basic* bathtub model, which is the focus of chapter 4, over the morning rush hour a fixed number of identical commuters travel an equal distance from home to work over an isotropic downtown area, with the velocity of traffic at a point in time depending on the density of traffic at that time. Thus, a particular commuter’s speed varies over the course of his journey, as traffic density changes.

Bathtub models have a serious weakness. The solution of their equilibria and optima is generally analytically intractable, giving rise to a class of DDEs, as described in the overview. Furthermore, there is little experience to draw on in their solution by numerical methods. The initial bathtub models, which were all basic bathtub models (assuming identical individuals), dealt with this intractability by making approximating assumptions that transform the delay differential equation into an ordinary differential equation (ODE) which is solvable using standard methods from the theory of ODEs. None of these models, has gained widespread acceptance in the community, since without solution of the “proper” model that they approximate, it is not possible to judge the accuracy of their approximated solutions.
We take a different approach to dealing with the analytical intractability. Rather than attempt to obtain an exact or approximate analytical solution, it moves directly to numerical solution. By drawing on the mathematical and economic structure of the problem, the solution algorithm we develop avoids most of the problems that would arise from application of generic numerical solution methods. The algorithm is tailored to a solution of equilibrium in the basic bathtub model under the assumption that the utility or trip cost function is a smooth function of its arguments. We illustrate application of the algorithm using two simple utility functions.

Before proceeding, we provide a brief description of the bathtub model. Consider a fixed number of commuters per unit area travel from home to work in the morning rush hour in an isotropic downtown area. Trip origins and destinations are uniformly distributed over the area and each commuter travels the same exogenous trip distance, \( L \). Commuters also have identical tastes, which are described by a utility function having as its arguments departure time and trip duration\(^2\). The congestion technology is described by a function exhibiting a negative relationship between velocity and traffic density per unit area. Traffic density at a point in time equals the cumulative entries to the road system (which equals cumulative departures from home) minus the cumulative exits from the road system (which equals cumulative arrivals at work) to that point in time. Furthermore, for any commuter, cumulative entries at the time she enters the street network equal cumulative exits at the time she exits it (a FIFO condition). Equilibrium is obtained when no commuter can increase her utility by departing at a different time, and all commuters travel. Since the

\(^2\)It is assumed that the commuter drives directly from home to work. Thus, one could just as well take utility to be a function of departure time and arrival time, or arrival time and trip duration.
model can be completely solved for from the entry rate function, the objective of the solution algorithm is to solve for the entry rate function (or functions) consistent with the equilibrium conditions.

In the process of developing the solution algorithm, we have derived some mathematical properties of the bathtub model. Some hold for all smooth utility functions; others hold under more restrictive conditions. We present these formally, in proposition-proof format, in a companion paper [23], which is to appear. In chapter 4 of this thesis, we shall provide these results as they arise during the discussion of the numerical algorithm construction.

1.3 Discontinuous Galerkin Methods

In recent years, high order accurate numerical schemes have attracted increasing attention in many computational fields. Among different high-order methods, the discontinuous Galerkin (DG) method is a class of finite element methods using completely discontinuous piecewise polynomial basis functions, which inherits the benefits of both finite element and finite volume methods (see [60, 30, 32, 33], and [31] for a historic review). Advantages of DG methods are many, including the local conservativity, the ability for easy handling of complicated geometries and boundary conditions, the flexibility for $hp$-adaptivity, efficient parallel implementation, and easy coordination with finite volume techniques, making the methods very attractive in a wide range of applications. DG methods have attracted increasing attention for applications requiring high performance computing, due to their high computational intensity and less data communication. The parallel efficiency has been
shown to be more than 80% for adaptive meshes and more than 99% for fixed meshes, as well as many developments in different parallel strategies (see [14] and [8]). The DG method also provides for simple adaptation with the predominately used finite volume techniques, which benefits those groups who already implement those methods. The aforementioned benefits make the use of the DG method quite promising for the nonlinear coupled BBM system, and traffic flow problems on networks, which are considered in chapter 2 and chapter 3, respectively.

The DG methods were later generalized to the local discontinuous Galerkin (LDG) methods by Cockburn and Shu in [34] to solve the convection-diffusion equations and partial differential equations with high order spatial derivatives, motivated by successful numerical experiments from Bassi and Rebay for the compressible Navier-Stokes equations [10]. Later, LDG methods have been designed for many nonlinear dispersive equations, and we refer to the review paper [71] for the latest development of the LDG method. Recently, there have been many study in designing LDG methods which can conserve the energy or Hamiltonian of the model numerically. An energy conserving LDG method for the generalized KdV equation is proposed in [15]. Energy conserving LDG methods have also been designed for other type of equations, including the second order wave equation [69, 29], Camassa-Holm equation [51], Degasperis-Procesi equation [50], etc. We make use of the LDG formulation for the BBM system in Chapter 2.

In the remainder of this section, we provide a short overview of the DG method using the one-dimensional scalar hyperbolic conservation law (1.1.1), with initial data given by \( u(0, x) = u_0(x) \). Given a spatial interval \( I = [a, b] \), we divide \( I \) into \( N \) subintervals and
label each cell as \( I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \) for \( j = 1, \ldots, N \), with \( a = x_{\frac{1}{2}} \) and \( b = x_{N+\frac{1}{2}} \). The center of each cell is given by \( x_j = \frac{1}{2}(x_{j-\frac{1}{2}} + x_{j+\frac{1}{2}}) \), with mesh size \( h_j = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}} \). We denote the maximal mesh size as \( h = \max_{1 \leq j \leq N} h_j \).

Now, multiply equation (1.1.1) by an arbitrary smooth function \( \phi \), integrate over the cell \( I_j \), and perform an integration by parts to obtain

\[
\int_{I_j} \partial_t u(t, x) \phi(x) \, dx - \int_{I_j} f(u(t, x)) \partial_x \phi(x) \, dx
+ f(u(t, x_{j+\frac{1}{2}})) \phi(x_{j+\frac{1}{2}}) - f(u(t, x_{j-\frac{1}{2}})) \phi(x_{j-\frac{1}{2}}),
\]

\[
\int_{I_j} u(0, x) \phi(x) \, dx = \int_{I_j} u_0(x) \phi(x) \, dx.
\]

The primary difference between traditional finite element methods and the DG method is the choice of solution space and test function space. We approximate the true solution \( u(t, x) \) by \( u_h \), which belongs to the finite dimensional piecewise polynomial space

\[
V_h^\kappa = \{ v : v|_{I_j} \in P^\kappa(I_j), \ j = 1, \ldots, N \},
\]

where \( P^\kappa(I_j) \) denotes the space of polynomials of degree at most \( \kappa \) on the cell \( I_j \). Functions in \( V_h^\kappa \) are allowed to have discontinuities across the cell interface. In (1.3.14), we replace the smooth function \( \phi \) by test functions \( \phi_h \in V_h^\kappa \), and \( u \) by the numerical solution \( u_h \).

We use the notation \((\phi_h)^-_{j+\frac{1}{2}}\) and \((\phi_h)^+_{j+\frac{1}{2}}\) to represent the limit value of \( \phi_h \) at \( x_{j+\frac{1}{2}} \) from the left cell \( I_j \) and the right cell \( I_{j+1} \), respectively. Then, the notations for the jump at the interface, and the average of the function are given by \( [\phi_h] = (\phi_h)^+ - (\phi_h)^- \) and \( \{\phi_h\} = \frac{1}{2}(\phi_h^+ + (\phi_h)^-) \), respectively. Now we can replace the nonlinear flux term \( f(u(t, x_{j+\frac{1}{2}})) \) by the numerical flux term

\[
\hat{f}_{j+\frac{1}{2}} = F(u_h(t, x_{j+\frac{1}{2}}), u_h(t, x_{j+\frac{1}{2}})),
\]

19
and define the numerical scheme as

\[
\int_{I_j} \partial_t u_h(t,x) \phi_h(x) \, dx - \int_{I_j} f(u_h(t,x)) \partial_x \phi_h(x) \, dx \\
+ \tilde{f}_{j+\frac{1}{2}} \phi_h(x_{j+\frac{1}{2}}^{-}) - \tilde{f}_{j-\frac{1}{2}} \phi(x_{j-\frac{1}{2}}^{+}),
\]

\[
\int_{I_j} u_h(0,x) \phi_h(x) \, dx = \int_{I_j} u_0(x) \phi_h(x) \, dx.
\]

The flux \( F(a,b) \) is taken to be a monotone flux, for example the Godunov or Lax-Friedrichs flux. We can then sum over the index \( j \) to get the scheme for the whole interval \( I \). To simplify the notation, we will use the following:

\[
(\phi, \psi)_{I_j} = \int_{I_j} \phi \psi \, dx, \quad (\phi, \psi) = \sum_{j=1}^{N} \int_{I_j} \phi \psi \, dx.
\]

Also, the \( L^1 \) and \( L^2 \) norms over the interval \( I \) are given by \( || \cdot ||_1 \) and \( || \cdot || \), respectively.
Chapter 2

Local Discontinuous Galerkin

Methods for the Coupled BBM System

In this chapter, we discuss the derivation and development of LDG methods to solve the coupled BBM system (1.2.5), which conserve or dissipate the Hamiltonian functional (1.2.11). To that effect, we develop two different choices of numerical flux, where one choice (the alternating flux, to be defined in Section 2.1) conserves the Hamiltonian for long time simulations, and the second flux (upwind flux) adds numerical dissipation and has a Hamiltonian decreasing property. Optimal error estimates are derived for the single BBM equation (1.2.10) and for the linearized coupled BBM system. We have also experimented with two different types of time discretizations, the implicit second order midpoint rule which conserves the Hamiltonian, and the standard Runge-Kutta methods. Proof is
provided to verify that the LDG method with the alternating flux, coupled with the midpoint rule temporal discretization, and a special nonlinear term discretization, conserves the discrete Hamiltonian exactly. Note that the Hamiltonian (1.2.11) is only guaranteed to be non-negative when $1 + \eta \geq 0$. Under this condition, the stability of the LDG method follows from the Hamiltonian conservation (or dissipation). When such condition does not hold, we observe numerically that both Hamiltonian conserving and dissipative methods perform well. Numerical evidence is also provided to show that the Hamiltonian conserving methods have a smaller phase and shape error for the long time simulations.

The organization of this chapter is as follows. In Section 2.1, we present an LDG method for the problem with two choices of numerical fluxes. We then prove the Hamiltonian conservation and dissipation properties for the alternating and upwind numerical fluxes, respectively. Section 2.2 is devoted to the error estimate of the proposed methods for the linearized equations. Temporal discretizations, along with their conservation property, are discussed in Section 2.3. Section 2.4 contains various numerical experiments that show the optimal convergence rates, long time simulations, solitary wave generations and collisions.

### 2.1 Local Discontinuous Galerkin Method

We now describe the semi-discrete LDG method for (1.2.5) by discretizing the space with the LDG method, and leave the time continuous. Time discretizations will be discussed in Section 2.3. We start by rewriting (1.2.5) as a first order system:
Here we introduce two new variables $p, q$ to replace the nonlinear terms, for the purpose of designing Hamiltonian preserving methods later. Using the abbreviated notation and periodic boundary condition, we formulate the LDG method for the system (2.1.1) as the following: find $u_h, \eta_h, p_h, q_h, r_h, s_h, w_h, v_h \in V^k_h$, such that

\begin{align}
(w_t + (\eta + q)_x &= 0, \\
v_t + (u + p)_x &= 0, \\
w &= u - \frac{1}{6} r_x, \\
v &= \eta - \frac{1}{6} s_x, \\
r &= u_x, \\
s &= \eta_x, \\
q &= \frac{1}{2} u^2, \\
p &= \eta u. \\
\end{align}

(2.1.1)

Here we introduce two new variables $p, q$ to replace the nonlinear terms, for the purpose of designing Hamiltonian preserving methods later. Using the abbreviated notation and periodic boundary condition, we formulate the LDG method for the system (2.1.1) as the following: find $u_h, \eta_h, p_h, q_h, r_h, s_h, w_h, v_h \in V^k_h$, such that

\begin{align}
((w_h)_t, \phi) - (\eta_h + q_h, \phi_x) - \sum_{j=1}^{N} (\tilde{\eta}_h + \tilde{q}_h)_{j+\frac{1}{2}} [\phi]_{j+\frac{1}{2}} &= 0, \\
(w_h, \psi) &= (u_h, \psi) + \frac{1}{6} (r_h, \psi_x) + \frac{1}{6} \sum_{j=1}^{N} (\tilde{r}_h)_{j+\frac{1}{2}} [\psi]_{j+\frac{1}{2}}, \\
(r_h, \varphi) &= -(u_h, \varphi_x) - \sum_{j=1}^{N} (\tilde{u}_h)_{j+\frac{1}{2}} [\varphi]_{j+\frac{1}{2}}, \\
(q_h, \zeta) &= \left( \frac{1}{2} u_h^2, \zeta \right), \\
((v_h)_t, \rho) - (u_h + p_h, \rho_x) - \sum_{j=1}^{N} (\tilde{u}_h + \tilde{p}_h)_{j+\frac{1}{2}} [\rho]_{j+\frac{1}{2}} &= 0, \\
(v_h, \theta) &= (\eta_h, \theta) + \frac{1}{6} (s_h, \theta_x) + \frac{1}{6} \sum_{j=1}^{N} (\tilde{s}_h)_{j+\frac{1}{2}} [\theta]_{j+\frac{1}{2}}, \\
(s_h, \lambda) &= -(\eta_h, \lambda_x) - \sum_{j=1}^{N} (\tilde{\eta}_h)_{j+\frac{1}{2}} [\lambda]_{j+\frac{1}{2}}, \\
(p_h, \vartheta) &= (\eta_h u_h, \vartheta),
\end{align}

(2.1.2a-h)
for all test functions $\phi, \psi, \varphi, \zeta, \rho, \theta, \lambda, \vartheta \in V_h^k$. The hat and tilde terms $\hat{u}_h, \tilde{u}_h, \tilde{\eta}_h$, etc., known as numerical fluxes, are the boundary terms at each cell interface obtained from the integration by parts. In the next two subsections, we will consider two different sets of numerical fluxes, namely the alternating and upwind fluxes, with different conservation properties.

2.1.1 Alternating Flux

The choice of alternating flux can be summarized in the following groups

\[
\begin{align*}
\hat{u}_h &= u_h^+, \\
\hat{\eta}_h &= \eta_h^-, \\
\hat{r}_h &= r_h^- \\
\hat{s}_h &= s_h^+ \\
\tilde{u}_h + \hat{p}_h &= u_h^+ + p_h^+, \\
\tilde{\eta}_h + \hat{q}_h &= \eta_h^- + q_h^-
\end{align*}
\]

where we can choose one bracketed group from row 1 for $\hat{u}_h, \hat{\eta}_h$, and one group from row 2 for the others. Note that the indices $j + \frac{1}{2}$ are dropped for convenience, as all fluxes above
are computed at the same point at the cell boundary. It is emphasized that the choice of flux is not unique, as one can choose up to four different pairs of fluxes, as stated above. In this paper, we choose the ones on the left, namely,

\[
\hat{u}_h = u_h^+, \quad \hat{\eta}_h = \eta_h^-,
\]
\[
\tilde{u}_h + \tilde{p}_h = u_h^+ + p_h^+,
\]
\[
\tilde{\eta}_h + \tilde{q}_h = \eta_h^- + q_h^-,
\]
\[
\hat{r}_h = r_h^-,
\]
\[
\hat{s}_h = s_h^+,
\]
(2.1.3)

with the periodic boundary conditions on the boundary. We now turn to establish the conservation of the Hamiltonian \( E(\eta, u) \) defined in (1.2.11).

**Theorem 1** The Hamiltonian

\[
E_h(t) = \frac{1}{2} \int_I \left[ \eta_h^2 + (1 + \eta_h)u_h^2 \right] dx
\]
(2.1.4)

is conserved by the semi-discrete LDG method (2.1.2) with the choice of alternating flux (2.1.3).

**Proof.** Choosing \( \phi = u_h + p_h - (s_h)_t/6 \) and \( \rho = \eta_h + q_h - (r_h)_t/6 \) in (2.1.2a) and (2.1.2e) respectively, we deduce

\[
\left( (w_h)_t, u_h + p_h - \frac{1}{6}(s_h)_t \right) - \left( \eta_h + q_h, \left( u_h + p_h - \frac{1}{6}(s_h)_t \right)_x \right) - \sum_{j=1}^{N} (\eta_h^- + q_h^-)_{j+\frac{1}{2}} \left[ u_h + p_h - \frac{1}{6}(s_h)_t \right]_{j+\frac{1}{2}} = 0, \quad (2.1.5a)
\]
\[
\left( (v_h)_t, \eta_h + q_h - \frac{1}{6}(r_h)_t \right) - \left( u_h + p_h, \left( \eta_h + q_h - \frac{1}{6}(r_h)_t \right)_x \right) - \sum_{j=1}^{N} (u_h^+ + p_h^+)_j \left[ \eta_h + q_h - \frac{1}{6}(r_h)_t \right]_{j+\frac{1}{2}} = 0. \quad (2.1.5b)
\]
Next, we take the time derivative of (2.1.2b) and (2.1.2f), and choose $\psi = u_h + p_h$ and $\theta = \eta_h + q_h$ to get

$$((w_h)_t, u_h + p_h) = ((u_h)_t, u_h + p_h) + \left(\frac{1}{6}(r_h)_t, (u_h + p_h)_x\right) + \frac{1}{6} \sum_{j=1}^{N} ( (r_h)_t )^-_{j+\frac{1}{2}} [u_h + p_h]_{j+\frac{1}{2}},$$

$$(2.1.6a)$$

$$((v_h)_t, \eta_h + q_h) = ((\eta_h)_t, \eta_h + q_h) + \left(\frac{1}{6}(s_h)_t, (\eta_h + q_h)_x\right) + \frac{1}{6} \sum_{j=1}^{N} ( (s_h)_t )^+_{j+\frac{1}{2}} [\eta_h + q_h]_{j+\frac{1}{2}}.$$  

$$(2.1.6b)$$

We add the equations in (2.1.5) and (2.1.6) respectively, and subtract the resulting equations to obtain

$$-\frac{1}{6} ((w_h)_t, (s_h)_t) - \frac{1}{6} ((v_h)_t, (r_h)_t) + ((u_h)_t, u_h + p_h) + ((\eta_h)_t, \eta_h + q_h)$$

$$= \frac{1}{6} \sum_{j=1}^{N} [( (r_h)_t (u_h + p_h) ] - [(r_h)_t] (u_h^+ + p_h^+) - (r_h)_t^- [u_h + p_h] ]_{j+\frac{1}{2}}$$

$$+ \frac{1}{6} \sum_{j=1}^{N} [( (\eta_h + q_h)(s_h)_t ] - [\eta_h + q_h] (s_h^+_t) - (\eta_h^- + q_h^-) [(s_h)_t] ]_{j+\frac{1}{2}}$$

$$- \sum_{j=1}^{N} [( (\eta_h + q_h)(u_h + p_h) ] - [\eta_h + q_h] (u_h^+ + p_h^+) - (\eta_h^- + q_h^-) [u_h + p_h] ]_{j+\frac{1}{2}}$$

$$= 0,$$  

$$(2.1.7)$$

after we integrate the complete derivative out.

Next, we take the time derivative of (2.1.2b) and (2.1.2f), and choose $\psi = (s_h)_t$ and $\theta = (r_h)_t$ to get

$$((w_h)_t, (s_h)_t) = ((u_h)_t, (s_h)_t) + \frac{1}{6} ((r_h)_t, (s_h)_{tx}) + \frac{1}{6} \sum_{j=1}^{N} ( (r_h)_t )^-_{j+\frac{1}{2}} [(s_h)_t]_{j+\frac{1}{2}},$$

$$(2.1.8a)$$

$$((v_h)_t, (r_h)_t) = ((\eta_h)_t, (r_h)_t) + \frac{1}{6} ((s_h)_t, (r_h)_{tx}) + \frac{1}{6} \sum_{j=1}^{N} ( (s_h)_t )^+_{j+\frac{1}{2}} [(r_h)_t]_{j+\frac{1}{2}},$$

$$(2.1.8b)$$
Similarly, taking \( \varphi = (\eta_h)_t \) and \( \lambda = (u_h)_t \) in the time derivative of (2.1.2c) and (2.1.2g):

\[
((r_h)_t, (\eta_h)_t) = -((u_h)_t, (\eta_h)_{tx}) - \sum_{j=1}^{N} ((u_h)_t)_{j+\frac{1}{2}}^-([(\eta_h)_t]_{j+\frac{1}{2}}),
\]

(2.1.9a)

\[
((s_h)_t, (u_h)_t) = -((\eta_h)_t, (u_h)_{tx}) - \sum_{j=1}^{N} ((\eta_h)_t)_{j+\frac{1}{2}}^-([(u_h)_t]_{j+\frac{1}{2}}).
\]

(2.1.9b)

Adding the equations in (2.1.8) and (2.1.9) together and using integration by parts, we obtain

\[
((w_h)_t, (s_h)_t) + ((v_h)_t, (r_h)_t) = \frac{1}{6} \sum_{j=1}^{N} \left( [(r_h)_t(s_h)_t] - [(r_h)_t](s_h)_t - [(s_h)_t] - (r_h)_t [(s_h)_t]_{j+\frac{1}{2}} \right)
+ \sum_{j=1}^{N} \left( [(\eta_h)_t(u_h)_t] - [(\eta_h)_t] (u_h)_t - (\eta_h)_t [(u_h)_t]_{j+\frac{1}{2}} \right)
= 0.
\]

(2.1.10)

Combining the equations (2.1.7) and (2.1.10), we have

\[
((u_h)_t, u_h) + ((\eta_h)_t, \eta_h) + ((u_h)_t, p_h) + ((\eta_h)_t, q_h) = 0,
\]

(2.1.11)

Therefore,

\[
\frac{d}{dt} \frac{1}{2} \int \left( \eta_h^2 + (1 + \eta_h) u_h^2 \right) \, dx = ((\eta_h)_t, \eta_h) + ((u_h)_t, u_h) + (u_h(u_h)_t, \eta_h) + ((\eta_h)_t, u_h^2/2)
= ((u_h)_t, u_h) + ((\eta_h)_t, \eta_h) + ((u_h)_t, p_h) + ((\eta_h)_t, q_h) = 0
\]

(2.1.12)

where the second equality comes from the equations (2.1.2b) and (2.1.2f).

\[ \blacksquare \]

### 2.1.2 Upwind Flux

The other set of numerical flux is the upwind flux, which is widely used for hyperbolic conservation laws to provide numerical dissipation. The choice of upwind flux for (1.2.5)
can be summarized as
\[
\tilde{u}_h = \{u_h\} - \frac{1}{2}[\eta_h], \quad \tilde{\eta}_h = \{\eta_h\} - \frac{1}{2}[u_h], \\
\hat{q}_h = \{q_h\} - \frac{1}{2}[p_h], \quad \hat{p}_h = \{p_h\} - \frac{1}{2}[q_h], \\
(\tilde{r}_h)_t = \{(r_h)_t\} - \frac{1}{2}[(s_h)_t], \quad (\tilde{s}_h)_t = \{(s_h)_t\} - \frac{1}{2}[(r_h)_t], \\
(\tilde{u}_h)_t = \{(u_h)_t\} + \frac{1}{2}[(\eta_h)_t], \quad (\tilde{\eta}_h)_t = \{(\eta_h)_t\} + \frac{1}{2}[(u_h)_t],
\]

(2.1.13)

With the above choices, we have the following theorem that shows the dissipation of the Hamiltonian \(E(\eta, u)\) over time.

**Theorem 2** For the semi-discrete LDG method (2.1.2) with the choice of upwind flux (2.1.13), the Hamiltonian \(E_h(t)\) satisfies
\[
\frac{d}{dt} E_h(t) = \frac{1}{2} \frac{d}{dt} \int_\mathbb{R} \left[ \eta_h^2 + (1 + \eta_h)u_h^2 \right] \, dx \leq 0,
\]

(2.1.14)

for all time.

**Proof.** We begin with the LDG method given by (2.1.2). In equations (2.1.2a) and (2.1.2e), the test functions \(\phi, \rho\) are taken to be \(\phi = u_h + p_h - \frac{1}{6}(s_h)_t\) and \(\rho = \eta_h + q_h - \frac{1}{6}(r_h)_t\). Summing up the resulting equations yields
\[
\left( (w_h)_t, u_h + p_h - \frac{1}{6}(s_h)_t \right) + \left( (v_h)_t, \eta_h + q_h - \frac{1}{6}(r_h)_t \right) \\
= \left( \eta_h + q_h, \left( u_h + p_h - \frac{1}{6}(s_h)_t \right)_x \right) + \sum_{j=1}^N (\tilde{\eta}_h + \hat{q}_h)_{j+\frac{1}{2}} \left[ u_h + p_h - \frac{1}{6}(s_h)_t \right]_{j+\frac{1}{2}} \\
+ \left( u_h + p_h, \left( \eta_h + q_h - \frac{1}{6}(r_h)_t \right)_x \right) + \sum_{j=1}^N (\tilde{u}_h + \hat{p}_h)_{j+\frac{1}{2}} \left[ \eta_h + q_h - \frac{1}{6}(r_h)_t \right]_{j+\frac{1}{2}}.
\]

(2.1.15)
Next, we take the sum of the time derivative of (2.1.2b) and (2.1.2f) with the test functions
\[ \psi = u_h + p_h - \frac{1}{6} (s_h)_t \text{ and } \theta = \eta_h + q_h - \frac{1}{6} (r_h)_t, \]
which leads to
\[
\begin{align*}
\left( (w_h)_t, u_h + p_h - \frac{1}{6} (s_h)_t \right) + \left( (v_h)_t, \eta_h + q_h - \frac{1}{6} (r_h)_t \right) \\
= \left( (\eta_h)_t, \eta_h + q_h - \frac{1}{6} (r_h)_t \right) + \frac{1}{6} \left( (s_h)_t, \left( \eta_h + q_h - \frac{1}{6} (r_h)_t \right)_x \right) \\
+ \frac{1}{6} \sum_{j=1}^{N} ((\hat{s}_h)_t)_{j+\frac{1}{2}} \left[ \eta_h + q_h - \frac{1}{6} (r_h)_t \right]_{j+\frac{1}{2}} \\
+ \left( (u_h)_t, u_h + p_h - \frac{1}{6} (s_h)_t \right) + \frac{1}{6} \left( (r_h)_t, \left( u_h + p_h - \frac{1}{6} (s_h)_t \right)_x \right) \\
+ \frac{1}{6} \sum_{j=1}^{N} ((\hat{r}_h)_t)_{j+\frac{1}{2}} \left[ u_h + p_h - \frac{1}{6} (s_h)_t \right]_{j+\frac{1}{2}}.
\end{align*}
\] (2.1.16)

Subtracting (2.1.15) from (2.1.16), as well as regrouping these terms, we have
\[
\begin{align*}
\left( (\eta_h)_t, \eta_h + q_h - \frac{1}{6} (r_h)_t \right) + \left( (u_h)_t, u_h + p_h - \frac{1}{6} (s_h)_t \right) \\
+ \frac{1}{6} \left( (s_h)_t, (\eta_h + q_h)_x \right) + \frac{1}{6} \left( \eta_h + q_h, (s_h)_t \right) + \frac{1}{6} \sum_{j=1}^{N} ((\hat{s}_h)_t) \left[ \eta_h + q_h \right] + ((\hat{\eta}_h + \hat{q}_h) \left[ (s_h)_t \right])_{j+\frac{1}{2}} \\
+ \frac{1}{6} \left( (r_h)_t, (u_h + p_h)_x \right) + \frac{1}{6} \left( u_h + p_h, (r_h)_t \right) + \frac{1}{6} \sum_{j=1}^{N} ((\hat{r}_h)_t) \left[ u_h + p_h \right] + ((\hat{u}_h + \hat{p}_h) \left[ (r_h)_t \right])_{j+\frac{1}{2}} \\
- \frac{1}{36} \left( (s_h)_t, (r_h)_t x \right) - \frac{1}{36} \left( r_h, (s_h)_t x \right) - \frac{1}{36} \sum_{j=1}^{N} ((\hat{s}_h)_t) \left[ (r_h)_t \right] + ((\hat{r}_h) \left[ (s_h)_t \right])_{j+\frac{1}{2}} \\
- (\eta_h + q_h, (u_h + p_h)_x) - (u_h + p_h, (\eta_h + q_h)_x) \\
- \sum_{j=1}^{N} ((\hat{\eta}_h + \hat{q}_h) \left[ u_h + p_h \right] + ((\hat{\eta}_h + \hat{q}_h) \left[ (s_h)_t \right])_{j+\frac{1}{2}}.
\end{align*}
\] (2.1.17)
Using the choice of upwind flux for $\bar{\eta}_h$, $\bar{u}_h$, $\bar{q}_h$, $\bar{p}_h$, $(\bar{r}_h)_t$, and $(\bar{s}_h)_t$ in (2.1.13), equation (2.1.17) reduces to

\[
\left((\eta_h)_t, \eta_h + q_h - \frac{1}{6}(r_h)_t\right) + \left((u_h)_t, u_h + p_h - \frac{1}{6}(s_h)_t\right) + \sum_{j=1}^{N} \left(\frac{1}{6}[u_h + p_h][(s_h)_t] - \frac{1}{6}[\eta_h + q_h][(r_h)_t] + \frac{1}{72}[(r_h)_t]^2 + \frac{1}{72}[(s_h)_t]^2 + \frac{1}{2}[u_h + p_h]^2 + \frac{1}{2}[\eta_h + q_h]^2 \right)_{j+\frac{1}{2}} = 0. \tag{2.1.18}
\]

Now, we take the test functions $\varphi = \frac{1}{6}(\eta_h)_t$ and $\lambda = \frac{1}{6}(u_h)_t$ in the time derivative of (2.1.2c) and (2.1.2g), and sum up the resulting equations to obtain

\[
\frac{1}{6}((r_h)_t, (\eta_h)_t) + \frac{1}{6}((s_h)_t, (u_h)_t) = -\frac{1}{6}((u_h)_t, (\eta_h)_t),\frac{1}{6}(u_h)_t, (u_h)_t) - \frac{1}{6} \sum_{j=1}^{N} \left(\frac{1}{6}[(u_h)_t][(s_h)_t] - \frac{1}{6}[\eta_h + q_h][(r_h)_t] + \frac{1}{12}[(u_h)_t]^2 + \frac{1}{12}[(\eta_h)_t]^2 \right)_{j+\frac{1}{2}} = 0, \tag{2.1.19}
\]

with the choices of flux for $(\bar{u}_h)_t$ and $(\bar{\eta}_h)_t$ in (2.1.13). Summing up (2.1.18) and (2.1.19), we obtain

\[
((\eta_h)_t, \eta_h + q_h) + ((u_h)_t, u_h + p_h) + \sum_{j=1}^{N} \left(\frac{1}{6}[u_h + p_h][(s_h)_t] - \frac{1}{6}[\eta_h + q_h][(r_h)_t] + \frac{1}{12}[(u_h)_t]^2 + \frac{1}{12}[(\eta_h)_t]^2 \right) + \frac{1}{72}[(r_h)_t]^2 + \frac{1}{72}[(s_h)_t]^2 + \frac{1}{2}[u_h + p_h]^2 + \frac{1}{2}[\eta_h + q_h]^2 \right)_{j+\frac{1}{2}} = 0, \tag{2.1.20}
\]

which leads to

\[
((\eta_h)_t, \eta_h + q_h) + ((u_h)_t, u_h + p_h) + \frac{1}{12} \sum_{j=1}^{N} [(u_h)_t]^2 + \frac{1}{12} \sum_{j=1}^{N} [(\eta_h)_t]^2 \leq 0, \tag{2.1.21}
\]
by Young’s inequality. The same procedure in the alternating flux proof can be used to establish the following inequality

\[
\frac{d}{dt} \frac{1}{2} \int_I \left( \eta_h^2 + (1 + \eta_h) u_h^2 \right) \, dx = ((u_h)_t, u_h) + ((\eta_h)_t, \eta_h) + ((u_h), p_h) + ((\eta_h), q_h)
\leq - \frac{1}{12} \sum_{j=1}^N [(u_h)_t]^2 - \frac{1}{12} \sum_{j=1}^N [(\eta_h)_t]^2 \leq 0,
\] (2.1.22)

which completes the proof. ■

2.2 Error Estimate for the Linearized Systems

In this section, we provide optimal error estimate for the linearized coupled BBM systems when the alternating flux is used. The linearized coupled systems take the form of

\[
\begin{cases}
\eta_t + u_x - \eta_{xxt} = 0, \\
u_t + \eta_x - u_{xxt} = 0,
\end{cases}
(x, t) \in [a_0, a_1] \times [0, T],
\] (2.2.23)

where we have suppressed the constant coefficient $1/6$ for the ease of presentation. The LDG methods (2.1.2) then reduce to

\[
\begin{align*}
((w_h)_t, \phi) - (\eta_h, \phi_x) - \sum_{j=1}^N ((\tilde{\eta}_h) [\phi])_{j+\frac{1}{2}} &= 0, & ((v_h)_t, \rho) - (u_h, \rho_x) - \sum_{j=1}^N ((\tilde{u}_h) [\rho])_{j+\frac{1}{2}} &= 0, \\
(w_h, \psi) &= (u_h, \psi) + (r_h, \psi_x) + \sum_{j=1}^N ((\tilde{r}_h) [\psi])_{j+\frac{1}{2}}, \\
(v_h, \theta) &= (\eta_h, \theta) + (s_h, \theta_x) + \sum_{j=1}^N ((\tilde{s}_h) [\theta])_{j+\frac{1}{2}}, \\
(r_h, \varphi) &= -(u_h, \varphi_x) - \sum_{j=1}^N ((\tilde{u}_h) [\varphi])_{j+\frac{1}{2}}, & (s_h, \lambda) &= -(\eta_h, \lambda_x) - \sum_{j=1}^N ((\tilde{\eta}_h) [\lambda])_{j+\frac{1}{2}}. & (2.2.24)
\end{align*}
\]
Let us introduce three projections to be used later. The first projection is the standard piecewise $L^2$ projection, which will be denoted as $P$, with
\[ \int_{I_j} (Pw(x) - w(x)) v(x) \, dx = 0 \quad \text{for all } v \in P^k(I_j), \quad j = 1, \ldots, N. \] (2.2.25)

We also define $P^+$ as a projection such that for a function $w$, $P^+w$ is the unique function in $V_h^k$ satisfying
\[ \int_{I_j} (P^+w(x) - w(x)) v(x) \, dx = 0 \quad \text{for all } v \in P^{k-1}(I_j), \] (2.2.26)
and
\[ P^+w \left( x^+_{j+\frac{1}{2}} \right) = w \left( x^-_{j+\frac{1}{2}} \right) \quad \text{for all } j. \] (2.2.27)

Similarly, $P^-$ is defined as
\[ \int_{I_j} (P^-w(x) - w(x)) v(x) \, dx = 0 \quad \text{for all } v \in P^{k-1}(I_j), \] (2.2.28)
and
\[ P^-w \left( x^-_{j+\frac{1}{2}} \right) = w \left( x^+_{j+\frac{1}{2}} \right) \quad \text{for all } j. \] (2.2.29)

All the projections mentioned previously satisfy the property (see [70] for details)
\[ ||\epsilon^w|| + h||\epsilon^w||_{\infty} + h^{\frac{1}{2}}||\epsilon^w||_{\Gamma_h} \leq Ch^{k+1}, \] (2.2.30)
where $\epsilon^w = Pw - w$ or $\epsilon^w = P^\pm w - w$, $\Gamma_h$ denotes the boundary points of all elements $I_j$, and $C$ is a constant depending on $w$ and independent of $h$.

We now provide the following theorem on the optimal error estimate for the linearized case when the alternating flux is used. One simplification, compared with the nonlinear problem, is that the linearized system yields an additional conserved quantity
\[ \int_I (\eta^2 + u^2 + \eta_x^2 + u_x^2) \, dx, \] (2.2.31)
which is helpful in the proof of the error estimate presented below. The approach presented here cannot be easily generalized to prove the optimal error estimate of the nonlinear systems, as the quantity (2.2.31) is not conserved by the nonlinear systems, therefore new ideas are needed for their error estimate.

**Theorem 3** Let \( u, \eta \) be the exact solutions to (2.2.23), and \( u_h, \eta_h \) be the numerical solution of the LDG scheme (2.2.24). Assume \( u, \eta \) have enough regularity, there holds the following error estimate:

\[
||u - u_h|| \leq C h^{k+1}, \quad ||\eta - \eta_h|| \leq C h^{k+1},
\]

(2.2.32)

where the constant \( C \) depends on the final time \( T \), \( k \) and the \( H^{k+1} \) norm of \( u, \eta \) up to time \( T \).

