This page is here to make the page numbers come out correctly.

Do not print this page.

Control and Optimization of the Burgers IVP and applications to Traffic Flow

A Thesis Presented to

The Faculty of the Mathematics Program

California State University Channel Islands

In (Partial) Fulfillment

of the Requirements for the Degree

Masters of Science

by

Mayra Emperatriz Sahagun

August, 2016

Signature page for the Masters in Mathematics Thesis of

APPROVED FOR THE MATHEMATICS PROGRAM

Cynthia Flores — August 9, 2016

Dr. Cynthia Flores, Thesis Advisor — Date

08/09/16

Dr. Jorge Garcia, Thesis Committee — Date

Selenne Bañuelos — 08/09/2016

Dr. Selenne Bañuelos, Thesis Committee — Date

APPROVED FOR THE UNIVERSITY

8-9-16

Dr. Gary A. Berg, AVP Extended University — Date

To my parents, Ignacio and Maria, in gratitude for their encouragement and support.

Acknowledgements

This thesis could not have been written without the constant advice and encouragement of my advisor, Cynthia Flores. Many thanks must also go to the members of my committee, Jorge Garcia and Selenne Bañuelos. I would also like to thank Marina Morales, Michael Ruiz, and Dana Cochran for many helpful conversations for support and encouragement throughout my time in graduate school. I would also like to thank Christopher Tull, Jennifer Sahagun, and Jesus Sahagun for their unconditional love and support.

Control and Optimization of the Burgers IVP and applications to Traffic Flow

by Mayra Emperatriz Sahagun

Abstract

Conservation laws are of fundamental importance to describing the physical world. Burgers equation is a fundamental partial differential equation that is a conservation law. Burgers equation with viscosity $\mu > 0$ is the following,

$$\begin{cases} u_t + uu_x = \mu u_{xx} & x \in \mathbb{R}, t > 0 \\ u(x,0) = u_0(x) & x \in \mathbb{R}. \end{cases}$$

A good example where Burgers equation is applicable is in the area of traffic, where the main variables are density, traffic speed, and velocity. The density, and velocity are the main variables expressed as a function of $x \in \mathbb{R}$ and time $t > 0$. Our goal is to approximate solutions to the initial value problem (IVP) for Burgers equation in the presence of additional external forces and compare our numerical simulations to real (actual) data collected from the Caltrans Performance Measurement system (PeMS). The data comes from

different sensors along the 101 South freeway from Woodland Hills to Universal Studio and consists of collected information such as traffic speed, velocity, and vehicle occupancy. The main goal in this thesis is to design a numerical model in terms of the collected traffic information along with a corresponding optimization problem whose solutions will help to improve traffic conditions.

This problem is reduced to minimizing an integral quantity. Specifically, we are studying the following quantity

$$\min_{u_*} \frac{1}{2} \int_0^T \int_0^1 (u(x,t) - u_d(x,t))^2 + C u_*^2(x,t) \; dxdt, \tag{1}$$

where $u(x,t)$ are solutions to Burgers equation, $u_d(x,t)$ represents our desired state and $u_*(x,t)$ represents the external forces that are added to the system. The main goal for minimizing this integral quantity is to achieve close solutions to our desired state. Choosing our desired state $u_d(x,t)$, then computing the corresponding external forces $u_*(x,t)$, and solutions $u(x,t)$ to the *forced* Burgers equation helps us achieve our main goal – improving traffic conditions on the 101 South freeway.

Contents

1. Introduction

Burgers equation is a fundamental partial differential equation that is a conservation law named after Johannes Martinus Burgers (1895-1981). It occurs in various areas in applied mathematics, such as modeling of gas dynamics, traffic flow, and captures formations of shock and of rarefied waves. This equation arises as a one dimensional simplification of the three dimensional system of equations that govern fluid dynamics. That is, Burgers equation is a simplified form of the Navier-Stokes equation. Elements of fluid flow and gas dynamics include compressibility, pressure, and velocity. As an example, gases are compressible fluids, unlike viscous fluids that are incompressible, i.e., liquids are incompressible fluids. Another example includes the velocity fields of moving gases that can be analyzed as the flow of air over an airplane wing or over a surface of an automobile. Burgers equation is a model for incompressible fluid flow and the main topic of this manuscript.

The initial value problem (IVP) to Burgers equation is,

$$\begin{cases} u_t + uu_x = \mu u_{xx} & x \in \mathbb{R}, t > 0 \\ u(x,0) = u_0(x) & x \in \mathbb{R}, \end{cases} \tag{2}$$

as presented in its viscid form. The function $u(x, t)$, is a function of two variables, $x \in \mathbb{R}$ and $t > 0$. Another important concept of Burgers equation is viscosity, represented by the parameter $\mu > 0$. Viscosity is the property of a fluid that resists the force tending to cause the fluid to flow. As an example, imagine a styrofoam cup with a hole at the bottom of the cup. If we pour honey into the cup we will find that the cup drains very slowly. This is because of honey's viscosity i.e., resistance is large compared to other liquid's viscosity's. If we were to fill the same cup with water, we will see that the water will drain much more quickly given its low viscosity. Burgers equation is studied in its viscous and inviscid forms, which depend on whether the effects of viscosity are considered i.e., $\mu > 0$ or $\mu = 0$.

The inviscid Burgers equation is defined by,

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0, \tag{3}$$

i.e., no viscosity is present.

The viscous Burgers equation can be linearized by the Cole-Hopf transformation as was seen in [1, 7]. In the viscous Burgers equation, x is the position; t is the time; $u(x, t)$ is a function given by its velocity, which is furthermore related to the fluid density; and, $\mu > 0$ is the viscosity parameter.

The modeling of traffic through Burgers equation is important because it can assist with traffic patterns and the prediction of an ideal departure time. We work with Burgers equation, a nonlinear conservation law that is used to model traffic flow. We are interested in the initial value problem of the Burgers equation, which can be solved analytically under certain circumstances. Our goal for this thesis is to stimulate traffic flow using numerical methods and to compare our methods to actual (real) data.

We will first introduce some important terminology and definitions that are used throughout this thesis.

1.1. **Notation and Terminology.** The following notation will be used throughout this thesis. For the sake of simplicity, we will use subscripts to denote partial differentiation,

$$\frac{d}{dt}u(x,t) = u_t(x,t) = u_t,$$

where the dependence on (x,t) will be omitted when it is clear from the context. The symbol $\int$ will be used to denote integration on the real line, suppressing the limits of integration when it is appropriate to do so.

Many of the definitions can be found in standard Functional analysis texts, for instance see the classical Rudin analysis texts or [6]. For the context of this thesis we will be working with function spaces that have important structures, namely one that admits an inner product between functions. We remind the reader the definition of an inner product space.