**Proof.** Let’s define the operators \( L(f, g) \) and \( R(f, g) \) as

\[
L(f, g) = (f, g_x) + \sum_{j=0}^{N} f_{j+\frac{1}{2}}^+[g]_{j+\frac{1}{2}}, \quad R(f, g) = (f, g_x) + \sum_{j=0}^{N} f_{j+\frac{1}{2}}^-[g]_{j+\frac{1}{2}}.
\]

(2.2.33)

Observe the following results of the above operators:

\[
L(f, g) + R(g, f) = 0,
\]

(2.2.34)

and

\[
L(\omega - \mathcal{P}^+ \omega, g) = 0, \quad R(\omega - \mathcal{P}^- \omega, g) = 0, \quad \text{for all } g \in V_h^k.
\]

(2.2.35)

We will denote the following notations for the error estimate: \( \xi^\omega = \mathcal{P} \omega - \omega_h, \epsilon^\omega = \mathcal{P} \omega - \omega, \) and \( e^\omega = \omega - \omega_h = \xi^\omega - \epsilon^\omega, \) where \( \omega \) stands for \( u, \eta, \) etc., and \( \mathcal{P} \) is a projection that is to be chosen later in the proof. With these shorthand notations, we can write out the error
equations based of the LDG method (2.2.24) for the linearized case

\[(e^w_t, \phi) = \mathcal{L}(e^\eta, \phi), \quad (2.2.36a)\]
\[(e^w, \psi) = (e^u, \psi) + \mathcal{L}(e^r, \psi), \quad (2.2.36b)\]
\[(e^r, \varphi) = -\mathcal{R}(e^u, \varphi), \quad (2.2.36c)\]
\[(e^v_t, \rho) = \mathcal{R}(e^u, \rho), \quad (2.2.36d)\]
\[(e^v, \theta) = (e^\eta, \theta) + \mathcal{R}(e^s, \theta), \quad (2.2.36e)\]
\[(e^s, \vartheta) = -\mathcal{L}(e^\eta, \vartheta). \quad (2.2.36f)\]

Subtract (2.2.36a) from the time derivative of (2.2.36b), and take the test functions \(\phi = \psi = \xi^u\) to obtain

\[(e^u_t, \xi^u) + \mathcal{L}(e^r_t, \xi^u) - \mathcal{L}(e^\eta, \xi^u) = 0.\]

Applying the same procedure to (2.2.36d) and (2.2.36e) with the test functions \(\phi = \psi = \xi^\eta\) yields

\[(e^\eta_t, \xi^\eta) + \mathcal{R}(e^s_t, \xi^\eta) - \mathcal{R}(e^u, \xi^\eta) = 0.\]

Taking the test functions \(\varphi = \xi^r_t, \vartheta = \xi^s_t\) in (2.2.36c) and (2.2.36f) gives

\[(e^r_t, \xi^r_t) + \mathcal{R}(e^u_t, \xi^r_t) + (e^s, \xi^s_t) + \mathcal{L}(e^\eta, \xi^s_t) = 0.\]
Combining these equations and separating the error term \( e^\omega \) as \( \xi^\omega - e^\omega \), we have

\[
\begin{align*}
(\xi^t_l, \xi^u) + (\xi^u, \xi^u) + (\xi^r, \xi^r_t) + (\xi^s, \xi^s_t) \\
= (\epsilon^t_l, \xi^u) + (\epsilon^u, \xi^u) + (\epsilon^r_t, \xi^r_t) + (\epsilon^s_t, \xi^s_t) \\
+ \mathcal{L}(\epsilon^r_t, \xi^u) + \mathcal{R}(\epsilon^s_t, \xi^u) - \mathcal{L}(\epsilon^u, \xi^u) - \mathcal{R}(\epsilon^u, \xi^u) + \mathcal{R}(\epsilon^s_t, \xi^s_t) + \mathcal{L}(\epsilon^u, \xi^s_t) \\
- \mathcal{L}(\xi^r_t, \xi^u) - \mathcal{R}(\xi^s_t, \xi^u) + \mathcal{L}(\xi^s_t, \xi^u) + \mathcal{R}(\xi^u, \xi^u) - \mathcal{R}(\xi^u, \xi^r_t) - \mathcal{L}(\xi^r_t, \xi^s_t),
\end{align*}
\]

(2.2.37)

where the last line disappears following (2.2.34). If we take the projections of the variables as:

\[
\mathbb{P}_r = \mathbb{P}_r \quad \mathbb{P}_s = \mathbb{P}_s, \quad \mathbb{P}u = \mathbb{P}^{-} u \quad \mathbb{P}\eta = \mathbb{P}^{+} \eta, \quad (2.2.38)
\]

and utilize the relation (2.2.35), the error equation (2.2.37) becomes

\[
\frac{1}{2} \frac{d}{dt} \left( ||\xi^\eta||^2 + ||\xi^u||^2 + ||\xi^r||^2 + ||\xi^s||^2 \right) = (\epsilon^t_l, \xi^u) + (\epsilon^u, \xi^u) + \mathcal{L}(\epsilon^r_t, \xi^u) + \mathcal{R}(\epsilon^s_t, \xi^u). \quad (2.2.39)
\]

Applying the results in [68, Lemma 2.4] to equations (2.2.36c) and (2.2.36f), we have the following estimates

\[
||\xi^\eta|| + h^{-1/2} ||\xi^\eta||_{\mathbb{G}^h} \leq C ||\xi^\delta||, \quad ||\xi^u|| + h^{-1/2} ||\xi^u||_{\mathbb{G}^h} \leq C ||\xi^r||. \quad (2.2.40)
\]

Therefore, we have

\[
\frac{1}{2} \frac{d}{dt} \left( ||\xi^\eta||^2 + ||\xi^u||^2 + ||\xi^r||^2 + ||\xi^s||^2 \right) \leq ||\xi^\eta||^2 + ||\xi^u||^2 + ||\xi^r||^2 + ||\xi^s||^2 + Ch^{k+1}, \quad (2.2.41)
\]

and the optimal error estimate (2.2.32) follows from the Gronwall’s inequality. ■
2.3 Time Discretization

In this section, we present two different temporal discretizations that will be used to discretize the semi-discrete LDG methods (2.1.2) in time.

2.3.1 SSPRK Methods

In applications, higher order time discretizations are often required. A standard choice of high order time discretization is the well known class of SSPRK methods presented in [44]. Such methods pose nonlinear stability properties in the discretization of hyperbolic conservation laws. A variety of methods are presented in [44] including explicit SSP methods, implicit SSP methods, as well as multi-step SSPRK methods. In our numerical examples, we will use the five stages SSPRK4 scheme that is provided in [44], and developed in [64], given by

\[
\begin{align*}
  u_h^{(1)} &= u_h^n + 0.391752226571890 \Delta t \mathcal{L}(u_h^n), \\
  u_h^{(2)} &= 0.444370493651235u_h^n + 0.555629506348765u_h^{(1)} + 0.368410593050371\Delta t \mathcal{L}(u_h^{(1)}), \\
  u_h^{(3)} &= 0.620101851488403u_h^n + 0.379898148511597u_h^{(2)} + 0.251891774271694\Delta t \mathcal{L}(u_h^{(2)}), \\
  u_h^{(4)} &= 0.178079954393132u_h^n + 0.821920045606868u_h^{(3)} + 0.544974750228521\Delta t \mathcal{L}(u_h^{(3)}), \\
  u_h^{n+1} &= 0.517231671970585u_h^{(2)} + 0.096059710526147u_h^{(3)} + 0.063692468666290\Delta t \mathcal{L}(u_h^{(3)}), \\
  &\quad + 0.386708617503269u_h^{(4)} + 0.226007483236906\Delta t \mathcal{L}(u_h^{(4)}),
\end{align*}
\]

where \( \mathcal{L}(\cdot) \) is the spatial discretization, which is taken to be the LDG method.
2.3.2 Midpoint Rule

As the SSPRK method dissipates the Hamiltonian, in this subsection, we describe an alternative midpoint rule time stepping scheme, which can conserve the discrete Hamiltonian in time.

Let \( \{t_n\}_{n=0}^M \) be a partition of \([0, T]\) where \( \Delta t^n = t^{n+1} - t^n \). Denote \( \omega^n_h \in V_h \) (where \( \omega = u, \eta, w, v \)) as the numerical solution at time step \( n \). We update in time via the following equation

\[
\omega_{n+1}^h = 2\omega_n^1 - \omega_n^h,
\]

where \( \omega_n^1 \) are the solutions of the following equations

\[
(D w_n^h, \phi) - (\eta_n^1 + p_n^1, \phi_x) - \sum_{j=1}^{N} (\tilde{u}_n^1 + \tilde{q}_n^1)_{j+\frac{1}{2}} [\phi]_{j+\frac{1}{2}} = 0,
\]

\[
(w_n^1, \psi) - (u_n^1, \psi) - \frac{1}{6} (r_n^1, \psi_x) - \frac{1}{6} \sum_{j=1}^{N} (\tilde{r}_n^1)_{j+\frac{1}{2}} [\psi]_{j+\frac{1}{2}} = 0,
\]

\[
(r_n^1, \varphi) + (u_n^1, \varphi_x) + \sum_{j=1}^{N} (\tilde{u}_n^1)_{j+\frac{1}{2}} [\varphi]_{j+\frac{1}{2}} = 0,
\]

\[
(q_n^1, \zeta) - \left( \frac{1}{4} \left( (u_n^1)^2 + (u_n^{1+1})^2 \right), \zeta \right) = 0,
\]

\[
(D v_n^h, \rho) - (u_n^1 + p_n^1, \rho_x) - \sum_{j=1}^{N} (\tilde{u}_n^1 + \tilde{p}_n^1)_{j+\frac{1}{2}} [\rho]_{j+\frac{1}{2}} = 0,
\]

\[
(u_n^1, \theta) - (\eta_n^1, \theta) - \frac{1}{6} (s_n^1, \theta_x) - \frac{1}{6} \sum_{j=1}^{N} (\tilde{s}_n^1)_{j+\frac{1}{2}} [\theta]_{j+\frac{1}{2}} = 0,
\]

\[
(s_n^1, \lambda) + (\eta_n^1, \lambda_x) + \sum_{j=1}^{N} (\tilde{\eta}_n^1)_{j+\frac{1}{2}} [\lambda]_{j+\frac{1}{2}} = 0,
\]

\[
(p_n^1, \vartheta) - (\eta_n^1, u_n^1, \vartheta) = 0,
\]
for all test functions $\phi, \psi, \varphi, \zeta, \rho, \theta, \lambda, \vartheta \in V_h^k$ and the notation $D$ defined as
\[
D\omega^n_h = \frac{\omega^{n+1}_h - \omega^n_h}{\Delta t^n} = \frac{\omega^{n,1}_h - \omega^n_h}{\Delta t^n/2}.
\] (2.3.44)

Notice that special care is needed in the implementation of the midpoint rule on the non-linear terms in (2.3.43d), as the straightforward approach
\[
(q^1_h, \zeta) - \left( \frac{1}{2} (u^1_h)^2, \zeta \right) = 0,
\]
will not yield a conservative time discretization. In the following theorem, we will show that the discrete Hamiltonian is conserved exactly by the proposed LDG methods with alternating fluxes and midpoint rule time discretization.

**Theorem 4** The solution to the midpoint rule LDG method (2.3.42) and (2.3.43), with the choice of alternating flux (2.1.3), conserves the discrete Hamiltonian functional
\[
E^n_h = \int \left[ (\eta^n_h)^2 + (1 + (\eta^n_h))(u^n_h)^2 \right] dx
\] (2.3.45)
for all $n$.

**Proof.** Choosing $\phi = u^1_h + p^1_h - \frac{1}{6} D s_h^n$ and $\rho = \eta^1_h + q^1_h - \frac{1}{6} D r_h^n$ in (2.3.43a) and (2.3.43e) respectively, we get
\[
\left( D w^n_h, u^n_h + p^n_h - \frac{1}{6} D s^n_h \right) - \left( \left( \eta^n_h + q^n_h \right), \left( u^n_h + p^n_h - \frac{1}{6} D s^n_h \right) \right)_x
- \sum_{j=0}^N (\tilde{\eta}^n_h + \tilde{q}^n_h) \left[ u^n_h + p^n_h - \frac{1}{6} D s^n_h \right] = 0,
\] (2.3.46a)
\[
\left( D v^n_h, \eta^n_h + q^n_h - \frac{1}{6} D r^n_h \right) - \left( \left( u^n_h + p^n_h \right), \left( \eta^n_h + q^n_h - \frac{1}{6} D r^n_h \right) \right)_x
- \sum_{j=0}^N (\tilde{u}^n_h + \tilde{p}^n_h) \left[ \eta^n_h + q^n_h - \frac{1}{6} D r^n_h \right] = 0.
\] (2.3.46b)
Then, following the exact same steps as in the proof of Theorem 1, we can derive

\[(\mathcal{D}u_h^n, u_h^{n,1}) + (\mathcal{D}q_h^n, p_h^{n,1}) + (\mathcal{D}\eta_h^n, \eta_h^{n,1}) + (\mathcal{D}q_h^n, q_h^{n,1}) = 0, \]

(2.3.47)

which is an analogue of equation (2.1.11). From the definition of \(\mathcal{D}\), we obtain

\[\mathcal{D}u_h^n u_h^{n,1} = \frac{u_h^{n+1} - u_h^n u_h^{n+1} + u_h^n}{\Delta t^n} = \frac{(u_h^{n+1})^2 - (u_h^n)^2}{2\Delta t^n}, \quad \mathcal{D}\eta_h^n \eta_h^{n,1} = \frac{(\eta_h^{n+1})^2 - (\eta_h^n)^2}{2\Delta t^n},\]

and also

\[\begin{align*}
(\mathcal{D}u_h^n, p_h^{n,1}) + (\mathcal{D}q_h^n, q_h^{n,1}) &= (\eta_h^{n+1} u_h^{n,1}, \mathcal{D}u_h^n) + \left(\frac{1}{4} \left( (u_h^n)^2 + (u_h^{n+1})^2 \right), \mathcal{D}q_h^n \right) \\
&= \left( \frac{\eta_h^{n+1} + q_h^n u_h^{n+1} + q_h^n}{2}, \frac{u_h^{n+1} - u_h^n}{\Delta t^n} \right) + \left( \frac{1}{4} \left( (u_h^n)^2 + (u_h^{n+1})^2 \right), \frac{\eta_h^{n+1} - \eta_h^n}{\Delta t^n} \right) \\
&= \frac{\eta_h^{n+1} (u_h^{n+1})^2 - \eta_h^n (u_h^n)^2}{2\Delta t^n},
\end{align*}\]

where the approximation of nonlinear terms in (2.3.43d) and (2.3.43h) is used in the first equality. Combining these together, we conclude the result that \(\mathcal{E}_h^{n+1} = \mathcal{E}_h^n\).

### 2.4 Numerical Experiments

In this section numerical results are presented for the proposed LDG method (2.1.2) for the coupled BBM system (1.2.5). We will numerically test or validate the issues including the convergence rate, Hamiltonian conservation or dissipation, and their long time behavior. The third order finite element LDG methods (polynomials of degree \(k = 2\)), coupled with the two different types of time discretizations given in the previous section, are implemented in the numerical experiments.
To check accuracy and convergence rates, exact solutions are needed. One set of exact solutions is the traveling wave solutions presented in [27]

\[ u(x, t) = 3k \text{sech}^2 \left( \frac{3}{\sqrt{10}} (x - kt - x_0) \right), \]
\[ \eta(x, t) = \frac{15}{4} \left( -2 + \cosh \left( 3\sqrt{\frac{2}{5}} (x - kt - x_0) \right) \right) \text{sech}^4 \left( \frac{3}{\sqrt{10}} (x - kt - x_0) \right), \]  
(2.4.48)

where \( k = \pm \frac{5}{2} \), and \( x_0 \) denote the center of the wave at \( t = 0 \). The initial condition can be obtained from (2.4.48), i.e., \( u_0(x) = u(x, 0) \) and \( \eta_0(x) = \eta(x, 0) \).

### 2.4.1 Accuracy Test

The accuracy of our proposed LDG methods will be tested for a combination of numerical fluxes and time discretizations. The exact traveling wave solutions (2.4.48) on the domain \([0, L]\) with \( L = 40 \) are used. We first test the LDG method with the alternating fluxes, combined with both fourth order RK method and the second order midpoint rule. The time step size of midpoint rule is taken as \( \Delta t = c \Delta x^2 \) to be consistent with the fourth order RK method. Their numerical errors and the orders of convergence of the variables \( \eta_h \) and \( u_h \) are listed in Tables 3.4.1-2.4.2. The exact parameters for each test case, including the time step size, spatial step size, are given in the caption of each table. From the tables, we can clearly see that the proposed LDG methods with alternating flux achieve the optimal convergence rates of 3 if the \( P^2 \) elements are used. To test the accuracy of the proposed LDG method with upwind flux and fourth order RK method, we implement it on the exact traveling wave solutions (2.4.48). We document the \( L^1 \) error of the LDG solutions and list the results in Table 2.4.3. Again, we observe the desired third order rate of convergence.
Table 2.4.1: Convergence rate test for the alternating flux in space and SSPRK4 in time for the traveling wave solutions (2.4.48)

<table>
<thead>
<tr>
<th>Nx</th>
<th>j</th>
<th>$|e^j|_{L^1}$ Order</th>
<th>$|e^u|_{L^1}$ Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0</td>
<td>1.6003e-00</td>
<td>9.3584e-01</td>
</tr>
<tr>
<td>80</td>
<td>1</td>
<td>1.5717e-01</td>
<td>6.9160e-02 3.75</td>
</tr>
<tr>
<td>160</td>
<td>2</td>
<td>1.5362e-02</td>
<td>5.0564e-03 3.77</td>
</tr>
<tr>
<td>320</td>
<td>3</td>
<td>1.7227e-03</td>
<td>5.2204e-04 3.27</td>
</tr>
<tr>
<td>640</td>
<td>4</td>
<td>2.0514e-04</td>
<td>6.4118e-05 3.02</td>
</tr>
</tbody>
</table>

Parameters: $k = 2, L = 40, \Delta x = \frac{1}{2^j}$ for $j = 0, \ldots, 4$, $\Delta t = .1\Delta x$, $T = 1$.

Table 2.4.2: Convergence rate test for the alternating flux in space and $2^{nd}$ order midpoint rule in time for the traveling wave solutions (2.4.48)

<table>
<thead>
<tr>
<th>Nx</th>
<th>j</th>
<th>$|e^j|_{L^1}$ Order</th>
<th>$|e^u|_{L^1}$ Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0</td>
<td>2.1994e-00</td>
<td>1.5848e-00</td>
</tr>
<tr>
<td>80</td>
<td>1</td>
<td>1.7709e-01</td>
<td>1.1434e-01 3.79</td>
</tr>
<tr>
<td>160</td>
<td>2</td>
<td>1.5581e-02</td>
<td>7.0977e-03 4.00</td>
</tr>
<tr>
<td>320</td>
<td>3</td>
<td>1.6858e-03</td>
<td>6.0759e-04 3.54</td>
</tr>
<tr>
<td>640</td>
<td>4</td>
<td>1.9711e-04</td>
<td>6.7434e-05 3.17</td>
</tr>
</tbody>
</table>

Parameters: $k = 2, L = 40, \Delta x = \frac{1}{2^j}$ for $j = 0, \ldots, 4$, $\Delta t = .1\Delta x^2$, $T = 1$, tolerance $= 10^{-10}$.

Table 2.4.3: Convergence rate test for the upwind flux in space and SSPRK4 in time for the traveling wave solutions (2.4.48)

<table>
<thead>
<tr>
<th>Nx</th>
<th>j</th>
<th>$|e^j|_{L^1}$ Order</th>
<th>$|e^u|_{L^1}$ Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0</td>
<td>1.6629e-00</td>
<td>1.0943e-00</td>
</tr>
<tr>
<td>80</td>
<td>1</td>
<td>1.1121e-01</td>
<td>1.1281e-01 3.28</td>
</tr>
<tr>
<td>160</td>
<td>2</td>
<td>8.0167e-03</td>
<td>7.2562e-02 3.17</td>
</tr>
<tr>
<td>320</td>
<td>3</td>
<td>7.3197e-04</td>
<td>1.5346e-03 3.03</td>
</tr>
<tr>
<td>640</td>
<td>4</td>
<td>7.8245e-05</td>
<td>1.9317e-04 2.99</td>
</tr>
</tbody>
</table>

Parameters: Parameters: $k = 2, L = 40, \Delta x = \frac{1}{2^j}$ for $j = 0, \ldots, 4$, $\Delta t = .1\Delta x$, $T = 1$.  

41
2.4.2 Comparisons of Numerical Flux and Time Discretization

In this section, we investigate the long time behavior of the proposed LDG methods to provide comparison of different choices of numerical flux and time discretizations.

First, we will compare the difference in the (Hamiltonian dissipating) RK and (Hamiltonian conserving) midpoint rule temporal discretizations, while keeping the spatial discretization as the LDG method with alternating flux. In this simulation, we take the exact traveling wave solutions (2.4.48) with \( t = 0 \) as the initial condition. The parameters in the initial condition are taken as follows: \( x_0 = 20 \) and \( L = 40 \), so that the peak of \( u(x,0) \) and the trough of \( \eta(x,0) \) are located at the center of the domain. We take the boundary conditions to be periodic, so the wave will return to the initial profile after one period (with \( T = 16 \)). We run the simulation to a final time of \( T = 60 \). Discretizations of space and time are taken to be \( \Delta x = 0.25 \) and \( \Delta t = 0.1 \Delta x^2 \). Figure 2.4.1 gives the numerical results of the comparison.

The alternating flux with the SSPRK4 and midpoint rule time discretizations achieve the correct order of accuracy for short time simulations, as indicated in the tables in the previous subsection. In this test, we run the simulation for a long time, up to \( T = 180 \) which corresponds to about eleven periods of the domain. The numerical solutions with various time discretization, compared with the exact solutions, at times \( T = 60, 130, \) and 180 are shown in Figure 2.4.1. At time \( T = 60 \), no visual difference among different solutions can be observed. When time increases to 130, we can see that the SSPRK4 method performs slightly better. At \( T = 180 \), the fourth order SSPRK4 method clearly outperforms
the other two, and midpoint rule is better than the second order SSPRK2 method with the same time step size. In Figure 2.4.2, we plot the time history of the $L^1$ error between the exact solution and the numerical solutions up to $T = 180$. The error of SSPRK2 remains the largest one, and the error of midpoint rule and SSPRK4 are similar until $T = 120$, after which, the midpoint rule error increases much faster than the SSPRK4 scheme. It should be reiterated that the alternating flux-SSPRK4 method is order 3 in space and order 4 in time, whereas the alternating flux-midpoint rule method is order 3 in space and order 2 in time. As time progresses, the time error is compounded and begins to dominate.

The lower plot in Figure 2.4.1 demonstrates the time history of the Hamiltonian that was proven for the midpoint rule with the alternating flux. It was shown in section 2.3 that the discrete Hamiltonian (2.3.45) is conserved exactly in time for the midpoint rule. The fluctuations in the conserved quantity for the midpoint rule are extremely small, on the order of $10^{-10}$, while the decay in the SSPRK4 method is on the order of $10^{-5}$. The deviation of the conserved quantity for the SSPRK2 is much larger, as seen in the plot. This test shows that, with the same spatial discretization, the Hamiltonian conserving midpoint rule performs better than the Hamiltonian dissipative SSPRK2 method, but the higher order SSPRK4 (which dissipates Hamiltonian slightly) provides the best numerical solution. In the numerical experiments below, we will fix the SSPRK4 as the temporal discretization.

We now turn to looking at the difference between the choices of numerical flux on the same traveling wave solution. We will keep the choice of time discretization fixed as the SSPRK4 method, and compare the results of the alternating and upwind fluxes. We use the same initial condition as in the comparison of time discretizations in the previous
paragraphs. The parameters of the simulation are taken to be: \( x_0 = 20, L = 40, \Delta x = 0.25, \) and \( \Delta t = 0.1 \Delta x. \) Both choices of flux are comparable in terms of \( L^1 \) error up to about \( T = 20. \) After this point, the discrepancy between the errors in the alternating flux and upwind flux become evident. In Figure 2.4.3, the comparison of both numerical solutions and the exact solution are provided at various times. We can observe that the upwind flux approximation is lagging behind the exact traveling wave solution, while the alternating flux approximation is essentially the same as the exact solution. The time history of the \( L^1 \) error and conserved Hamiltonian is shown in Figure 2.4.4, which shows a larger error produced by the upwind flux. In terms of the Hamiltonian, the approximations verify that the Hamiltonian is constant for the alternating flux approximation, and the Hamiltonian is decreasing over time for the upwind flux.

### 2.4.3 Solitary Wave Generation

In this subsection, we perform the test to generate solitary waves (with \( 1 + \eta \geq 0 \)), as was done in [27] for their finite difference scheme. Generating such clean solitary waves is difficult in the physical laboratory setting, but can be easily done numerically using the procedure given in [27], and outlined as follows. This solitary wave solution will also be used in the next subsection to test the head-on collisions of solitary waves.

The coupled BBM system (1.2.5) is provided with the following initial condition

\[
\eta(x, 0) = N_0 \text{sech}^2 \left( \frac{1}{2} \sqrt{\frac{3N_0}{\kappa}} (x - x_0) \right),
\]

\[
u(x, 0) = \eta(x, 0) - \frac{1}{4} \eta^2(x, 0),
\] (2.4.49)
with the parameters taken as $x_0 = 12$, $\kappa = 1 + \frac{1}{2}N_0$, and $N_0 = 0.7$, so that the peaks of $u(x,0)$ and $\eta(x,0)$ are located close to the left boundary of the domain. We take the boundary conditions to be periodic, and the length of the domain to be $L = 120$. Other numerical parameters are taken as $\Delta x = 0.25$ and $\Delta t = 0.1\Delta x$.

The initial wave (2.4.49) is provided to the LDG method and run up to a particular time, denoted as $T$. Here we chose $T = 48$ for the first run, but this value is arbitrary. The wave is evolved until the oscillations that are produced near the peaks are of small amplitude. Since we are using periodic boundary conditions, the only requirement of $T$ is that the simulation must be stopped before any dispersive tails coming from the right interfere with the main wave (alternatively, one can use Dirichlet boundary conditions). The longer domain ($L = 120$ as opposed to $L = 40$ in previous tests) allows for the wave to travel farther before the dispersive tails become an issue. Once at time $T = 48$, the solution on the entire domain is clipped near the peak, such that the main peaks are spliced out of the domain. We inspect the amplitude of $\eta(x,T)$ and choose an subinterval of length 20 which contains the main peak, and save the LDG approximation from the subinterval$^1$. The saved subinterval is then shifted and reloaded as the initial condition, so that the peak is again located at $x_0 = 20$, and then set $u(x,0) \equiv 0$ and $\eta(x,0) \equiv 0$ for the remaining portion of the domain. The clock is reset to $t = 0$, and the LDG code is run again up to a time where the main peak can be clipped again, using the described procedure. This setup allows the wave that was evolved in a previous run to be used as the initial condition in the subsequent run, but without the dispersive tails. The simulation is ran for a second time,

$^1$The length of this subinterval is arbitrary. We wish to capture the peak in the interval, and have the “tails” near the peak to be close to zero.
up until $T = 48$. This procedure can be repeated as many times as possible, to reduce the magnitude of the dispersive tails. Alternatively, one can use a domain and larger value of $T$. We have also used this method with $L = 400$, yielding similar results.

Figure 2.4.5 shows the evolution and filtering of the solitary wave by the LDG method with alternating flux and SSPRK4 time discretization. In Figure 2.4.5a, the initial condition (2.4.49) for $\eta(x, t)$ is plotted. Figure 2.4.5b shows the evolution of $\eta(x, t)$, which is the wave profile (Recall $u(x, t)$, which is not shown, is the horizontal velocity of the wave) at time $T = 48$. The dispersive tails emanating behind the main peak are evident. We can now excise the main peak from this wave profile, which can be used for another filtering step. The filtering step reduces the magnitude of the oscillation, as can be seen in Figure 2.4.5d, where on visual inspection, the oscillations cannot be seen. The magnitude of the oscillations in this figure is on the order of $10^{-5}$ after one filtering step.

### 2.4.4 Solitary Wave Collision

In this example, we present a simulation for the head-on collision of two solitary waves of the same height, using the clean solitary waves generated in the previous subsection. We use the functions given in (2.4.49) to generate a right moving solitary wave, and we use

$$\tilde{\eta}(x, 0) = \eta(L - x, 0),$$

$$\tilde{u}(x, 0) = -u(L - x, 0),$$

to generate a left moving solitary wave, where $\eta(x, 0)$ and $u(x, 0)$ denote the functions in (2.4.49). Other parameters in the simulation are taken as follows: $\Delta x = 0.5$, $\Delta t = 0.1\Delta x$, and $L = 200$ for the length of the domain.
We apply the LDG method with alternating flux and SSPRK4 time discretization for the simulation. The figures in Figure 2.4.6 track the movement of the peaks moving towards each other, combining to give a single peak, and then the waves moving past each other, keeping the same shape post-collision. Figure 2.4.6a shows the initial condition. Figure 2.4.6b records the solution of $\eta(x, T)$ when the solitary waves traveled up to $T = 40$. We can observe that the amplitude and shape of the wave are kept well, and no oscillations are present. In Figure 2.4.6c, the waves have interacted at time $T = 60$, as the two main peaks are visible, with the depression in the center. At time $T = 61.5$, the two waves merge into one, as shown in Figure 2.4.6d. In Figures 2.4.6e and 2.4.6f, we observe after the interaction, the single peak separates into two solitary waves moving away from each other, and with nearly the same shape and amplitude as the waves before the collision.
Figure 2.4.1: Wave profiles for time discretization comparison

(a) Comparison between the SSPRK2, SSPRK4, and midpoint rule time discretizations for $u(x, T = 60)$, using the exact solitary wave initial condition; (b) Comparison for $\eta(x, T = 60)$; (c) Comparison for $u(x, T = 130)$; (d) Comparison for $\eta(x, T = 130)$; (e) Comparison for $u(x, T = 180)$; (f) Comparison for $\eta(x, T = 180)$. 

48
Figure 2.4.2: Error and conserved quantity plots for time discretization comparison

(a) $L^1$ errors over time for $u(x,t)$, for the LDG numerical approximation using each time scheme; (b) $L^1$ errors for $\eta(x,t)$; (c) The computed value of the Hamiltonian (2.3.45) plotted over time for each choice of time discretization.
Figure 2.4.3: Wave profiles for alternating and upwind flux comparison.

(a) Comparison between the alternating and upwind flux choices for \( u(x, T = 60) \), using the exact solitary wave initial condition. (b) Comparison for \( \eta(x, T = 60) \). (c) Comparison for \( u(x, T = 130) \). (d) Comparison for \( \eta(x, T = 130) \). (e) Comparison for \( u(x, T = 180) \). (f) Comparison for \( \eta(x, T = 180) \).
Figure 2.4.4: Error and conserved quantity plots for alternating and upwind flux comparison

(a) $L^1$ errors over time for $u(x, t)$, for the LDG numerical approximation using each numerical flux. (b) $L^1$ errors for $\eta(x, t)$. (c) The computed value of the Hamiltonian (2.3.45) plotted over time for each choice of time discretization.
Figure 2.4.5: Clean solitary wave generation for $\eta(x, t)$

(a) Initial condition for the solitary wave generation (b) Eq. (1.2.5) evolves up to $T = 48$ for the first run. (c) The peak of the solitary wave is excised from (b) and inserted so that the peak is located at $x_0 = 12$, as in figure (a). (d) The initial profile from (c) evolved to $T = 48$, and no visible oscillations are present.
Figure 2.4.6: Solitary wave collision for $\eta(x, t)$

(a) Waves at $T = 0$. (b) Waves at $T = 40$. (c) Waves at $T = 60$. (d) Waves at $T = 61.5$.

(e) Waves at $T = 67.5$. (f) Waves at $T = 80$. 

53
Chapter 3

Discontinuous Galerkin Method for
the Aw-Rascle Traffic Model on
Networks

The introductory section 1.2.2 provided the history of the theoretical aspects of traffic flow modeling. The organization of the chapter is as follows. A brief review of the AR traffic flow model on a network is given. In section 3.1 we define the problem of interest, the network AR model and notational conventions that will be important for the remainder of the paper. Section 3.2 outlines the theoretical work that has been done in developing coupling conditions that satisfy the Riemann problem at the junction. In Section 3.3.1, we present a DG method for the network problem, with detailed implementation of the coupling conditions at the junction presented in sections 3.3.2-3.3.9. Section 3.4 contains various numerical experiments that show the optimal convergence rates, comparisons of
different coupling conditions, and the behavior of the second order model. As comparison, we also implement the first order LWR model on networks with DG methods using Godunov flux, and the comparison of LWR and AR models on networks with the proposed DG method is also provided. We also compare how different coupling conditions affect the numerical scheme for the AR model. In addition, we present numerical results showing the capacity drop effect for the two incoming roads and one outgoing road case.

3.1 Notation and Conventions

In this subsection, we give concrete definitions and notational conventions that will hold for the remainder of the paper. In [42] and [47], the notational convention gives the following definition for incoming and outgoing roads. Each road $k$ is described as the interval $[a_k, b_k]$ where $a_k = -\infty$, $b_k = 0$ for incoming roads, and $a_k = 0$, $b_k = +\infty$ for outgoing roads. They assume that traffic goes in the direction from $a_k$ to $b_k$, thus traffic has positive speed in that direction. This notational convention can be visualized in Figure 3.1.1a, where the vertex diagram and interface diagram provide a visual of the junction.

An alternative way to represent the roads, used in [19], where we apply the position mapping $x \mapsto -x$ to the incoming road. The mapping allows the incoming roads to be represented on a positive interval, so that all roads are directed out of the interface at $x = 0$. To simplify the situation further for the numerical work, we consider each road on the unit interval $[0, 1]$, exiting the junction. Each road has a dotted line portion to the left of the junction which represents a “ghost cell” for the solution of the left state of the Riemann problem at the junction. This notational convention can be visualized in right subfigure
Figure 3.1.1: Notation at the junction

(a) The left subfigure of (a) gives the direction of traffic where the arrows that point to the vertex (Edges 1 and 2) represent incoming roads, and the arrows pointing away from it are outgoing roads, for the case of 2 incoming roads and 2 outgoing roads. This notation is used in [42]. The right subfigure of (a) gives the junction in terms of the interface, where the vertex represents $x = 0$. (b) The left subfigure of (b) gives the situation after the $x \mapsto -x$ has been applied to the incoming roads, so the direction of all traffics points outwards from $x = 0$. In the right subfigure of (b), each road points out of the vertex at $x = 0$, where the dotted line to left represents the “ghost cell”. This notation is used in [19].

For the remainder of the paper, we will use the notational convention in Figure 3.1.1b. For the remainder of the paper, we will use the notational convention in Figure 3.1.1b on the unit interval.

For the traffic network problem, we apply the AR model given by (1.2.13), on each road. Assume that there are a total of $m$ roads, so that initial conditions $u_0^{(k)}(0, x)$ and $q_0^{(k)}(0, x)$ are prescribed on each road $k = 1, \ldots, m$. Let there be $\tilde{m}$ incoming roads and $\hat{m}$ outgoing roads such that $m = \tilde{m} + \hat{m}$. We must also provide coupling conditions
at the junction, denoted as $\Phi(U^{(1)}(t, 0^+), \cdots, U^{(m)}(t, 0^+)) = 0$, where $U = (u, q)^T$ denotes the vector of unknown variables. The coupling conditions describe how traffic moves from incoming roads to outgoing roads at the interface. Putting together (1.2.13), the initial conditions, and coupling conditions, we can describe the AR traffic network model on outgoing roads as

$$
\begin{cases}
\partial_t u^{(k)}(t, x) + \partial_x \left( q^{(k)}(t, x) - u^{(k)}(t, x)p(u^{(k)}(t, x)) \right) = 0 & (t, x) \in \mathbb{R}^+ \times [0, 1] \\
\partial_t q^{(k)}(t, x) + \partial_x \left( \frac{(q^{(k)}(t, x))^2}{u^{(k)}(t, x)} - q^{(k)}(t, x)p(u^{(k)}(t, x)) \right) = 0 & (t, x) \in \mathbb{R}^+ \times [0, 1] \\
u^{(k)}(0, x) = u_0^{(k)}(0, x) & \text{for } x \in [0, 1] \\
q^{(k)}(0, x) = q_0^{(k)}(0, x) & \text{for } x \in [0, 1] \\
\Phi(U^{(1)}(t, 0^+), \cdots, U^{(m)}(t, 0^+)) = 0 & t \geq 0,
\end{cases}
$$

for $k = \tilde{m} + 1, \ldots, \tilde{m}$, and incoming roads as

$$
\begin{cases}
\partial_t u^{(k)}(t, x) - \partial_x \left( q^{(k)}(t, x) - u^{(k)}(t, x)p(u^{(k)}(t, x)) \right) = 0 & (t, x) \in \mathbb{R}^+ \times [0, 1] \\
\partial_t q^{(k)}(t, x) - \partial_x \left( \frac{(q^{(k)}(t, x))^2}{u^{(k)}(t, x)} - q^{(k)}(t, x)p(u^{(k)}(t, x)) \right) = 0 & (t, x) \in \mathbb{R}^+ \times [0, 1] \\
u^{(k)}(0, x) = u_0^{(k)}(0, x) & \text{for } x \in [0, 1] \\
q^{(k)}(0, x) = q_0^{(k)}(0, x) & \text{for } x \in [0, 1] \\
\Phi(U^{(1)}(t, 0^+), \cdots, U^{(m)}(t, 0^+)) = 0 & t \geq 0,
\end{cases}
$$

for $k = 1, \ldots, \tilde{m}$, where $u^{(k)}(t, x)$ and $q^{(k)}(t, x)$ are the density and pseudo-momentum on the $k^{\text{th}}$ road, and the fluxes are given by $q^{(k)}(t, x) - u^{(k)}(t, x)p(u^{(k)}(t, x))$ which is the flux of the density, and $\frac{(q^{(k)}(t, x))^2}{u^{(k)}(t, x)} - q^{(k)}(t, x)p(u^{(k)}(t, x))$ which is the flux of the pseudo-momentum.
Note that there is a negative sign in front of the flux term in Eq. (3.1.2), which comes from the position mapping $x \mapsto -x$ to update the computational domain from $[-1,0]$ to $[0,1]$ on the incoming roads. The initial conditions for the density and pseudo-momentum are prescribed on each road with the functions $u_0^{(k)}$ and $q_0^{(k)}$, with the coupling conditions $\Phi(U^{(k)}(t,0^+)) = 0$ defined for each set of PDEs and depends on both the incoming and outgoing roads, hence $k = 1, \ldots, m$.

3.2 Coupling Conditions

Coupling conditions prescribe equations that must be satisfied at the junction. These conditions model conservation of quantities, determine how much flow moves from road to road, and how quantities can be maximized or minimized. Different coupling conditions for second-order models of traffic flow have been proposed in the literature [42, 45, 46], and all of the methods described above provide a unique solution to the Riemann problem at the junction. Below, we will review them individually. First, there are some coupling conditions that are accepted by all of the papers we consider. Those coupling conditions are:

- The flux of the density must be conserved. This condition is required, as the number of vehicles that enter the junction from incoming roads must leave on an outgoing road. That is, vehicles cannot be created or destroyed at the junction.

- Waves produced at the junction must have positive speed, assuming that we orient all roads out of the junction (see Figure 3.1.1b). This consideration is taken so that the solution will satisfy the boundary conditions at the junction, and results in a physical model.
At this point we now review possible coupling conditions, as various authors have considered
different possibilities at the junctions and the choice of coupling condition has been shown
to slightly affect the solution determined by the method. In [42], the coupling conditions
considered include, in addition to the two above, the following

- There exists a traffic distribution matrix $A$, stating what percentage of the flux of the
density on each incoming road moves to each outgoing road.

- The sum of the flux of the density on incoming roads is maximized.

- In some situations, the four coupling conditions above do not provide a unique solu-
tion. The last condition employed can be either: (i) maximize velocity on outgoing
roads, (ii) maximize density on outgoing roads, or (iii) minimize the total variation
of density on outgoing roads. According to [42], the choice between (i)-(iii) does not
matter as any choice recovers the same solution.

This model can be applied to the general $\tilde{m}$-incoming and $\hat{m}$-outgoing roads case, with the
restriction that $\tilde{m} \geq \hat{m}$ (except for the special case of 1-incoming and 2-outgoing roads).

In [45], the coupling conditions considered include, in addition to the canonical
two, the following three rules

- The quantity $\frac{q}{u}$, which describes the “behavior” of drivers, must be the same before
and after the junction.

- The sum of the flux of the densities on incoming roads is maximized.

- In some cases, the rules above do not provide a unique solution. The last condition is
to maximize velocity on outgoing roads.
The coupling conditions in [45] are developed only for junction problems where there are \( \tilde{m} \)-incoming roads and 1-outgoing road, and 1-incoming road and \( \hat{m} \)-outgoing roads. Note that the conditions are very similar, but the coupling conditions in [45] includes a different condition, which ensures the quantity \( \frac{q}{u} \) is the same on each side of the junction. Note that this conservation is not the same as conservation of the flow, as the total flow of the quantity \( \frac{q}{u} \) may not be the same on both sides of the junction. Instead, each vehicle tends to conserve their quantity, \( \frac{q}{u} \), through the junction. Another distinct difference between the two sets of coupling conditions is the order in which variables are fixed or maximized on outgoing roads. In [45], maximization of the flux of the density is the final coupling condition implemented, whereas in [42] the maximization is the first coupling condition that is computed to set up a system of nonlinear equations. There are some important physical consequences of these choices.

Another set of coupling conditions given in [46] incorporate microscopic behavior at the junction to model junction merge behavior, and includes

- The flux of the pseudo-momentum \( q \) must be conserved.

- The sum of the flux of the densities on incoming roads is maximized, subject to a traffic distribution matrix \( A \).

- In the case for 2 incoming roads and 1 outgoing road (or more generally when there is more than one incoming road), a “mixture rule” corresponding to microscopic considerations of how vehicles merge at the junction.

We do not discuss the details of [46] here, as we do not consider these coupling conditions in this paper.
3.3 Discontinuous Galerkin Method

3.3.1 DG method for the AR Model

In this section, we construct the DG method for (3.1.1) and (3.1.2) by discretizing the space with the DG method, and using strong stability preserving (SSP) Runge-Kutta (RK) methods in time. As the outgoing and incoming road cases are the same up to a sign difference, we will just consider the outgoing road formulation as the incoming road formulation can be defined similarly.

We can write (3.1.1) and (3.1.2) in the following general form of a system of conservation laws

$$U_t + F(U)_x = 0,$$

(3.3.3)
on each road, where $F(U)$ is the flux term. Utilizing the conservative form of the AR model, we have the following variables

$$U = \begin{pmatrix} u \\ u(v + p(u)) \end{pmatrix} = \begin{pmatrix} u \\ q \end{pmatrix}, \quad F(U) = \begin{pmatrix} uv \\ uv(v + p(u)) \end{pmatrix} = \begin{pmatrix} q - up(u) \\ \frac{q^2}{u} - qp(u) \end{pmatrix}. \quad (3.3.4)$$

Using the variables in (3.3.4), we formulate the DG method for the system (3.1.1) and (3.1.2) as the following: find $U_h^{(k)} \in V_h^\kappa$, such that

$$\left( \left( \begin{pmatrix} U_h^{(k)} \\ \phi_h \end{pmatrix} \right)_t, \phi_h \right)_{I_j} = \left( F(U_h^{(k)}) \right)_{I_j} - \hat{F}_{j+1/2}(U_{h,j+1/2}^{(k)}, U_{h,j+1/2}^{(k),+}) \hat{\phi}_{h,j+1/2}^- + \hat{F}_{j-1/2}(U_{h,j-1/2}^{(k)}, U_{h,j-1/2}^{(k),+}) \hat{\phi}_{h,j-1/2}^+,$$

(3.3.5)
for all test functions $\phi_h \in V^\kappa_h$. The $\hat{F}_{j+1/2}$ terms represent the numerical fluxes, which come from the boundary terms at each cell interface obtained from the integration by parts. There is not a unique choice of the numerical flux in general. For our numerical experiments, we will use the Lax-Friedrichs flux which is given by

$$
\hat{F}_{j+1/2}((U^{(k),-}_{h,j+1/2}, U^{(k),+}_{h,j+1/2}) = \frac{1}{2}(F(U^{(k),-}_{h,j+1/2}) + F(U^{(k),+}_{h,j+1/2}) - \frac{\alpha}{2}(U^{(k),+}_{h,j+1/2} - U^{(k),-}_{h,j+1/2}),
$$

(3.3.6)

where $\alpha = \max \{|\lambda_1|, |\lambda_2|\}$ is the Lax-Freidrichs constant. The $\lambda_j$ are the eigenvalues of Jacobian matrix of $F(U)$, which are given by $\lambda_1 = v = \frac{1}{u} (q - u^{\gamma+1})$ and $\lambda_2 = v - \gamma u^{\gamma}$. The framework presented in this paper can also be applied to other choices of numerical fluxes as well.

The discretization of the spatial domain using the DG method handles all of the numerical fluxes at the edge of each cell, but a problem arises for the flux $\hat{F}_{1/2}$ of the very first cell, as this is where the junction is located. To determine the flux $\hat{F}_{1/2}$, we need to implement the appropriate coupling conditions as outlined in section 3.2. In next section, we will explain in details how we numerically compute these fluxes at the junction.

Spurious artificial oscillations can appear near discontinuities in the numerical solution where shock-waves appear in hyperbolic problems. When such oscillations appear, total variation bounded (TVB) limiters can be applied to control these artifacts and to achieve total variation stability. The limiter should not change the cell averages when the limiter is applied, and the accuracy of the method for smooth solutions should remain the same. In this paper, we apply the TVB limiter outlined in [63].
The DG formulation (3.3.5) for the AR model is solved in time utilizing the SSPRK3 method [44], which is given as

\[
(U_h^{(k)})^{(1)} = (U_h^{(k)})^i + \Delta t^i L \left( (U_h^{(k)})^i \right)
\]

\[
(U_h^{(k)})^{(2)} = \frac{3}{4} (U_h^{(k)})^i + \frac{1}{4} (U_h^{(k)})^{(1)} + \frac{1}{4} \Delta t^i L \left( (U_h^{(k)})^{(1)} \right)
\]

\[
(U_h^{(k)})^{i+1} = \frac{1}{3} (U_h^{(k)})^i + \frac{2}{3} (U_h^{(k)})^{(2)} + \frac{2}{3} \Delta t^i L \left( (U_h^{(k)})^{(2)} \right), \quad (3.3.7)
\]

where \( L \) represents the spatial operator, which denotes the right side of (3.3.5), and \( \Delta t^i \) is the numerical time step.

### 3.3.2 Implementation of the Coupling Conditions

In this section we describe the implementation of the coupling conditions for the AR model on a network in the general case, by extending the first order method developed in [19] to the high order DG framework. We start with describing the main idea to evaluate the numerical flux on the junction. In subsection 3.3.3, the methodology to decide the admissible region is presented, the maximization of density flux coupling condition is discussed in subsection 3.3.4, and in the proceeding subsections 3.3.5-3.3.8 we provide the specific coupling conditions we implement for the 1-1, 1-2, 2-1, and 2-2 junction cases, respectively. Finally, in 3.3.9, we discuss the optimization algorithms used in solving the maximization/minimization problems.

On each road, we utilize the DG method (3.3.5) with the Lax-Friedrichs flux (3.3.6) for the spatial discretization, and SSPRK3 scheme (3.3.7) in time. To construct DG method for the AR model on a network, it requires the implementation of coupling conditions at the left hand boundary \( x = 0 \), in other words, we need to determine \( \hat{F}_{1/2}(U_h^{(k),-}, U_h^{(k),+}) = \)

63
\( \hat{F}_{1/2}((U_h^{(k)})_L, (U_h^{(k)})_R) \), with \((U_h^{(k)})_L\) and \((U_h^{(k)})_R\) denoting the left and right states at the junction on the \(k^{th}\) road. Note that \((U_h^{(k)})_R\) is the numerical solution in the first cell and \((U_h^{(k)})_L\) is the numerical solution in the "ghost cell". For the case with no junction involved, \((U_h^{(k)})_L\) comes from the given boundary condition. In our problem, it needs to be solved from the coupling conditions at the junction.

To determine the unknown values for the left hand states, \((U_h^{(k)})_L\), and subsequently the flux \(\hat{F}_{1/2}((U_h^{(k)})_L, (U_h^{(k)})_R)\), we extend the technique developed in [19] for the first order finite difference method to the DG framework. First, rewrite the DG method (3.3.5) as

\[
\left( \left( \frac{U_h^{(k)}}{t} \right)_j, \phi_h \right)_{I_j} = - \left( F(U_h^{(k)})_{x, j}, (\phi_h) \right)_{I_j} - \left( \hat{F}_{j+1/2}(U_h^{(k),-}, U_h^{(k),+}) - F(U_h^{(k),-}) \right) \phi_{h,j+1/2}^-
\]

\[
+ \left( \hat{F}_{j-1/2}(U_h^{(k),-}, U_h^{(k),+}) - F(U_h^{(k),+}) \right) \phi_{h,j-1/2}^+.
\]

By introducing the notation of "fluctuations"

\[
\left( D_{j+1/2}^{(k)} \right)^- = \hat{F}_{j+1/2}(U_h^{(k),-}, U_h^{(k),+}) - F(U_h^{(k),-})_1,
\]

\[
\left( D_{j-1/2}^{(k)} \right)^+ = -\hat{F}_{j-1/2}(U_h^{(k),-}, U_h^{(k),+}) + F(U_h^{(k),+})_1,
\]

the DG method becomes

\[
\left( \left( \frac{U_h^{(k)}}{t} \right)_j, \phi_h \right)_{I_j} = - \left( F(U_h^{(k)})_{x, j}, (\phi_h) \right)_{I_j} - \left( D_{j+1/2}^{(k)} \right)^- \phi_{h,j+1/2}^- - \left( D_{j-1/2}^{(k)} \right)^+ \phi_{h,j-1/2}^+.
\]

(3.3.9)

When \(\phi_h = 1\), it reduces to \(\overline{U_h^{(k)}}_{t} | I_j = - \left( D_{j+1/2}^{(k)} \right)^- - \left( D_{j-1/2}^{(k)} \right)^+\), which shares the form of the residual distribution method [1]. At the junction, the fluctuations \(D_{1/2}^{(k)}\) are defined by

\[
\left( D_{1/2}^{(k)} \right)^- = \hat{F}_{1/2} - F \left( (U_h^{(k)})_L \right), \quad \left( D_{1/2}^{(k)} \right)^+ = -\hat{F}_{1/2} + F \left( (U_h^{(k)})_R \right),
\]

64
and the \( (D_{1/2}^{(k)})^{-} \) term contains all information of backward moving waves at the interface of the cell. One of the main coupling conditions outlined in traffic flow is that only waves entering the junction (right-moving waves or waves with positive speed) are admissible, as outlined in section 3.2. Ideally, to satisfy the coupling condition of admissible waves, we would like to require that \( (D_{1/2}^{(k)})^{-} = 0 \), as this would mean there would be no information traveling backwards into the junction. If the Godunov flux with the exact Riemann solver is employed at the junction (as in [26] for the LWR model), this condition will be satisfied exactly. With the choice of Lax-Friedrichs (LF) flux (or other numerical flux corresponding to different approximate Riemann solvers), this condition cannot be enforced exactly, instead we relax it to the minimization problem

\[
\min \sum_{k=1}^{m} \left\| (D_{1/2}^{(k)})^{-} - \left( (U_h^{(k)})_L, (U_h^{(k)})_R \right) \right\|^2, \tag{3.3.10}
\]

subject to the coupling conditions

\[
\Phi \left( (U_h^{(1)})_I, \ldots, (U_h^{(m)})_I \right) = 0, \tag{3.3.11}
\]

where the unknowns \((U_h^{(k)})_L\) are the left states coming from the variables \(u_h\) and \(q_h\) in the ghost cell, \((U_h^{(k)})_R\) are the right states coming from the first cell inside the domain, and \((U_h^{(k)})_I = (U_h^{(k)})_I \left( (U_h^{(k)})_L, (U_h^{(k)})_R \right)\) are the intermediate values at \(x = 0\) by solving the Riemann problem with left and right values \((U_h^{(k)})_{L,R}\) approximately. We could choose the intermediate values as

\[
(U_h^{(k)})_I \left( (U_h^{(k)})_L, (U_h^{(k)})_R \right) = \frac{1}{2} \left( (U_h^{(k)})_R + (U_h^{(k)})_L \right) - \frac{1}{2\alpha} \left( F \left( (U_h^{(k)})_R \right) - F \left( (U_h^{(k)})_L \right) \right). \tag{3.3.12}
\]
For the LF flux, the left-moving (backwards) fluctuation takes the form of

\[
(D_{1/2}^{(k)})^- = \frac{1}{2} \left( F\left( (U_h^{(k)})_R \right) - F\left( (U_h^{(k)})_L \right) \right) - \frac{\alpha}{2} \left( (U_h^{(k)})_R - (U_h^{(k)})_L \right).
\] (3.3.13)

The solution of the minimization problem (3.3.10) and (3.3.11) produces the values \((U_h^{(k)})_L\) in the ghost cells. Therefore, once the optimization problem is solved, we have the last component (flux \(\hat{F}_{1/2}\)) to update the numerical solution of the DG method (3.3.5) and (3.3.6) at the given time step.

In the following subsections, we consider the coupling conditions discussed in [42] and [45] which will be defined in (3.3.11) for the cases of 1-1, 1-2, 2-1, and 2-2 junction types, where the first number is the amount of incoming roads and the second number is the amount of outgoing roads. For the situation of the 1-1 junction and 1-2 junction, the coupling conditions from both papers appear to produce practically same result, which will be shown in section 3.4. The 2-1 and 2-2 junction cases are more complicated, as there is mixing from incoming roads to outgoing roads. The 2-1 junction case, as we shall see, will provide an example where the two sets of coupling conditions differ. The 2-2 case is only considered with the coupling conditions from [42], as the coupling conditions in [45] are only provided for the \(\tilde{m} - 1\) and \(1 - \tilde{m}\) junction types.

Before proceeding further, we give an overview of the procedure for the implementation of the coupling conditions. All of the coupling conditions, except for the maximization of density flux and preference for waves of positive speed, can be described via nonlinear algebraic equations. The two exceptions are formulated through optimization problems. The preference for waves of positive speed condition, when using the LF flux, is defined through the minimization of fluctuations (3.3.10), when solved will provide the unknown
left states at the junction. Prior to being able to solve the minimization problem (3.3.10),
the coupling conditions (3.3.11) must be defined as a system of equations, which requires
the maximization of density flux problem to be solved beforehand.

The maximization of the density flux problem returns the maximized fluxes which
are used to construct nonlinear equations used in the definition of (3.3.11). To solve the
maximization of density flux, we must first determine the admissible values for the flux
of the density. The admissible values of the density flux must be consistent with the
preference for waves of positive speed condition, which in turn is determined by the states
in the ghost cells of the left states through the junction Riemann problem. For example,
in [42], the admissible region for the flux of the density must be determined on each road,
consistent with the coupling condition requirements. The admissible region is a function of
\((U_h^{(k)})_R\) and \(\gamma\) in the \((u,q)\)-plane with respect to some computed functions. The admissible
region computation is also different for incoming and outgoing roads, therefore each case
is treated separately\(^1\). It should be noted that the maximization problem may not return
the maximum possible flux of the density, as heavily congested traffic will not allow for the
maximum possible flux to be achieved. We now give the details of the admissible region in
the next subsection, and those of the maximization of density flux in Section 3.3.4.

3.3.3 Determining the Admissible Regions

The computation of the admissible regions for the two types of coupling conditions con-
sider in this paper are discussed in [42] and [45]. To demonstrate the computation of the
\(^1\)The maximization of density flux is used for the 1-2, 2-1, and 2-2 junction cases. The 1-1 junction case is
simple and does not require this additional condition to obtain a unique solution, whereas the other junction
types will require this coupling condition.
admissible regions for the AR model, we provide an overview of sections 5.1 and 5.2 in [42]. The admissible regions provide the constraints on the maximization of flux problem coupling condition, which must be solved to determine a nonlinear equation for the coupling condition vector.

First, define the following regions $D, D_1$ and $D_2$ in the $(u, q)$-plane as

$$D_1 = \{(u, q) \in D \mid q \geq (\gamma + 1)u^{\gamma+1}\}, \quad (3.3.14)$$

$$D_2 = \{(u, q) \in D \mid q \leq (\gamma + 1)u^{\gamma+1}\}, \quad (3.3.15)$$

$$D = \{(u, q) \in \mathbb{R}^+ \times \mathbb{R}^+ \mid u^{\gamma+1} \leq q \leq u\}. \quad (3.3.16)$$

In order to satisfy the coupling condition that the waves produced by the half-Riemann problem must have positive speed, we must enforce the admissible region $\Omega_{inc}^k$, for the density flux $\delta_k$ on the incoming roads to be

$$\Omega_{inc}^k = \begin{cases} 
0, \gamma \left(\frac{1}{\gamma + 1}\right)^{\frac{\gamma+1}{\gamma}} \left(q_{R}^{(k)} \left(u_{R}^{(k)}\right)^{\frac{\gamma+1}{\gamma}}\right), & \text{if } (u_{R}^{(k)}, q_{R}^{(k)}) \in D_2, \\
0, q_{R}^{(k)} - (u_{R}^{(k)})^{\gamma+1}, & \text{if } (u_{R}^{(k)}, q_{R}^{(k)}) \in D_1. 
\end{cases} \quad (3.3.17)$$

For what is to follow in subsection 3.3.4, we introduce a new notation for the upper bounds of the admissible regions. We define

$$\delta_{k}^{\text{max}} = \begin{cases} 
\gamma \left(\frac{1}{\gamma + 1}\right)^{\frac{\gamma+1}{\gamma}} \left(q_{R}^{(k)} \left(u_{R}^{(k)}\right)^{\frac{\gamma+1}{\gamma}}\right), & \text{if } (u_{R}^{(k)}, q_{R}^{(k)}) \in D_2, \\
q_{R}^{(k)} - (u_{R}^{(k)})^{\gamma+1}, & \text{if } (u_{R}^{(k)}, q_{R}^{(k)}) \in D_1, 
\end{cases} \quad (3.3.18)$$

where $\delta_{k}^{\text{max}}$ represents the maximum possible density flux on incoming road $k$. The maximum possible flux does not necessarily mean that $\delta_{k}^{\text{max}}$ will be achieved in the maximization of flux, as severe congestion could cause less than optimal flux at the junction.
Second, we determine the admissible region $\Omega_k^{\text{out}}$ for the density flux on outgoing roads. One function of importance is the curve of the second family (CSF) going through $(u_R^{(k)}, q_R^{(k)})$, given by $q = \frac{q_R^{(k)}}{u_R^{(k)}} u + u^{\gamma+1} - (u_R^{(k)})^\gamma u$. This curve is determined through analyzing the Riemann problem for the AR model (details are provided in [42]). Again, we must have waves produced by the half-Riemann problem to have positive speed, thus the admissible region $\Omega_k^{\text{out}}$, for the density flux $\delta_k$ on the outgoing roads is

$$\Omega_k^{\text{out}} = \begin{cases} 
0, \gamma \left( \frac{1}{\gamma + 1} \right)^{\frac{\gamma+1}{\gamma}} , & \text{if CSF is in } D_1, \\
0, \frac{1}{u_R^{(k)}} \left( d_R^{(k)} - (u_R^{(k)})^{\gamma+1} \right) \left( 1 + \left( u_R^{(k)} \right)^\gamma - \frac{q_R^{(k)}}{u_R^{(k)}} \right)^{\frac{1}{\gamma}} , & \text{otherwise.}
\end{cases}$$

(3.3.19)

Similar to the incoming road case, the $\delta_k^{\max}$ for outgoing roads are defined as

$$\delta_k^{\max} = \begin{cases} 
\gamma \left( \frac{1}{\gamma + 1} \right)^{\frac{\gamma+1}{\gamma}} , & \text{if CSF is completely in } D_1, \\
\frac{1}{u_R^{(k)}} \left( d_R^{(k)} - (u_R^{(k)})^{\gamma+1} \right) \left( 1 + \left( u_R^{(k)} \right)^\gamma - \frac{q_R^{(k)}}{u_R^{(k)}} \right)^{\frac{1}{\gamma}} , & \text{otherwise.}
\end{cases}$$

(3.3.20)

A similar analysis of the Riemann problem with the coupling conditions outlined in [45] is also done, except that the admissible regions give restrictions on the values of the density and pseudo-momentum. As the theory of admissible regions are not the main focus of this paper, we omit the details of the computations. The details are provided in [45].

With the admissible regions for the density flux $\delta_k$ on both incoming and outgoing roads, and the maximum possible fluxes, we now have all the components necessary to solve the maximization of density flux optimization problem.
3.3.4 Maximization of flux of the density

Before proceeding with the case-specific maximization problems, we provide a general overview of the maximization flux coupling condition, and how the condition fits into (3.3.11). First, we use the notation \( \hat{\delta}_k \) to represent the solution to the maximization of flux problem, which is the maximal attained (not necessarily the maximum possible) flow leaving the incoming road \( k \). The value of \( \hat{\delta}_k \) has bounds such that \( \hat{\delta}_k \in [0, \delta_{k,\text{max}}] \) is enforced, where \( \delta_{k,\text{max}} \) represents the maximum possible flux determined through the admissible region computation described in 3.3.3. The \( \hat{\delta}_k \) values for each road \( k \) are dependent upon the value of \( \gamma \) in the pressure term and the right and left states of the density and pseudo-momentum for each road. The solution of the maximization problem returns the \( \hat{\delta}_k \) for incoming roads.

To get the fluxes \( \hat{\delta}_k \) for the outgoing roads, the traffic distribution matrix \( A \) is applied to the vector of \( \hat{\delta}_k \) for incoming roads. We then set up the equations that are nonlinear in \( u^{(k)}_L \)

\[
q^{(k)}_I - (u^{(k)}_I)^{\gamma+1} - \hat{\delta}_k = 0,
\]

for \( k = 1, \ldots, m \), where \( u^{(k)}_I \) and \( q^{(k)}_I \) are given by the interface equation (3.3.12). The other coupling conditions are already provided in terms of nonlinear equations. Putting these coupling conditions in vector form, we get the coupling condition vector \( \Phi \) in (3.3.11). We can now consider the specific maximization problems for the 1-2, 2-1, and 2-2 cases.

For the 1-2 junction case, the maximization of flux coupling condition is given by:

\[
\hat{\delta}_1 = \max_{\delta_1} \delta_1, \quad \text{subject to} \quad \begin{cases} 
\delta_1 & \in [0, \delta_{1,\text{max}}] \\
A \cdot \delta_1 & \in [0, \delta_{2,\text{max}}] \times [0, \delta_{3,\text{max}}]
\end{cases},
\]

(3.3.21)

where \( A \) is the exogenous traffic distribution matrix, and \( \hat{\delta}_1 \) is the flux of the density on the
incoming road which solves the maximization problem. Recall that the admissible regions 
\([0, \delta_{k}^{\text{max}}]\) are, in general, a function of the right state \((U_{h}^{(k)})_{R} = (u_{R}^{(k)}, q_{R}^{(k)})^{T}\) and the value of \(\gamma\), through the admissible region computation. To get the value for the outgoing roads, we just apply the traffic distribution matrix \(A\) to this vector:

\[
[\hat{\delta}_{2}, \hat{\delta}_{3}] = A \cdot [\hat{\delta}_{1}] = [\alpha_{1,2}, \alpha_{1,3}] \cdot [\hat{\delta}_{1}] = [\alpha_{1,2}\hat{\delta}_{1}, \alpha_{1,3}\hat{\delta}_{1}],
\]

which states that the fraction \(\alpha_{1,2}\) vehicles from incoming road 1 travel to outgoing road 2, and the fraction \(\alpha_{1,3}\) vehicles from incoming road 1 travel to outgoing road 3, such that \(\alpha_{1,2} + \alpha_{1,3} = 1\).

The maximization of flux for 2-1 junction case is slightly more complicated. One can imagine a freeway merge where two different freeways merge into a single one, and if there is a large number of vehicles attempting to enter from each incoming road, congestion sets in and traffic backs up. We consider two possibilities depending upon the values of \(\delta_{1}^{\text{max}}, \delta_{2}^{\text{max}},\) and \(\delta_{3}^{\text{max}}\) determined from the admissible region calculation. If \(\delta_{1}^{\text{max}} + \delta_{2}^{\text{max}} < \delta_{3}^{\text{max}}\), then there will not be severe congestion, as both incoming road fluxes can be maximal and the condition \(\hat{\delta}_{1} + \hat{\delta}_{2} = \hat{\delta}_{3} \in [0, \delta_{3}^{\text{max}}]\) will be satisfied. In this case, we simply have that \(\hat{\delta}_{k} = \delta_{k}^{\text{max}}\) for \(k = 1, 2\), and \(\hat{\delta}_{3} = \hat{\delta}_{1} + \hat{\delta}_{2}\). In terms of the traffic distribution matrix, which in this case is given by \(A = [1, 1]\), all vehicles must travel through the junction to the outgoing road such that

\[
\hat{\delta}_{3} = A \cdot [\hat{\delta}_{1}, \hat{\delta}_{2}]^{T} = [1, 1] \cdot [\hat{\delta}_{1}, \hat{\delta}_{2}]^{T} = \hat{\delta}_{1} + \hat{\delta}_{2}.
\]

If on the other hand, \(\delta_{1}^{\text{max}} + \delta_{2}^{\text{max}} > \delta_{3}^{\text{max}}\), then it may be the case that the sum of the fluxes on the incoming roads will be greater than road 3 can accommodate (ie. \(\delta_{3}^{\text{max}}\),
and the required condition in (3.3.23) may produce results where \( \hat{\delta}_3 \not\in [0, \delta_3^{\text{max}}] \) if we use the \( \delta_k^{\text{max}} \) for \( k = 1, 2 \). To remedy this situation, we introduce a fixed “merge coefficient” called \( q^* \), which states that the percentage \( q^* \) of vehicles can enter from road 1 and \( 1 - q^* \) can enter from road 2. For our numerical experiments, we take the value of \( q^* = \frac{1}{2} \), but other choices can be made. Now, the maximization of flux of the density can be given as

\[
\hat{\delta} = \left[ \hat{\delta}_1, \hat{\delta}_2 \right] = \max_{\delta_1, \delta_2} \delta_1 + \delta_2, \quad \text{subject to} \quad \begin{cases} \delta_1 \in [0, \delta_1^{\text{max}}] \\ \delta_2 \in [0, \delta_2^{\text{max}}] \\ \delta_1 + \delta_2 \in [0, \delta_3^{\text{max}}] \\ \frac{\delta_1}{q^*} = \frac{\delta_2}{1 - q^*} \end{cases} , \quad (3.3.24)
\]

where \( [\hat{\delta}_1, \hat{\delta}_2] \) are the flux of the densities on the 2 incoming roads which solves the maximization problem. We then apply (3.3.23) to get the other 2 fluxes.

Finally, the maximization of flux for 2-2 junction case is similar to the 1-2 junction case with an additional incoming road, and can be written as

\[
\hat{\delta} = \left[ \hat{\delta}_1, \hat{\delta}_2 \right] = \max_{\delta_1, \delta_2} \delta_1 + \delta_2, \quad \text{subject to} \quad \begin{cases} \delta_1 \in [0, \delta_1^{\text{max}}] \\ \delta_2 \in [0, \delta_2^{\text{max}}] \\ [\alpha_{1,1}, \alpha_{1,2}] \cdot [\delta_1, \delta_2]^T \in [0, \delta_3^{\text{max}}] \\ [\alpha_{2,1}, \alpha_{2,2}] \cdot [\delta_1, \delta_2]^T \in [0, \delta_4^{\text{max}}] \end{cases} , \quad (3.3.25)
\]

where \( \alpha_{i,j} \) are the entries of the 2x2 traffic distribution matrix \( A \). Again, this optimization provides the values of \( \hat{\delta}_1 \) and \( \hat{\delta}_2 \), from which we apply \( A \) to get \( [\hat{\delta}_3, \hat{\delta}_4]^T = A \cdot [\hat{\delta}_1, \hat{\delta}_2]^T \) for the other two roads.
3.3.5 1 incoming road, 1 outgoing road

For the simple 1-1 junction case, we treat a single road as if there were a junction located on the interior of the said single road. We can compare the junction model directly to the case where we treat the 1-1 junction as a single interval, and just apply DG normally. Both methods should return the same solution, and this could be used as the first step to validate the junction model.

For the network case, we could implement the conservation of the flux of the density and obtain the unique solution as stated previously. We consider the following coupling conditions using the intermediate approximation values at the junction via (3.3.12), which is given as

$$
\Phi = \begin{bmatrix}
q^{(1)}_I - (u^{(1)}_I)^{\gamma+1} - \left( q^{(2)}_I - (u^{(2)}_I)^{\gamma+1} \right) \\
\frac{q^{(1)}_I}{u^{(1)}_I} - \frac{q^{(2)}_I}{u^{(2)}_I}
\end{bmatrix} = \begin{bmatrix} 0 \\
0
\end{bmatrix},
$$

(3.3.26)

where the first condition is the conservation of the flux of the density for the interface values, and the second one is the conservation of the quantity \( \frac{q_I}{u_I} \) at the junction for each roads. These two conditions are enough in the simple 1-1 junction case to provide a unique solution to the problem.

3.3.6 1 incoming road, 2 outgoing road

In the 1-2 case, six unknowns need to be determined, \( u^{(k)}_L, q^{(k)}_L \), for \( k = 1, 2, 3 \). The admissible regions are first computed through the procedure outlined in section 3.3.3, from which we can determine the upper bound on the flux of the density for each road, denoted as \( \delta^\text{max}_k \) for \( k = 1, 2, 3 \). Then using the method outlined in 3.3.4, the equations for the maximization
of flux can be determined. For the case for 1 incoming road and 2 outgoing roads we can define the coupling conditions $\Phi_1$ from [42] as

$$
\Phi_1 = \begin{bmatrix}
q_i^{(1)} - (u_i^{(1)})^{\gamma+1} - \delta_1 \\
q_i^{(2)} - (u_i^{(2)})^{\gamma+1} - \delta_2 \\
q_i^{(3)} - (u_i^{(3)})^{\gamma+1} - \delta_3 \\
q_i^{(2)} - u_i^{(2)} \\
q_i^{(3)} - u_i^{(3)}
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}.
$$

(3.3.27)

The first three lines enforce the flux of the density being equal to the value determined from the maximization problem (3.3.21). The last two conditions enforce the maximization of speed on outgoing roads condition, which states that $q_i^{(k)} = u_i^{(k)}$ must hold for outgoing roads.

We also consider the coupling condition given in [45]. The maximization of density flux problem for this set of coupling conditions is slightly different in implementation than that of those developed for the conditions in [42]. For coupling conditions in [45], we still compute an admissible region but the maximization problem is not implemented over the value $\delta_k$, but instead its constituent components $q^{(k)}$ and $u^{(k)}$. That is to say, for the 1-2 case we have

$$
\hat{\delta}_1 = \max_{q^{(1)},u^{(1)}} q^{(1)} - (u^{(1)})^{\gamma+1}, \quad \text{subject to } \begin{cases}
(u^{(k)},q^{(k)})^T \in \text{AdR}, & \text{for } k = 1, \ldots, m \\
q^{(1)} - (u^{(1)})^{\gamma+1} = \sum_{k=2}^{3} q^{(k)} - (u^{(k)})^{\gamma+1} \\
\frac{q^{(1)}}{u^{(1)}} = \frac{q^{(2)}}{u^{(2)}} = \frac{q^{(3)}}{u^{(3)}} = \tilde{\epsilon}
\end{cases},
$$

(3.3.28)
where AdR is the admissible region for the density and pseudo-momentum (see [45] for
details on the computation of the admissible region). The value of $\tilde{\epsilon}$ is defined to be $\frac{q}{u}$ which
is determined through this optimization, so that the quantity is the same on each side of the
junction. This value is then the *fixed* value of $\tilde{\epsilon}$, describing the behavior of drivers, which
is used in the minimization of fluctuations problem. Once we have $\tilde{\delta}_1$, applying (3.3.22)
determines $\tilde{\delta}_2$ and $\tilde{\delta}_3$.

Having the setup with the maximization of density flux complete, the coupling
conditions (3.3.11) utilized in [45] are given as

$$
\Phi_2 = \begin{bmatrix}
q_I^{(1)} - \left(u_I^{(1)}\right)^{\gamma+1} - \tilde{\delta}_1 \\
q_I^{(2)} - \left(u_I^{(2)}\right)^{\gamma+1} - \tilde{\delta}_2 \\
q_I^{(3)} - \left(u_I^{(3)}\right)^{\gamma+1} - \tilde{\delta}_3 \\
\frac{q_I^{(1)}}{u_I^{(1)}} - \frac{q_j^{(2)}}{u_j^{(2)}} \\
\frac{q_I^{(1)}}{u_I^{(1)}} - \frac{q_j^{(3)}}{u_j^{(3)}} \\
\frac{q_I^{(1)}}{u_I^{(1)}} - \tilde{\epsilon}
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix},
$$

(3.3.29)

where we have added conditions 4 and 5 to enforce the conservation of the value of the
quantity $\frac{q}{u}$ across the junction, and condition 6 to enforce the equality of $\frac{q}{u}$ with the fixed
value of $\tilde{\epsilon}$.

### 3.3.7 2 incoming road, 1 outgoing road

In the 2-1 case, we also need to determine 6 unknowns, $u_L^{(k)}, q_L^{(k)}$, for $k = 1, 2, 3$. This
situation is more complicated than the previous two, as there is mixing involved at the
junction, since vehicles are entering from both incoming roads into a single outgoing road which can cause severe congestion. As with the 1-2 case, the admissible regions are first computed to determine $\delta_k^{\text{max}}$, which was outlined in section 3.3.3. Using the method outlined in 3.3.4, we obtain the equations from the maximization of density flux condition. The coupling conditions $\Phi_1$ can thus be defined from [42] as

$$
\Phi_1 = \begin{bmatrix}
q^{(1)}_I - (u^{(1)}_I)^{\gamma+1} - \hat{\delta}^1 \\
q^{(2)}_I - (u^{(2)}_I)^{\gamma+1} - \hat{\delta}^2 \\
q^{(3)}_I - (u^{(3)}_I)^{\gamma+1} - \hat{\delta}^3 \\
q^{(3)}_I - u^{(3)}_I 
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
0 \\
0 
\end{bmatrix},
$$

(3.3.30)

where the first 3 conditions are enforcing the values of the flux of the density on each road to be that of the solution of the maximization problem (3.3.24). The last condition enforces the maximization of speed condition.