Definition 1. *An **inner product space** is a vector space X with an inner product defined on X. An inner product on X is a mapping of $X \times X$ into the scalar field K of X such that, for every pair of vectors x and y there is associated a scalar which is written $\langle x, y \rangle$ and is called the inner product of x and y, such that $\forall$ vectors $x, y, z \in X$ and scalars $\alpha \in K$ we have,*

$$(i)\langle x+y, z\rangle = \langle x, z\rangle + \langle y, z\rangle$$

$$(ii)\langle \alpha x, y\rangle = \alpha\langle x, y\rangle$$

$$(iii)\langle x, y\rangle = \overline{\langle y, x\rangle}$$

$$(iv)\langle x, x\rangle \geq 0$$

$$(v)\langle x, x\rangle = 0 \Longleftrightarrow x = 0$$

Next we will take a look at what it means for our function to be in the space $L^2(\mathbb{R})$.

Definition 2. *The $L^2(\mathbb{R})$ space is the space of square-integrable functions for which the integral of the square of the absolute value is finite. Hence, if f is a measurable function such that*

$$\int_{-\infty}^{\infty} |f(x)|^2 \, dx < \infty,$$

then f is in the $L^2(\mathbb{R})$ space.

The following remark is another way to re-write the above definition that will be necessary for this thesis.

Remark 1. *We are interested in $\mathbb{R}$ valued functions. Let $L^2(I) = \{f \in M$ s.t. $f : I \to K$ and $\int_{x \in I} |f(x)|^2 \; dx < \infty\}$, where the domain $I \subset \mathbb{R}$ and $K = \mathbb{R}$. In this thesis we will consider $I = [0, 1]$.*

The space $L^2(\mathbb{R})$ admits an inner product given by the next definition.

Definition 3. *The $L^2(\mathbb{R})$ inner product is defined as*

$$\langle f, g \rangle = \int_{-\infty}^{\infty} f(x)\overline{g}(x) \, dx.$$

The inner product presented above satisfies the conditions presented in definition 1. The ability to define an inner product allows us to discuss when functions are orthogonal or perpendicular.

Definition 4. *Given an inner product on a vector space V, we say that two vectors $v, u \in V$ are* ***orthonormal*** *if their inner product is zero and their norm with respect to inner product is unitary.*

Mathematically, orthonormal means we have a collection $\{v_1, \cdots, v_k\} \subset H$, where H is a Hilbert space (definition 6), such that when $i \neq j$ then $\langle v_i, v_j \rangle = 0$. That is, the vectors are mutually perpendicular and are require to have length one hence, $\sqrt{|\langle v_i, v_i \rangle|} = 1$ when $i = j$.

The $L^2(\mathbb{R})$ space as well as $L^2(I)$ forms a Hilbert space with an inner product as given in definition 3. We now recall the definition of a Hilbert space for which we need to know the definition of a complete metric space.

Definition 5. *A* ***complete*** *metric space is a metric space in which every Cauchy sequence converges.*

Definition 6. *A* ***Hilbert space*** *is a complete inner product space, that is, it is complete in the metric defined by the inner product.*

Definition 7. *A* ***Schauder basis*** *is a sequence $\{x_i\}$ in X, a normed vector space, with the property that every $x_i \in X$ has a unique representation of* *the form* $x = \sum_{i=1}^{\infty} a_i x_i$ *for all* $a_i \in \mathbb{R}$*, where the convergence is understood with respect to the norm. In other words,*

$$\lim_{N\to\infty} ||x - \sum_{i=1}^{N} a_i x_i||_X \to 0.$$

Much of this thesis will be considered by working in the space $L^2[0,1]$ (recall remark 1). We highlight properties of this space below.

Example 1. *The space of square integrable functions on the unit interval* $L^2[0,1]$ *is complete.*

For any $f \in L^2[0,1]$ there exists $\{g_n\}$ that is Cauchy such that, $||f - g_n||_{L^2[0,1]} \to 0$ as $n \to \infty$.

We now direct your attention to Rolle's Theorem that will be important for Theorem 2.

Theorem 1 (Rolle's Theorem)**.** *If a real-valued function* f *is continuous on a proper closed interval* $[a,b]$*, differentiable on the open interval* (a,b) *and* $f(a) = f(b)$*, then there exists at least one* c *in the open interval* (a,b) *such that* $f'(c) = 0$.

An interesting phenomenon occurs when two characteristic curves belonging to a given PDE intersect (see definition 8). In order to understand this behavior, we need to remind the reader of the following fundamental theorems, whose proof is provided for convenience.

Theorem 2. *The Mean Value Theorem states that if $f(x)$ is defined and continuous on the interval $[a, b]$ and differentiable on (a, b), then there is at least one number $c \in (a, b)$ such that,*

$$f'(c) = \frac{f(b) - f(a)}{b - a}.$$

Proof. The equation of a secant through $(a, f(a))$ and $(b, f(b))$ is

$$y - f(a) = \frac{f(b) - f(a)}{b - a}(x - a).$$

Adding $f(a)$ to both sides we obtain,

$$y = \frac{f(b) - f(a)}{b - a}(x - a) + f(a).$$

Let $g(x) = f(x) - [\frac{f(b)-f(a)}{b-a}(x - a) + f(a)]$. Note that $g(a) = g(b) = 0$, we know that g is continuous on $[a, b]$ and differentiable on (a, b) since f is. Using theorem 1 there exists $c \in (a, b)$, such that $g'(c) = 0$. However,

$$g'(x) = f'(x) - \frac{f(b) - f(a)}{b - a} = 0.$$

Then, $g'(c) = f'(c) - \frac{f(b)-f(a)}{b-a} = 0$ adding $\frac{f(b)-f(a)}{b-a}$ to both sides we obtain,

$$f'(c) = \frac{f(b) - f(a)}{b - a},$$

which is what we wanted. $\square$

Now we will state the Fudamental Theorem of Calculus whose proof will also be provided.

Theorem 3. *The Fundamental Theorem of Calculus states that if f is continuous on the closed interval $[a, b]$ and F is the indefinite integral of f on $[a, b]$, then $\int_a^b f(x)dx = F(b) - F(a)$.*

Proof. Let f be continuous and differentiable function, define $F(x)$ with the property that $F'(x) = f$. We are going to define

$$G(x) = \int_a^x f(t)dt.$$

Applying the second Fundamental Theorem of Calculus, tells us that

$$G'(x) = f(x).$$

So $F'(x) = G'(x)$. Hence,

$$(F - G)' = F' - G' = f - f = 0.$$

We know that if two functions have the same derivative then they differ only by a constant, namely

$$F(x) - G(x) = c.$$

Adding $G(x)$ to both sides we obtain,

$$F(x) = G(x) + c.$$

Now we are going to compute $F(b) - F(a)$ to see if it equals the definite integral. We have,

$$F(b) - F(a) = (G(b) + c) - (G(a) + c). \tag{4}$$

Looking at the right hand side of equation (4) and distributing the $(-)$ sign we have,

$$G(b) + c - G(a) - c.$$

We now combine like terms to obtain,

$$G(b) - G(a).$$

Substituting what G is and integrating we have,

$$\int_a^b f(t)dt - \int_a^a f(t)dt,$$

to obtain,

$$\int_a^b f(t)dt - 0.$$

Therefore, $F(b) - F(a) = \int_a^b f(x)dx$.