We can also implement the coupling conditions from [45], where the maximization problem is slightly different from that of the 1-2 junction case. The maximization problem in this case is defined as

$$
\hat{\delta}_3 = \max_{q^{(1),(2)},u^{(1),(2)}} \sum_{k=1}^2 q^{(k)} - (u^{(k)})^{\gamma+1}, \quad \text{subject to} \quad \begin{cases}
(u^{(k)}, q^{(k)})^T \in \text{AdR} \\
\frac{q^{(1)} - (u^{(1)})^{\gamma+1}}{\beta_1} = q^{(3)} - (u^{(3)})^{\gamma+1} \\
\frac{q^{(2)} - (u^{(2)})^{\gamma+1}}{\beta_2} = q^{(3)} - (u^{(3)})^{\gamma+1} \\
\frac{q^{(3)}}{u^{(3)}} = \beta_1 \frac{q^{(1)}}{u^{(1)}} + \beta_2 \frac{q^{(2)}}{u^{(2)}} \equiv \hat{\epsilon}
\end{cases},
$$

(3.3.31)

where the $\beta_k$ are the coefficients from $\beta_k = \delta_k^{\text{max}} / \sum_{j=1}^2 \delta_j^{\text{max}}$ for $k = 1, 2$, and AdR is the
admissible region from [45]. From the definition of $\beta_k$, we can obtain the values for the maximized flux by $\hat{\delta}_k = \beta_k \delta_3$ for $k = 1, 2$. Similar to the 1-2 junction case, define the coupling conditions $\Phi_2$ from [45] as

$$
\Phi_2 = \begin{bmatrix}
q_I^{(1)} - (u_I^{(1)})^{\gamma+1} - \delta_1 \\
q_I^{(2)} - (u_I^{(2)})^{\gamma+1} - \delta_2 \\
q_I^{(3)} - (u_I^{(3)})^{\gamma+1} - \delta_3 \\
\beta_1 \frac{q_I^{(1)}}{u_I^{(1)}} + \beta_2 \frac{q_I^{(2)}}{u_I^{(2)}} - \frac{q_I^{(3)}}{u_I^{(3)}} - \hat{\epsilon}
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
0 \\
0
\end{bmatrix}, \quad (3.3.32)
$$

where the first three conditions in $\Phi_2$ correspond to conservation of the flux of the density, and the fourth and fifth representing the conservation of the quantity $q/u$ through the junction. The values of $\beta_1$ and $\beta_2$ are the coefficients describing what percentage of the flow into the outgoing road is coming from incoming road 1 and incoming road 2 which are defined after (3.3.31). The constants $\beta_k$ essentially play the same role as $q^*$ and $1 - q^*$ in (3.3.24), but do not necessarily have to be defined as a priori constants. For example, we could fix the $\beta_k$ at the start of the implementation as constants, or define $\beta_k$ as in the sum after (3.3.31). The choice is not unique as stated in [45]; the constants simply give preference to one road over another if $\beta_1 \neq \beta_2$ in the 2-1 junction case. These constants are introduced due to the possibility that the flow exiting the incoming roads is too great for the outgoing road to accept, which is why the introduction of these extra constants is required.
3.3.8 2 incoming road, 2 outgoing road

In the 2 incoming roads and 2 outgoing roads case, we must determine 8 unknowns, \( u^{(k)}_L \), \( q^{(k)}_L \), for \( k = 1, 2, 3, 4 \). The coupling conditions for the 2-2 junction case were only treated in [42], whereas the coupling conditions in [45] considered the \( \tilde{m} - 1 \) and \( 1 - \tilde{m} \) junction cases only. So for this section we will only consider one set of coupling conditions, \( \Phi \).

For the case for 2 incoming roads and 2 outgoing roads we can define the coupling conditions as

\[
\Phi = \begin{bmatrix}
q^{(1)}_I - (u^{(1)}_I)^{\gamma+1} - \hat{\delta}_1 \\
q^{(2)}_I - (u^{(2)}_I)^{\gamma+1} - \hat{\delta}_2 \\
q^{(3)}_I - (u^{(3)}_I)^{\gamma+1} - \hat{\delta}_3 \\
q^{(4)}_I - (u^{(4)}_I)^{\gamma+1} - \hat{\delta}_4 \\
q^{(3)}_I - u^{(3)}_I \\
q^{(4)}_I - u^{(4)}_I
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}, \tag{3.3.33}
\]

where the first four equations are the result of the maximization problem (3.3.25), and the last two equations are the maximization of speed condition on the two outgoing roads. The traffic distribution matrix for the 2-2 junction case is given by

\[
A = \begin{bmatrix}
\alpha_{1,1} & \alpha_{1,2} \\
\alpha_{2,1} & \alpha_{2,2}
\end{bmatrix},
\]

where \( \alpha_{1,1} + \alpha_{2,1} = 1 = \alpha_{1,2} + \alpha_{2,2} \). The relationship between the flux of the density at the junction is given by \( [\hat{\delta}_3, \hat{\delta}_4]^T = A \cdot [\hat{\delta}_1, \hat{\delta}_2]^T \), where the \( \hat{\delta}_1 \) and \( \hat{\delta}_2 \) are the solutions to the maximization of the flow of the density on incoming roads condition, defined in (3.3.25).
3.3.9 Optimization Problems

For the implementation of the coupling conditions described in this section, there are two optimization problems to be solved: the maximization problem in Subsection 3.3.4 which maximizes the flux of the density on incoming roads subject to various constraints, and the minimization of the fluctuations given by (3.3.10).

For the maximization problem, we use the `scipy.optimize.minimize` package in Python, where we implement the Sequential Least Squares Quadratic Programming (SLSQP) method [55, 56]. The maximization problem involves maximizing an objective function subject to nonlinear equality constraints \( \Phi \), with bounds on the values of \( u_L^{(k)} \) and \( q_L^{(k)} \). Due to the nonlinear equality constraints and bounds, the SLSQP solver can handle both constraints and bounds, and has also been robust in terms of initial conditions in all numerical tests for the maximization problem. Prior to the maximization of flux, we determine the admissible region for the flux of the density. The initial guess for the solver is taken to be the midpoint of the admissible region for the fluxes \( \delta_1 \) and \( \delta_2 \), depending on which junction case is being considered.

For the minimization problem (3.3.10), we implement the optimization algorithm via a Lagrange multiplier method described in [19]. The Armijo rule is utilized to determine the step size of the method. To simplify the notation, for the remainder of this subsection we drop the dependence on \( h \) and \( k \) so that \( U_L = (U_h^{(k)})_L \). To solve (3.3.10), the Lagrange functional is defined as

\[
\mathcal{L}(U_L, \mu) = g(U_L) + \mu^T\Phi(U_L),
\]

(3.3.34)
where
\[
g(U_L) = \sum_{k=1}^{m} \left\| D_{1/2}^{(k)} (U_L, U_R) \right\|^2 ,
\]
\[
\tilde{\Phi}(U_L) = \Phi(U_I(U_L, U_R)).
\]
(3.3.35)

At local minima the derivatives of (3.3.34) with respect to \(U_L\) and the Lagrange multipliers \(\mu\) are equal to zero:
\[
\nabla_\mu \mathcal{L}^T[\nu] = \nu^T \tilde{\Phi} = 0,
\]
\[
\nabla_{U_L} \mathcal{L}^T[\zeta] = \nabla_{U_L} g^T \zeta + \mu^T \nabla_{U_L} \tilde{\Phi} \zeta = 0.
\]
(3.3.36)

The first equation in (3.3.36) represents the coupling conditions, and the second equation in (3.3.36) is a linear system for the Lagrange multipliers \(\mu\). The system is overdetermined in this case as \(c < mn\), where \(c\) is the number of coupling conditions. We can compute the minimizing solution for \(\mu\) by the following equation
\[
\mu = -\left( \nabla_{U_L} \tilde{\Phi} \nabla_{U_L} \tilde{\Phi}^T \right)^{-1} \nabla_{U_L} \tilde{\Phi} \nabla_{U_L} g,
\]
(3.3.37)
and the update for the values of the left-hand state in the ghost cells is given by
\[
\tilde{U}_L^{(i+1)} = U_L^{(i)} - \alpha \left( \nabla_{U_L} g + \nabla_{U_L} \tilde{\Phi}^T \mu \right),
\]
(3.3.38)
where the Lagrange multipliers are found via (3.3.37). The step size \(\alpha\) can be determined through any line search algorithm, and the initial value \(U_L^{(0)}\) is used to start the update step in (3.3.38), so that we can determine the temporary value of \(U_L^{(i+1)}\), which we define as \(\tilde{U}_L^{(i+1)}\), such that the coupling conditions are satisfied, i.e. \(\tilde{\Phi} (\tilde{U}_L^{(i+1)}) = 0\). Then we use the temporary value of \(\tilde{U}_L^{(i+1)}\) as the initial value for the minimization of \(g(U_L)\) in (3.3.36), (3.3.10), and (3.3.11).
In the implementation for the initial condition of the solver, we define some vector $\xi$ of pseudo-random numbers from a uniform distribution on either $[-0.0001, 0.0001]$ or $[-0.001, 0.001]$, and define the initial value for the minimization problem as $U^0_L = U^0_R + \xi$, where $U^0_R$ is the vector of values of the states on right-hand side of the junction in the first cell of the domain.

In the numerical examples of Section 3.4, we use the Lagrange multiplier method as the optimizer of choice. The optimizer has shown to be robust in finding local minima given our initial values, but is more time consuming than the SLSQP algorithm. For the SLSQP algorithm, we specify the objective function $g(U_L)$, its Jacobian, and the coupling conditions $\Phi$ as equality constraints as arguments to the solver, and the solver returns the $U_L$ which minimize the objective function. We have chosen the Lagrange multiplier method for our numerical experiments due to the robustness on initial condition choice even though computation is more expensive. This is due to the fact that for some cases, the SLSQP method was sensitive to initial values for some junction types, and the method did not converge in all numerical experiments. When the SLSQP algorithm does converge, it is much faster than the Lagrange multiplier method, and converges to the same solution. As efficient treatment of optimization problems is not the goal of our paper, the Lagrange multiplier method has suited our needs for optimization. In any case, any choice of a nonlinear optimization solver that accepts bounds and constraints can be utilized here if speed is a concern.

$^2$Numerical tests have shown nearly identical results, pulling $\xi$ from either domain.
3.4 Numerical Experiments for the AR Model

In this section numerical results are presented for the proposed DG method (3.3.5) for the AR model for different types of network junctions. In subsection 3.4.1, we present a convergence rate test to show the optimal convergence rate of the DG scheme for each polynomial space from $P^0$ to $P^3$ elements using coupling conditions from [42]. In subsection 3.4.2, we consider the 1-1, 1-2, 2-1, and 2-2 junction types where we also consider coupling conditions provided in [42] for the AR model. The numerical results of the AR model plotted against the Godunov method for the LWR model (1.1.1), are also provided in this section, along with comparing the situation with different pressure terms for the AR model. Subsection 3.4.3 addresses the comparison of the different coupling conditions provided in subsection 3.2. Of special interest is subsection 3.4.4, where the numerical results are provided to support the case that capacity drop phenomenon can be observed with the second order AR model and the coupling conditions in [45].

3.4.1 Accuracy Test

In this section, we test the accuracy of our proposed high order DG method (3.3.5), with the Lax-Friedrichs flux and the SSPRK3 temporal scheme (3.3.7). For the accuracy test, we take the CFL constant to be CFL = 0.1 for the $P^0$ and $P^1$ solution spaces, and CFL = 0.05 for the $P^2$ and $P^3$ solution spaces, so that the spatial error dominates for the higher order polynomial spaces. We then define the time step in terms of the spatial step size as $\Delta t = CFL \Delta x$. The accuracy test is based upon comparing the solution at consecutive mesh sizes and computing the $L^1$ error.
We consider the 1-1 junction case for the AR model (3.1.1) and (3.1.2), with coupling conditions given by (3.3.26), and the initial conditions given by

\[ u(0, x) = 0.5 + 0.25 \cos(\pi x) \]
\[ v(0, x) = 1 - u(0, x). \]  

(3.4.39)

The computational domain is \([0, 1]\) for the incoming road and outgoing road, with the junction is located at \(x = 0\). The spatial mesh used for each \(P^\kappa\) is \(\Delta x = \frac{1}{160}\), and periodic boundary conditions are implemented. The solutions are computed up to time \(T = 0.1\). As seen in Table 3.4.1, we observe the \((k+1)^{\text{st}}\) order convergence rate of the DG method for \(P^\kappa\) \((\kappa = 0, 1, 2, 3)\) solution spaces for (3.3.5). The \(P^1_{\text{ref}}\) solution in Figure 3.4.2 is obtained by using a refined mesh size of \(\Delta x = 1/2560\), with the DG method implemented on a single continuous interval \([-1, 1]\), with no junction.

3.4.2 Numerical Experiments Comparing LWR and AR Models

In this section numerical results are presented for the proposed DG method (3.3.5) applied to AR model, and the DG method with the Godunov flux developed in [26] for the LWR models. In the first set of numerical test for each junction type, we do the same numerical test cases and initial conditions as in [26], to verify that the second order AR model can replicate the first order LWR model that was tested in the network case in [26]. Note that, if we take \(v(t, x) = 1 - u(t, x)\) and \(\gamma = 1\) in the pressure term \(p(u) = u^\gamma\), the AR model reduces to the LWR model. We take these parameters for the first test case of each junction type.
P0

<table>
<thead>
<tr>
<th>N</th>
<th>j</th>
<th>$|e^u|_{L^1}$</th>
<th>Order</th>
<th>$|e^q|_{L^1}$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0</td>
<td>1.2582e-02</td>
<td></td>
<td>1.2582e-02</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>6.3520e-03</td>
<td>0.98</td>
<td>6.3519e-03</td>
<td>0.98</td>
</tr>
<tr>
<td>40</td>
<td>2</td>
<td>3.1976e-03</td>
<td>0.99</td>
<td>3.1975e-03</td>
<td>0.99</td>
</tr>
<tr>
<td>80</td>
<td>3</td>
<td>1.6029e-03</td>
<td>0.99</td>
<td>1.6029e-03</td>
<td>0.99</td>
</tr>
<tr>
<td>160</td>
<td>4</td>
<td>8.0250e-04</td>
<td>0.99</td>
<td>8.0250e-04</td>
<td>0.99</td>
</tr>
<tr>
<td>320</td>
<td>5</td>
<td>4.0151e-04</td>
<td>0.99</td>
<td>4.0151e-04</td>
<td>0.99</td>
</tr>
</tbody>
</table>

P1

<table>
<thead>
<tr>
<th>N</th>
<th>j</th>
<th>$|e^u|_{L^1}$</th>
<th>Order</th>
<th>$|e^q|_{L^1}$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0</td>
<td>5.8382e-04</td>
<td></td>
<td>5.8378e-04</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>1.4462e-04</td>
<td>2.01</td>
<td>1.4462e-04</td>
<td>2.01</td>
</tr>
<tr>
<td>40</td>
<td>2</td>
<td>3.6042e-05</td>
<td>2.00</td>
<td>3.6042e-05</td>
<td>2.00</td>
</tr>
<tr>
<td>80</td>
<td>3</td>
<td>9.0033e-06</td>
<td>2.00</td>
<td>9.0033e-06</td>
<td>2.00</td>
</tr>
<tr>
<td>160</td>
<td>4</td>
<td>2.2503e-06</td>
<td>2.00</td>
<td>2.2503e-06</td>
<td>2.00</td>
</tr>
<tr>
<td>320</td>
<td>5</td>
<td>5.6256e-07</td>
<td>2.00</td>
<td>5.6256e-07</td>
<td>2.00</td>
</tr>
</tbody>
</table>

P2

<table>
<thead>
<tr>
<th>N</th>
<th>j</th>
<th>$|e^u|_{L^1}$</th>
<th>Order</th>
<th>$|e^q|_{L^1}$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0</td>
<td>3.1880e-05</td>
<td></td>
<td>3.1880e-05</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>4.7253e-06</td>
<td>2.75</td>
<td>4.7253e-06</td>
<td>2.75</td>
</tr>
<tr>
<td>40</td>
<td>2</td>
<td>6.9243e-07</td>
<td>2.77</td>
<td>6.9243e-07</td>
<td>2.77</td>
</tr>
<tr>
<td>80</td>
<td>3</td>
<td>1.0045e-07</td>
<td>2.78</td>
<td>1.0045e-07</td>
<td>2.78</td>
</tr>
<tr>
<td>160</td>
<td>4</td>
<td>1.4417e-08</td>
<td>2.80</td>
<td>1.4417e-08</td>
<td>2.80</td>
</tr>
<tr>
<td>320</td>
<td>5</td>
<td>2.0473e-09</td>
<td>2.81</td>
<td>2.0474e-09</td>
<td>2.81</td>
</tr>
</tbody>
</table>

P3

<table>
<thead>
<tr>
<th>N</th>
<th>j</th>
<th>$|e^u|_{L^1}$</th>
<th>Order</th>
<th>$|e^q|_{L^1}$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0</td>
<td>4.4062e-07</td>
<td></td>
<td>4.4062e-07</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>2.5307e-08</td>
<td>4.12</td>
<td>2.5307e-08</td>
<td>4.12</td>
</tr>
<tr>
<td>40</td>
<td>2</td>
<td>1.5541e-09</td>
<td>4.02</td>
<td>1.5541e-09</td>
<td>4.02</td>
</tr>
<tr>
<td>80</td>
<td>3</td>
<td>9.6444e-11</td>
<td>4.01</td>
<td>9.6444e-11</td>
<td>4.01</td>
</tr>
<tr>
<td>160</td>
<td>4</td>
<td>5.9881e-12</td>
<td>4.00</td>
<td>5.9881e-12</td>
<td>4.00</td>
</tr>
<tr>
<td>320</td>
<td>5</td>
<td>3.8495e-13</td>
<td>3.95</td>
<td>3.8507e-13</td>
<td>3.95</td>
</tr>
</tbody>
</table>

Table 3.4.1: Accuracy test for the system (1.2.13)

The 1-1 junction case is implemented for the accuracy test. The $L^1$ errors and orders for $P^0$, $P^1$, $P^2$, and $P^3$ solution spaces are given for the variables $u(x,T)$ and $q(x,T)$. Parameters: $\Delta x = 1/N$, $\Delta t = CFL\Delta x$, $T = 0.1$.  

84
Figure 3.4.2: One incoming and one outgoing road case with initial and boundary data given in (3.4.39).

To emphasize the difference between the LWR model and the AR model, we can then change the value of $\gamma$ in the pressure term as is done in [54], which is carried out in subsection 3.4.2.4. The numerical experiments in [54] are for the single road Riemann problem case. In those examples, the velocity of vehicles in the AR model is shown to be faster than that of the LWR model. This behavior arises from an additional intermediate state that does not appear in the first order LWR model, which implies that the vehicles in the AR model have a less densely packed distribution than vehicles in the LWR model. Similar to the single road case, we consider the same type of approach in the network case. These numerical experiments are carried out for the cases of 2 incoming roads and 1 outgoing road.
3.4.2.1 One Incoming Road and Two Outgoing Roads

In the first test with one incoming road and two outgoing roads, we use the following initial conditions

\[ u^{(1)}(0, x) = 0.1, \]  
\[ u^{(2)}(0, x) = \begin{cases} 0.1, & \text{if } x \in [0, 0.2] \cup [0.4, 0.6] \cup [0.8, 1], \\ 0.2, & \text{otherwise} \end{cases} \]  
\[ u^{(3)}(0, x) = 0.1, \]

with the coupling conditions given by (3.3.27), and with each road defined on the spatial interval \( x \in [0, 1] \). The traffic distribution matrix is \( A = [0.5, 0.5]^T \). We use the mesh size \( \Delta x = 1/160 \), and temporal step size \( \Delta t = CFL\Delta x \), where CFL is the number given in Subsection 3.4.1 for each \( P^k \) element space. The reference solution is obtained by running the Godunov method for the LWR model, as outlined in [26], with \( P^0 \) element and a spatial step size of \( \Delta x = 1/5120 \). The numerical results are given in Figure 3.4.3, which shows that the proposed higher order DG scheme with the Lax-Friedrichs flux provides good numerical results and can resolve shocks and rarefactions well. Numerical results also demonstrate a good agreement with the reference solution computed by the method in [26] for the LWR model.
### 3.4.2.2 Two Incoming Roads and One Outgoing Road

For the next test with two incoming roads and one outgoing road, we use the following initial conditions

\[
\begin{align*}
  u^{(1)}(0, x) &= \begin{cases} 
  0.1, & \text{if } x \in [0, 0.2] \cup [0.4, 0.6] \cup [0.8, 1], \\
  0.2, & \text{otherwise ,}
  \end{cases} \\
  u^{(2)}(0, x) &= 0.1 + 0.05 \sin(5\pi x), \\
  u^{(3)}(0, x) &= 0.1,
\end{align*}
\]

with the coupling conditions given by (3.3.30), and with each road defined on the spatial interval \(x \in [0, 1]\). We use the mesh size \(\Delta x = 1/160\), and temporal step size \(\Delta t = 160\). The reference solution is obtained by running the Godunov method for the LWR model, as outlined in [26], with \(P^0\) element and a spatial step size of \(\Delta x = 1/5120\). The results are given in Figure 3.4.4. Similar to the 1-2 junction case, the numerical experiment shows that the higher order DG scheme again provides better numerical results than the first order methods, and agree well with the reference solution.
3.4.2.3 Two Incoming Roads and Two Outgoing Roads

For the test with two incoming roads and two outgoing roads, we use the following initial conditions

\[ u^{(1)}(0, x) = \begin{cases} 0.1, & \text{if } x \in [0, 0.2] \cup [0.4, 0.6] \cup [0.8, 1], \\ 0.2, & \text{otherwise}, \end{cases} \tag{3.4.46} \]

\[ u^{(2)}(0, x) = 0.2 + 0.1 \sin(5\pi x), \tag{3.4.47} \]

\[ u^{(3)}(0, x) = 0.5, \tag{3.4.48} \]

\[ u^{(4)}(0, x) = 0.5, \tag{3.4.49} \]

with the coupling conditions given by (3.3.33), and with each road defined on the spatial interval \( x \in [0, 1] \). The traffic distribution matrix that we use is given as

\[ A = \begin{bmatrix} 0.4 & 0.3 \\ 0.6 & 0.7 \end{bmatrix}. \]

We use the mesh size \( \Delta x = 1/160 \), and temporal step size \( \Delta t = CFL\Delta x \). The reference solution is obtained by running the Godunov method for the LWR model, as outlined in [26], with \( P^0 \) element and a spatial step size of \( \Delta x = 1/5120 \). The results are given in Figure 3.4.5. In this more complicated situation, the higher order DG methods capture the complicated structure on the outgoing roads much better the first order \( P^0 \) method, which exhibits numerical diffusion at the discontinuities. Again, our results agree well with the reference solution by the method in [26].
3.4.2.4 Two Incoming Roads and One Outgoing Road with Different Pressure Terms

The goal of this test is to demonstrate the differences between the first order LWR model and second order AR model. We will use DG methods to solve both models. For the LWR model, the Godunov flux is used, and the Lax-Friedrichs flux is used for the AR model. Two cases of the AR model will be tested, one with $\gamma = 1$ and $\gamma = 2$, in the pressure term $p(u) = u^\gamma$. For the $\gamma = 1$ case, we will recover the LWR model as in the previous cases, while for $\gamma = 2$, we will observe the second order model effects. For this test with two incoming roads and one outgoing road, we use the following initial conditions

\[
    u^{(1)}(0, x) = \begin{cases} 
    0.5, & \text{if } x \in [0, 0.5], \\
    0.8, & \text{if } x \in (0.5, 1.0], 
\end{cases} \tag{3.4.50}
\]

\[
    u^{(2)}(0, x) = u^{(1)}(0, x), \tag{3.4.51}
\]

\[
    u^{(3)}(0, x) = u^{(1)}(0, x), \tag{3.4.52}
\]

with the coupling conditions given by (3.3.30), and with each road defined on the spatial interval $x \in [0, 1]$. We use the $P^2$ element space, the mesh size $\Delta x = 1/160$, and temporal step size $\Delta t = 0.05\Delta x$. The DG method with $P^0$ element space is used for the LWR model, with a spatial step size of $\Delta x = 1/5120$. The results at $T = 0.15$ are given in Figure 3.4.6 and the results at $T = 0.30$ are given in Figure 3.4.7. The first row of figures gives the density profiles on each road, while the second row gives the velocity profiles on each road. The AR model with $\gamma = 1$ in the pressure term should recover the LWR model, and at both times $T = 0.15$ and $T = 0.30$, we see that the AR model recovers the LWR model well.
When the AR model is used with $\gamma = 2$, we start to see the effects of using a second order model over the first order model. We can see that the density is more distributed over the incoming roads, and the speeds of the vehicles are greater. The behavior observed follows from the different wave speeds, which are affected by the pressure term in the AR model. The performance of the AR model in this numerical experiment agrees with the behavior outlined in [54], which reports similar results. Also of note in our junction model with the AR model case of $\gamma = 2$, the numerical experiment shows less congestion at the junction at the junction than the LWR case, as the density is distributed differently over the interval.

### 3.4.3 Comparisons of Different Coupling Conditions

In this section numerical results are presented for the proposed DG method (3.3.5) applied to the AR model, with different types of coupling conditions in [42] and [45].

#### 3.4.3.1 One Incoming Road and Two Outgoing Roads

For the first test with one incoming road and two outgoing roads, we use the following initial conditions

\[
    u^{(1)}(0, x) = 0.1, \quad (3.4.53)
\]

\[
    u^{(2)}(0, x) = \begin{cases} 
    0.1, & \text{if } x \in [0, 0.2] \cup [0.4, 0.6] \cup [0.8, 1], \\
    0.2, & \text{otherwise},
\end{cases} \quad (3.4.54)
\]

\[
    u^{(3)}(0, x) = 0.1, \quad (3.4.55)
\]

with the coupling conditions given by (3.3.27) (in [42]) and (3.3.29) (in [45]) to compare the effects of the choice of coupling condition. Each road defined on the spatial interval
$x \in [0, 1]$. We use the mesh size $\Delta x = 1/160$, and temporal step size $\Delta t = \text{CFL}\Delta x$. The reference solution is obtained by running the Godunov method for the LWR model, as outlined in [26], with $P^0$ element and a spatial step size of $\Delta x = 1/5120$. The results are given in Figure 3.4.8. The numerical results for both coupling conditions in this case are nearly identical, and both resolve the solution structure well for the high order method.

### 3.4.3.2 Two Incoming Roads and One Outgoing Road

For the next test with two incoming roads and one outgoing road, we use the following initial conditions

$$u^{(1)}(x, 0) = 0.1 \quad (3.4.56)$$

$$u^{(2)}(x, 0) = \begin{cases} 
0.1 & \text{if } x \in [0, 0.2] \cup [0.4, 0.6] \cup [0.8, 1] \\
0.2 & \text{otherwise}
\end{cases} \quad (3.4.57)$$

$$u^{(3)}(x, 0) = 0.1 \quad (3.4.58)$$

with the coupling conditions given by (3.3.30) and (3.3.32) to compare the effects of the choice of coupling condition. Each road defined on the spatial interval $x \in [0, 1]$. We use the mesh size $\Delta x = 1/160$, and temporal step size $\Delta t = \text{CFL}\Delta x$. The reference solution is obtained by running the Godunov method for the LWR model, as outlined in [26], with $P^0$ element and a spatial step size of $\Delta x = 1/5120$. The results are given in Figure 3.4.9. As with the 1-2 junction case, the numerical results for both coupling conditions are nearly identical, and both are able to achieve good results with a complicated solution structure on the outgoing roads for the high order method.
The comparison of coupling conditions tests thus far appear to illustrate that the coupling conditions in [42] and [45] give similar, if not the same, numerical results. This can be explained by the fact that the initial conditions chosen for the numerical tests represent traffic conditions that do no introduce severe congestion at the junction. In the next subsection, we show that the coupling condition developed in [45] gives capacity drop when severe congestion is present, while the coupling condition in [42] does not generate the same results.

### 3.4.4 Capacity Drop Phenomenon Test

In this subsection, we show the capacity drop phenomenon when applying the coupling conditions given in [45]. Capacity drop is a situation in the outflow of congested traffic is significantly lower than the maximum achievable flow at the same location. Consider the 2-1 junction case. If the flow from the two incoming roads is large enough that the outgoing road cannot accommodate the incoming flows, the result is flows leaving incoming roads are lower than optimal, and the flow entering the outgoing road from the junction rises, then falls due to congestion. In [45], it is stated that the coupling conditions in (3.3.32) may be able to replicate the capacity drop phenomenon.

More recently, in [57] and [48], examples are constructed to show that the capacity drop phenomenon can be achieved. The papers also state that the choice of initial conditions are sensitive to the observation of the phenomenon, and observations made from real data should be used to calibrate the model. For our example, we follow the general method that is used in the previously mentioned papers to construct an appropriate initial condition. We
will consider a constant density and velocity on Incoming Road 2 and Outgoing Road 3. We will then have density starting at a relatively low value and increase steeply, to overwhelm the outgoing road with an influx of vehicles, so that the resulting flow drops with an increase of density. As is done in [48], we track the density, density flux, and \( w = q/u \) on Incoming Road 1, and density and density flux on Outgoing Road 3. We implement the following initial conditions,

\[
\begin{aligned}
  u^{(1)}(0, x) &= \begin{cases} 
  0.1 + 0.7 \sin \left( \frac{5}{2} \pi x \right), & \text{if } x \in [0, \zeta], \\
  0.4, & \text{for } x \in (\zeta, 1],
  \end{cases} \\
  u^{(2)}(0, x) &= 0.2, \\
  u^{(3)}(0, x) &= 0.1,
\end{aligned}
\]

where \( \zeta = 0.4 - \left( \frac{2}{5\pi} \right) \arcsin \left( \frac{3}{7} \right) \). For the velocity functions, we take \( v(0, x) = 1 - u(0, x) \) as in the other numerical tests. The coupling conditions are given by (3.3.32), and with each road defined on the spatial interval \( x \in [0, 1] \). We use the mesh size \( \Delta x = 1/160 \), and temporal step size \( \Delta t = 0.05 \Delta x \). The simulation is implemented up to \( T = 0.655 \), with \( \gamma = 2 \) in the pressure term, with \( P^2 \) elements. We also take \( \beta_1 = \beta_2 = 0.5 \) for the merge constants in the coupling conditions instead of the given definition, as is done in [48]. For comparison, we use the second set of coupling conditions (3.3.30) (in [42]) which do not give the capacity drop phenomenon.

The main numerical results to present the capacity drop pattern are provided in Figure 3.4.11. The plots of the fluxes on Incoming Road 1 for the coupling conditions (3.3.32) and (3.3.30), are provided in Figures 3.4.11a and 3.4.11b, respectively. Both sets of coupling conditions observe lower than optimal flux values entering the junction when
congestion sets in, representing vehicles backing up on Incoming Road 1. Consider traveling on a freeway where two freeways merge together, where lanes entering the merge are densely populated with traffic. From the point of those sitting in traffic, it is easy to see that the flow through the junction slows down tremendously when density of vehicles is high, and is less than optimal.

In Figures 3.4.11c and 3.4.11d, the plots of the fluxes on Outgoing Road 3 are given for the coupling conditions (3.3.32) and (3.3.30), respectively. These plots support the conclusion that capacity drop is observed when using coupling conditions in (3.3.32). We observe that flow from the junction onto Outgoing Road 3 increases and then decreases once congestion sets in for the coupling conditions (3.3.32), whereas the coupling conditions in (3.3.30) have increasing flux and reach a near constant state. This behavior matches the capacity drop behavior which was reported in the discrete test case in [48]. Additionally, observe in Figure 3.4.10, that density is non-decreasing and velocity is non-increasing over time. Also note that in Figures 3.4.11e and 3.4.11f, that the quantity \( w = \frac{q}{u} \) decreases, then increases slightly for the coupling conditions in (3.3.32) which have a change in driver behaviors due to the coupling condition involving the quantity \( w = \frac{q}{u} \). In contrast, the coupling conditions in (3.3.30) have a near constant value. The final density profiles for each road at time \( T = 0.655 \) are given in 3.4.12. It is clear from numerical tests that the two sets of coupling conditions provide different results when severe congestion is introduced. Additionally, we reproduce the capacity drop phenomenon for our continuous model with DG, and replicate similar qualitative results to the discrete test case reported in [48].
Figure 3.4.3: One incoming and two outgoing roads - density plots

Initial and boundary data given by (3.4.40)-(3.4.42), coupling condition (3.3.27), $\alpha = 0.5$, and $T = 0.25$.

(a) Incoming Road 1;
(b) Outgoing Road 2;
(c) Outgoing Road 3.
Figure 3.4.4: Two incoming and one outgoing roads - density plots

Initial data given by (3.4.43)-(3.4.45), coupling condition (3.3.30), and $T = 0.25$. (a) Incoming Road 1; (b) Incoming Road 2; (c) Outgoing Road 3.
Initial data given by (3.4.46)-(3.4.49), coupling condition (3.3.33), and $T = 0.25$. (a) Incoming Road 1; (b) Incoming Road 2; (c) Outgoing Road 3; (d) Outgoing Road 4.
Figure 3.4.6: Two incoming and one outgoing roads - AR-LWR comparison, $T = 0.15$

Initial data is given by (3.4.50)-(3.4.52), coupling condition (3.3.30), and $T = 0.15$. (a) Incoming Road 1 density; (b) Incoming Road 2 density; (c) Outgoing Road 3 density; (d) Incoming Road 1 velocity; (b) Incoming Road 2 velocity; (c) Outgoing Road 3 velocity.
Figure 3.4.7: Two incoming and one outgoing roads - AR-LWR comparison, $T = 0.30$

Initial data is given by (3.4.50)-(3.4.52), coupling condition (3.3.30), and $T = 0.30$. (a) Incoming Road 1 density; (b) Incoming Road 2 density; (c) Outgoing Road 3 density; (d) Incoming Road 1 velocity; (b) Incoming Road 2 velocity; (c) Outgoing Road 3 velocity.
Figure 3.4.8: Comparison of coupling conditions between [42] and [45] - One incoming and two outgoing roads

Initial data given by (3.4.53)-(3.4.55), coupling conditions (3.3.27) and (3.3.29), $\alpha = 0.5$, and $T = 0.25$. (a) Incoming Road 1; (b) Outgoing Road 2; (c) Outgoing Road 3.
Figure 3.4.9: Comparison of coupling conditions between [42] and [45]. Two incoming and one outgoing roads.

Initial data given by (3.4.56)-(3.4.58), coupling conditions (3.3.30) and (3.3.32), $\alpha = 0.5$, and $T = 0.25$. (a) Incoming Road 1; (b) Incoming Road 2; (c) Outgoing Road 3.
Figure 3.4.10: Capacity drop test: density and velocity plots over time

Two incoming roads and one outgoing road case with initial data (3.4.59)-(3.4.61), coupling conditions (3.3.30) and (3.3.32), with $P^2$ elements, and $T = 0.655$. (a) Incoming Road 1 density over time with coupling conditions (3.3.32); (b) Incoming Road 1 density over time with coupling conditions (3.3.30); (c) Incoming Road 1 velocity over time with coupling conditions (3.3.32); (b) Incoming Road 1 velocity over time with coupling conditions (3.3.30);
Figure 3.4.11: Capacity drop test: density flux and $w = \frac{q}{u}$ plots over time

Two incoming roads and one outgoing road with initial data (3.4.59)-(3.4.61), coupling conditions (3.3.30) and (3.3.32), with $P^2$ elements, and $T = 0.655$. (a) Incoming Road 1 density flux using (3.3.32); (b) Incoming Road 1 density flux using (3.3.30); (c) Outgoing Road 3 density flux using (3.3.32); (d) Outgoing Road 3 density flux using (3.3.30); (e) Incoming Road 1 $w = \frac{q}{u}$ using (3.3.32); (f) Incoming Road 1 $w = v + p(u) = \frac{q}{u}$ using (3.3.30).
Two incoming roads and one outgoing road with initial data (3.4.59)-(3.4.61), coupling conditions (3.3.30) and (3.3.32), with $P^2$ elements, and $T = 0.655$. (a), (c), (e) Incoming Road 1, Incoming Road 2, Outgoing Road 3 densities with coupling conditions (3.3.32), respectively; (b), (d), (f) Incoming Road 1, Incoming Road 2, Outgoing Road 3 densities with coupling conditions (3.3.30), respectively.
Chapter 4

Basic Bathub Model

The layout for the final chapter is as follows. Section 4.1 presents a technical description of the basic bathtub model. Section 4.2 begins with a roadmap of the various theoretical results that are required for the solution algorithm. The section also provides a description of the numerical solution algorithm, with the relevant theoretical items included, in order to illustrate how the theory fits with the numerical work. Section 4.3 treats the existence and uniqueness of the equilibrium (or equilibria). Finally, section 4.4 gives numerical examples of the solution algorithm for two choices of the utility function.

4.1 The Bathtub Model

A fixed number of commuters per unit area travel from home to work in an isotropic downtown area over the morning rush hour. Trip origins and destinations are uniformly distributed over the area and each commuter travels the same exogenous trip distance, $L$. Commuters have identical tastes, which are described by a utility function with departure
time and trip duration as its arguments. The congestion technology is described by a function exhibiting a negative relationship between velocity and traffic density per unit area. Traffic density at a point in time equals cumulative entries to the road system (which equals cumulative departures from home) minus cumulative exits from the road system (which equals cumulative arrivals at work) to that point in time. Furthermore, cumulative entries at the time a commuter enters the road network equal cumulative exits at the time they exit it, a FIFO condition. Equilibrium is obtained when no commuter can increase their utility by departing at a different time, and all commuters travel. Since the model can be completely solved for from the entry rate function, solving the model reduces to solving for the entry rate function or functions consistent with the equilibrium conditions.

- The demand side of the model

A fixed number, $N$, of commuters per unit area travel from home to work in the morning rush hour over an isotropic downtown area. They have the same tastes that are described by the smooth utility function, $U(t, T)$, where $t$ is the departure time from home and $T$ is trip duration. The dependence of the utility function on $t$ captures scheduling preferences. The utility function has the properties that: i) there is increasing marginal disutility to travel time, $U_T < 0$, $U_{TT} < 0$; ii) $U_{tt} < 0$, with $U_t$ being positive for early departure times, zero at the most preferred departure time (conditional on $T$), and negative for late departure times; iii) it is smooth. The trip duration function is obtained by inverting the equal utility condition, $U(t, T(t)) = u$, to give $T(t; u)$. We assume as well that the utility function is quasi-concave, which implies that the trip duration function is strictly concave;
i.e. $\dot{T}(t; u) < 0$.\footnote{Differentiate $U(t, T(t)) = u$ twice. Differentiating once gives $U_t + U_T \ddot{T} = 0$, so that $\ddot{T} = -U_t/U_T$ (\dot{T}(t) therefore equals the marginal rate of substitution between trip duration and departure time). Differentiating a second times yields $U_{tt} + 2U_t T + U_{TT} T^2 + U_T \ddot{T} = 0$. Thus, $\dddot{T} = -\left[U_{tt} + 2U_t T + U_{TT} T^2\right]/U_T = -\left[U_{tt} - 2U_t U_t/\dot{T} + U_{TT}(U_t/\dot{T})^2\right]/U_T$, which is strictly negative by quasi-concavity of the utility function.} Commuters differ in their home and work locations. Work and home locations are uniformly distributed over the downtown area, and each commuter travels the same distance, $L$, between home and work. Each commuter chooses when to leave home so as to maximize their utility, taking as given the equilibrium relationship between trip duration and departure time, $T(t)$.

We summarize the main properties of the demand side as:

\begin{itemize}
  \item[(A-1): (i)] There is increasing marginal disutility to travel time: $U_T < 0$ and $U_{TT} > 0$.
  \item[(ii)] $U_u < 0$, with $U_t$ being positive for early departures, zero at the most preferred departure time (conditional on $T$), and negative for late departures.
  \item[(iii)] The function is quasi-concave.
\end{itemize}

Where $u$ denotes the equilibrium trip utility, the trip duration function is obtained by inverting the equal trip utility condition, $U(t, T) = u$, to give $T(t; u)$, and, under (A-1), has the properties that $1 + T(t; u) > 0$, and that $\dddot{T}(t; u) < 0$.

Some later results are obtained with the following additional assumption on tastes:

\begin{itemize}
  \item[(A-2): (i)] $\dot{T}_u(t; u) > 0$.
\end{itemize}

Since $\dot{T}(t; u)$ is the premium that a commuter is willing to pay in terms of travel time for a unit time later departure, $\dot{T}_u(t; u) > 0$ when this premium is increasing in utility, which is a normality condition.
Commuters differ in their home and work locations. Work and home locations are uniformly
distributed over the downtown, and each commuters travels the same distance, \( L \), between
home and work. Each commuter decides when to depart so as to maximize her utility,
taking as given the equilibrium relationship between trip duration and departure time.

- **The supply side of the model**

Traffic congestion is described by a function relating velocity to traffic density: \( v = v(k) \).
The fundamental identity of traffic flow is that the flow, \( q \), equals density times velocity.
Combining the identity and the velocity function yields \( q(k) = kv(k) \), which is the model’s
macroscopic fundamental diagram and provides an alternative description of the form of
traffic congestion in the model. It is assumed that the MFD has the following properties:

**(A-3):**

(i) \( q(0) = 0 \).

(ii) \( q(k_j) = 0 \), where \( k_j \) is jam density.

(iii) it is smooth.

(iv) there is a unique maximum flow, termed capacity flow, \( q_c \), which occurs at capacity
density, \( k_c \).

The evolution of traffic density over the rush hour is described by the following boundary
conditions, functions, and differential equation.

(i) \( k(t) = 0 \) where \( t \) is the time of the first departure from home (boundary condition).

(ii) \( k(T) = 0 \), where \( T \) is the time of the last arrival at work (boundary condition).

(iii) \( E(t) \) denotes cumulative entries to the road by time \( t \) and \( X(t) \) denotes cumulative
    exits from the road by time \( t \), \( k(t) = E(t) - X(t) \) (conservation of vehicles).

108
(iv) \( X(t + T(t)) = E(t) \), which states that cumulative arrivals by time \( t + T(t) \) equals cumulative exits by time \( t \) (FIFO condition).

(v) \( e(t) \equiv \dot{E}(t) \) is the rate at which cars enter the road system and \( x(t) \equiv \dot{X}(t) \) is the rate at which cars exit the road system (definition), so that from (iii), \( \dot{k}(t) = e(t) - x(t) \).

- **Equilibrium**

**E-1**: A **u-trip-timing equilibrium** is an entry/departure rate function, \( e(t) \), and exit/arrival rate function \( x(t) \), a density function, \( k(t) \), and a trip duration function \( T(t) \) such that:

(i) No commuter can increase her utility by altering her departure time. This implies that utilized is equalized over the departure set, \( D \), and is no higher outside the departure set:

\[
U(t, T(t)) = u \quad \text{for} \quad t \in D
\]
\[
U(t, T(t)) \leq u \quad \text{for} \quad t \notin D,
\]

where \( u \) is the exogenous equilibrium utility level.

(ii)

\[
X(t + T(t)) = E(t),
\]

which, as explained above, combines the conservation of cars and a FIFO condition.

Differentiation of (4.1.2) with respect to \( t \) yields

\[
x(t + T(t))(1 + \dot{T}(t)) = e(t),
\]
(iii) Since all commuters travel an exogenous distance $L$, and since at each point in time all cars travel at the same speed, which depends on density at that point in time, trip duration as a function of departure time is given implicitly by the condition

$$\int_{t}^{t+T(t)} v(k(s)) \, ds = L,$$

which states that trip distance equals the integral of velocity over the duration of the trip.

(iv) The entry rates, exit rates, and densities are non-negative.

We also define a second type of trip-timing equilibrium:

(E-2): An $N$-trip-timing equilibrium is defined in the same way as a $u$-trip-timing equilibrium, except that the level of utility is taken as endogenous and the population of commuters as exogenous, $N$. Thus, there is the additional condition\(^2\) that the equilibrium utility level be such that

$$\int_{t(u)}^{t^*(u)} e(t; u) \, dt = N \text{ (or } N\text{),}$$

where $t(u)$, $t^*(u)$, and $e(t; u)$ are the time of the first departure, the time of the last departure, and the entry rate function, associated with a $u$-trip-timing equilibrium.

A $u$-trip-timing equilibrium is the analog of an open-city equilibrium in urban economic theory, while an $N$-trip-timing equilibrium is the analog of a closed-city equilibrium in urban economic theory. Finally, we also define

(E-3): A full trip-timing equilibrium is defined in the same way as an $N$-trip-timing equilibrium, except a demand function replaces the exogenous population of commuters.

\(^2\)If we are considering the $N$-trip-timing equilibrium, then population is exogenous and we use (4.1.5) with $N$. Equation (4.1.5) also holds in the $u$-trip-timing equilibrium, except the $N(= N(u))$ is endogenous and depends on the exogenous utility.
The algorithm employed in [4] constructs a \( u \)-trip-timing equilibrium. As shall be shown below, the method of construction proves the existence and uniqueness of \( u \)-trip-timing equilibrium.

One may define the user cost of a trip, conditional on \( u \), \( uc(u) \) to be the maximum possible level of trip utility minus the actual trip utility, where the maximum possible level of utility is \( \rho \equiv \max_t U(t, 0) \), where \( \max_t U(t, 0) \) is the trip utility obtained from a trip of zero duration taken at the utility-maximizing time. The function relating equilibrium user cost to equilibrium utility is

\[
\hat{uc}(u) \equiv \rho - u, 
\] (4.1.6)

so that user cost monotonically decreasing in \( u \). The uniqueness of the \( u \)-trip-timing equilibrium implies that there is a unique population of commuters, associated with each level of \( u \). Thus, we may write

\[
N = \hat{N}(u) 
\] (4.1.7)

The existence and uniqueness of \( N \)-trip-timing equilibrium and of full trip-timing equilibrium will be discussed in section 4.3.

### 4.2 Solution Algorithm

The complicated nature of the mathematics of the bathtub model requires many moving parts in the derivation of the uniqueness of equilibrium. Many results that require the use of the assumptions laid out in section 4.1 are quite technical. A bird’s eye view of the steps to prove the existence and uniqueness of a \( u \)-trip-timing equilibrium is provided in Table
4.2.1. The table provides a list of theorems from this section regarding the $u$-trip-timing equilibrium with the result, and the corresponding Figure (if applicable) to give a geometric interpretation for the result.

<table>
<thead>
<tr>
<th>Item #</th>
<th>Result</th>
<th>Figure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notational conventions</td>
<td></td>
<td>Figure 4.2.1</td>
</tr>
<tr>
<td>5</td>
<td>Connectedness of the departure set</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Continuity of velocity $v(t; u)$</td>
<td>Figure 4.4.8</td>
</tr>
<tr>
<td>7, 8, 9</td>
<td>Sign of $\frac{\partial G_i}{\partial t}$, sign of $\frac{\partial G_i}{\partial u}$, location of $G_{i+1}$ with respect to $G_i$</td>
<td>Figure 4.2.3</td>
</tr>
<tr>
<td>10</td>
<td>Utility is strictly lower outside the departure interval $D$</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>$v(t; u)$ is non-negative</td>
<td>Figure 4.4.8</td>
</tr>
<tr>
<td>14</td>
<td>Entry rate is non-negative in $D$</td>
<td>Figure 4.4.8</td>
</tr>
<tr>
<td>15, 16, 17</td>
<td>$H(t, u, L) = 0$ is $C^1$, sign of $\frac{\partial H}{\partial t}$, sign of $\frac{\partial H}{\partial u}$</td>
<td>Figure 4.2.3</td>
</tr>
<tr>
<td>18</td>
<td>Uniqueness of $u$-trip-timing equilibrium</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.2.1: Outline of theorems for the existence and uniqueness of equilibrium

The central feature of our solution strategy is to solve for equilibrium as a function of $u$. One advantage of this procedure is that the equal utility condition can be inverted to give equilibrium trip duration as a function of departure time and the exogenous utility level, $T(t; u)$. Another advantage is that, as we shall see, for every value of $u$, there is a unique solution, which is not generally the case for $t$, $t^*$, or $N$. Intuitively, if the equilibrium trip duration function is known, it should be possible to solve for the time of the first departure, the time of the last departure, and the equilibrium entry rate function over $D$.

We define the refinement procedure in which the time of the first and last departure, and the equilibrium entry rate function are computed, such that the computed trip length $L$ is within some tolerance to the exogenous trip length $L$, as the inner loop of the solution.\footnote{We define the “entry rate function” to be entries as a function of time over the whole departure interval, $D$.}
algorithm. Once this refinement is completed, the population $N$ is computed, then the inner loop is repeated for different values of exogenous utility. The function $N(u)$ can then be constructed, from which the equilibrium (or equilibria) as a function of the exogenous population can be determined. We define the repeated inner loop computation for different exogenous utility levels as the outer loop of the solution algorithm. This intuition is sound, but it turns out to be more difficult than we had initially expected to solve completely for the full equilibrium given the equilibrium trip duration function.

The main element of the algorithm is a more explicit version of (4.1.4), which gives the dependencies on the endogenous value of $t$, and the exogenous utility $u$, specifically

$$\int_t^{t+T(t,u)} v(s; t, u) \, ds = L \quad \text{for } t \in D$$

which we term the distance condition. It states that, in equilibrium, for all departure times in the departure set, the integral of velocity over the duration of the trip equals the common, exogenous trip distance $L$. More specifically, (4.2.8) states this equilibrium condition for the exogenous utility level $u$. We shall show shortly how the unique velocity function $v(k(s; t, u))$ is computed. Additionally, we will show that for each $u$, the calculation of the $t$ or $t'$s that solve (4.2.8).

Before moving to the main equations of the algorithm, there are two results that require proof. The algorithm from [4] requires the derivative with respect to $t$ of (4.2.8). For the derivative to be well defined, the departure set $D$ must be connected for all $t \in D$. In Theorem 5, we only provide a proof for the case of discrete entries, not for a continuum of entries. We are fairly confident that the continuum case can be proven as a limit of the discrete case, as we have a tentative line of argument, but a more rigorous presentation
is required. We have assumed a smooth concave trip duration function, $T(t; u)$, which will be helpful in the proceeding analysis. The following theorem and conjecture establish, respectively: (i) the departure set is connected based upon the mentioned properties of the trip duration function, and (ii) continuity of the velocity function.

**Theorem 5** The departure set $\mathcal{D}$ is connected.

**Proof.** See Appendix A, Result 19. ■

**Conjecture 6** The velocity function $v(t, u)$ is continuous.

**Proof.** See Appendix A, Conjecture 20. ■

With Theorem 5 established, and Conjecture 6 taken as an assumption, the derivative of (4.2.8) is well defined and given by

$$v(k(t + T(t; u); t, u))(1 + \dot{T}(t; u)) - v(k(t; t, u)) = 0 \quad \text{for } t \in \text{Int}(\mathcal{D}). \quad (4.2.9)$$

Observe that (4.2.9), which we term the velocity condition, holds for any common trip length, $L$. Put alternatively, in moving from (4.2.8) to (4.2.9), a constant of integration is dropped. Eq. (4.2.9) indicates how a commuter’s velocity at the end of their trip must be related to their velocity at the start of their trip and their departure time, conditional on $t$ and $u$.

Before proceeding to the “cycle” structure of the problem, we outline the main ideas of the solution algorithm. We proceed in three steps:

(a) Determine the number of entry cycles (which we will define in the next subsection, 4.2.1) for a given equilibrium utility level $u$.

---

4By smooth, we mean that the function is $C^\infty$, that is, the function is continuously differentiable for all orders in the domain of the variables $t$ and $u$.
(b) In the inner loop, solve for the equilibrium entry rate function, conditional on starting values for \( u \) and \( t \). The loop continues updating the values of \( t \) and \( t^* \), with \( u \) fixed, until the computed trip length, \( L \), is within a provided tolerance of the exogenous trip distance \( L \) for the commuter who departs at \( t_1 \). At this point, \( t \) and \( t^* \) are known.

(c) Using the computed values for \( t \) and \( t^* \) as functions of \( u \), the complete solution as a function of \( u \) is calculated, including \( N(u) \) for the fixed \( u \). The level of utility is then decreased by a provided \( \Delta u \), and the inner loop is repeated.

Now we introduce the detailed cycle structure that is inherently present due to the delay differential equation, and explain how to determine the number of cycles.

### 4.2.1 Cycle Structure and Cycle Counting

Before proceeding to the structure of the algorithm, some terminology is in order. In working out the implications of the delay difference equation (4.2.9), we apply terminology from the literature on delay differential equations (Bellen and Zennaro [12]), specifically the terms cycles and breakpoints. Recall that \( t_1 \) is the first departure, \( u_\text{ex} \) is the exogenous utility level, and \( T(t; u) \) is trip duration as a function of departure time, conditional on \( u \).

The first cycle starts at \( t_1 \), the time the first commuter departs, and ends at \( t_1 + T(t_1; t, u) \), the time the first commuter arrives. If departures continue beyond \( t_1 + T(t_1; t, u) \), the first cycle is said to be a full cycle, and the second cycle begins at \( t_1 + T(t_1; t, u) \) and ends at \( t_1 + T(t_1; t, u) + T(t_1 + T(t_1; t, u); t, u) \), which is the arrival time of the commuter who departs at the time that the first commuter arrives. If, alternatively, departures cease before the first commuter arrives, the first cycle is said to be a partial cycle. Defining the third, fourth,
etc. cycles, analogously, the departure interval comprises \( I \) cycles.

Since the notation becomes cumbersome for large numbers of full entry cycles, with some abuse of notation, for the rest of this section we shall let \( t \) denote some time in the first entry cycle. We then define the following condensed notation: 
\[
t_1(t; u) \equiv t, t_2(t; u) \equiv t + T(t; u), t_3(t; u) \equiv t + T(t; u) + T(t + T(t; u)), \text{ and so on.}
\]
Thus, \( t_1(t; u) \) is the time in the first entry cycle corresponding to the time \( t \) in the first entry cycle, which is just \( t \); \( t_2(t; u) \) is the time in the second entry cycle corresponding to time \( t \) in the first entry cycle, and so on. Note that a first order condition can be obtained by taking a time derivative of (4.1.4). This condition naturally divides the departure set into entry cycles. Using the condensed notation, the first full entry cycle runs from \( t_1(t; u) \) to \( t_2(t; u) \), the period over which the first commuter to depart travels; the second full entry cycle runs from \( t_2(t; u) \) to \( t_3(t; u) \), the travel period of the commuter who departs when the first commuter arrives; and so on.

Our solution algorithm is formulated in terms of entry, but not exit, cycles. For the remainder of the chapter, we shall use the term “cycle” to refer to an “entry cycle”. We distinguish between full and partial cycles. In a full cycle, entry occurs throughout the cycle; in a partial cycle, which is always the last (entry) cycle, entry occurs over only an initial part of the cycle. Suppose, for the point of illustration, that the departure set comprises two full cycles, followed by a partial cycle. We say that there are three cycles. Let \( I \) denote the number of cycles, which is normally comprised of \( I - 1 \) full cycles, followed by a single partial cycle, and \( i \) is the index of the entry cycle.

With the simplified notation, we define the points \( t_i(t; u) \) as primary breakpoints. Now, consider the time of the last exit, \( \bar{t} \), which occurs in cycle \( I + 1 \). The individual who
exits at time $\bar{t}$ in cycle $I + 1$, entered in the previous cycle, $I$, at the point we call $t^*$, or the time of the last entry. We can follow successive cycles in the reverse direction until we reach the first cycle, ie. cycle $I = 1$. The value in the first cycle which corresponds to $t^*$ in cycle $I$ and to $\bar{t}$ in cycle $I + 1$, is what we define as $t' \equiv t_1(t'; u)$. Thus, using the simplified notation, we have $t^* = t_I(t'; u)$ and $\bar{t} = t_{I+1}(t'; u)$. We refer to the $t_i(t'; u)$ as the secondary breakpoints. Both primary and secondary breakpoints play a role in the solution, as we shall see, and are the cause of the discontinuities in the entry rate function.

Using the notion of primary and secondary breakpoints, we can further decompose the cycles into subcycles. Each cycle (except when $t^*$ occurs at a breakpoint) contains two subcycles, the first extending from $t_i(t; u)$ to $t_i(t'; u)$, which we term the first subcycle, and the second extending from $t_i(t'; u)$ to $t_{i+1}(t; u)$ which is referred to as the second subcycle. Figure 4.2.1 illustrates the notation and terminology with respect to cycles.

The above definitions for breakpoints and cycles can be used to write the equilibrium conditions in a concise way; note that

$$t_{i+1}(t; u) = t_i(t; u) + T(t_i(t; u); u),$$

(4.2.10)

so that

$$t'_{i+1}(t; u) = t'_i(t; u)(1 + \dot{T}(t_i(t; u); u)), \quad \text{where } t'_i(t; u) \equiv \frac{dt_i(t; u)}{dt}, \text{etc.} \quad (4.2.11)$$

Throughout the remaining sections of this chapter, we will use the condensed entry cycle notation to simplify mathematical expressions. For example, we can rewrite (4.2.9) using

\footnote{With some abuse of notation, in (4.2.11), $t'(\cdot)$ refers to the derivative with respect to $t$, which is a function. This is not to be confused with $t'$, which is a time in the first cycle.}
Figure 4.2.1: Terminology and notation for time and cycles

Notes: The rush hour is divided into cycles. The first cycle starts when the first commuter departs from home, $t = t_1(t'; u)$ and ends when he arrives at work, at which time the second cycle begins, $t_2(t'; u)$, and so on. The breakpoints are marked with dots.

The point in time at which the $i$th cycle ends and the $(i+1)$st begins is called the $(i+1)$st breakpoint. $t'$ is defined such that $t_{f}(t'; u) = t^{*}$; i.e., $t'$ is the time in the first cycle corresponding to the last departure. Within each cycle, the first subcycle extends from $t_i(t'; u)$ to $t_i(t'; u)$ and the second subcycle from $t_i(t'; u)$ to $t_{i+1}(t'; u)$. The figure shows the case of 3 cycles. The red (top) bar denotes where entries occur relative to the cycle location, and the blue (bottom) bar denotes where exits occur.
the cycle indexing notation as

\[ v(k(t_{i+1}(t; u); t, u)(1 + \hat{T}(t_i(t; u); u))) - v(k(t_i(t; u); t, u)) = 0 \]  

(4.2.12)

for \( t \in (t_i, t + T(t; u)) \) and \( i = 1, \ldots, I \), from which it follows that equilibrium velocity throughout the departure and arrival intervals may be calculated from the equilibrium velocity over the first cycle.

For the remainder of the chapter, to avoid confusion, we use the generic term, “breakpoint,” to refer to a primary breakpoint. Both types of breakpoints are central to the numerical solution algorithm. In the general problem where \( t \) is endogenous, the first \( I - 1 \) cycles will be full cycles, with the last cycle being a partial cycle. But if the time of the last departure coincides with the \((i + 1)\)st breakpoint, there are \( I \) full cycles and no partial cycle.

The algorithm requires knowing how many entry cycles there are in equilibrium. To establish the number of cycles, we proceed by repeatedly appealing to (4.2.12). Consider first the situation where there is one full cycle and no partial cycle. One boundary condition is that the velocity at the time of the first entry is \( v_f \); the other is that the velocity at the time of the last exit is also \( v_f \). Since the last exit occurs two full cycles after the first entry for this case, then applying (4.2.12), we have

\[ v_f = v(k(t_1(t; u); t, u)) = v(k(t_2(t; u); t, u)(1 + \hat{T}(t_1(t; u); u))) \]

\[ = v(k(t_3(t; u); t, u)) \prod_{i=1}^{2} (1 + \hat{T}(t_i(t; u))) \]

\[ = v_f \left( \prod_{i=1}^{2} (1 + \hat{T}(t_i(t; u))) \right), \]  

(4.2.13)
from which it follows that \( \prod_{i=1}^{2} (1 + \dot{T}(t_i(t; u))) = 1 \). This relation gives the locus of \((t, u)\) for which in equilibrium there is exactly one full entry cycle, and can be computed once the exogenous trip duration function \(T(t; u)\) is defined. The argument extends straightforwardly to an arbitrary number of full cycles.

Before proceeding to the general definition of the relationship determined by (4.2.13), we introduce the following definitions which will be used throughout the remainder of the paper

\[
A_i(t; u) \equiv k(t_i(t; u)), \quad B_i(t; u) \equiv (1 + \dot{T}(t_i(t; u); u)) > 0, \quad \text{and} \quad C_i(t; u) \equiv -\ddot{T}(t_i(t; u); u) > 0,
\]

and noting that (4.2.12) may be rewritten as

\[
v(A_{i+1}(t; u))B_i(t; u) = v(A_i(t; u)) \quad \text{for} \quad i = 1, \ldots, I,
\]

we can apply (4.2.15) recursively to give the compact version of (4.2.12)

\[
v(A_{i+1}(t; u)) = \frac{v(A_1(t; u))}{F_i(t; u)} \quad \text{for} \quad i = 1, \ldots, I,
\]

where

\[
F_i(t; u) = \prod_{j=1}^{i} B_j(t; u).
\]

Using the definitions in (4.2.14) and (4.2.17), we can define the general formula for the locus of \((t, u)\) for cycle \(I\) as

\[
G_I(t, u) \equiv F_{I+1}(t, u) - 1 = \prod_{j=1}^{I+1} B_j(t, u) - 1 = \prod_{j=1}^{I+1} \left( 1 + \dot{T}(t_j(t, u); u) \right) - 1,
\]

where \(G_I(t, u) = 0\) is exactly the locus of \((t, u)\) for which there are \(I\) full entry cycles. We refer to (4.2.18) as the \(G\)-functions. This equation proved useful in the numerical solution
algorithm in [4] for the $u$-trip-timing equilibrium since, as we shall see, it permits the pre-
determination of the number of entry cycles in solving for the equilibrium velocity and entry
function, which will be discussed later in this section.

Given the chosen pair of exogenous utility value and an initial guess of the en-
dogenous $t$, we determine the number of cycles from (4.2.18). Since (4.2.18) determines
functions in $(t, u)$-space, the chosen pair can be located in the plane with respect to the
$G$-functions. We can now establish some results about the $G$-functions. Specifically, we
want to establish monotonicity of the partial derivatives of the $G$-functions, if possible. We
would like to be able to describe how the relationship between the $G$-functions ultimately
affect the properties of the equilibrium.

**Theorem 7** The partial derivative $\frac{\partial G_i}{\partial t}(t, u) < 0$ for $i = 1, \ldots, I + 1$.

**Proof.** See Appendix A, Result 22. ■

The analogous result in the $u$ variable can also be shown, with the additional
restriction on the sign of $\dot{T}_u(t; u)$. The sign of the partial derivative of the $G$-functions with
respect to the utility level $u$ is given in the following theorem.

**Theorem 8** For $i = 1, \ldots, I+1$, a sufficient condition for $\frac{\partial G_i(t, u)}{\partial u} > 0$ is that $\dot{T}_u(t_i(t); u) > 0$; otherwise, its sign is in general indeterminate.

**Proof.** See Appendix A, Result 23. ■

Using the previous two monotonicity results, we can prove a result on how the
$G$-functions are aligned in $(t, u)$-space. The orientation of the $G$-functions is of importance,
as their positions will determine how many cycles there are for a given set of equilibrium $t$
and $u$. 121
Theorem 9  In \((t, u)\)-space, \(G_{i+1}(t, u) = 0\) lies to the left of \(G_i(t, u) = 0\).

Proof. See Appendix A, Result 24. ■

Theorem 9 is of particular significance, as it allows us to determine the number of cycles for a given \(t\) and \(u\). The \(G\)-functions as defined in (4.2.18) can be computed \textit{a priori} in the numerical algorithm, as the \(G\)-functions only require the first derivative of the travel duration function, \(\dot{T}(t, u)\), which is exogenous. Contour plots can be generated for the \(G_i(t, u) = 0\), which we stated previously, is the locus of \((t, u)\) for which there are \(i\) full entry cycles. The curves \(G_i(t, u) = 0\) are the zero level sets for fixed \(t\) and \(u\) of a more general surface in \(\mathbb{R}^3\). An example of the level sets is given in Figure 4.2.3. The \(G_1(t, u) = 0\) curve corresponds to the pair of \((t, u)\) values for which there are one full entry cycle. The region to the right of the \(G_1(t, u) = 0\) curve corresponds to \((t, u)\) pairs where there is one partial cycle of entries, while the region between the \(G_1(t, u) = 0\) and \(G_2(t, u) = 0\) curves correspond to \((t, u)\) pairs where there is one full cycle and one partial cycle of entries. The remaining regions are analogous. In Figure 4.2.3, only the first four \(G\)-functions are provided but any number can be chosen. In numerical experiments using the logarithmic utility function\(^6\), which will follow in section 4.4, it is shown that for the wide range of exogenous \(u\) considered, the number of cycles increases from one partial cycle of entries to 3 full cycles and a partial cycle of entries as \(u\) is decreased to a certain point. Then as \(u\) continues to decrease, the number of cycles decrease. The implication of this phenomenon is that there \textit{may not} be an equilibrium with four or more full cycles, but we have not shown this result analytically for general utility functions.

\(^6\)Note that limit on the number of cycles may not be a general result of arbitrary utility functions satisfying the assumptions from the beginning of the chapter.
Once a utility function that satisfies the technical conditions is specified, the $G$-functions can be determined exclusively without any other computation. By definition, the $G$-functions are only the product of the derivative of the trip duration function, which is just an inversion of the utility function. At this point, for any fixed value of $t$ and $u$, we can determine exactly how many cycles of entries will be present.

One could start with an arbitrary fixed value of $t$ and $u$, but there would be no guarantee that (4.1.4) would be satisfied for the exogenous value of $L$. At the beginning of the algorithm, we compute the maximum level of utility corresponding to one commuter who travels at free flow speed for distance $L$. This maximum level of equilibrium utility, $u_{\text{max}}$ provides a natural starting point for the inner loop to begin, as it provides an upper bound on the feasible equilibrium utility levels. Since only one commuter entered the system, we also know there exists a partial cycle of entries for the equilibrium utility $u_{\text{max}}$. Now how to find the endogenous time of the first departure, $t$? The inner loop provides us that solution.

4.2.2 The Inner Loop

The inner loop solves for the equilibrium entry rate function conditional on $t$ and $u$, and updates the value of $t$ until the trip length condition (4.1.4) is satisfied within a specified tolerance. Thus to obtain the full equilibrium solution conditional on $u$, we need to determine, for each $u$, the $t$ or $t'$s consistent with the exogenous trip length, $L$. For clarity, the functional dependence on $t$ and $u$ will be dropped until further notice\textsuperscript{7}.

\textsuperscript{7}In full notation, we would write for example, the velocity function as $v(k(t_i(t; u); t; u))$. In the simplified notation, we drop the last two arguments, so the functions will be given as $v(k(t_i(t; u)))$. This change occurs in the density, entry, and exit rate functions.
Define the function
\[
H(t, u; L) = \int_t^{t+T(t,u)} v(\tilde{k}(t_1(t; u))) \, dt - L, \tag{4.2.19}
\]
where \(\int_t^{t+T(t,u)} v(\tilde{k}(t_1(t; u))) \, dt\) is the algorithm’s computed trip distance, \(L\), which gives the distance traveled by the first commuter to depart, conditional on \(t\) and \(u\). The function \(\tilde{k}(t_1(t; u))\) is defined to be the density at time \(t_1(t; u)\) conditional on the exogenous utility level and the exogenous time of the first entry, computed via the algorithm. By construction, \(L\) is the same as the distance traveled by all other commuters. Therefore the function \(H(t, u; L) = 0\) implicitly defines the set of \((t, u)\) consistent with trip distance equaling the exogenous trip distance \(L\). We shall refer to \(H(t, u; L) = 0\) simply as the \(H\)-function. The inner loop computes the \(H\)-function and continues until the value of (4.2.19) is within the chosen tolerance, for the specified exogenous trip length \(L\).

With the inner loop refinement mechanism defined, we now develop a method to construct the endogenous velocity, density, and entry rate functions to compute (4.2.19).

First, differentiate (4.2.12) with respect to \(t\), which we can do thanks to Theorem 5 and Conjecture 6, yields
\[
v'(k(t_{i+1}(t; u)))\dot{k}(t_{i+1}(t; u))t'_{i+1}(t; u)B_i(t; u) - v(k(t_{i+1}(t; u)))C_i(t; u)t'_i(t; u)
- v'(k(t_i(t; u)))\dot{k}(t_i(t; u))t'_i(t; u) = 0, \quad \text{for } t \in \mathcal{D}, \tag{4.2.20}
\]
which we refer to as the acceleration condition. Equation (4.2.20) is a delay differential equation with an endogenous delay, relating \(k(t_i(t; u); t; u)\), \(\dot{k}(t_{i+1}(t; u); t; u)\), and \(k(t_{i+1}(t; u); t; u)\).

\(^8\text{The algorithm runs based on the estimated value of } t \text{ as stated earlier. Therefore, the computed trip distance } L \text{ is almost never the exogenous trip length } \underline{L}. \text{ The construction of the } G\text{-functions described in section 4.2.1 allow for a good starting value of } t \text{ for the algorithm, and allows us to know when to switch from one partial cycle to one full and one partial entry cycle, etc. The loop runs until the computed value is within some tolerance of } \underline{L}.\)
\[ \dot{k}(t_{i+1}; t; u), \text{ since given } u, \text{ the values of } T(t; u), \dot{T}(t; u), \text{ and } \ddot{T}(t; u) \text{ are known for any value of } t. \] It is the delay differential equation (4.2.20) that causes the mathematics to be both unfamiliar and complex. In order to simplify the algebra we shall assume that the congestion technology takes the form of Greenshields’ Relation, which specifies a negative linear relationship between velocity and density:

\[ v(k) = v_f \left(1 - \frac{k}{k_j}\right), \tag{4.2.21} \]

where \(v_f\) is free-flow velocity and \(k_j\) is jam density. This simplifies the algebra since the \(v'(k(\cdot))\) terms in (4.2.20) equal \(-v_f/k_j\), a constant. We also make two normalizations, setting \(v_f = 1\) and \(k_j = 1\), hence measuring velocity as a ratio of free-flow velocity and density as a ratio of jam density. Substitute out \(\dot{k}(t_{i+1}; t; u) = e(t_{i+1}; t; u) - x(t_{i+1}; t; u); \text{ then substitute out } x(t_{i+1}; t; u) = e(t_i; t; u)/B_i; \text{ which is obtained by differentiating } X(t_{i+1}; t; u) = E(t_i; t; u); \text{ use } t'_{i+1}(t; u) = t'_{i}(t; u) + \dot{T}(t_i; t; u); \text{ and finally use (4.2.12) to substitute out for } v(k(t_{i+1}; t; u); t, u), \text{ yielding}

\[ -e(t_{i+1}; t; u)B_i^2(t; u) + e(t_i; t; u))B_i(t; u) \]

\[ - \frac{v(k(t_i; t; u))C_i(t; u)}{B_i(t; u)} + e(t_i; t; u)) - \frac{e(t_{i-1}(t; u))}{B_{i-1}(t; u)} = 0. \tag{4.2.22} \]

The delay differential equation from (4.2.20) can now be written as the following

\[ -e(t_{i+1}(t; u))B_i(t; u)^2 + e(t_i; t; u)))(1 + B_i(t; u)) - v(A_1(t; u)) \frac{C_i(t; u)}{F_i(t; u)} - \frac{e(t_{i-1}(t; u))}{B_{i-1}(t; u)} = 0, \tag{4.2.23} \]

where we have used the definitions from (4.2.14) in (4.2.22). Since the \(B_i, C_i, \text{ and } F_i\) functions are exogenous and the \(v(\cdot)\) function is the equilibrium velocity function for the
first cycle, (4.2.23) indicates how the entry rates at corresponding points in time in three contiguous cycles must be related in order to satisfy the acceleration condition (4.2.20).

In order to utilize (4.2.23) for the system of equations, we need to prove that there exist no entries before or after the departure interval. To that end, we prove that the utility outside the departure interval is strictly lower than the equilibrium utility level, so that no individuals depart before \( t_1(t; u) \) or after \( t_1(t'; u) \).

**Theorem 10** Utility is strictly lower than the equilibrium value of \( u \) outside the departure interval.

**Proof.** See Appendix A, Result 21. ■

Eq. (4.2.23) defines a system of equations. There are two cases to consider. In case A, there is only one partial cycle. In case B, there is more than one full cycle of entries, corresponding to the situation where there exist two subcycles for each full cycle, as discussed earlier. It should also be mentioned that there are analytical results that can be obtained for Case A, which will be discussed after Cases A and B.

- **Case A: One partial cycle**

In this case, since \( I = 1 \), with \( i = 1 \), there are entries in the first subcycle of cycle 1, but there are no entries with \( i = 0 \) and \( i = 2 \), so that (4.2.23) reduces to

\[
e(t_1(t; u))(1 + B_1(t; u)) - v(k(t_1(t; u)))\frac{C_1(t; u)}{B_1(t; u)} = 0.
\]

(4.2.24)

Since density is zero at \( t \), we have that \( e(t_1(t; u)) = v_j \frac{C_1(t; u)}{(1 + B_1(t; u))B_1(t; u)} > 0 \). We now time step forward in increments of \( \Delta \). At time \( t + \Delta \), the number of entries that have occurred is approximately \( e(t_1(t; u))\Delta \). This is the density of traffic on the road at \( t + \Delta \),
which is strictly positive. We then use (4.2.24) with \( t = \bar{t} + \Delta \) to calculate \( e(\bar{t}; \bar{u}) + \Delta \), which is strictly positive, and the process is repeated until \( t' = t_1(t'; \bar{u}) \) is reached. The value of \( t' \) is determined by how the dynamics affect the last cycle. Recall that \( \bar{t} \) is the time of the last exit, which is defined by the right-hand boundary condition described in section 4.2.1. The value of \( \bar{t} \) is the \( t \) that satisfies \( v(\bar{k}(\bar{t}+T(\bar{t}; \bar{u}))) = v(\bar{k}(t_{I+1}(t'; \bar{u}))) = v_f \), which relates the time \( \bar{t} \) in the \((I + 1)\)st cycle to the value of \( t' \) in the first cycle. This also provides the time of the last entry, \( t^* = t_1(t'; \bar{u}) \) in the \( I \)th cycle. Therefore, the algorithm time-steps forward until the right-hand boundary condition is satisfied in the \((I + 1)\)st cycle, thus the values of \( t', \bar{t}, \) and \( t^* \) can be computed.