$\square$

We direct the readers attention to the next corollary. It briefly mentions integration by parts and how this technique is preformed.

Corollary 1. *Integration by parts is a technique for performing indefinite integration $\int udv$ or definite integration $\int_a^b udv$ by expanding the differential of a product of functions $d(uv)$ and expressing the original integral in terms of a known integral $\int vdu$.*

Integration by parts can be justified by considering the product rule of differentiation with,

$$d(uv) = udv + vdu,$$

and integration by both sides,

$$\begin{aligned}\int d(uv) &= uv \\ &= \int udv + \int vdu.\end{aligned}$$

Rearranging the top we achieve,

$$\int udv = uv - \int vdu.$$

For the simplicity of this thesis we will be using integration by parts specifically, when we get to the section of discretization when we have boundary terms of the form,

$$\int_0^1 udv = uv \Big|_0^1 - \int_0^1 vdu. \tag{5}$$

Looking at equation (5) we will be interested in ensuring the boundary terms vanish. Later in this thesis 16, we will work with an appropriate collection of functions, namely $\varphi_j = \sqrt{2}\sin(j\pi x)$. Since the function $\sin(2j\pi x)$ is periodic, anytime that x is 0 or 1 the quantity at the boundary would be 0.

The inviscid Burgers equation (3) describes a conservation equation. More generally it is a quasilinear hyperbolic equation. The solutions to this conservation equation can be constructed by the methods of characteristics which yields an ordinary differential equation. For readers interested in learning more about this methods of characteristics, see [1].

The fundamental theorems and corollaries mentioned previously will help us justify proposition 1, and figure out what it means for a shock wave corresponding to the inviscid Burgers IVP to break. In the following proposition we find an exact time labeled as T_b at which a shock wave breaks.

Definition 8. *Given a PDE, it is possible to define curves in the domain where the solution is constant and can be found by solving an ODE. We call these* ***characteristic curves****.*

Proposition 1. *If we solve the solution to (2) with smooth initial data $u_0(x)$ for which $u_0'(x)$ is somewhere negative, will exhibit a wave breaking phenomenon at time.*

$$T_b = -\frac{1}{min\ u_0'(x)}.$$

Proof. Burgers equation is $u_t + uu_x = 0$, and can be written as an inner product. Namely,

$$= \langle 1, u \rangle \cdot \langle u_t, u_x \rangle$$

$$= \langle 1, u \rangle \cdot \nabla u = 0.$$

The function $u(x,t)$ evaluated along the curve $x(t)$, in the direction of $(1, u)$ is a constant, and ∇u describes the maximum rate of change. Hence, satisfying

$$x'(t) = u(x(t), t).$$

Equation (2) states that $u(x(t), t)$ is a constant, which we will call $u_0(x(0))$. Since $u(x(t), t) = \mathit{constant}$, in particular

$$\begin{aligned} u(x(t), t) &= u(x(0), 0) \\ &= u_0(x_0). \end{aligned}$$

Using 3 we get the following,

$$x(t) = u_0(x(0))t + c_1. \tag{6}$$

We will substitute $t = 0$ to (6) where,

$$x(t)\Big|_{t=0} = c_1.$$

Since $t = 0$ we obtain,

$$x(0) = c_1.$$

Therefore,

$$u_0(x(0))t + x(0) = \alpha,$$

where α is a constant.

Consider what were to happen when two characteristic lines intersect. Say we have two characteristic equations:

$$x_1(t) = u_0(x(0))t + x(0),$$

$$x_2(t) = u_1(x(1))t + x(1).$$

Lets observe what happens when these two equations equal each other,

$$u_0(x(0))t + x(0) = u_1(x(1))t + x(1).$$

Solving for t is going to give us exactly the time when these two characteristic equations intersect. Hence we solve for t showing the algebraic manipulations below,

$$u_0(x(0))t + x(0) = u_1(x(0))t + x(1),$$

$$u_0(x(0))t + x(0) - u_1(x(1))t - x(1) = 0,$$

$$t(u_0(x(0)) - u_1(x(1))) = x(1) - x(0),$$

$$t = \frac{x(1) - x(0)}{u_0(x(0)) - u_1(x(1))},$$

$$t = -\frac{x(1) - x(0)}{u_1(x(1)) - u_0(x(0))}.$$

Applying the Mean Value Theorem to our problem there exists $c \in (x(0), x(1))$ such that,

$$u_0'(c) = \frac{u_0(x(1)) - u_0(x(0))}{x(1) - x(0)}.$$

We can see from equation (2) and (6), that t and $u_0'(c)$ are negative reciprocals of each other. Hence,

$$t = -\frac{1}{u_0'(c).}$$

Since u_0' is negative, the first instance the two lines will meet is given by the wave breaking time T_b, which is

$$T_b = -\frac{1}{\min\, u_0'(c)}.$$

$\square$

In the sections that follow, we will deduce a linear system of equations by projecting solutions to (2) to a finite dimensional space, and then formulating a corresponding linear system to solve. In order to deduce this, we are presented with the challenge of identifying a collection of orthonormal functions in $L^2[0,1]$, which we denote by $G = \{\varphi_\lambda\}_{\lambda \in I}$, and I is an index set that satisfies some basic properties. When a finite collection contained in G is considered, $\Phi_N = \{\varphi_1, ..., \varphi_N\}$ we formulate (2) in the span of Φ_N.

We now briefly mention the Conjugate Gradient Method that will be the main tool for solving linear systems of equations.

Definition 9. *In mathematics, the* ***Conjugate Gradient Method*** *is an algorithm for the numerical solution of particular systems of linear equations, namely those whose matrix is positive-definite.*

Originally the conjugate gradient method was developed as a direct method that later became popular for its properties. The conjugate gradient method can be used to solve unconstrained optimization problems, such as the minimization problem presented in this thesis. The use of the conjugate gradient method for these types of problems was studied in [1] and are not the central focus of this thesis. In this work we expand its use to solve traffic control problems.

This is essential because without the theorems and propositions that were mentioned previously, we would not have been able to understand this thesis.

2. Deducing Burgers Equation as a model for traffic flow

Traffic is something that everyone does not like to be in. Other people find traffic to be the time to reflect on what they did throughout the day, or reflect on things that they have planned to do during the week. In this section, we will be deriving Burgers equation as a traffic flow model with the aim of developing numerical methods that give insight as to improving traffic conditions.