Now define \( \hat{e}(t(t; \bar{u})) \) to be the entry rate at time \( t_i(t; \bar{u}) \) conditional on the exogenous utility level and the exogenous time of the first entry, computed per this procedure. We can then write

\[
\hat{e}(t_1(t; \bar{u}))(1 + B_1(t; \bar{u})) - v(\hat{k}(t_1(t; \bar{u})))\frac{C_1(t; \bar{u})}{B_1(t; \bar{u})} = 0, \tag{4.2.25}
\]

from which it follows that the entry rate is strictly positive over \( D \). The entry rate also increases discontinuously from zero at \( \bar{t} \), and then decreases discontinuously to zero at \( t^* = t_1(t'; \bar{u}) = t' \).

**Case B: Greater than one full cycle**

In this case, since \( I > 1 \), there are entries in cycles 1 through \( I \). Using the cycle structure presented in Figure 4.1, there exists a natural way to separate the problem into two parts. We will restructure (4.2.23) to be used for the set of first subcycles, then for the set of second subcycles.

Recall that the first subcycles occur in the subcycles between the points \( t_i(t; \bar{u}) \) and
The equation \( t_i(t'; u) \), so there are a total of \( I \) first subcycles, in which there are entries over all of the first subcycles for \( i = 1, 2, \ldots, I \). In the 1st first subcycle, \( i = 1 \), there are no entries before \( t = t_1(t'; u) \), therefore (4.2.23) reduces to

\[
- e(t_1(t; u))B_1(t; u)^2 + e(t_1(t; u))(1 + B_1(t; u)) - v(k(t_2(t; u)))C_1(t; u) = 0. \tag{4.2.26}
\]

In the last of the first subcycles where \( i = I \), the opposite situation is true, i.e. there are no entries after \( t^* = t_I(t'; u) \). Thus (4.2.23) reduces to

\[
e(t_I(t; u))(1 + B_I(t; u)) - v(k(t_{I+1}(t; u)))C_I(t; u) - \frac{e(t_{I-1}(t; u))}{B_{I-1}(t; u)} = 0. \tag{4.2.27}
\]

For the intermediate \( i = 3, \ldots, I - 1 \), the full form of (4.2.23) is applied.

Defining \( A_i(t; u) \equiv k(t_i(t; u)) \), the system of entry rate equations for the first subcycles, (4.2.23), (4.2.26), and (4.2.27) can be written in matrix form as:

\[
\begin{bmatrix}
(1 + B_1) & -B_1^2 & 0 & 0 & 0 & \cdots & 0 \\
-\frac{1}{m_1} & (1 + B_2) & -B_2^2 & 0 & 0 & \cdots & 0 \\
0 & -\frac{1}{m_2} & (1 + B_3) & -B_3^2 & 0 & \cdots & 0 \\
0 & 0 & \ddots & \ddots & \ddots & \ddots & 0 \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & -\frac{1}{m_{I-2}} & (1 + B_{I-1}) & -B_{I-1}^2 \\
0 & 0 & \cdots & 0 & 0 & -\frac{1}{m_{I-1}} & (1 + B_I)
\end{bmatrix}
\begin{bmatrix}
e(t_1(t; u)) \\
e(t_2(t; u)) \\
e(t_3(t; u)) \\
\vdots \\
e(t_{I-1}(t; u)) \\
e(t_I(t; u))
\end{bmatrix}
= \begin{bmatrix}
v(A_2)C_1 \\
v(A_3)C_2 \\
v(A_4)C_3 \\
\vdots \\
v(A_{I-1})C_{I-1} \\
v(A_I)C_I
\end{bmatrix} \tag{4.2.28}
\]

Substituting (4.2.17) into (4.2.28), and substituting the numerical approximation \( \hat{e}(t_i(t; u)) \)

\footnote{Note that \( i = 2 \) is not included in the intermediate subcycles case. For this special case, \( I = 2 \), and only (4.2.26) and (4.2.27) are required, so then \( i = I = 2 \) corresponds to (4.2.27) being used in the second cycle. Thus the 2×2 matrix of (4.2.28) is applied for the first subcycles.}
for the true entry rate \( e(t_i(t'; u)) \) gives

\[
\begin{bmatrix}
(1 + B_1) & -B_1^2 & 0 & 0 & 0 & \cdots & 0 \\
-\frac{1}{\pi} & (1 + B_2) & -B_2^2 & 0 & 0 & \cdots & 0 \\
0 & -\frac{1}{\pi} & (1 + B_3) & -B_3^2 & 0 & \cdots & 0 \\
0 & 0 & \ddots & \ddots & \ddots & \ddots & 0 \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & -\frac{1}{\pi_{i-2}} & (1 + B_{i-1}) & -B_{i-1}^2 \\
0 & 0 & \cdots & 0 & 0 & \frac{1}{\pi_{i-1}} & (1 + B_i)
\end{bmatrix}
\begin{bmatrix}
\tilde{e}(t_1(t'; u)) \\
\tilde{e}(t_2(t'; u)) \\
\tilde{e}(t_3(t'; u)) \\
\vdots \\
\tilde{e}(t_{i-1}(t'; u)) \\
\tilde{e}(t_i(t'; u))
\end{bmatrix}
= 
\begin{bmatrix}
\frac{v(A_1)C_1}{\pi_1} \\
\frac{v(A_1)C_2}{\pi_2} \\
\frac{v(A_1)C_3}{\pi_3} \\
\vdots \\
\frac{v(A_1)C_{i-1}}{\pi_{i-1}} \\
\frac{v(A_1)C_i}{\pi_i}
\end{bmatrix}
\tag{4.2.29}
\]

Conditional on \( t \), the entry rate function over the first subcycle of the first cycle, \( \tilde{e}(t_1(t'; u)) \), the corresponding density function, \( \tilde{\kappa}(t_1(t'; u)) \), and the corresponding velocity function, \( v(\tilde{\kappa}(t_1(t'; u))) \), may be computed by time-stepping forward in \( t \) until the appropriate boundary condition is satisfied. The boundary condition is \( v(t_{I+1}(t'; u)) = v_f \), i.e. the velocity reaches free-flow velocity in the \((I + 1)\)st cycle. We can then locate the value of \( t_1(t'; u) \) knowing \( t_{I+1}(t'; u) \), via (4.2.10).

Note that the computation for the entry rates requires the discretization of time over only the first and second subcycles of the first cycle. The points in time for the subsequent cycles \( i = 2, \ldots, I + 1 \) are automatically determined by relation (4.2.10). So then, the functions \( \tilde{e}(t_i(t'; u)), \tilde{\kappa}(t_i(t'; u)) \), etc. are also determined at these points. The computations for the set of first subcycles is complete\(^{10}\).

Now we turn to the case for the set of second subcycles. The second subcycles occur in the subcycles between the points \( t_i(t'; u) \) and \( t_{i+1}(t'; u) \), so there are a total of \( I - 1 \) second

\(^{10}\)A special case which occurs when there are only full cycles, only requires the computation of (4.2.29), as \( t' \) will be a breakpoint. This situation is discussed in section 4.2.4.
subcycles, in which there are entries over all of the second subcycles for \( i = 1, 2, \ldots, I - 1 \). In the 1st second subcycle, \( i = 1 \) for \( I > 2 \), there are no entries before \( t = t_1(t; u) \), therefore (4.2.23) reduces to (4.2.26). In the special case when \( I = 2 \), then there is only one second subcycle, which corresponds to (4.2.24), which is equivalent to the 1 \times 1 case of (4.2.31).

For the case when \( I = 3 \), the same reasoning for the equations used in footnote 9 for the \( I = 2 \) first subcycles is used, hence the 2 \times 2 case of (4.2.31) is applied. In the last of the first subcycles where \( I > 3 \) and \( i = I - 1 \), there are no entries after \( t^* = t_I(t'; u) \), then

\[
\hat{e}(t_{I-1}(t; u))(1 + B_{I-1}(t; u)) - v(\hat{k}(t_I(t; u)))C_{I-1}(t; u) - \frac{\hat{e}(t_{I-2}(t; u))}{B_{I-2}(t; u)} = 0 \quad (4.2.30)
\]

is used. For the intermediate second subcycles \( i = 4, \ldots, I - 2 \), the full form of (4.2.23) is used. In terms of the matrix equation (4.2.29), the \( I \)th row and \( I \)th column are deleted, yielding

\[
\begin{bmatrix}
(1 + B_1) & -B_1^2 & 0 & 0 & \cdots & 0 \\
-\frac{1}{B_1^2} & (1 + B_2) & -B_2^2 & 0 & \cdots & 0 \\
0 & -\frac{1}{B_2^2} & (1 + B_3) & -B_3^2 & \cdots & 0 \\
0 & 0 & \ddots & \ddots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & -\frac{1}{B_{I-2}^2} & (1 + B_{I-1})
\end{bmatrix}
\begin{bmatrix}
\hat{e}(t_1(t; u)) \\
\hat{e}(t_2(t; u)) \\
\hat{e}(t_3(t; u)) \\
\vdots \\
\hat{e}(t_{I-1}(t; u))
\end{bmatrix}
= \begin{bmatrix}
v(A_1)C_1 \\
v(A_1)C_2 \\
v(A_1)C_3 \\
\vdots \\
v(A_1)C_{I-1}
\end{bmatrix}
\quad (4.2.31)
\]

Conditional on \( t \), the entry rate function over the second subcycle of the first cycle, \( \hat{e}(t_1(t; u)), \hat{k}(t_1(t; u)), \) and \( v(\hat{k}(t_1(t; u))) \), may be computed by time-stepping forward for all \( t \) values in the interval \( (t_1(t'; u), t_2(t'; u)) \).

Thus, we have derived the entry rate function conditional on \( t \) and \( u \). We have done this without imposing the restriction that the entry rate be non-negative over \( D \).
The following results establish positivity of the entry rate, velocity, and density functions.

**Theorem 11** The velocity function $v(t,u)$ is non-negative over the rush hour. By the normalized Greenshields’ Relation, $k(t,u)$ is also non-negative.

**Proof.** See Appendix A, Result 25. □

Before proceeding to the proof of positive entry rates, some notation and definitions are in order due to the technical nature of the proof. Throughout the proof when the inequality symbol, $\preceq$, is used with a matrix, we mean the inequality holds element-wise. For example, if a matrix $A$ satisfies $A \preceq 0$, then $a_{ij} \leq 0$ for all $i$ and $j$. We also define two special types of matrices:

**Definition 12** A matrix $A = (a_{ij})$ for $1 \leq i,j \leq n$ is said to be a $Z$-matrix of size $n \times n$ if $a_{ij} \leq 0$ for all $1 \leq i,j \leq n$ and $i \neq j$.

**Definition 13** Let $A$ be a $Z$-matrix, and either one of two conditions hold

- every real eigenvalue of $A$ is positive, or
- all leading principal minors of $A$ are positive,

then $A$ is said to be an $M$-matrix. Furthermore, $A^{-1}$ exists with $0 \preceq A^{-1}$.

**Theorem 14** The entry rate is non-negative over the departure interval.

**Proof.** See Appendix A, Result 26. □
From Theorem 14, a simple corollary follows; that the exit rate $x(t, u)$ is non-negative. The proof is straightforward, as the derivative of the equilibrium FIFO condition (4.1.3) gives that $x(t + T(t)) = e(t)/(1 + \dot{T}(t))$, so exits must be positive if entries are positive.

Now, consider the case of one full cycle followed by a partial cycle. At $t_1(t; u)$ the entry rate increases discontinuously from zero; at $t_1(t'; u)$ the entry rate changes discontinuously as the equation system determining the entry rate switches from having one rather than two equations; at $t_2(t; u)$ the exit rate increases discontinuously from zero, and to maintain the continuity of the velocity function required by the equal-trip-cost condition, the entry rate changes discontinuously too. And so on. Thus, discontinuities in the entry and exit rates at the primary breakpoints between entry cycles, and between subcycles within each cycle (secondary breakpoints) are properties of the solution, despite the smoothness of the congestion and utility functions.

- **Analytical Solution for Case A**

Recall Case A, where there are only entries in first cycle (also including the case of one full cycle). The entry rate is governed by (4.2.24), which can be rewritten in terms of cumulative entries, $E(t)$, to give the ODE

$$\dot{E}(t_1(t; u))(1 + B_1(t_1(t; u))) - (1 - E(t_1(t; u))) \frac{C_1(t_1(t; u))}{B_1(t_1(t; u))} = 0,$$

(4.2.32)

where the normalized Greenshields’ Relation and $k(t) = E(t)$ are used. The latter holds, since there are no exits in the first cycle. The ODE (4.2.32) has the exact solution in terms
of the original exogenous functions:

\[
E(t) = \begin{cases} 
1 - \tilde{C} \frac{B_1(t; u)}{1 + B_1(t; u)} & \text{for } t \in [t_1(t; u), t_1(t' = t^*; u)] \\
E(t^*; u) & \text{for } t \in [t_1(t' = t^*; u), t_2(t'; u)],
\end{cases}
\]

\[
e(t) = \begin{cases} 
\tilde{C} \frac{C_1(t; u)}{(1 + B_1(t; u))^2} & \text{for } t \in [t_1(t; u), t_1(t' = t^*; u)] \\
0 & \text{for } t \in [t_1(t' = t^*; u), t_2(t'; u)],
\end{cases}
\]

where the constant \( \tilde{C} \) is found using the initial condition \( E(t) = 0 \) is given by

\[\tilde{C} = \frac{1 + B_1(t; u)}{B_1(t; u)}.\]

Recall that in the case of one partial cycle, \( t_1(t'; u) = t^* \). Using the FIFO condition,
\( k(t_1(t; u)) = E(t_1(t; u); u) \), and \( k(t_2(t; u)) = E(t_2(t; u); u) - X(t_2(t; u)) \). Since \( E(t; u) \) is
cumulative entries, \( E(t_2(t; u); u) = E(t^*; u) \). Therefore, density in the second cycle is given
by \( k(t_2(t; u)) = E(t^*; u) - X(t_2(t; u)) \), where \( X(t_2(t; u)) = E(t_1(t; u)) \), again by the FIFO
condition. The velocity function is easily determined through the normalized version of
(4.2.21): \( v(t; u) = 1 - k(t; u) \). Finally, we can compute the normalized population as a
function of the equilibrium utility level, \( N(u) \), by using (4.1.5), which yields:

\[N(u) = \tilde{C} \left[ \frac{B_1(t; u)}{1 + B_1(t; u)} - \frac{B_1(t^*; u)}{1 + B_1(t^*; u)} \right]. \quad (4.2.33)\]

Note that the solution presented here is \textit{general}, as a specific utility function was not
used; it need only satisfy the conditions presented in section 2. When the number of cycles
is greater than one, the endogenous delay is present during the first cycle via (4.2.26), and
thus the delay differential equation is recovered, as in Case B. Therefore, the same kind of
computation is no longer possible.

But before moving to the outer loop, we make a number of remarks.
4.2.3 Remarks

1. We have employed the boundary condition that traffic density is zero immediately before the first commuter departs \( (v(k(t^-)) = v_f) \) in numerically solving the function \( v(\hat{k}(t_1(t^-))) \).

We have also employed the boundary condition that traffic density is zero immediately after the last commuter exits \( (v(k((t^*)^+ + T((t^*)^+;u))) = v_f) \) in solving for \( t^*(t;u) = t_I(t';u) \).

The derivations of the previous section drew heavily on (4.2.9), the velocity condition, and its derivative, (4.2.20), the acceleration condition. They did not, however, employ (4.1.4), the distance condition, from which both (4.2.9) and (4.2.20) are derived. Since (4.2.9) is derived through time differentiation of (4.1.4), a constant of integration is lost from in moving from (4.1.4) to (4.2.9). That constant of integration is the exogenous trip distance, \( L \), which is used to obtain a complete solution of equilibrium, conditional on \( u \). The previous subsection solved for the unique equilibrium, conditional on \( t \) and \( u \), which corresponds to some trip distance, but not generally the exogenous trip distance. In [23] we prove that, conditional on \( u \), there is a one-to-one mapping from \( t \) to \( L \), a result that is used in the next subsection. Thus, there is a unique equilibrium conditional on \( u \) and \( L \). Calculating the population corresponding to this equilibrium generates the function \( N(u;L) \). The equilibrium or equilibria with an exogenous trip distance and an exogenous population then correspond to utility levels that solve \( N = N(u;L) \). We shall see in the next section that there may be two utility levels corresponding to an exogenous population and trip distance, one corresponding to aggregate congestion, the other to aggregate hypercongestion.

2. This is a convenient point for a digression on the relationship between the utility function employed in this paper and the user cost function. The easiest way to relate them is
to assume, first, that the utility function $U(t, T(t))$ is in fact a subutility function of a mixed direct-indirect total utility function having $t, T(t)$, and $y$, income, as its arguments: $V(t, T(t), y)$, and, second, that the total utility function is additively separable between $U(t, T(t))$ and $y$. Then $V(t, T(t), y) = U(t, T(t)) + y$. Since the marginal utility of income is then one, $U(t, T(t))$ is measured is money units. Now define $\nu$ to be the utility that would be achieved if trip duration were zero and if utility were maximized with respect to $t$: $\nu = \max_t U(t, 0)$. Then $\nu - U(t, T(t))$ equals the money-metric loss in utility from a commuting trip with travel duration $T$ and departure time $t$ compared to a commuting trip of zero duration and departure at the optimal time. Define this to be the user cost. Since all commuters experience the same utility in equilibrium, $\underline{u}$, one may define $\tilde{c}(\underline{u}) = \nu - \underline{u}$ to be the equilibrium user cost. Then this function and the function $N(\underline{u})$ together give a relationship between user cost and population, which is the user cost “function” $c = c(N)$.

3. The solution of the last subsection did not take into account several inequality conditions that equilibrium must satisfy: i) neither the entry rate nor the exit rate can be negative; ii) traffic density cannot be negative; and iii) utility be below the equilibrium utility level everywhere outside the departure interval. Theorem 14 shows that the mathematical structure of the problem implies that the entry rate is strictly positive over the departure interval, which implies via (4.1.3) and (4.2.9) that the exit rate is strictly positive from the time of the first exit to the time of the last exit. Since cumulative exits lag cumulative entries, since cumulative entries are strictly positive over the interior of the departure interval, and since the number of entries over the rush hour equals the number of exits, density cannot be negative. By construction, the entry rate function solved for in the previous subsection,
which depends on $t$ and $u$, has the property that utility is constant over the departure interval. Furthermore, Theorem 10 shows that the utility is everywhere lower outside the departure interval, so that (4.1.1) is satisfied.

4. The dynamic structure of the problem is unfamiliar. If traffic density were instead determined as the solution to an ordinary differential equation, the equation of motion would have the form $\dot{k}(t) = f(e(t), k(t))$, and the solution would be obtained by combining this equation of motion with appropriate boundary conditions on functions of the state variable. Here instead, the equation of motion takes the form $\dot{k}(t + T(t; u)) = e(t + T(t; u)) - x(t + T(t; u)) = e(t + T(t; u)) - e(t)/(1 + \dot{T}(t; u))$. The boundary conditions are unfamiliar too. In the simplest form of a delay differential equation, the boundary condition is an exogenous initial history, the analog of which here would be the entry function over the first cycle. But in the problem at hand, the initial history has to be solved for by the conditions that the first entry cycle generates a final exit cycle, with the property that all the commuters exiting during the final exit cycle satisfy the equilibrium trip-timing conditions. To see this, consider starting with an arbitrary non-negative entry function over the first cycle. The entry function over the first cycle determines the exit function over the second cycle, and the entry function over the second cycle is chosen so that all commuters in the first cycle satisfy the trip timing condition. This process continues until the last entry, but then there are no subsequent entries to choose to satisfy the equilibrium trip-timing condition for commuters still traveling at this time. Considering the complexity of the dynamic structure of the problem, it is perhaps surprising that the solution procedure is as “simple” as it is.
4.2.4 The Outer Loop

Recall Walters’ analysis of steady-state traffic flow with MFD congestion. It solves for the user cost and flow as a function of traffic density, and then plots user cost against flow as density increases, as shown in Figure 4.2.2, obtaining the user cost curve. Equilibrium is then characterized as points of intersection of the demand curve and the user cost curve.

![Figure 4.2.2: Walters’ user cost diagram.](image)

Our procedure here is analogous, except that we solve for user cost and the commuter population as a function of utility, and then plot user cost against commuter population to obtain the user cost curve. In Walters’ construction, there is always a backward-sloping portion of the user cost curve, which corresponds to hypercongestion. In our construction, depending on the form of the utility function, there may or may not be a backward-sloping portion of the user cost curve, which corresponds to aggregate hypercongestion.
The possibility of aggregate hypercongestion has not, to our knowledge, been noted before, but is not difficult to understand intuitively. In our central example, the utility function is such that the earliest feasible departure from home occurs at 5 am, while the latest feasible arrival at work occurs at 11 am. These two constraints, along with the exogenous trip length, imply a maximum feasible commuter population that can be accommodated over the rush hour. Now suppose that demand is high and price-sensitive. The only way that equilibrium can be achieved is with a user cost that is higher than that consistent with congested traffic flow.

When aggregate hypercongestion is a possibility, the number of equilibria associated with an exogenous population is a priori unknown, which creates numerical problems. However, even with the possibility of aggregate hypercongestion, there is at most one equilibrium consistent with an exogenous user cost or utility level. For this reason, we generate the user cost curve by starting out with the maximum level of utility and then incrementally lowering utility for each utility level, determining the population associated with the corresponding unique equilibrium.

We proceed as stated above, by incrementally lowering $u$ from $u_{\text{max}}$, and for each $u$ solving for the $t$ satisfying (4.2.19) using the procedure described in subsection 4.2.2. At each equilibrium utility level, $u$, the population is computed by using (4.1.5). The outer loop repeats as long as the number of cycles remains constant. The number of cycles is computed at the beginning of the loops, as discussed in subsection 4.2.1. Once the number of cycles changes, the matrices in subsection 4.2.2 either increase or decrease by one dimension in the number of equations and the number of variables, i.e. a row and column is either
added or deleted. The matrix size increases (decreases) when the number of cycles increases (decreases). Details of the \( G \)-functions and \( H \)-function are outlined in the following section.

### 4.2.5 Details of the \( G \)-functions and \( H \)-function

The procedure developed in the inner loop to compute the integral term in (4.2.19) requires knowing the number of entry cycles over the departure set. At the end of section 4.2.1, it was shown how to solve for the number of entry cycles for a particular \((t, u)\). But to perform this exercise for each guess of the value of \( t \) that solves (4.2.19) for a particular \( u \), and to repeat this for each value of \( u \) investigated is inefficient. This inefficiency led to the development of the \( G \)-functions and \( H \)-function.

Recall the shorthand notation from the previous subsections:

\[
F_I(t) \equiv \prod_{i=1}^{I} B_i(t, u) \quad \text{and} \quad B_i \equiv 1 + \hat{T}(t_i(t; u)).
\]

We can rewrite the \( G \)-functions (4.2.18) in terms of the shorthand notation, specifically

\[
G_I(t, u) \equiv F_{I+1}(t, u) - 1 = \left[ \prod_{i=1}^{I+1} B_i(t, u) \right] - 1 = \left[ \prod_{i=1}^{I+1} \left( 1 + \hat{T}(t_i(t; u)) \right) \right] - 1,
\]

so that \( G_I(t, u) = 0 \) is the set of \((t, u)\) for which there are \( I \) full entry cycles in equilibrium, as stated before.

Two general properties of the \( G \)-functions bear note.

1. Recall that the \( G \)-functions are defined without reference to trip length. Suppose that \( U = \max_t U(t, 0) \equiv \nu \). This can only be achieved with zero trip length. But with zero trip length, everyone can travel at the utility-maximizing time however many entry cycles there are. Thus, the \( G \)-functions share the point \((\nu, \arg \max_t U(t, 0))\) in common.
2. As \( u \) decreases from \( \nu \), in equilibrium the number of full entry cycles increases to a maximum number of cycles and then decreases. Despite the increasing then decreasing nature of the number of full entry cycles, as \( u \) decreases, the earlier is the time of the first departure.

Figure 4.2.3 displays this construction for the central numerical example. Suppose, for example that we have solved for the \( H \)-function from \( u = u_{\text{max}} \) to \( u = 10 \), and now wish to solve for \( t \) satisfying (4.2.19) with \( u = 9 \). From inspection of the Figure, the solution entails three cycles, two full cycles, followed by a partial cycle. We could then determine our initial guess of the equilibrium value of \( t \) for \( u = 10 \). Applying (4.2.19), taking into account that the solution entails two full cycles followed by a partial cycle, it would be found that the trip distance so computed exceeds the exogenous trip length, which would indicate that the equilibrium value of \( t \) for \( u = 9 \) is lower than that for \( u = 10 \).

In the central numerical example, both the \( H \)-function and the \( G \)-functions are positively sloped over the entire economically relevant region of \( t-u \) space. These properties are specific to the example. Appealing to Figure 4.2.3, we can plot the equilibrium \( t \) for each given exogenous \( u \) that satisfies the equilibrium conditions, and more specifically (4.2.19). The generation of the \( H \)-function in the \((t,u)\)-space allows the algorithm to know exactly when to switch to a different size matrix equation (4.2.29), based upon the number of cycles, ie. its location with respect to the \( G \)-functions.

With a way to determine the equilibrium, we can now move to the question of whether the \( u \)-trip-timing equilibrium is unique. To provide the answer, we establish some properties of the \( H \)-function. Since the \( H \)-function provides all of the \((t,u)\) pairs such that an equilibrium exists, if we can show that for each \( u \) there is a single \( t \) then the equilibrium
Figure 4.2.3: Utility versus time plot

The first four $G_i(t, u)$ functions, and $H(t, u, L)$ for the case for $L = 4$. 

The diagram shows the utility $u$ versus time $t$ for different values of $L$. The functions $G_i(t, u)$ and $H(t, u, L)$ are plotted, with break points indicated for each function.
is unique. In terms of Figure 4.2.3, this corresponds to proving that the $H$-function cannot “bend upwards”; in other words, a horizontal line denoting a fixed utility cannot be crossed more than once by the $H$-functions. Similar to the theorems 7 and 8 for the $G$-functions, we have analogous results for the signs of the partial derivatives of the $H$-function. To that end, we must first prove that the $H$-function (4.2.19) is continuously differentiable.

**Theorem 15** The $H$-function defined in (4.2.19) is a continuously differentiable function.

**Proof.** See Appendix A, Result 27. ■

Before establishing signs of the partial derivatives of the $H$-function, it will be useful to derive a new expression from $H(t,u,L) = 0$. Since we have established the continuity and the piece-wise differentiability of $v(t,u)$ in both variables, we can totally differentiate $H(t,u,L) = 0$ with respect to $t$ and $u$:

$$
\left\{ \begin{array}{l}
v(k(t + T(t,u); t,u)) + \int_{L}^{L+T(t,u)} \left[ \frac{\partial}{\partial t} v(k(t; t,u)) \right] dt \\
+ \left\{ v(k(t + T(t,u); t,u))T_u(t,u) + \int_{L}^{L+T(t,u)} \left[ \frac{\partial}{\partial u} v(k(t; t,u)) \right] dt \right\} du 
\end{array} \right\} = 0.
$$

(4.2.35)

Now, from (4.2.9) the sum of first two terms in the curly brackets before $dt$ equals zero. We also take, $v'(k) = -1$ by normalized Greenshields’ Relation. Thus, (4.2.35) reduces to

$$
\left\{ \begin{array}{l}
- \int_{L}^{L+T(t,u)} \left[ \frac{\partial k}{\partial t} (t; t,u) \right] dt \\
\end{array} \right\} dt
$$

$$
+ \left\{ v(k(t + T(t,u); t,u))T_u(t,u) - \int_{L}^{L+T(t,u)} \left[ \frac{\partial k}{\partial u} (t; t,u) \right] dt \right\} du = 0.
$$

(4.2.36)
Using the reduced form of the total derivative of the distance condition (4.2.36), we are now ready to prove the following theorems relating to the signs of the partial derivatives of the $H$-function.

**Theorem 16** The $H$-function defined in (4.2.19) is monotonic in the $t$ variable for a fixed value of $u$. In other words, the $H$-function is monotonic in $t$: \( \frac{\partial H}{\partial t}(t, u) > 0 \).

**Proof.** See Appendix A, Result 28. ■

**Conjecture 17** The sign of \( \frac{\partial H}{\partial u}(t, u) \) is negative.

**Proof.** See Appendix A, Conjecture 29. ■

The results given in Theorem 16 and Conjecture 17 imply the ansatz provided at the beginning of the section. Precisely, the $H$-function is monotonic in the $t$ variable which implies that in $(t, u)$-space the $H$-function may curve left and right, but never bend upward. In other words, we can never have the situation where there exists two $t$ for a single exogenous $u$.

Since the solution algorithm solves for the entry rate function conditional on $u$ from the time of the first departure to the time of the last departure, it is straightforward to calculate and record the equilibrium commuter population as a function of $u$.

Figure 4.2.4 gives a flowchart that summarizes the full algorithm employed, including both the inner and the outer loop. The computer program that generates the examples is given at the hyperlink: http://math.ucr.edu/~buli/Bathtub_Code/Bathtub_Code.html. The program follows exactly the algorithm as described in this section. The program can be readily adapted to treat other utility functions that satisfy the properties assumed.
in section 4.1, as long as they yield closed-form formulas of the \( \dot{T} \) and \( \ddot{T} \) function. The program is written to be easy to understand and to avoid numerical difficulties, rather than to be efficient in terms of run time.

4.3 Existence and uniqueness of equilibrium

The uniqueness and existence of the \( u \)-trip-timing equilibrium was constructed numerically in [4]. In subsection 4.3.1, we establish the remaining analytical properties of the bathtub model to prove the existence and uniqueness of a \( u \)-trip-timing equilibrium. Existence follows from the construction of the equilibrium. Subsection 4.3.2 provides the results for the \( N \)-trip-timing equilibrium and the results for the full trip-timing equilibrium are provided in 4.3.3.

4.3.1 \( u \)-trip-timing equilibrium

At this point, we have proven the results laid out in the introduction of this subsection. We have proven positivity of the velocity, density, and entry rate functions for times in the departure set, as well as established properties of the \( H \)-function which is constructed via the algorithm in [4]. Using these results, along with the assumptions (A-1)-(A-4) from section 4.1, we establish the uniqueness of the \( u \)-trip-timing equilibrium.

**Theorem 18** For each level of utility \( u \) less than the maximum utility level \( u_{\text{max}} \), there exists a single \( t \) satisfying the \( H \)-function.

**Proof.** The existence of a \( t \) satisfying the \( H \)-function in (4.2.19) is guaranteed by construction with the algorithm developed in [4]. The existence follows from the unique solution of
Figure 4.2.4: Flowchart outlining the algorithm for a fixed number of cycles.
the matrix system (4.2.29) at each time step for a fixed $t$. The algorithm alters the $t$ in an outer loop to recover one of the “constants of integration” that was lost in deriving the acceleration condition (4.2.20), until the equilibrium conditions are satisfied.

Now we must show that the value of $t$ that is determined for a particular value of equilibrium utility $u$, is unique. The result follows from Theorems 7-9, 16, and 17. We have established that the $G$-functions do not intersect each other and that $G_{I+1}$ always lies to the left of $G_I$ in the $(t-u)$-plane (Theorem 9), as well as the sign of the partial derivatives of the $G$-functions (Theorems 7 and 8). This implies that the $G$ functions are monotonic in both variables and can have horizontal or vertical tangents in the $(t,u)$-plane, but cannot bend back on themselves in either direction. The results with the $H$-function, Theorem 16 and Theorem 17, show that the $H$-function is monotonic in the $t$ variable only, so it may bend backwards in the $(t-u)$-plane but cannot bend “upwards.” Recall that all of the proofs rely on the underlying assumptions (A-1)-(A-4) on the utility function, specifically the properties of the trip duration function $T(t)$. We have that the equilibrium $(t,u)$ are the tuples that satisfy the $H$-function. From the restrictions of the partial derivatives on the $H$ and $G$ functions, for a single $u$ the $H$-function can only cross the $G$-functions at a single $u$. This implies that for a fixed value of $u$ there can be only one value of $t$. If we suppose that there are two values of $t$ for one value of $u$, monotonicity would be violated, hence a contradiction. As only one $t$ exists for each $u$, each pair $(t,u)$ is unique, therefore the equilibrium is unique. ■
4.3.2 \textit{N-trip-timing equilibrium}

We can reformulate the \textit{u}-trip-timing equilibrium problem in section 4.3.1 to an \textit{N}-trip-timing equilibrium problem by taking the utility \textit{u} as exogenous, and the population \textit{N} as the new endogenous variable. The equilibrium problem in terms of \textit{t} and \textit{N} has a slightly different form of (4.1.5), given by

\[
\int_{t(N)}^{t^*(N)} e(t; \textit{u}) \, dt = N(\textit{u}) \tag{4.3.37}
\]

where \textit{t}(\textit{N}), \textit{t}^*(\textit{N}), and \textit{e}(\textit{t}; \textit{u}) are the same as in the \textit{u}-trip-timing equilibrium except now the functions are dependent upon the endogenous \textit{N} value. There is also functional relationship between \textit{N} and \textit{u}. From the algorithm developed in [4], in the outermost loop the utility \textit{u} was lowered incrementally from \textit{u}_{max}. For each value of \textit{u}, we determined the equilibrium time of the first departure, from which we could determine the equilibrium population \textit{N} via (4.1.5).

For the \textit{N}-trip-timing equilibrium, one could reformulate the algorithm to take \textit{N} in the outermost loop. The population commuting must be positive, and given that the traffic system can accommodate only a finite number of vehicles at any given time, \textit{k}_c, there must exist some maximum population, \textit{N}_{max}, that can commute over the rush hour. In theory, the algorithm could be modified to start with only a small population (\textit{N} near 0) and incrementally raise population until the upper bound of population is reached. In the development of the algorithm in [4], this type of algorithm was considered as it is more natural to work with population over utility, but the equilibrium conditions outlined in Section 4.1 become much more complicated in terms of \textit{N}. Both formulations are equivalent,
but from our numerical experience, using an outer loop with $u$ was more reasonable to solve computationally and derive analytical results. One reason for this, as will be discussed shortly is the uniqueness of the equilibrium. In the $u$-trip-timing equilibrium, for each $u$, there was at most one $t$. From the algorithm, we can compute the commuting population $N$. The results show that for each value of utility $u$ there is a unique population $N$, but the converse is not true. As stated before, the traffic system has a finite maximum capacity which implies a maximum value of $N$. In an algorithm similar to [4] where the role of $u$ was replaced by $N$, a value of $N$ that is greater than $N_{\text{max}}$ would never produce an equilibrium.

A second problem arises in an algorithm of this type; for one value of $N$, there can exist two values of $u$ consistent with equilibrium.

Since we can determine the value of $N$ contingent upon a choice of utility $u$ and the problem formulation in terms of $N$ and $u$ are equivalent as described above, the relationship between $u$ and $N$ can be constructed. In [4], for the numerical examples we considered\textsuperscript{11}, we showed that in $(u,N)$-space, the curve relating the two variables is backward bending. For each value of equilibrium utility, $u \in (-\infty,u_{\text{max}}]$, there exists exactly one value of equilibrium population $N$, but for a fixed value of population $N$ there can exist two $u$ values, a single $u$ (corresponding to $N = 0$ or $N = N_{\text{max}}$), or no equilibrium value of $u$.

In Figure 4.3.5 below, the user cost curve generated from the logarithmic utility function is given, which shows the backward-bending relationship. From this derivation, it is clear that if $N = N_{\text{max}}$, then there will exist exactly one equilibrium, hence a unique $N$-trip-timing equilibrium. The value of $N_{\text{max}}$ acts as a sort of bifurcation point, as the number

\textsuperscript{11}The results mentioned here are specific to the functional forms for the numerical examples presented in [4]. We have not ruled out the possibility that the user cost as a function of $N$ is monotonically increasing in $N$ such that user cost approaches infinity as $N$ approaches $N_{\text{max}}$. 
equilibria on either side of this value is different. If for example the value of $N > N_{\max}$, the population that wishes to commute is greater than the maximum allowable, then no equilibrium exists. This is an example of non-existence of equilibrium. If on the other hand, $N < N_{\max}$, then there are two equilibrium values of $u$ which will give the same commuting population, a congested equilibrium and a hypercongested equilibrium. In this case, the vertical line denoting the population would intersect the user-cost curve twice. Figure 4.3.5 summarizes the possible cases.