In mathematics, traffic flow is the study of interactions between vehicles, drivers, pedestrians, cyclists, travelers, and infrastructures. With the aim of understanding and developing an optimal road network with efficient movement of traffic, and minimal traffic congestion problems. However, in this thesis traffic flow is generally treated as a one dimensional pathway (travel lane). In fact, multiple lanes traveling in one direction are treated as a single lane. There are two important dependent variables to analyze and visualize in a traffic flow: speed, and density. The speed at which the vehicle is driving which we will denote as $v(x,t)$, and density of cars we will denote as $p(x,t)$.

The speed in traffic flow is defined as the distance covered per unit time. The speed of every vehicle is impossible to track on a roadway however, the average speed is based on the sampling of vehicles over a period of time. The density is defined as the number of vehicles per unit length. The flow is the number of vehicles that are passing a reference point per unit of time.

2.1. **Deriving a model for traffic flow using Burgers equation.** We will begin by considering the density of cars instead of looking at individual cars. As we said before, the density of vehicles is the number of vehicles per unit length (distance) and should be measured at a point in the lane for a specific moment in time. Mathematically, this leads to the density function $\rho(x,t)$ such that, the number of cars between x_1 and x_2 at time t_0 is the following,

$$\int_{x_1}^{x_2} \rho(x,t_0)dx. \tag{7}$$

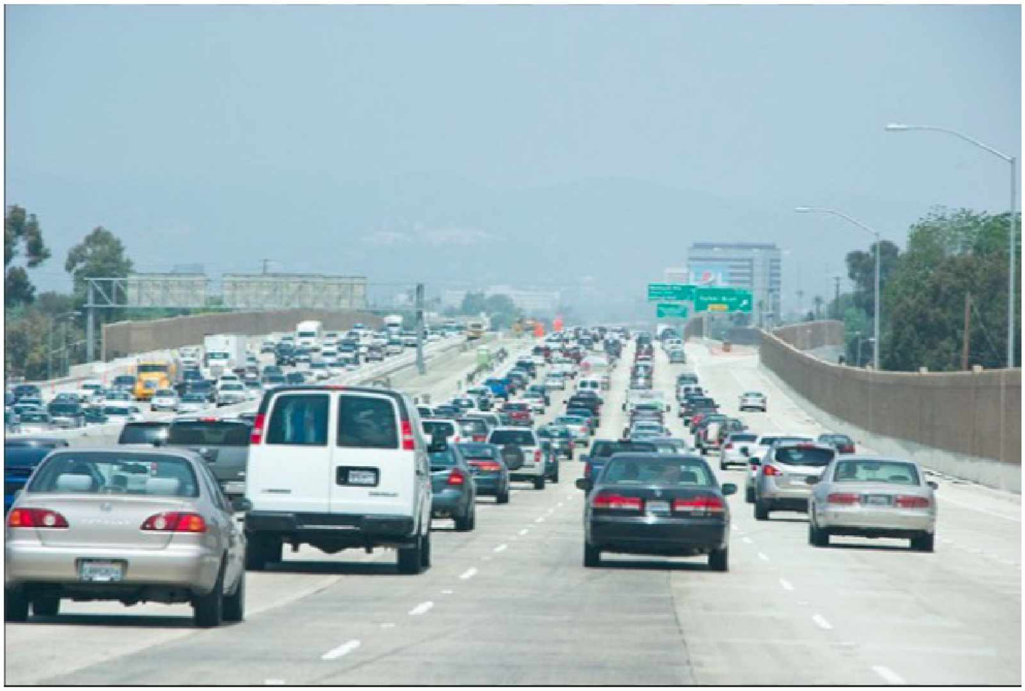

FIGURE 1. Depicting the flow of cars

Velocity will be $v(x,t)$ at which the vehicles are traveling. Let's consider the quantity $\rho(x,t)v(x,t)$ as the number of cars which passes through a point in space call it x, at a given time t.

We are interested in establishing a relationship between ρ and v. We will choose a fixed interval $[x_1, x_2]$ that measures the number of cars which are in the interval, and also figure out how the quantity is changing over time. Since the vehicles are flowing on an interval, the quantity changes as vehicles enter or exit the interval. (See figure 1.)

We can represent these facts as a mathematical equation that later can formally be used,

$$(8) \qquad \frac{d}{dt}\int_{x_1}^{x_2} \rho(x,t)dx = \rho(x_1,t)v(x_1,t) - \rho(x_2,t)v(x_2,t).$$

Considering equation (8) means we have expressed the the rate at which number of cars between x_1 and x_2 changes by considering the number of cars exiting and taking away the number of cars entering.

Integration on (9) over an interval of time implies the following,

$$\begin{aligned}\int_{t_1}^{t_2}\int_{x_1}^{x_2} \partial_t\rho(x,t)dxdt &= \int_{t_1}^{t_2}[\rho(x_1,t)v(x_1,t) - \rho(x_2,t)v(x_2,t)]dt \\ &= -\int_{t_1}^{t_2}\int_{x_1}^{x_2} \frac{d}{dx}(\rho(x,t)v(x,t))dxdt.\end{aligned}$$

Using the Fundamental Theorem of Calculus we obtain,

$$\int_{t_1}^{t_2}\int_{x_1}^{x_2} \partial_t\rho(x,t)dxdt = -\int_{t_1}^{t_2}\int_{x_1}^{x_2} \partial_x(\rho(x,t)v(x,t))dxdt.$$

Since the integral of these functions are equal over arbitrary interval, we can conclude that

$$(9) \qquad \partial_t\rho(x,t) = -\partial_x[\rho(x,t)v(x,t)],$$

which implies,

$$\rho_t(x,t) + (\rho(x,t)\ v(x,t))_x = 0. \tag{10}$$

Equation (10) is interesting because it almost looks like Burgers equation without viscosity!

If the road is empty, meaning $\rho = 0$ the cars are traveling a maximum speed of v_{max}; however, if the maximum density ρ_{max} is reached, the cars are bumper to bumper meaning that drivers are going to reduce their speed eventually to zero. We will introduce an equation that will model the facts mentioned above, and provides a linear relation between the velocity and density function namely,

$$v(\rho) = v_{max}\Big(1 - \frac{\rho}{\rho_{max}}\Big). \tag{11}$$

By adding an initial condition to equation (9) and using v as a function of ρ by equation (11), we can deduce our main initial value problem. We define the conservation of cars problem as given by

$$\begin{cases} \rho(x,t)_t + [v_{max}\rho(x,t)(1 - \frac{\rho(x,t)}{\rho_{max}}])_x = 0, & x \in \mathbb{R},\ t > 0 \\ \rho(x,0) = \rho_0(x), & x \in \mathbb{R}. \end{cases} \tag{12}$$

We want to establish (12) in regards to its conservative behavior, that the quantity of cars on the road is constant over time. This establishes the following proposition.

Proposition 2. *The Burgers IVP is conservative. Given a function ρ such that is satisfies* (11) *and has compact support, then the quantity*

$$\int_{\mathbb{R}} \rho(x,t)\ dx$$

is a constant over time.

Proof. To begin, we need to show that the quantity does not change over time. We can deduce that its derivative is zero. Differentiating with respect to time we have,

$$\begin{aligned}\frac{d}{dt}\int_{\mathbb{R}} \rho(x,t)\ dx &= \int_{\mathbb{R}} \partial_t \rho(x,t)\ dx \\ &= -\int_{\mathbb{R}} \partial_x [v_{max}\rho(x,t)(1-\frac{\rho(x,t)}{\rho_{max}})]\ dx \\ &= v_{max}\rho(x,t)(1-\frac{\rho(x,t)}{\rho_{max}})\Big|_{|x|\to\infty} \\ &= 0.\end{aligned}$$

This follows from (11) and the compact support of ρ. $\square$

Corollary 2. *The initial amount of cars is conserved over time. That is,*

$$\int_{\mathbb{R}} \rho(x,t)\ dx = \int_{\mathbb{R}} \rho_0(x)\ dx.$$

Readers interested in more details about the conservative nature of equation (12) are directed to [7]. We now consider the minimization problem arising from the traffic flow PDE that recently derived, because this is the main goal for this thesis.

3. Statement of the Minimization Problem

The minimization problem related to the optimal control problem that will be stated in this section motivated by avoiding the wave breaking time, T_b presented in proposition 1, associated with Burgers equation. Applications consist of introducing numerically computed force terms to apply to the Burgers model. Moreover, the solutions to the control problem may give insight to controlling undesirable traffic situations.

Specifically, our goals for this minimization problem are related to the following integral quantity over an appropriate class of functions. The associated control problem to Burgers equation is given to us by the minimization problem stated below,

$$\min_{u_*} \frac{1}{2} \int_0^T \int_0^1 (u(x,t) - u_d(x,t))^2 + C u_*^2(x,t) dx dt, \tag{13}$$

subject to the condition that

$$u_*(x,t) = u_t - \epsilon u_{xx} + u u_x, \tag{14}$$

where u_d represents the desired state, and u_* is the external (control) forcing term. Noted above is the parameter for viscosity $\epsilon > 0$, alluding to the choice of small viscosity, as in the traditional vanishing viscosity method. For any given u_* we want to approximation a solution to (14). Overall our objective is to figure out a way to approximate a solution to the optimization problem for Burgers equation, namely, by finding the appropriate minimum u_* using equation (13). The first thing that we will do is discretize the the domain of u so we can arrive to a finite dimensional formulation corresponding to (13). Then we will use the conjugate gradient method (see Definition 9) to approximate these solutions. Recall that $\Phi_N = \{\varphi_1, \cdots, \varphi_N\}$. In this chapter, we will often refer to the following contrained initial value problem,

$$\begin{cases} \min_{u_*} \frac{1}{2} \int_0^T \int_0^1 (u(x,t) - u_d(x,t))^2 + C u_*^2(x,t) \; dxdt \\ u_*(x,t) = u_t - \epsilon u_{xx} + u u_x, \\ u(x,0) = u_0(x) & x \in (0,1). \end{cases} \tag{15}$$

In definition 7 we presented what a Schauder basis is and the properties that have to hold. For $j \in \mathbb{Z}$ we introduce an important collection of functions given by,

$$\varphi_j = \sqrt{2}\sin(j\pi x), \tag{16}$$

and $\varphi_j(0) = 0, \varphi_j(1) = 0$.

Moreover, the collection $G = \{\varphi_n\}$ for $n \in \mathbb{Z}$, and $G \subset L^2[0,1]$ will serve as a basis for solutions to (15). That is, if $f \in L^2[0,1]$ then, $||f - \sum_{i=1}^{n} a_i\varphi_i||_{L^2} \to 0$ as $n \to \infty$ and $a_i \in \mathbb{R}$. An important property of G is summarized in the following proposition.

Proposition 3. *The collection φ_n are orthogonal where $\varphi_n(x) = \sqrt{2}\sin(n\pi x)$. Moreover,*

$$\langle \varphi_n, \varphi_m \rangle = \begin{cases} 1 & \text{if } n = m, \\ 0 & \text{otherwise} \end{cases}$$

with respect to $L^2[0,1]$ inner product.

Proof. We are looking at the case where $n \neq m$ and $n, m \in \mathbb{Z}$,

$$\int_0^1 [\sqrt{2}\sin(n\pi x)][\sqrt{2}\sin(m\pi x)]\ dx.$$

We can take the constant out to have,

$$2\int_0^1 [\sin(n\pi x)][\sin(m\pi x)]\ dx.$$

We will be using a trig identity that is,

$$\sin(\alpha)\sin(\beta) = \frac{1}{2}[\cos(\alpha-\beta) - \cos(\alpha+\beta), \tag{17}$$

where $\alpha = n\pi x$ and $\beta = m\pi x$. Using equation (17) we have,

$$2\int_0^1 \frac{1}{2}[\cos(n\pi x - m\pi x) - \cos(n\pi x + m\pi x)]dx,$$

$$\dagger = \int_0^1 [\cos(n\pi x - m\pi x) - \cos(n\pi x + m\pi x)]dx.$$

Using u-substitution on the second part of the integral we are going to let $u = x(n\pi + m\pi)$ and $du = n\pi + m\pi dx$ to have the following:

$$\begin{aligned}
&= \int_0^1 \cos(n\pi x - m\pi x) - \frac{1}{n\pi + m\pi}\int_0^1 \cos(u)\ du, \\
&= \int_0^1 \cos(n\pi x - m\pi x) - \frac{\sin(u)}{n\pi + m\pi}\Big|_0^1.
\end{aligned}$$

Now looking at the first part of the integral, we are going to use another u-substitution. Let $s = x(n\pi - m\pi)$ and $ds = n\pi - m\pi dx$ to obtain,

$$= \frac{1}{n\pi - m\pi}\int_0^1 \cos(s)ds - \frac{\sin(u)}{n\pi + m\pi}\Big|_0^1,$$

$$= \frac{\sin(s)}{n\pi - m\pi} - \frac{\sin(u)}{n\pi + m\pi}\Big|_0^1.$$

Completing our substitution we get,

$$= \frac{\sin(xn\pi - xm\pi)}{\pi(n-m)} - \frac{\sin(xn\pi + xm\pi)}{\pi(n+m)}\Big|_0^1.$$