4.3.3 Full trip-timing equilibrium

A full trip-timing equilibrium can be defined in a similar way as the previous subsection on the $N$-trip-timing equilibrium, except we use a demand function instead of an exogenous population of commuters. The full trip-timing equilibrium corresponds to a point of intersection of the inverse demand function and the user cost function. In terms of the algorithm, the full trip-timing equilibrium would still have the user cost curve constructed in Figure 4.3.6, in the previous subsection, except we have a demand function replacing the vertical line representing a fixed population.

The shape of the demand function will determine the number of equilibria for the problem, as the user cost curve has the same backward-bending shape. If the demand function is linear, for example, then there is a possibility of having one equilibrium or three equilibria. If the demand curve has the correct slope, then the demand curve could intersect the user cost curve once, either in the congested or hypercongested region but not both. In the second case, if the slope of the demand curve is steep we could intersect the user cost
curve twice in the hypercongested region and once in the congested region. The shape of
the user cost curve does not allow for two intersection points, so this case is not possible.
If the demand curve lies everywhere below the user cost curve, a unique equilibrium exists
with no travel, therefore, existence is guaranteed in the full trip-timing equilibrium.

4.4 An Extended Numerical Example

This section demonstrates the implementation of the algorithm in a test case. We generated
two extended numerical examples, one for a logarithmic utility function, and another for
an exponential utility function. In most of the section, we focus on the extended numerical
example with logarithmic utility since it was computationally more straightforward and
its results easier to present and explain, but in subsection 4.4.2 we shall briefly contrast
the results from the logarithmic utility function with those obtained from the exponential
utility function.

Subsection 4.4.1 presents the logarithmic utility function and derives and explains
its properties, and at the end gives the parameters used in the example. The only pa-
parameter that varies between runs is the exogenous population. Subsection 4.4.2 reports
on the results with the logarithmic utility function for three populations. Subsection 4.4.3
briefly contrasts the results obtained with the exponential utility function to those obtained
with the logarithmic utility function. Finally, subsection 4.4.4 comments on computational
aspects of the extended numerical example.
4.4.1 The Logarithmic Utility Function

The logarithmic utility function is

\[ U(t, T(t)) = r_0 \log(r_1 t) + s_0 \log(s_1 (M - (t + T(t)))) \] (4.4.38)

It is the sum of two sub-utility functions. The sub-utility function on the left gives the “at home” utility function. A commuter derives infinitely negative utility if he leaves home at \( t = 0 \) and thereafter derives diminishing marginal utility from staying at home longer. In particular, his utility from being at home is the log of the time he leaves home. The sub-utility function on the right gives the “at work” utility function. A commuter derives infinitely negative utility if he arrives at work at \( t = M \), and derives diminishing marginal utility from arriving at work earlier. In particular, his utility from being at work is the log of the time he arrives at work earlier than \( M \). The simplicity of the utility function makes its properties easy to derive, and its intuitiveness makes its properties easy to understand.

The most obvious implication of the utility function is that the rush hour cannot start before \( t = 0 \) or end after \( t = M \). Since the congestion technology is such that there is a maximum or capacity flow, this puts an upper bound on the population the street system can accommodate over the rush hour. Recall that the way the algorithm proceeds is to successively lower the utility level, for each utility level computing the corresponding equilibrium, including the equilibrium population. The process generates a user cost curve. Since there is a strict lower bound on departure time and a strict upper bound on arrival time, the only way a very low equilibrium utility can be achieved is for the median commuter to have a very high trip duration, requiring severe hypercongestion, and hence flow
considerably below capacity flow, for most of his trip. Thus, the logarithmic form of the utility function results in a user cost curve with a backward-sloping portion corresponding to aggregate hypercongestion.

**Functions**

\[
U(t, T(t)) = r_0 \log(r_1 t) + s_0 \log(s_1 (M - (t + T(t))))
\]

\[
T(t) = M - t - \frac{1}{s_1} \exp \left( \frac{1}{s_0} (u - r_0 \log(r_1 t)) \right) = M - t - \frac{1}{s_1} \exp \left( \frac{u}{s_0} \right) (r_1)^{-\frac{r_0}{s_0}} t^{-\frac{r_0}{s_0}}
\]

\[
\dot{T}(t) = -1 + \frac{r_0}{s_0 s_1} \exp \left( \frac{u}{s_0} \right) (r_1)^{-\frac{r_0}{s_0}} t^{-\frac{r_0}{s_0}} - 1
\]

\[
\ddot{T}(t) = \frac{r_0}{s_0 s_1} \left( - \frac{r_0}{s_0} - 1 \right) \exp \left( \frac{u}{s_0} \right) (r_1)^{-\frac{r_0}{s_0}} t^{-\frac{r_0}{s_0}} - 2
\]

Table 4.4.2: Logarithmic utility and travel duration functions

Table 4.4.2 displays the logarithmic utility and travel duration functions. It has all the properties that we have assumed for the general utility function employed in the analysis of the previous section; in particular \(1 + \dot{T}(t; u) > 0\) and \(\ddot{T}(t; u) < 0\) for all \(t \in (0, k)\). We also note that, with the logarithmic utility function, \(\dddot{T}(t; u) > 0\) (which guarantees that the \(G_i(\cdot)\) functions are all positively sloped for all \(i\)), \(\dddot{T}(t; u) < 0\), and \(\ddot{T}(t; u) > 0\). Given the simple form of the utility function, it is not surprising that its properties are so simple.

There is no empirical work on the basis of which to choose the parameters of the utility function. We obtained the parameters instead through experimentation, with the aim of getting reasonable results. They were

\[
r_0 = 15 \quad r_1 = \frac{1}{2} \quad s_0 = 18 \quad s_1 = 1 \quad M = 6.
\]

The time units are hours; thus, one may view 5 am as the earliest feasible departure, and 11
am as the latest feasible arrival. In order to simplify the algebra, in the previous subsection results were derived using normalized units. But in presenting the results, we work with unnormalized units:

\[ v_f = 15 \text{ mph} \quad k_j = 10^6 \text{ vehicles} \quad L = 4 \text{ miles} \]

For interpretation of the results, it will be useful to work out some of the implications of these parameters. As a simple point of reference, consider the situation (which is not consistent with equilibrium) where the entry rate is such that density is at capacity density over the rush hour. With Greenshield’s Relation, capacity density is \(0.50 \times 10^6\), capacity flow is \(3.75 \times 10^6\), and capacity velocity is \(7.5 \text{ mph}\). In steady state, the inflow = outflow of the street system equals flow divided by trip length or \(0.94 \times 10^6\). Since the duration of a trip at the velocity corresponding to capacity flow is 0.53, the maximum feasible duration of the departure interval is 5.47, and the corresponding maximum feasible number of commuters over the rush hour is \(5.13 \times 10^6\). The maximum equilibrium number of commuters is significantly smaller than this, since in order to achieve equilibrium, velocity must be higher than capacity velocity (and flow correspondingly lower) in the shoulders of the rush hour and lower than capacity velocity (and flow correspondingly lower) during the peak.

Recall that, where \(\nu = \max_t U(t, 0)\) and \(u_{\text{max}} = \max_t U(t, L/v_f)\), \(\nu - u_{\text{max}}\) is trip cost at free-flow travel speed. With the assumed utility function, it is simple to calculate\(^{12}\) that \(\nu = 25.99\) and \(u_{\text{max}} = 24.49\), implying that trip cost at free-flow speed is 1.50 monetary

\(^{12}\)The utility-maximizing departure time with \(T = 0\) solves \(r_0/(t - s_0) = 0\), yielding \(t = r_0 M/(r_0 + s_0)\), so that \(\nu = r_0 \log(r_1 r_0 M/(r_0 + s_0)) + s_0 \log(M(1 - s_1 r_0/(r_0 + s_0))) = 25.99\). The utility-maximizing departure time with \(T = 0.2667\) solves \(r_0/(t - s_0)(M - 0.2667 - t) = 0\), yielding \(t = r_0 (M - 0.2667)/(r_0 + s_0)\), so that \(u_{\text{max}} = r_0 \log(r_1 r_0 (M - 0.2667)/(r_0 + s_0)) + s_0 \log((M - 0.2667)(1 - s_1 15/11^{15})) = 3.97027 + 20.5226 = 24.493.\)
units. Since free-flow travel time equals $4/15$ hrs, and since the cost of travel time is around $20/\text{hr}$, the trip cost in dollars is $5.33$, so that the money unit may be taken to be around $3.55$.

4.4.2 Numerical Results with the Logarithmic Utility Function

We present numerical results for three levels of utility, high, medium, and low. The example with high utility has only one partial entry cycle and exhibits only very modest congestion. The example with medium utility has three full entry cycles and one partial entry cycle, and, though corresponding to the congested portion of the aggregate user cost curve, still exhibits quite severe hypercongestion at the peak of the rush hour, with a minimum speed somewhat under 4 mph. The example with low utility corresponds to the hypercongested portion of the aggregate user cost curve. It exhibits very severe hypercongestion over almost the entire rush hour, with a minimum speed of about 1 mph.

- **high utility** ($u = 24.43, N = 4.12 \times 10^4$)

Figure 4.4.7 displays five panels. The top panel displays $T(t), \dot{T}(t)$, and $\ddot{T}(t)$ against clock time. The panel in the middle on the left plots velocity against time, and that on the right density over time; since under Greenshields’ Relation velocity is a negative linear function of density, the two graphs are closely related. The panel at the bottom left gives the distance traveled within successive entry cycles as a function of time; in this example, there is just one partial cycle\(^{13}\). The panel at the bottom right gives the entry and exit rates over time.

---

\(^{13}\)The derivative of the distance function for each entry cycle gives the velocity function over that entry cycle. Plotting cumulative distance against time would give the time-space diagram for vehicles over the rush hour.
Clock time is normalized so that $t = 0$ corresponds to 5 am. Breakpoints are indicated by circles. The vertical axes are scaled to make the results easy to read.

The five panels together tell a simple story of equilibrium rush-hour dynamics in a situation where there is generous capacity relative to the population of commuters, which is $N = 4.11892 \times 10^4$. The rush hour is short, lasting only about 0.4 hours or 24 minutes, and travel exhibits little congestion, with a minimum speed of around 14.4 mph, the free-flow speed being 15 mph.

There is only one partial entry cycle, which starts at $t = 2.542$ and ends at $t^* = 2.662$. Since trip length is 4 miles and speed is around 14.4 mph, trip duration averages around 0.274 hrs, first rising for early commuters to offset the increasingly desirable departure time and then falling for late commuters to offset the increasingly undesirable departure time. Since the trip duration of the first commuter to depart exceeds the period over which commuters depart, there is a period in the middle of the rush hour when all commuters are on the roads, after the last entry but before the first exit, which explains the flat portions of the velocity and density curves in the middle of the rush hour. The entry rate, calculated from (4.2.25), adjusts so as to ensure that utility is constant over the departure interval; it declines slowly over the departure interval. In contrast, the exit rate increases slowly over the exit interval, as traffic speed increases.

* medium utility ($u = 20.000, N = 2.246 \times 10^6$)

Figure 4.4.8 displays the equilibrium traffic dynamics with the considerably higher population of commuters of $N = 2.246 \times 10^6$, using the same five panels. The vertical axes of some
of the panels are scaled differently from the corresponding panels in Figure 4.4.7 to improve readability. The equilibrium rush-hour traffic dynamics are markedly different from those of the previous example. The rush hour lasts 2.859 hours, extending from \( t = 1.353 \) to \( t = 4.212 \), and there are three full entry cycles and one partial entry cycle. The separate cycles are displayed in different colors. Traffic is hypercongested at the peak.

The most striking feature of this case is the discontinuities in the entry and exit rates, particularly since the utility and congestion functions are smooth. The source of these discontinuities was explained in the previous section. Apart from the discontinuities in the entry and exit rates at the breakpoints, the rush-hour traffic dynamics accord with intuition. Velocity falls in the early morning rush hour and increases in the late morning rush hour. As the population of commuters increases, one expects both the rush hour to lengthen and average velocity to fall, and that is what is observed. Also, since average velocity is lower, the duration of the entry cycles is longer.

- low utility \((u = -25, \bar{N} = 1.656 \times 10^6)\)

This case is displayed in 4.4.9. Apart from the extremely low velocity at the peak of the rush hour, indicating severe hypercongestion, a cursory look at the rush-hour traffic dynamics suggests that they are well behaved. But on closer examination, the equilibrium in this case is anomalous.

The rush hour is longer than in the previous example, despite the considerably lower population, and there is one full and one partial entry cycle, whereas in the previous example there were three full and one partial entry cycle. At the start of the rush hour, the entry rate is very high, causing hypercongestion to set in almost immediately. Speed is so low
that the full entry cycle is over two hours long, so that even the first commuter to travel has an average speed of less than 2 mph. Because traffic is so slow at the end of the first entry cycle, the exit rate at the start of the second entry cycle is low, requiring only a lower entry rate to satisfy the equilibrium trip-timing condition. The second, partial entry cycle has an even longer duration. Together these results indicate severe hypercongestion at the aggregate level.

Figure 4.4.10a plots the aggregate user cost curves, \( c(N) \), for the logarithmic utility function. The points plotted in the figure are not the primary and secondary breakpoints described earlier, but instead indicate levels of population at which there is a switch in the number of entry cycles, which we define as switch-points. Letting \((m, n)\) denote a switch from \(m\) to \(n\) full cycles with movement up the user cost curve, the indicated breakpoints are \((0,1)\), \((1,2)\), \((2,3)\), \((3,2)\), and \((2,1)\). The breakpoint at the top of the Figure, beyond the plotted portion of the user cost curve is \((1,0)\).

One might think that aggregate hypercongestion is an artifact of the logarithmic utility function, which effectively restricts the length of the rush hour, but Figure 5.4b shows that aggregate hypercongestion also arises with the exponential utility function that Fosgerau has employed, which places no effective restrictions on the length of the rush hour.

Now add an exogenous population level to the user cost curve. By definition, any intersection of the exogenous population line and the user cost curve, of which there are two, is an equilibrium. The obvious question is which of the two equilibria is stable. The stability of an equilibrium is defined with reference to the adjustment process, but whatever the adjustment process, equilibria alternate between stable and unstable. Here the
adjustment process is the day-to-day departure time decision of the individual commuter. With any such process that is reasonable, the congested equilibrium is stable, which implies that the hypercongested equilibrium is unstable. However, stable, hypercongesed equilibria can occur if the exogenous population constraint is replaced by a downward sloping demand function, as in Arnott and Inci (2010) but there in the context of equilibriums steady-state traffic flow. It remains to be seen whether rush-hour traffic equilibrium in any of the world’s most congested cites is hypercongested at the aggregate level and stable.

4.4.3 Numerical Results with the Exponential Utility Function

The exponential utility function is

\[ U(t, T(t)) = \left( \frac{A_0}{a_1} \right) (1 - e^{-a_1 t}) + \left( \frac{B_0}{b_1} \right) \left( 1 - e^{-b_1 (M - t - T(t))} \right). \] (4.4.39)

The properties of this utility function are recorded in the companion paper, Buli and Arnott (2017). Suffice it to say here that the exponential utility function satisfies all the conditions imposed on the utility function in the general analysis of the previous section, and has the same qualitative properties as those listed in Table 4.4.2 for the logarithmic utility function. The main difference between the exponential and logarithmic utility functions is that the logarithmic utility function effectively constrains the length of the rush hour, whereas the exponential utility function does not.

In the numerical example for the exponential utility function, the parameters of the utility function were\(^{14}\) \(A_0 = 5, a_1 = 1/3, B_0 = 10,\) and \(b_1 = 1/2.\) They were chosen\(^{15}\) Hjorth et al. (2015) estimate the parameters. Even though they impose the parameter restriction that \(a_1 = b_1,\) they find that the parameter estimates have large standard errors and are highly correlated.
to give sensible numerical results. The other parameters are the same as those for the logarithmic utility function example.

Figure 4.4.10a displays the user cost curve with the logarithmic utility function, and Figure 4.4.10b the user cost curve with the exponential utility function. The most interesting feature of the exponential utility function example is that, even though it does not impose a constraint on the length of the rush hour, aggregate hypercongestion nonetheless occurs.

4.4.4 Computational Experience

Many methods were attempted to numerically solve the delay differential equation in (4.2.20) for the velocity function \( v(\cdot) \) directly, that satisfied the boundary conditions, the equilibrium timing condition, and the trip length condition. At first, we experimented with polynomial interpolation methods, which were applied between breakpoints\(^{15}\). In one of the polynomial methods, we used successive derivatives on (4.1.4), to get a system of equations, whose number of equations was equal to the number of interpolation points. In a second method, we attempted to partition the time domain between breakpoints by using Chebyshev nodes for polynomial interpolation. Unfortunately, these methods did not provide convergence when the polynomial degree was increased, and the velocity function was discontinuous at the breakpoints.

A global polynomial method was attempted, to create one large polynomial that would interpolate over the whole rush hour domain. We utilized the derivatives of (4.1.4),

\(^{15}\)The values of the breakpoints are exactly known from the exogenous nature of the \( T(t; u) \) function for a fixed equilibrium value of utility, so the value of the velocity function is known exactly.
which can be calculated, assuming enough regularity on \( v(\cdot) \). This method was also unsuccessful, as there was no apparent convergence, and depending on the number of Chebyshev nodes, large oscillations developed. Only later did we realize that the structure of the delay differential equation implies discontinuities in the entry rate function, hence only a \( C^1 \) velocity function. Polynomial approximation between the breakpoints is also unsatisfactory, since there are discontinuities in the entry rate function at the secondary breakpoints whose location is \textit{a priori} unknown.

We next looked at a simple time-stepping method. The algorithm we developed solved the problem going forward, with \( u \) fixed and with an initial guess of \( t \), and the approximated velocity function was computed on the first cycle to create the velocity function over subsequent cycles. At that point, the trip length \( L \) is computed. From here, we can determine how to change \( t \) based on if the computed value of \( L \) is greater than or less than \( L \). Then, the utility function \( U(t, T(t)) \) itself is used to compute the new level of utility. If the \( u \) is not within tolerance, the process is repeated. Once the tolerance is satisfied, the \( t \) can be used as the starting point, to run the problem backwards. Then the whole process is repeated in the backwards direction, and repeated forwards and backwards until convergence is achieved. The method employed above also failed to converge.

With the failure of these methods, we gradually developed the solution procedure described in section 4.2 that takes full account of the mathematical structure of the problem. Our motive for recounting our earlier failures is to warn that standard numerical methods used to solve smooth ordinary or partial differential equations may not work well when applied to the bathtub model or to other models whose dynamics are described by
delay differential equations. The main reason seems to be that standard methods assume
smoothness of the solution, but DDE’s generate discontinuities in the solution even when
the underlying functions are smooth. Another motive for recounting our earlier failures is
positive. Even though the bathtub model is analytically intractable, it can be solved nu-
merically without encountering computational difficulties by using customized algorithms
that account for its particular mathematical structure.

For all of the computational results, the algorithm was run on a DELL® XPS 13
laptop with an Intel® Core™ i5-5200U CPU and 8 GB memory using MATLAB®. All
results were computed in double precision, with loop tolerances set to at least $10^{-4}$ for the
loop where $L$ is computed as given in the flowchart, Figure 4.2.4. Depending on the number
of entry cycles, the increments of utility are taken to be in the range of $\Delta u = 0.00001$ to
$\Delta u = 0.01$. Finer scales were used in the congested region of the user-cost curve, and
the coarser scales were used in the hypercongested region. For the time increments, we
only divide the first cycle between the first two breakpoints, as the points in cycle 1 are
mapped to subsequent cycles via the $t_i(t; u)$ mechanism. In the numerical work, the number
of time increments range from 10,000 to 400,000, where more points are required in the
hypercongested region. Special care is taken to choose the number of time increments, and
how much the value of $t$ is changed in the inner loop as $N$ gets closer to $N_{\text{max}}$. When
$H(t, u, L)$ becomes steep, the number of entry cycles begins to decrease. The steep slope
of $H(t, u, L)$ affects the algorithm, as small decreases in $t$ can cause large changes in $u$. We
thus take $\Delta u$ to be coarser and time increments to be smaller, to keep the same tolerance.

\footnote{Since the distance between breakpoints is constantly changing, one can hold the number of time incre-
ment points to be held fixed, while the $\Delta t$ changes; or one can hold $\Delta t$ fixed and let the number of time
increments change. Numerical experimentation has shown both methods to yield similar results.}
Three possible situations for the $N$-trip-timing equilibrium for different exogenous populations $\overline{N}$. (a) Case I: the exogenous population is less than the maximum possible commuting population $\overline{N} < N_{\text{max}}$. In this case there exists two equilibria; one congested and one hypercongested, (b) Case II: the exogenous population is equal to the maximum possible commuting population $\overline{N} = N_{\text{max}}$. In this case there exists a single equilibrium, (c) Case III: the exogenous population is greater than the maximum possible commuting population $\overline{N} > N_{\text{max}}$. In this case there exists no equilibrium.
Three possible situations for the full-trip-timing equilibrium for different demand functions.

(a) Case I: the demand curve may have a slope such that it only intersects the lower part of the user cost curve, (b) the demand curve may have a slope such that it only intersects the upper part of the user cost curve, (c) Case III: the demand curve may have a slope such that it intersects the upper part of the user cost curve twice and the lower part once.
The outputs for equilibrium are $u = 24.43473, \bar{t} = 2.54243, N = 4.11892 \times 10^4, t^* = 2.66181, \text{ and } \bar{t} = 2.93739$. Figure 4.4.7a plots the functions $T(t), \dot{T}(t), \ddot{T}(t)$. Figure 4.4.7b plots the velocity function, $v(t)$. Figure 4.4.7c displays the density function, $k(t)$. Figure 4.4.7d plots the trip length of the individual who leaves at $t = \bar{t}$. Figure 4.4.7e displays the entry and exit rates, $e(t)$ and $x(t)$, respectively.
The outputs for equilibrium are $u = 20, t = 1.35298, N = 2.24556 \times 10^6, t^* = 3.77835,$ and \( \bar{t} = 4.21218 \). Figure 4.4.8a plots the functions $T(t), \dot{T}(t),$ and $\ddot{T}(t)$. Figures 4.4.8b and 4.4.8c plot the velocity function and density function, respectively. Figure 4.4.8d plots the time-space line of a commuter who departs at $t_i(t), i = 1, \ldots, 4$. Figure 4.4.8e plots the entry and exit rates, $e(t)$ and $x(t)$, respectively.
Figure 4.4.9: The case of one full and one partial entry cycle: hypercongestion

The outputs for equilibrium are $u = -25, \bar{t} = 0.078698, N = 1.65562 \times 10^6, t^* = 3.95950, \bar{t} = 5.85886$. Figure 4.4.9a plots the functions $T(t)$, $\dot{T}(t)$, and $\ddot{T}(t)$. Figures 4.4.9b and 4.4.9c plot the velocity function and density function, respectively. Figure 4.4.9d plots the time-space line of the first commuter to depart. Figure 4.4.9e displays the entry and exit rates, $e(t)$ and $x(t)$, respectively.
Figure 4.4.10: The user cost curve for the logarithmic utility function

Figure 4.4.10b displays the user cost curve for the exponential utility function.
Chapter 5

Conclusions

5.1 Summary

In chapter 2, we constructed LDG methods with both alternating flux and upwind flux for the coupled BBM system. This system has a Hamiltonian which is conserved for all time. We have shown that the LDG method with the alternating flux, coupled with the midpoint rule time discretization, provides a method which conserves the discrete Hamiltonian exactly, and the LDG method with the upwind flux dissipates the Hamiltonian. An optimal error estimate for the linearized system has been provided. Numerical examples are presented to illustrate the accuracy of the proposed methods in simulating the coupled BBM system.

In chapter 3, we constructed a general framework of DG methods to solve the AR model on networks, (3.1.1) and (3.1.2). Arbitrary numerical fluxes can be used, which does not require the exact solution to the complicated Riemann problem, and in this paper we consider the Lax-Friedrichs flux as the motivating example. This is an extension of the
first order method developed in [19] for the junction problem to high order DG methods. Numerical examples are provided to demonstrate the high-order accuracy, and comparison of results between the first-order LWR model and the second-order AR model. The ability of the model to capture the capacity drop phenomenon is also provided.

The primary aim of chapter 4 was to develop a method to solve numerically for equilibrium in the basic (identical individuals) bathtub model when the underlying utility and congestion functions are smooth. The difficulty in doing so results from the unfamiliar mathematics of the bathtub model, which requires the solution of a delay differential equation. Except for special cases, which were reviewed earlier, analytical solution is not possible, so that computation is necessary. When we started our work on the paper, we first went to literature on the computational solution of delay differential equations with an exogenous initial history. In contrast, in the bathtub model the initial history – either the entry function or the density function over the first cycle – has to be solved for as part of the overall equilibrium. Our broad conception was that we would add an outer loop that would iteratively converge to the equilibrium initial history. But we encountered numerical and conceptual difficulties. It was not until we developed a customized algorithm that builds on the mathematical properties of the bathtub model that we had success.

The method that we developed has three elements. The first is to invert the tripping equilibrium condition that \( U(t, T(t)) = u \) over the departure interval to give the function \( T(t; u) \), which gives trip duration as a function of departure time consistent with the exogenous utility level. The solution entails entry cycles. The first commuter departs at \( t \), which is the beginning of the first entry cycle, and arrives at \( t + T(t; u) \), which is the
end of the first entry cycle and the beginning of the second, etc. The second element is to
derive computationally the entry rate function consistent with the equilibrium trip duration
function for a particular $u$ and with a given time of first entry, $t$. The entry rate function
is solved by: i) twice differentiating the distance condition that, over the departure set,
the integral of velocity over the equilibrium trip duration equal the exogenous trip distance,
which we termed the acceleration condition; ii) substituting (4.2.9) and the FIFO conditions
into the acceleration condition to obtain a system of linear equations relating the entry rate
at a point in time to velocity at that point in time and the entry rate at the corresponding
points in time in the cycle ahead and the cycle behind; and iii) applying the boundary
conditions that traffic density is zero at the times of the first entry and of the last arrival.
The solution has the property that all trip distances are the same, but not generally equal
to the exogenous trip distance. The third element is to solve for the $t$ consistent with the
exogenous trip distance. The solution obtains a unique solution for each value of $u$.

While not difficult to understand or to apply, our solution method is complex. We
conjecture that it is nonetheless the simplest solution method that avoids computational
problems. On one hand, the complexity is discouraging. On the other hand, from the user’s
point of view, the computational method is a reliable black box.

We started with the maximum utility level (which is achieved when population
is zero) and gradually lowered the utility level, obtaining the equilibrium solution for each
utility level. Since there is a unique population of commuters for each utility level, and
since user cost can be straightforwardly related to utility, our procedure generates a user
cost curve, relating user cost to the commuter population.
5.2 Ongoing and Future Work

In the BBM system problem discussed in chapter 2, one outstanding analytical result that requires proof is the error estimate for the fully nonlinear coupled BBM system. We were able to prove the optimal order for the linearized BBM system, and we observe optimal order of accuracy with the nonlinear BBM system in numerical experiments. Traditional analysis for error estimates for this type of problem were not successful, so a different approach for the proof is needed for the fully nonlinear case. We also hope that in future work, the choices of flux and methods presented here could be used to develop DG methods for a wider class of problems associated with the full $abcd$-Boussinesq system which include additional high-order spatial derivative terms.

For the Aw-Rascle (AR) traffic network model considered in chapter 3, future work and extensions include incorporating stochastic components in a systematic way to the system. We would like to determine how stochasticity can capture more complicated behavior seen in real traffic data. Another important avenue for extension is to use our proposed model to model a large network of connected junctions to model a real freeway system, which implements parallelization to take advantage of the benefits of the DG method. We can also extend the model by adding source terms to the model to take into account commuter heterogeneity or controlled entry rates into the system.

Our solution method for the bathtub model in chapter 4 derives the user cost curve computationally. If the aim is to solve for equilibrium for a particular population level, the method is obviously inefficient. The difficulty in solving for equilibrium with an
exogenous population level is that the number of equilibria is unknown a priori. Depending on the form of the utility and congestion functions, there are three qualitative possibilities: i) There is no upper bound on the population of commuters that the street system can accommodate; ii) there is an upper bound, which equilibrium approaches asymptotically as user cost increases; iii) there is an upper bound that is achieved at a finite user cost, and higher user costs correspond to populations below this upper bound. The two examples that we presented illustrate possibility (iii). The paper [6] illustrates possibility (i) with the $\alpha - \beta - \gamma$ user cost function, and it is natural to conjecture that this asymptotic property applies with smooth approximations of that cost function. Through analysis of the asymptotic properties of the bathtub model as user cost approaches infinity, it should be possible to determine a priori which of the three possibilities arise.

The next item on the research agenda is to solve for the social optimum in the basic bathtub model. Formally, the optimum problem can be stated in a form that differs from the equilibrium only in that (4.1.1), the trip-timing equilibrium condition, is replaced by the condition that the marginal social cost of a trip is the same throughout the departure interval and weakly higher outside the departure interval. Via the Envelope Theorem, marginal social cost at a particular departure time can be calculated as the increase in total cost from inserting a commuter at that time, holding fixed the entry rate function. The difficulty lies in obtaining a manageable expression for the marginal social cost function. Since there will be entry cycles in the social optimum, discontinuities in the entry rate function should be expected, which will likely create difficulties for the application of generic numerical optimization procedures that assume smoothness.
Bibliography


Appendix A

Appendix - Proofs from chapter 4

Result 19 The departure set $D$ is connected.

Proof. First, divide the departure interval $D$ into cells of unit distance starting with the
time of the first departure. Assume that the departure set is not connected, such that the
hole occurs at marker $t_n$ during the first cycle where travel time is increasing. The travel
time at marker $t_n$, denoted $T_n$ is the same as that at location $t_{n-1}$ with the subtraction of
$\tau_{n-1}$ and the addition of $\tau_{n-1+L}$, where we define $\tau_n = v(k_n)^{-1}$. The travel time at $n$ is
also equal to the travel time at location $n+1$ with the addition of $\tau_n$ and subtraction of
$\tau_{n+L}$. We summarize those trip times as

$$T_n = T_{n-1} + \tau_{n-1+L} - \tau_{n-1} \quad \text{or} \quad T_{n-1} = T_n - \tau_{n-1+L} + \tau_{n-1}$$

$$T_n = T_{n+1} - \tau_{n+L} + \tau_n \quad \text{or} \quad T_{n+1} = T_n + \tau_{n+L} - \tau_n.$$ \hspace{1cm} (A.1)

Combining the two equations in (A.1) together, we get

$$T_{n-1} - T_{n+1} = \tau_{n-1} + \tau_n - \tau_{n+L} - \tau_{n-1+L}.$$ \hspace{1cm} (A.2)
Using the definition of the \( \tau_j \), and the setup in Figure A.0.1, we have that

\[
\tau_n = v(k_n)^{-1} = v(k_{n-1} - e_{n-L})^{-1}
\]

\[
\tau_{n-1} = v(k_{n-1})^{-1}
\]

\[
\tau_{n+L} = v(k_{n+L})^{-1} = v(k_{n-1} + e_{n+L})^{-1}.
\]

Also using (A.1), we can write \( T_n \) as

\[
T_n = \frac{T_{n-1}}{2} + \frac{T_{n+1}}{2} - \frac{\tau_{n+L} - \tau_n}{2} - \frac{\tau_{n-1} - \tau_{n-1+L}}{2} \quad (A.3)
\]

We also have the following relationships about time and travel time, and density and entries at each discrete level

\[
t_n = t_{n-1} + \tau(k_{n-1})
\]

\[
k_n = k_{n-1} - e_{n-L}.
\]

We can combine two consecutive time relations to obtain

\[
t_{n+1} - t_{n-1} = \tau(k_n) + \tau(k_{n-1}),
\]

such that

\[
t_n = \frac{t_{n-1}}{2} + \frac{t_{n+1}}{2} + \frac{\tau(k_{n-1}) - \tau(k_n)}{2}.
\]

Note that by Assumption (A-1) from section 4.1, we have that \( \tau(k_{n-1}) - \tau(k_n) = 0 \) over the first cycle, and is positive otherwise. This implies that \( t_n \geq \frac{1}{2}(t_{n+1} + t_{n-1}) \), which states that the equilibrium \( t_n \) must lie to the right of the midpoint of \( t_{n-1} \) and \( t_{n+1} \).

If we can prove that the value of the equilibrium travel time \( T_n \) which occurs at \( t_n \) is less than the value at the midpoint of \( T_{n-1} \) and \( T_{n+1} \), the proof is complete, as this
argument will violate the concavity condition for the trip time function $T$ since we have already shown that $t_n$ lies to the right of the midpoint of $t_{n-1}$ and $t_{n+1}$. Now, observe equation (A.3). Using the definition of $\tau_i$ and the velocity condition, we have

$$\frac{\tau_{n+L} - \tau_n}{v(k_{n+L})} - \frac{1}{v(k_n)} = \frac{(1 + \dot{T}(t_n))}{v(k_n)} - \frac{1}{v(k_n)} = \frac{\dot{T}(t_n)}{v(k_n)} > 0,$$

where the inequality comes from the fact that we have assumed we are on the portion of the travel duration function $T(t; u)$ where travel time is increasing\(^1\), hence $\dot{T}(t_n)$ is positive.

For the second difference of $\tau$'s, we have

$$\frac{\tau_{n-1} - \tau_{n-1+L}}{v(k_{n-1})} - \frac{1}{v(k_{n-1}+L)} = \frac{1}{v(k_{n-1})} - \frac{(1 + \dot{T}(t_{n-1}))}{v(k_{n-1})} = \frac{\dot{T}(t_{n-1})}{v(k_{n-1})}.$$

Now plugging these into (A.3), we have

$$T_n = \frac{T_{n-1}}{2} + \frac{T_{n+1}}{2} - \frac{1}{2} \left( \frac{\dot{T}(t_{n-1})}{v(k_{n-1})} - \frac{\dot{T}(t_n)}{v(k_n)} \right)$$

$$= \frac{T_{n-1}}{2} + \frac{T_{n+1}}{2} - \frac{1}{2} \left( \frac{\dot{T}(t_{n-1})}{v(k_{n-1})} - \frac{\dot{T}(t_n)}{v(k_{n-1} - e_{n-L})} \right),$$

where the term in parenthesis must be positive since $T(t; u)$ is increasing at a decreasing rate in the early rush hour, and $v(k_{n-1}) < v(k_{n-1} - e_{n-L})$ since velocity is negatively related to density. Therefore, we have shown that $T_n$ must be less than $\frac{1}{2}(T_{n+1} + T_{n-1})$ in equilibrium, which is a contradiction to the concavity assumption on $T(t; u)$. Figure A.0.1 provides a visual representation of the proof. In the Figure, the wedge notation represents the midpoints, ie. $t^\wedge = \frac{1}{2}(t_{n+1} + t_{n-1})$ and $T^\wedge = \frac{1}{2}(T_{n+1} + T_{n-1})$. As the concavity assumption on the trip duration function must hold, if $T_n$ ever lies below the blue line

\(^1\)From the symmetry of the problem, we believe an analogous argument could be applied working backwards from the time of the last exit and using exits rather than entries, to show a similar result. We have not rigorously proved this case, thus the proof has yet to be generalized to the full case where the times occur in the late portion of the rush hour.
(the line that connects any two points of the trip duration function), then the concavity assumption will be violated.

![Figure A.0.1: Discrete departure set diagram](image_url)

**Conjecture 20**  
The velocity function $v(t,u)$ is continuous in $t$.

**Remark:** For the purposes of this thesis, the continuity of velocity is assumed. A line of argument can be constructed to show that certain classes of discontinuities can be ruled out. One such example is entry masses. For example, consider an entry mass at time $t_I(\tilde{t})$ in the $I^{th}$ cycle, where $\tilde{t}$ corresponds to the time in the first cycle such that the entry mass occurs at $t_I(\tilde{t})$ in cycle $I$. Then the commuter who departs an infinitesimal time after the last entry mass $t_I(\tilde{t}) + \delta t$ in cycle $I$ travels with the entry mass for their whole trip, except
during the interval \((t_I+1, t_I+1 + \delta t)\), as the entry mass has exited. This commuter experiences the benefit of higher utility (and therefore higher velocity) over the last portion of their trip, which corresponds to a lower travel time implying a contradiction. We leave the rigorous proof to future work.

Figure A.0.2 summarizes the case of an entry mass in the last cycle, and provides a geometric interpretation for the conjecture. The black ticks on the time axis represent the beginning of cycles, and the red ticks on time axis correspond to the first and second sub-cycle divisions. Note that there is a strict decrease in velocity at \(t_I\) where the mass enters, and a strict increase at \(t_I+1\) when the mass exits. As there is an entry mass at \(t_I\) and an exit mass at \(t_I+1\), there is a jump discontinuity in velocity due to the relationship \(\dot{k} = e(t) - x(t)\) and Greenshields’ Relation. The slopes of the velocity in the Figure are drawn in an arbitrary way, as the function properties are determined by the numerical algorithm, only the behavior at the entry and exit mass times must follow the discontinuous decrease and increase patterns described above.

Result 21 Utility is strictly lower than the equilibrium value of \(u\) outside the departure interval.