Finding a common denominator we have,

$$= \frac{(n+m)\sin(xn\pi - xm\pi) - (n-m)\sin(xn\pi + xm\pi)}{\pi(n^2 - m^2)}\Big|_0^1$$

$$= \frac{n\sin(xn\pi - xm\pi) + m\sin(xn\pi - xm\pi) - [n\sin(xn\pi + xm\pi) - m\sin(xn\pi + xm\pi)}{\pi(n^2 - m^2)}\Big|_0^1$$

$$= \frac{(n+m)[\sin(xn\pi - xm\pi)] - (n+m)[\sin(xn\pi + xm\pi)]}{\pi(n^2 - m^2)}\Big|_0^1$$

$$= 0.$$

Lets look at the case where $n = m$. We have the following,

$$\begin{aligned} 2\int_0^1 \sin^2(n\pi x)\ dx &= 2\int_0^1 \frac{1}{2}[1-\cos(2m\pi x)]\ dx \\ &= \int_0^1 1-\cos(2m\pi x)\ dx \\ &= \int_0^1 1\ dx - \int_0^1 \cos(2m\pi x)\ dx \\ &= x - \frac{\sin(2m\pi x)}{2m\pi}\Big|_0^1. \end{aligned}$$

From which we deduce,

$$1 - \frac{\sin(2m\pi)}{2m\pi} = 1,$$

where $m \in \mathbb{Z}$, and $m \neq 0$. Hence we conclude,

$$2\int_0^1 \sin^2(n\pi x)dx = 0.$$

Since we have shown both cases when $n = m$ and $n \neq m$, we can conclude that $\varphi_n(x)x = \sqrt{2}\sin(n\pi x)$ is orthogonal with respect to the L^2 inner product. $\square$

Here are some special identities that we encountered when we started to discretized the control problem, which are mentioned below.

Observe this integral quantity $\int_0^1 \varphi_i'(x)\varphi_j'(x)dx$. We are going to look at the case when $i = j$, where $i, j \in \mathbb{Z}^+$. We turn our attention to,

$$\begin{aligned}
\int_0^1 \sqrt{2}j\pi\cos(i\pi x)\sqrt{2}i\pi\cos(j\pi x)\ dx &= 2(ij\pi^2)\int_0^1 \cos(i\pi x)\cos(j\pi x)dx \\
&= 2(j\pi)^2\int_0^1 \cos(j\pi x))^2 dx \\
&= 2(j\pi)^2\int_0^1 \frac{\cos(2j\pi x)+1}{2}dx \\
&= \frac{(j\pi)^2\sin(2j\pi x)}{2j\pi} + x\Big|_0^1 \\
&= 1.
\end{aligned}$$

Now lets take a look at the case where $i \neq j$, we have,

$$\begin{aligned}
\int_0^1 \sqrt{2}j\pi\cos(j\pi x)\sqrt{2}i\pi\cos(i\pi x)\ dx &= 2\int_0^1 ij\pi^2\cos(j\pi x)\cos(i\pi x)\ dx \\
&= 2ij\pi^2\int_0^1 \cos(j\pi x)\cos(i\pi x)\ dx \\
&= 2ij\pi^2\int_0^1 \frac{1}{2}(\cos(j\pi x + i\pi x) + \cos(j\pi x - i\pi x))\ dx \\
&= ij\pi^2[\frac{-\sin(j+i)\pi x}{(j+i)\pi} - \frac{\sin(j-i)\pi x}{(j-i)\pi}] \\
&= 0.
\end{aligned}$$

Lastly, looking at the integral $-\int_0^1 \varphi_j(x)\varphi_i(x)\ dx$, it follows immediately from Proposition 3. For completeness, we mention some of the details of the computations. When $i = j$ we have,

$$\begin{aligned}
-\int_0^1 \sqrt{2}\sin(j\pi x)\sqrt{2}\sin(j\pi x)\ dx &= -2\int_0^1 \sin^2(j\pi x)\ dx \\
&= -2\int_0^1 [\frac{1-\cos(2j\pi x)}{2}]\ dx \\
&= -1\int_0^1 [1-\cos(2j\pi x)]\ dx \\
&= -1[1-\frac{\sin(2j\pi x)}{j\pi}\Big|_0^1 \\
&= -1.
\end{aligned}$$

When $i \neq j$ we have the following,

$$\begin{aligned}
-\int_0^1 \sqrt{2}\sin(j\pi x)\sqrt{2}\sin(i\pi x)dx &= -2\int_0^1 \sin(j\pi x)\sin(i\pi x)dx \\
&= -2\int_0^1 [\frac{1}{2}\cos((j-i)\pi x)-\cos((j+i)\pi x)]dx \\
&= -1\int_0^1 \cos((j-i)\pi x)-\cos((j+i)\pi x)dx \\
&= -1[\frac{\sin(j-i)\pi x}{(j-i)\pi}-\frac{\sin(j+i)\pi x}{(j+i)\pi}]\Big|_0^1 \\
&= 0.
\end{aligned}$$

3.1. **Discretization of the Control problem.** In order to begin the discretization lets recall a family of test functions that we mentioned earlier in this section, $G = \{\varphi_j\}$ such that $\varphi(0) = \varphi(1) = 0$ for $j \in \mathbb{Z}$. Looking at (14) and multiply $\varphi(x)$ to each term we obtain,

$$u_*(x,t)\varphi(x) = u_t(x,t)\varphi(x) + u(x,t)u_x(x,t)\varphi(x) - \epsilon u_{xx}(x,t)\varphi(x). \tag{18}$$

Next, we will integrate from 0 to 1 to obtain the following,

$$\int_0^1 u_*(x,t)\varphi(x)dx = \int_0^1 u_t(x,t)\varphi(x) + u(x,t)u_x(x,t)\varphi(x)dx - \int_0^1 \epsilon u_{xx}(x,t)\varphi(x)dx. \tag{19}$$

Remark 2. *We are strictly working with the domain form* $[0,1]$ *in the case of traffic we can think of* $[0,1]$ *as the start of the freeway to the end of the freeway.*

We will now separate (19) into three separate parts. For convenience, we label these integrals as the following,

(A) $\displaystyle\int_0^1 u_*(x,t)\varphi(x)\ dx$

(B) $\displaystyle\int_0^1 u_t(x,t)\varphi(x) + u(x,t)u_x(x,t)\varphi(x)\ dx$

(C) $\displaystyle -\epsilon\int_0^1 u_{xx}(x,t)\varphi(x)\ dx.$

Lets take a look at (B) we have,

$$\int_0^1 u_t(x,t)\varphi(x) + u(x,t)u_x(x,t)\varphi(x)dx = \int_0^1 u_t\varphi(x)dx + \int_0^1 \frac{\partial}{\partial x}\Big(\frac{u(x,t)^2}{2}\Big)\varphi(x)\ dx$$
$$= \frac{d}{dt}\left\{\int_0^1 u(x,t)\varphi(x)dx\right\} + \frac{u(x,t)^2}{2}\varphi(x)\Big|_0^1$$
$$- \int_0^1 \frac{u(x,t)^2}{2}\varphi'(x)\ dx.$$