Proof. We shall show that \(\dot{U}(t) < 0\) for \(t > t^*\). Now, from differentiation of (4.1.4),

\[ v(k(t + T(t); t, u))(1 + \dot{T}(t)) - v(k(t; t, u)) = 0. \]

Solving for \(\dot{T}(t)\), and noting that \(v(k(t + T(t); t, u)) = v_f\) for \(t \in (t^*, t^* + T(t^*, u))\), we then have the following

\[ \dot{T}(t) = \frac{v(k(t; t, u))}{v_f} - 1 \]

\[ \ddot{T}(t) = \frac{v'(k(t; t, u))\dot{k}(t; t, u)}{v_f} = \frac{x(t; t, u)}{v_f} > 0. \]
The functions $T(t)$ and $\dot{T}(t)$ are continuous at $t^*$, as the density function $k(t)$ is continuous at $t^*$. At the equilibrium utility, $u$, we know that $\ddot{T}(t; u) < 0$, but (A.4) shows that $\ddot{T}(t) > 0$, which implies that $t^*$ is an inflection point for the travel time function $T(t)$. Therefore for $t \in (t^*, t^* + T(t^*; u))$, the $\ddot{T}$ inequalities hold, and the inequality $\ddot{T}(t) > \ddot{T}(t; u)$ holds as well.

From the inequality constraint on the $\dot{T}$ functions, it must be the case that $T(t) > T(t; u)$ for $t \in (t^*, t^* + T(t^*; u))$, which then implies that $U(t, T(t)) < U(t, T(t; u)) = u$. Thus, we have shown that $\dot{U}(t)$ is negative for $t > t^*$, which completes the proof. An analogous argument establishes that utility is strictly lower than $u$ for entry at $t < t^*$. \[\Box\]

**Result 22** The partial derivative $\frac{\partial G_i}{\partial t}(t, u) < 0$ for $i = 1, \ldots, I + 1$.  

183
Proof. Note that

\[ G_1(t, u) = F_2(t; u) - 1 = B_1(t; u)B_2(t; u) - 1 \]

\[ \frac{\partial G_1(t, u)}{\partial t} = \frac{\partial B_1(t; u)}{\partial t}B_2(t; u) + B_1(t; u)\frac{\partial B_2(t; u)}{\partial t}. \]  \hspace{1cm} (A.5)

Since

\[ \frac{\partial B_i(t; u)}{\partial t} = \ddot{\bar{T}}(t_i(t; u); u)t_i'(t; u), \]  \hspace{1cm} (A.6)

and since

\[ t_{i+1}'(t; u) = t_i'(t; u)(1 + \bar{T}(t_i(t; u); u)) \quad \text{where} \quad t_i'(t, u) = \frac{dt_i(t; u)}{dt}, \text{ etc.,} \]  \hspace{1cm} (A.7)

applying equation (A.7) recursively to the right-hand side of (A.6), and using (4.2.17), we obtain

\[ t_i' = F_i(t; u) \]  \hspace{1cm} (A.8)

\[ \frac{\partial B_i(t; u)}{\partial t} = \ddot{\bar{T}}(t_i(t; u); u)F_i(t; u). \]  \hspace{1cm} (A.9)

Substituting (A.9) into (A.5), we have

\[ \frac{\partial G_1(t, u)}{\partial t} = \ddot{\bar{T}}(t_1(t; u); u)F_2(t; u) + \ddot{\bar{T}}(t_2(t; u); u)F_2(t; u)B_1(t; u) < 0, \]  \hspace{1cm} (A.10)

where the inequality follows from the bounds \( F_i(t; u), B_i(t; u) > 0 \) and \( \ddot{\bar{T}}(t_i(t; u); u) < 0 \) for all \( i \). Taking the partial derivative of (4.2.18) with respect to \( t \), and substituting in (A.9), we get

\[ \frac{\partial G_2(t, u)}{\partial t} = \frac{\partial G_1(t, u)}{\partial t}B_3(t, u) + (G_1(t, u) + 1)\frac{\partial B_3(t; u)}{\partial t} \]

\[ = \frac{\partial G_1(t, u)}{\partial t}B_3(t, u) + F_1(t; u)\ddot{\bar{T}}(t_3(t; u); u)F_3(t, u) < 0, \]
where the inequality follows from (A.10), \( F_i(t, u), B_i(t, u) > 0 \), and \( \dot{T}(t_i(t; u); u) < 0 \). Then through recursion,

\[
\frac{\partial G_i(t, u)}{\partial t} < 0 \quad \text{for } i = 1, \ldots, I + 1
\]

\[\Box\]

**Result 23** For \( i = 1, \ldots, I+1 \), a sufficient condition for \( \frac{\partial G_i(t, u)}{\partial u} > 0 \) is that \( \dot{T}_u(t_i(t); u) > 0 \); otherwise, its sign is in general indeterminate.

**Proof.** We first derive an expression for \( \dot{T}_u(t_i(t); u) \) in terms of the utility function. We write \( U(t, T(t; u)) = u \). Differentiation with respect to \( u \) gives

\[
U_T(t, T(t; u)) T_u(t; u) = 1, \quad \text{so that } T_u(t; u) = \frac{1}{U_T(t, T(t; u))} < 0.
\]

Then

\[
\dot{T}_u(t; u) = \frac{\partial \dot{T}}{\partial u}(t; u) = \frac{\partial}{\partial t} \frac{\partial T}{\partial u}(t; u) = \frac{\partial}{\partial t} \frac{1}{U_T(t, T(t; u))} = \left[ \frac{1}{U_T} \right]^2 \left( U_{TT} + U_T \dot{T} \right).
\]

In the language of bid-rent theory, \( \dot{T}(t; u) \) is the premium a commuter is willing to pay in terms of travel time for a unit time later departure. We have \( \dot{T}_u(t; u) > 0 \) when this premium is increasing in utility, which, as can be seen from its expression in terms of the utility function, is a normality condition.

Before proceeding, we clarify a potential ambiguity in notation. In the expression \( 1 + \dot{T}(t_i(t; u); u) \), we write \( \frac{\partial}{\partial u}(1 + \dot{T}(t_i(t; u); u)) \) to denote the derivative with respect to \( u \) taking both \( u \)'s into account, and \( \dot{T}_u(t_i(t; u); u) \) to denote the derivative with respect to the second argument of the function \( \dot{T}(t_i(t; u); u) \).
Since $G_1(t; u)$ and $\frac{\partial G_1}{\partial u}(t; u)$ are defined as

$$G_1(t, u) = B_1(t; u)B_2(t; u) - 1 = F_2(t, u) - 1$$

$$\frac{\partial G_1}{\partial u}(t, u) = \frac{\partial B_1}{\partial u}(t; u)B_2(t; u) + B_1(t; u)\frac{\partial B_2}{\partial u}(t; u),$$  \hspace{1cm} (A.11)

we now have that

$$\frac{\partial B_1}{\partial u}(t; u) = \frac{\partial}{\partial u}(1 + \hat{T}(t_1(t; u); u))$$

$$= \hat{T}_u(t; u) \quad \text{(since } t_1(t; u) = t)$$

and

$$\frac{\partial B_2}{\partial u}(t; u) = \frac{\partial}{\partial u}(1 + \hat{T}(t_2(t; u); u))$$

$$= \hat{T}(t_2(t; u); u)\frac{\partial t_2}{\partial u}(t; u) + \hat{T}_u(t_1(t; u); u).$$

Differentiating (A.7) with respect to $u$ yields

$$\frac{\partial}{\partial u} t_{i+1}(t; u) = \frac{\partial}{\partial u} t_i(t; u) + T_u(t_i(t; u); u) + \hat{T}(t_i(t; u); u)\frac{\partial}{\partial u} t_i(t; u)$$

$$= \left[ \frac{\partial}{\partial u} t_i(t; u) \right] (1 + \hat{T}(t_i(t; u); u)) + T_u(t_i(t; u); u). \hspace{1cm} (A.12)$$

Note that if $\frac{\partial t_i}{\partial u} (t; u) < 0$, then $\frac{\partial t_{i+1}}{\partial u} (t; u) < 0$. Since $\frac{\partial t_1}{\partial u} (t; u) = 0$, then

$$\frac{\partial t_i}{\partial u} (t; u) < 0 \quad \text{for all } i = 2, \ldots, I + 1. \hspace{1cm} (A.13)$$

Since from (A.12), $\frac{\partial t_2}{\partial u} = T_u(t; u)$, then

$$\frac{\partial B_2}{\partial u}(t; u) = \frac{\partial}{\partial u} \left( 1 + \hat{T}(t_2(t; u), u) \right)$$

$$= \hat{T}(t_2(t; u); u)T_u(t; u) + \hat{T}_u(t_2(t; u); u).$$
Substituting the above results into \( (A.11) \) yields

\[
\frac{\partial G_1}{\partial u}(t, u) = \dot{T}_u(t; u)B_2(t; u) + B_1(t; u) \left[ \ddot{T}(t_2(t; u); u)T_u(t; u) + \dot{T}_u(t_2(t; u); u) \right], \tag{A.14}
\]

If \( \dot{T}_u(t; u) > 0 \), then \( \frac{\partial G_1}{\partial u}(t, u) > 0 \) as we know the sign of all terms in \( (A.14) \) from the above arguments; otherwise, \( \frac{\partial G_1}{\partial u}(t, u) \) is in general of indeterminate sign. Differentiating the second line of \( (4.2.18) \) with respect to \( u \), we get

\[
\frac{\partial G_2}{\partial u}(t, u) = \frac{\partial}{\partial u} \left[ B_3(t, u)(G_1(t, u) + 1) - 1 \right] = \frac{\partial B_3}{\partial u}(t, u)F_1(t, u) + B_3(t, u)\frac{\partial G_1}{\partial u}(t, u) \\
= \left[ \dot{T}(t_3(t; u); u)\frac{\partial t_3}{\partial u}(t, u) + \dot{T}_u(t_3(t; u); u) \right] F_1(t, u) + B_3(t, u)\frac{\partial G_1}{\partial u}(t, u) > 0, \tag{A.15}
\]

where all of the terms in \( (A.15) \) are positive, except \( \dot{T}(t_3(t; u); u) \) and \( \frac{\partial t_3}{\partial u}(t, u) \), but both are negative, so their product is positive. Thus we have \( \frac{\partial G_2}{\partial u}(t, u) > 0 \). Applying the above procedure recursively gives the result.

**Result 24** In \((t, u)\)-space, \( G_{i+1}(t, u) = 0 \) lies to the left of \( G_i(t, u) = 0 \).

**Proof.** Consider the equilibrium solution corresponding to \( I \) full entry cycles for a utility level \( u \), labeling the corresponding \( B_i \)'s as \( B_1, B_2, \ldots \), and the equilibrium solution corresponding to \( I + 1 \) full entry cycles for the same utility level, labeling the corresponding \( B_i \)'s as \( B_1^*, B_2^*, \ldots \). Label the starting time for the former equilibrium solution \( t \) and for latter equilibrium \( t^* \).

The proof proceeds by contradiction. Suppose that \( t^* > t \). Then since \( \dot{T}(\cdot; u) < 0 \),

\[
\dot{T}(t_1(t; u); u) > \dot{T}(t_1(t^*; u); u),
\]

which implies that \( B_1 > B_1^* \). Also, since \( t_{i+1}(t; u) = t_i(t; u) + \)

187
\( T(t_i(t; u); u) \), thus by taking the derivative, we obtain
\[
   t_{i+1}'(t; u) = t_i'(t; u) + \hat{T}(t_i(t; u); u)t'_1(t; u).
\]
Applying this with \( i = 1 \) yields
\[
   t_2'(t; u) = t_1'(t; u) + \hat{T}(t_1(t; u); u)t_1'(t; u) + 1 + \hat{T}(t_1(t; u); u) > 0,
\]
which implies that \( t_2(t^*; u) > t_2(t) \). Since \( \hat{T}(\cdot; u) < 0 \), this implies in turn that \( \hat{T}(t_2(t; u); u) > \hat{T}(t_2(t^*; u); u) \), so then \( B_2 > B_2^* \). Applying the same recursive formula for \( t_{i+1}'(t; u) \) for \( i = 2 \) yields
\[
   t_3'(t; u) = t_2'(t; u) + \hat{T}(t_2(t; u); u)t_2'(t; u) = t_2'(t; u)(1 + \hat{T}(t_2(t; u); u)) = (1 + \hat{T}(t_1(t; u); u))^2 > 0.
\]
Since \( \hat{T}(\cdot; u) < 0 \), this implies in turn that \( \hat{T}(t_3(t; u); u) < \hat{T}(t_3(t^*; u); u) \), thus \( B_3 > B_3^* \). By recursion, \( B_i > B_i^* \) for \( i = 1, \ldots, I + 1 \).

Now, \( G_{I+1}^* = \left( \prod_{i=1}^{I+1} B_{I+2}^* \right) B_{I+1}^* - 1 \) and \( G_I = \left( \prod_{i=1}^{I+1} B_i \right) - 1 \). Also, \( B_{I+2}^* < 1 \), since the velocity at the start of the exit cycle must exceed that at the end. Thus, \( G_I > G_{I+1}^* \).

But, by assumption, \( \hat{t} \) is the equilibrium rush-hour start time with \( I \) cycles and \( \hat{t}^* \) is the equilibrium rush-hour start time with \( I + 1 \) cycles, so that \( G_I = G_{I+1}^* = 1 \), which establishes the contradiction.  

**Result 25** The velocity function \( v(t; u) \) is non-negative for \( t \in \mathcal{D} \). By the normalized Greenshields’ Relation, \( k(t; u) \) is also non-negative for \( t \in \mathcal{D} \).

**Proof.** If there is one partial cycle of entries, or exactly one full cycle of entries, the analytical solution for \( e(t; u) \) is available, and entries are non-negative, thus by the matrix equation (4.2.29) it must be the case that \( v(t; u) \) is positive for all \( t \in \mathcal{D} \).
Now we consider the more general case when the number of entry cycles is greater than one. Given the left and right boundary conditions \( v(t, u) = v(t, \bar{u}) = v_f > 0 \), and \( 1 + \dot{T}(t_i(t, u), u) > 0 \), by (4.2.9) we necessarily have that the velocity must be strictly positive at all primary and secondary break points. By the properties of the utility function, we also have that \( 1 + \dot{T}(t, u) > 0 \) and \( \ddot{T}(t, u) < 0 \) for all \( t \in D \), thus \( 1 + \dot{T}(t, u) \) is a monotonically decreasing function, implying that \( \frac{1}{1 + \dot{T}(t, u)} \) is monotonically increasing for all \( t \in D \).

Given the above criteria on the first and second derivative of the trip duration function, there also must exist a unique point, say \( \hat{t} \in D \), such that \( \dot{T}(\hat{t}, u) = 0 \), which implies that

\[
v(\hat{t} + T(\hat{t}, u)) = v(\hat{t}).
\]

For \( t \in [\ell, \hat{t}] \), the velocity condition (4.2.9) implies that \( v(t, u) \) must be monotonically decreasing in this time interval, as we have the following bounds on the coefficient: \( 0 < \frac{1}{1 + \dot{T}(t, u)} < 1 \), as well as the monotonicity of the \( \dot{T}(t, u) \) function. Therefore, since the break points in \( t \in [\ell, \hat{t}] \) are positive and the monotonic decreasing property holds, it would be impossible for \( v \) to be negative in this region, as the velocity function would have to decrease, become negative, and increase to become positive to reach the break point as \( v(t, u) \) is continuous by Lemma 20. Since the problem is symmetric, working from right boundary \( t = \bar{t} \), we would also require that \( v(t, u) \) be positive on \( [\hat{t} + T(\hat{t}, u), \bar{t}] \) by the same argument.

This leaves the region \((\hat{t}, \hat{t} + T(\hat{t}))\) in question. We must have that

\[
\frac{1}{1 + \dot{T}(\ell, u)} > 1, \quad (A.16)
\]

as the value of \( \hat{t} \) is unique, and the quantity in (A.16) is monotonically increasing in \( t \). That
is not to say that there is a minimum at \( \hat{t} \), as the computation of \( v(t, u) \) is given by the quantity \( 1 - k(t, u) \) by Greenshields’ Relation, so \( v(t, u) \) can still be decreasing after \( \hat{t} \).

The interval \([\hat{t}, \hat{t} + T(\hat{t}, u)]\) can span at most two cycles (and at the lowest, one cycle if the endpoints correspond to the break points, in which case we would be done with the proof). In this case, we have shown that velocity must be positive between \( \hat{t} + T(\hat{t}, u) \) and the proceeding break point. If this is true, then it must be true that between \( \hat{t} \) and the break point in \([\hat{t}, \hat{t} + T(\hat{t}, u)]\), the velocity is positive since (4.2.9) must hold. A similar argument would prove positivity of velocity between the interior break point and \( \hat{t} + T(\hat{t}, u) \).

In summary, we have proved that \( v(t, u) \) is positive across all three intervals that make up the departure interval \( D \). This also implies that we have established that \( v(t, u) \in [0, v_f = 1] \), and via Greenshields’ Relation \( k(t, u) \in [0, k_j = 1] \). We have thus established the lemma.

**Result 26** *The entry rate is non-negative over the departure interval.*

**Proof.** The proof of strictly positive entry rates will be shown by induction. In general, there is no analytical formula for the eigenvalues of a tridiagonal matrix, so proving (i) in the definition is all but hopeless. We will instead prove the equivalent statement (ii). The leading principal minors are the determinants of the matrices that are obtained by deleting the last \( n - \kappa \) rows and columns for \( \kappa = 1, \ldots, n \). The coefficient matrix is given in (4.2.29)\(^2\). We can say that there must exist a minimum of the velocity function in \([\hat{t}, \hat{t} + T(\hat{t}, u)]\), since \( v(t, u) > v(\hat{t}, u) \) outside this interval, and the velocity function is continuous, thus the minimum is at least the endpoints, but could be lower.
for the general case. Computing the first few determinants, we see that

\[ D_1 = 1 + B_1 \]
\[ D_2 = 1 + B_2 + B_1B_2 = 1 + (1 + B_1)B_2 \]
\[ D_3 = 1 + B_3 + B_2B_3 + B_1B_2B_3 = 1 + (1 + (1 + B_1)B_2)B_3 \]
\[ D_4 = 1 + B_4 + B_3B_4 + B_2B_3B_4 + B_1B_2B_3B_4 = 1 + (1 + (1 + (1 + B_1)B_2)B_3)B_4, \quad (A.17) \]

where the dependence on \( t \) and \( u \) is dropped. Recall that \( B_i(t, u) > 0 \) for all values of \( t \), therefore the first four determinants are strictly positive. The coefficient matrix in (4.2.29) is a tridiagonal matrix, which is a special kind of structure that allows for the closed form solution of the determinant in the general \( n \times n \) case to be

\[ D_n = a_nD_{n-1} - c_{n-1}b_{n-1}D_{n-2}, \quad (A.18) \]

where \( D_n \) is the determinant with tridiagonal structure

\[
D_n = \begin{vmatrix}
  a_1 & b_1 & 0 & 0 & 0 & \cdots & 0 \\
  c_1 & a_2 & b_2 & 0 & 0 & \cdots & 0 \\
  0 & c_2 & a_3 & b_3 & 0 & \cdots & 0 \\
  0 & 0 & c_3 & a_4 & \cdots & \cdots & 0 \\
  \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \\
  0 & 0 & 0 & \cdots & \cdots & \cdots & b_{n-1} \\
  0 & 0 & 0 & 0 & 0 & c_{n-1} & a_n
\end{vmatrix}
\]

(A.19)

From (4.2.29), the structure implies that \( a_n = (1 + B_n) \), \( b_{n-1} = -B_{n-1}^2 \), and \( c_{n-1} = -\frac{1}{B_{n-1}} \).

Also note the simplification of the product of “\( b_{n-1} \)” and “\( c_{n-1} \)” terms from (A.18), which
we have
\[ c_{n-1}b_{n-1} = \left( -\frac{1}{B_{n-1}} \right) (-B_{n-1}^2) = B_{n-1}. \]  
(A.20)

The goal of the proof will be to show, that in general, the determinant formula (A.18) is equivalent to the nested structure in (A.17) so that positivity of the determinant is guaranteed. Establishing this fact would then show that all leading principal minors are positive, hence the matrix in (4.2.29) is an \( M \)-matrix. For the matrix in (4.2.29) to be an \( M \)-matrix, the off-diagonal entries must be non-positive. The non-positivity of off diagonal entries is easily verified as the matrix is tridiagonal and the first upper and lower bands are
\[ b_{n-1} = -B_{n-1}^2, \text{ and } c_{n-1} = -\frac{1}{B_{n-1}}, \] which are strictly negative. All that remains is to show that the leading principal minors are positive.

With the necessary definitions and structural details completed, we begin the strong induction. The base case for \( n = 1 \) (and \( n = 2, 3, 4 \)) is given in (A.17), which can be written in terms of (A.18), or as the equivalent nested structure in (A.17). For the assumption step, we assume that the equivalency between (A.18) and the nested structure is true for all finite positive integers \( 1 \leq n \leq \kappa \):
\[ D_n = a_n D_{n-1} - c_{n-1} b_{n-1} D_{n-2} = 1 + (1 + (1 + (1 + \ldots)B_{n-2})B_{n-1})B_n > 0 \]  
(A.21)

where by definition, \( D_0 = 1 \) and \( D_{-1} = 0 \). Now, we must establish that the equivalency is true for the \( n = \kappa + 1 \) case, as it is not clear from the formula that the determinant \( D_n \) must be positive, whereas the positivity of the nested structure is clear. For the case where
\[ n = \kappa + 1, \text{ we have the} \]

\[ D_{\kappa+1} = a_{\kappa+1}D_\kappa - c_\kappa b_\kappa D_{\kappa-1} \]

\[ = (1 + B_{\kappa+1}) [1 + (1 + (1 + \ldots)B_{\kappa-2})B_{\kappa-1})B_\kappa] \]

\[ - B_\kappa [1 + (1 + (1 + \ldots)B_{\kappa-3})B_{\kappa-2})B_{\kappa-1}] \]

\[ = 1 + B_{\kappa+1} + \{B_\kappa + (1 + (1 + \ldots)B_{\kappa-2})B_{\kappa-1})B_\kappa\} \]

\[ + \{(1 + (1 + (1 + \ldots)B_{\kappa-1})B_\kappa)B_{\kappa+1}\} - B_\kappa - (1 + (1 + (1 + \ldots)B_{\kappa-2})B_{\kappa-1})B_\kappa \]

\[ = 1 + B_{\kappa+1} + (1 + (1 + (1 + \ldots)B_{\kappa-2})B_{\kappa-1})B_\kappa B_{\kappa+1} \]

\[ = 1 + (1 + (1 + (1 + \ldots)B_{\kappa-2})B_{\kappa-1})B_\kappa B_{\kappa+1}, \quad (A.22) \]

where in lines two and three of (A.22), we use the induction hypothesis that the equivalency statement is true for \( n = \kappa \) and \( n = \kappa - 1 \). After careful distribution of the \( 1 + B_{\kappa+1} \) and \( -B_\kappa \) term in lines 4 and 5 of (A.22), we can cancel out the first bracketed term in line 4 above to get line 6. Factoring out a copy of \( B_{\kappa+1} \) from line 6, the nested structure is the result. This completes the induction step, and verifies that the leading principal minors of the coefficient matrix are positive. Condition (ii) of the definition of the \( M \)-matrix is complete, therefore, the matrix in (4.2.29) is an \( M \)-matrix.

Therefore, the inverse matrix of the coefficient matrix in (4.2.29) has entries that are greater than or equal to zero. At time \( t = t_1 \), all of the values in the column vector on the right-hand side of (4.2.29) are positive, as \( C_1, F_1 > 0 \) for all \( t \) as stated previously, and \( v(A_1) = v_f > 0 \). So we have that the column vector in (4.2.29) is strictly positive at \( t = t_1 \), and the inverse coefficient matrix has non-negative entries, so then the result of multiplying the inverse matrix by the column vector results in the entry rate column vector.
to have strictly positive entries. Since the velocity function $v$ is continuous by Lemma 20, and positive by Lemma 25, the repeated application of the inverse of the matrix in (4.2.29) to the column vector on the right-hand side will be non-negative for all $t \in \mathcal{D}$.

The above proof establishes that the entry rate is non-negative at the first break point $t$, and between break points. In general, we have the left and right limits at a secondary break point

$$\lim_{t \to t_i^-} e(t, u),$$

$$\lim_{t \to t_i^+} e(t, u),$$

which can be computed via (4.2.29). Recall that the matrix used to the left of a secondary break point is of size $I \times I$, whereas the matrix used on the right side of a secondary break point is of size $(I - 1) \times (I - 1)$. Looking at the set of equations, it is not hard to imagine that these limits will not have the same value in general, making the entry rate function $e(t, u)$ discontinuous at each secondary break point, which implies that the velocity and density functions are only of class $C^0$. Nevertheless, the entry rate values at the secondary break points will still be positive due to (4.2.29). A similar argument can be made for primary break points, with the full $I \times I$ matrix being applied on the right limit, and the $(I - 1) \times (I - 1)$ matrix applied to the left side of the primary break point.

Thus we have shown the entry rate is non-negative at the break points and between all break points. This completes the proof.

Result 27 The H-function defined in (4.2.19) is a continuously differentiable function.

Proof. The proof follows from the continuity of the velocity function which was proved
in Theorem 20. Since the velocity function is continuous in both arguments, we take the derivative of the $H$-function (4.2.19) with respect to $t$ to get exactly (4.2.9), which is continuous since the velocity function is continuous and trip duration function is smooth.

If we take the derivative of (4.2.19) with respect to $u$, we get

$$\frac{\partial H}{\partial u}(t, u) = v(k(t + T(t; u); t; u))T_u(t; u) - \int_{t}^{t + T(t; u)} \frac{\partial k}{\partial u}(s; t; u) \, ds.$$  \hspace{1cm} (A.23)

The first term is continuous as the product of continuous functions is continuous. The integral will be continuous, since $k$ is continuous and therefore its partial derivative would contain, at most, jump discontinuities. The theory of Lebesgue integration provides us with the continuity of the integral through the fundamental theorem of calculus. Using the sum of continuous functions being continuous, we establish that (A.23) is continuous. This completes the proof. ■

**Result 28** The $H$-function defined in (4.2.19) is monotonic in the $t$ variable for a fixed value of $u$. In other words, the $H$-function is monotonic in $t$: \( \frac{\partial H}{\partial t}(t, u) > 0. \)

**Proof.** From (4.2.36), we have that

$$\frac{\partial H}{\partial t}(t, u) = - \int_{t}^{t + T(t; u)} \left[ \frac{\partial k}{\partial t}(t; t; u) \right] \, dt.$$  \hspace{1cm} (A.24)

We will establish that the function $\frac{\partial k}{\partial t}(t; t; u)$ is negative. Let the $\delta t$ denote the incremental change in $t$. Then the density function after the incremental change, which we represent as $k_\delta$ is

$$k_\delta(t + \delta t; t + \delta t, u) = 0$$

195
and since from Theorem 26, the entry rate is positive, \( e(t; t, u) > 0 \), then

\[
k(t + \delta t; t + \delta t, u) = k(t; t, u) + e(t; t, u)dt > 0.
\]

Since over the interval \((t, t + T(t, u))\), \( \dot{k}(t; t, u) = e(t; t, u) \) and \( e(t; t, u) \) is strictly positive and finite, \( k(t; t, u) \) is continuous over the interval. Similarly, \( k(t; t + \delta t, u) \) is continuous over the interval \((t + \delta t, t + \delta t + T(t + \delta t, u))\).

Suppose that \( k(t; t, u) \) and \( k(t; t + \delta t, u) \) do not intersect in the interior of the interval \((t + \delta t, t + T(t, u))\). Since \( k(t + \delta t; t + \delta t, u) > k(t; t + \delta t, u) \), then \( k(t; t + \delta t, u) > k(t; t, u) \) for all \( t \) in the interior of the interval \((t + \delta t, t + T(t, u))\). Conditional on this supposition, the lemma is established.

Now suppose, to the contrary, that \( k(t; t, u) \) and \( k(t; t + \delta t, u) \) intersect at some \( t \) in the interior of the interval \((t + \delta t, t + T(t, u))\). Let \( t' \) denote the earliest time at which the two functions intersect. Then from (4.2.29), each the elements in the matrix on the left hand side and in the column vector on the right hand side has the same value, whether the starting time is \( t \) or \( t + \delta t \). Thus, \( e(t'; t, u) = e(t'; t + \delta t, u) \) so that \( k(t' + dt; t, u) = k(t' + dt; t + \delta t, u) \).

The same argument applies for successive increments of time \( dt \) moving forward over this time interval. Hence, \( k(t; t + \delta t, u) > k(t; t + \delta t, u) \) for all \( t \) in the interval \((t + \delta t, t')\), while \( k(t; t + \delta t, u) = k(t; t + \delta t, u) \) for all \( t \) in the interval \((t', t + T(t, u))\). These two results establish that, conditional on this alternative supposition, the lemma is established.

Since the two suppositions are collectively exhaustive, the lemma is established unconditionally.

\[\blacksquare\]

**Conjecture 29** The sign of \( \frac{\partial H}{\partial u}(t, u) \) is negative.
Remark: Some discussion of Conjecture 29 is required. From (4.2.36), we have that

\[
\frac{\partial H}{\partial u}(t, u) = v(k(t + T(t, u); t, u))T_u(t, u) - \int_t^{t+T(t, u)} \left[ \frac{\partial k}{\partial u}(t, t, u) \right] dt. \tag{A.25}
\]

We know that the velocity function is non-negative by Lemma 25, and that \( T_u(t, u) < 0 \), so if we can show that \( \frac{\partial k}{\partial u}(t; t, u) > 0 \), then we will have proven the sign of \( \frac{\partial H}{\partial u}(t, u) \). Since we do not have bounds on the quantities in (A.25) in generality, if we cannot prove the sign of \( \frac{\partial k}{\partial u}(t; t, u) \). Numerical evidence for the logarithmic and exponential utility functions suggest that the conjecture is true. Intuition also suggests that the sign must be negative, as we expect

\[
\left( \frac{du}{dt} \right)_{H=0} = -\frac{\partial H}{\partial t} \frac{\partial H}{\partial u} > 0, \tag{A.26}
\]

and we know from Result 28 that the numerator is positive. For the purposes of this dissertation, we assume the conjecture to be true. A general proof appears to be complicated, as it will require dissecting the numerical algorithm to determine how the entry rate function is determined, and its properties.
Appendix B

Appendix - Utility functions from chapter 4

B.1 Properties of the Logarithmic Utility Function

In [6], the $\alpha - \beta - \gamma$ cost function was implemented, and results showed the presence of departure masses. At first, we attempted to use the $\alpha - \beta - \gamma$ cost function which has a slope discontinuity, as well as a parabolic function for costs. In both instances, those two choices yielded discontinuities in the second derivative of $\ddot{T}(t; u)$, which is used heavily in the computations of equilibrium. To counteract this effect, utility functions were chosen such that derivatives of $T(t; u)$ were smooth in both variables. With smooth utility functions, all derivatives of $T(t; u)$ with respect to $t$ and $u$ can be computed, thus we need not worry about discontinuities in derivatives of the trip duration function.
We use two different choices of utility function, the first is the logarithmic utility function developed in [4], and the second is the exponential utility function developed in [41]. The utility functions have the general properties that: i) there is increasing marginal disutility to travel time, $U_T(t, T(t)) < 0$, $U_{TT}(t, T(t)) < 0$; and ii) $U_t(t, T(t)) < 0$, with $U_t(t, T(t))$ being positive for early arrivals, zero for the most preferred arrival time, and negative for late arrivals. These choices also assume that the utility function is quasi-concave. An assumption is also made, that the utility function generates a strictly concave travel time function, $T(t)$.

All parameters are chosen to be positive constants\(^1\) (eg. $A_0, \alpha_0$, etc.). From these two relationships, we can derive the travel time function $T(t)$ for each utility function. under the assumption that $U(t, T(t)) = u$ in the departure interval, where $u$ is the equilibrium utility which is a constant. The two functions and their specific properties are given in the following subsections.

### B.2 Logarithmic Utility Function

The logarithmic utility function was formulated to preserves almost all of the same properties\(^2\) as the exponential utility function, with one major difference. Both utility functions are convex with respect to time, $t$, and the parameters of the logarithmic utility function are chosen in such a way that the maximum utility is of the same magnitude to that of the exponential utility function. The main difference between the utility functions is that the

\(^1\)It should be reiterated that the parameters were derived through numerical experiment, not empirical results.

\(^2\)It can also be shown that important bounds also hold with the logarithmic utility function, ie. $(1 + \dot{T}(t)) > 0$ and $\ddot{T}(t) < 0$. 

199
logarithmic utility function restricts the rush hour to the interval \((0, M)\), so that \(0 < t\) and \(t + T(t) < M\), where \(M\) is the latest possible arrival time. The logarithmic utility functions has the property that when \(t = 0\) or \(t + T(t) = M\), then \(U(t, T(t)) = -\infty\). The utility function, the travel time function \(T(t)\) and its derivatives are summarized in the following table.

<table>
<thead>
<tr>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>(U(t, T(t)) = \alpha_0 \log(\alpha_1 t) + \beta_0 \log(-\beta_1 (t + T(t) - M)))</td>
</tr>
<tr>
<td>(T(t) = M - t - \frac{1}{\beta_1} \exp\left(\frac{1}{\beta_0} (u - \alpha_0 \log(\alpha_1 t))\right) = M - t - \frac{1}{\beta_1} \exp\left(\frac{u}{\beta_0}\right) (\alpha_1)^{-\frac{\alpha_0}{\beta_0}} t^{-\frac{\alpha_0}{\beta_0}})</td>
</tr>
<tr>
<td>(\dot{T}(t) = -1 + \frac{\alpha_0}{\beta_0 \beta_1} \exp\left(\frac{u}{\beta_0}\right) (\alpha_1)^{-\frac{\alpha_0}{\beta_0}} t^{-\frac{\alpha_0}{\beta_0} - 1})</td>
</tr>
<tr>
<td>(\ddot{T}(t) = \frac{\alpha_0}{\beta_0 \beta_1} \left(-\frac{\alpha_0}{\beta_0} - 1\right) \exp\left(\frac{u}{\beta_0}\right) (\alpha_1)^{-\frac{\alpha_0}{\beta_0}} t^{-\frac{\alpha_0}{\beta_0} - 2})</td>
</tr>
</tbody>
</table>

Table B.2.1: Logarithmic utility and travel time functions

Note that the logarithmic utility function is only defined in the interval \((0, M)\), where the rush hour occurs, and is also smooth in this region. Therefore, no difficulties should arise in discontinuities in \(\ddot{T}(t)\). The first, second, and mixed second partial derivatives of the \(G\) and \(H\) functions, with respect to \(t\) and \(u\), arise in computations for determining various properties of the \(G\) and \(H\) functions (proofs provided in Appendix A). The signs of the partial derivatives establish the properties of interest.

The partial derivatives of the \(G\) functions can be calculated analytically, by differentiating (4.2.18). Table B.2.2 provides the signs of various partial derivatives which can be verified by taking the appropriate derivatives of functions of Table 4.4.2.
We know that when the $G$ and $H$ functions intersect, there exists an integer number of entry cycles. If it can be shown that at least one of the $G_i$ functions intersects the $H$ function, $t$

### B.3 Exponential Utility Function

The parameters of the exponential utility function are the positive constants $A_0, a_1, B_0, b_1$. Once the $T(t)$ function for travel time is defined, all other time derivatives of $T(t)$ are known. Note that all the derivative, $\dot{T}(t), \ddot{T}(t)$, etc. are smooth functions in the departure interval. The exponential utility function is well defined even outside the interval $(0, M)$, where we have used 0 as the “reference time” and $M$ as the “latest feasible” arrival time. One consequence of the latter property of the exponential utility function, is that the rush hour can become extremely long; much longer than the 6 hour time interval we initially
gave \((M = 6)\) for the logarithmic utility function. The utility function, the travel time function \(T(t)\) and its derivatives are summarized in the following table.

### Functions

\[
U(t, T(t)) = \left( \frac{A_0}{a_1} \right) (1 - e^{-a_1 t}) + \left( \frac{B_0}{b_1} \right) \left( 1 - e^{b_1(t+T(t)-M)} \right)
\]

\[
T(t) = M - t + \frac{1}{b_1} \log \left( 1 - b_1 \left( \frac{u - A_0}{a_1} (1 - \exp(-a_1 t)) \right) \right)
\]

\[
\dot{T}(t) = -1 + \frac{A_0 a_1}{-A_0 b_1 + \exp(a_1 t)(A_0 b_1 + a_1 (B_0 - b_1 u))}
\]

\[
\ddot{T}(t) = - \frac{a_1^2 A_0 (A_0 b_1 + a_1 (B_0 - b_1 u)) \exp(a_1 t)}{(-A_0 b_1 + \exp(a_1 t)(A_0 b_1 + a_1 (B_0 - b_1 u))^2}
\]

#### Table B.3.3: Exponential utility and travel time functions

The exponential utility function is given below. The travel time function, \(T(t)\), and its first two derivatives are also provided, as both are heavily used in the algorithm.

Under the assumptions that are required for the utility functions, and the analytic expressions in the table, one can easily deduce that \(1 + \dot{T}(t) > 0\) and \(\ddot{T}(t) < 0\) for all \(t\) in the departure interval. These bounds are essential for the proofs in chapter 4.