Looking at (C) and using integration by parts we have,

$$\begin{aligned}
\text{-}\epsilon \textstyle\int_0^1 u_{xx}(x,t)\varphi(x)\ dx &= -\epsilon\left(u_x(x,t)\varphi(x)\Big|_0^1 - \int_0^1 u_x(x,t)\varphi'(x)\ dx\right) \\
&= \epsilon\int_0^1 u_x(x,t)\varphi'(x)\ dx \\
&= \epsilon\int_0^1 \frac{\partial}{\partial x}u(x,t)\frac{d}{dx}\varphi(x)\ dx.
\end{aligned} \tag{20}$$

Looking at (20) and noting the vanishing behavior at the boundary we have,

$$-\epsilon\int_0^1 u_{xx}(x,t)\varphi(x)\ dx = \epsilon\int_0^1 \frac{\partial}{\partial x}u(x,t)\frac{d}{dx}\varphi(x)\ dx.$$

Here is a brief explanation on how we obtain the right hand side of the integral. We first apply the chain rule and lastly do integration by parts together with the properties of G that ensures the boundary terms to disappear.

Substituting (A), (B), and (C) to (19) we have,

(21)
$$\frac{d}{dt}\int_0^1 u(x,t)\varphi(x)dx - \int_0^1 \frac{u(x,t)^2}{2}\varphi'(x)\ dx + \epsilon\int_0^1 \frac{\partial}{\partial x}u(x,t)\frac{d}{dx}\varphi(x)dx = \int_0^1 u_*(x,t)\varphi(x)dx.$$

Moving forward we take a finite sub-collection of G denoted $\Phi_N = \{\varphi_1, \ldots, \varphi_N\}$. We will approximate u by u_h and u_* by functions of the form,

$$u_h(x,t) = \sum_{j=1}^{N} u_j(t)\varphi_j(x), \tag{22}$$

$$u_*(x,t) = \sum_{j=1}^{N} u_{*j}(t)\varphi_j(x). \tag{23}$$

Recall, $\varphi_j(x) = \sqrt{2}\sin(j\pi x)$. We set $\vec{u}(t) = [u_1(t), u_2(t), ..., u_N(t)]$, and $\vec{u}_*(t) = [u_1(t), u_2(t), ..., u_n(t)]$ where $i = 1, 2, 3, ...N$. Then, $\varphi_1(x) = \sqrt{2}\sin(\pi x)$, $\varphi_2(x) = \sqrt{2}\sin(2\pi x), ..., \varphi_N(x) = \sqrt{2}\sin(N\pi x)$.

We fix a member of $\{\Phi_N\}$, φ_i, for i fixed. Typically, the functions are called *test functions* and we may refer to them as such for the remainder of this work.

Looking at (C) we are now going to substitute the projection and fix the index i to obtain the following,

$$\begin{aligned}\epsilon\int_0^1 \frac{\partial}{\partial x}(\sum_{j=1}^N u_j(t)\varphi_j(x))\varphi_i'(x)dx &= \epsilon\int_0^1 \frac{\partial}{\partial x}(u_1(t)\varphi_1(x)+....+u_N(t)\varphi_N(x))\varphi_i'(x)dx \\ &= \epsilon\int_0^1 (\sum_{j=1}^N u_j(t)\varphi_j'(x))\varphi_i'(x)dx.\end{aligned}$$

We define the matrix,

$$(A_h)_{ij} = \int_0^1 \varphi_i'\varphi_j' \, dx.$$

With this notation presented we write (C) as,

$$(A_h)_{ij}\ (\vec{u}) = \epsilon\sum_{j=1}^N u_j(t)\int_0^1 \varphi_j'(x)\varphi_i'(x)dx.$$

A question that we can ask ourselves is what are the best choices for the coefficients $u_1(t), ...u_n(t)$. Working with (21) and plugging the finite projections into the first part of (B) we have the coefficients in the expansion of (22) satisfying,

$$
(24) \qquad \frac{d}{dt}\int_0^1 u_h(x,t)\varphi_i(x)dx = \frac{d}{dt}\int_0^1 \Big(\sum_{j=1}^{N} u_j(t)\varphi_j(x)\Big)\varphi_i(x)dx.
$$

All we did was substitute what the projection of $u_h(x,t)$ is to (24). When, $i = 1$ we have that,

$$
\frac{d}{dt}\int_0^1 u_1(t)(\varphi_1(x))^2 dx = \frac{d}{dt}u_1(t).
$$

Then when $i = 2$ we have,

$$
\frac{d}{dt}\int_0^1 u_2(t)(\varphi_2(x))^2 dx = \frac{d}{dt}u_2(t).
$$

For each i we have that, $i = n$,

$$
\frac{d}{dt}\int_0^1 u_n(t)(\varphi_n(x))^2 dx = \frac{d}{dt}u_n(t).
$$

We define the matrix,

$$
(M_h)_{ij} = \int_0^1 \varphi_i(x)\varphi_j(x)dx.
$$

With the notation above we can rewrite (24) as,

$$(M_h)_{ij}\frac{d}{dt}(\vec{u}) = \frac{d}{dt}\sum_{j=1}^{N} u_j(t)\int_0^1 \varphi_i(x)\varphi_j(x)dx,$$

where $\frac{d}{dt}\ \vec{u}(t) = [\frac{d}{dt}u_1(t), \frac{d}{dt}u_2(t), ..., \frac{d}{dt}u_n(t)]^T$.

At the second part of (B), we are now going to substitute the projection and fix index i to obtain the following,

(25)
$$\begin{aligned}\int_0^1 u(x,t)u_x(x,t)\varphi(x)\ dx &= \int_0^1 \frac{\partial}{\partial_x}\Big(\frac{u(x,t)^2}{2}\Big)\varphi(x)\ dx\\ &= -\frac{1}{2}\int_0^1 \Big(\sum_{j=1}^{N} u_j\varphi_j\Big)^2\varphi_i'(x)\ dx.\end{aligned}$$

Remark 3. *The following identity is noted,* $\Big(\sum_{j=1}^N u_j\varphi_j\Big)^2 = \sum_{j=1}^N\sum_{k=1}^N\Big(u_j(t)u_k(t)\Big)$.

We can pull $\sum_{j=1}^N\sum_{k=1}^N\Big(u_j(t)u_k(t)\Big)$ because it is in terms of t and not in terms of x to have the following,

$$\frac{1}{2}\sum_{j=1}^{N}\sum_{k=1}^{N}\Big(u_j(t)u_k(t)\Big)\int_0^1 \frac{d}{dx}\varphi_i(x)\varphi_j(x)\varphi_k(x).$$

Looking at (25) we have,

(26)
$$\int_0^1 u(x,t)u_x(x,t)\varphi(x)\ dx = \frac{1}{2}\sum_{j=1}^{N}\sum_{k=1}^{N}\Big(u_j(t)u_k(t)\Big)\int_0^1 \frac{d}{dx}\varphi_i(x)\varphi_j(x)\varphi_k(x)\ dx.$$

We can define the vector (only depends on i),

$$(N_h(\vec{u}(t)))_i = \sum_{j=1}^{N} \sum_{k=1}^{N} \int_0^1 \frac{d}{dx} \varphi_i(x) \varphi_k(x) \varphi_j(x) \; dx \Big(u_k(t) u_j(t) \Big).$$

That is equation (26) can be expressed as $\Big(N_h(\vec{u}(t)) \Big)_i$ for each $i = 1, \cdots, N$.

We are going to approximate (23). That is, we substitute the following integral,

$$\int_0^1 u_*(x,t) \varphi_i(x) dx,$$

with its projection, where $i = 1, 2, 3...$ to obtain,

$$\int_0^1 \Big(\sum_{j=1}^{N} u_{*j}(t) \varphi_j(x) \Big) \varphi_i(x) \; dx. \tag{27}$$

We can pull $\sum_{j=1}^{N} u_{*j}(t)$ because it is in terms of t and not in terms of x to have the following,

$$\sum_{j=1}^{N} u_{*j}(t) \int_0^1 \varphi_j(x) \varphi_i(x) \; dx.$$

We can define the matrix to be,

$$(B_h)_{ij} = - \int_0^1 \varphi_j(x) \varphi_i(x) \; dx,$$

where $\vec{u}_*(t) = [u_{1*}(t), \cdots, u_{N*}(t)]$, with this notation (27) can be expressed by $B_h \vec{u}_*(t)$.

Putting all matrices together we obtain a system of ordinary differential equations. Hence,

$$M_h \frac{d}{dt}\vec{u}(t) + A_h \vec{u}(t) + N_h(\vec{u}(t)) + B_h \vec{u_*}(t) = 0,$$

where $M_h, A_h \in \mathbb{R}^{(n-1)\times(n-1)}, B_h \in \mathbb{R}^{(n-1)\times(n+1)}$ and $N_h(\vec{u}(t)) \in \mathbb{R}^{n-1}$ are the matrices and $t \in (0,1)$.

Inserting the approximations (22), (23) into (13) and performing calculations like those applied to terms (A), (B), (C), we arrive at the semi-discretization given by,

(28)

$$\min_{u_*} \int_0^T \frac{1}{2}\vec{u}(t)^T M_h \vec{u}(t) + (g_h(t))^T \vec{u}(t) + \frac{C}{2}\vec{u_*}(t)^T Q_h \vec{u_*}(t)\ dt + \int_0^T \int_0^1 \frac{1}{2} u_d^2(x,t)\ dxdt.$$

The matrix and vector entries are the following,

$$(Q_h)_{ij} = \int_0^1 \varphi_j(x)\varphi_i(x)\ dx$$

$$(g_h(t))_i = -\int_0^1 u_d(x,t)\varphi_i(x)\ dx.$$

Since $u_d^2 > 0$, we are interested mainly in the following,

$$\min_{u_*} \int_0^T \frac{1}{2}\vec{u}(t)^T M_h \vec{u}(t) + (g_h(t))^T \vec{u}(t) + \frac{C}{2}\vec{u_*}(t)^T Q_h \vec{u_*}(t)\ dt,$$

where $\vec{u}(t)$ is the solution of to,

$$\begin{cases} M_h \frac{d}{dt}\vec{u}(t) + A_h \vec{u}(t) + N_h(\vec{u}(t)) + B_h \vec{u_*}(t) = 0,\ t \in (0,1) \\ \vec{u}(0) = \vec{u_0} \end{cases}$$

and $\vec{u_0} = (u_0(h), \ldots, u_0(1-h))^T$, and $h > 0$ determines the domain the step size.

In order to approximate the entries to $(g_h(t))_i$ we apply the composite trapezoid rule. This is a numerical algorithm that approximates a definite integral since u_d and φ_i are known functions; recall (13) and (16). See [2].

Below we state the fully discretized problem which is given by,

$$(29) \qquad \min_{\vec{u}_{*0},\cdots,\vec{u}_{*N+1}} \sum_{i=0}^{N+1} \frac{\Delta t_{i-1} + \Delta t_i}{2} \left(\frac{1}{2}\vec{u}_i^T u_h^T \vec{u}_i^T + (g_h)_i^T \vec{u}_i + \frac{C}{2}\vec{u}_i^T Q_h \vec{u}_{*i} \right),$$

where $\vec{u}_1, \cdots, \vec{u}_{N+1}$ is a solution to,

(30)

$$\left(M_h + \frac{\Delta t_i}{2} A_h\right)\vec{u}_{i+1} + \frac{\Delta t_i}{2} N_h(\vec{u}_{*i+1}) + \frac{\Delta t_i}{2} B_h \vec{u}_{*i+1} + (-M_h + \frac{\Delta t_i}{2} A_h)\vec{u}_i \cdots + \frac{\Delta t_i}{2} N_h(\vec{u}_i) + \frac{\Delta t_i}{2} B_h \vec{u}_{*i} = 0$$

$i = 0, \cdots, N$ and $\vec{u}_0$ is given.

For more details on the fully discretization problem you may look at [1]. We direct your attention to the conjugate gradient method that is very useful and important. It is what we used to discretized our problem. On the next section, we will be mentioning the conjugate gradient method and how it is so useful.

3.2. **Conjugate Gradient Method.** The purpose of this section is to be able to understand the Conjugate Gradient Method. Once we understand this method, the connection between finite dimensional space and MATLAB program will be a lot easier.

The conjugate gradient method was originally developed as a direct to solve $n \times n$ positive definite linear system. The conjugate gradient method is useful when employed as an iterative approximation method for solving large sparse system with nonzero entries occurring in predictable patterns.

We will use the inner product notation as

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x}^t \boldsymbol{y}, \tag{31}$$

$\boldsymbol{x}$ and $\boldsymbol{y}$ are n-dimensional vectors.

Theorem 4. *The vector $\boldsymbol{x}^*$ is a solution to the positive definite linear $A\boldsymbol{x} = \boldsymbol{b}$ if and only if $\boldsymbol{x}^*$ produces the minimal value of $g(x) = \langle \boldsymbol{x}, A\boldsymbol{x} \rangle - 2\langle \boldsymbol{x}, \boldsymbol{b} \rangle$.*

Proof. To begin, $\boldsymbol{x}$ must be chosen. This $\boldsymbol{x}$ is an approximate solution to $A\boldsymbol{x}^* = \boldsymbol{b}$, and $\boldsymbol{x} \neq 0$. Let $\boldsymbol{m} = \boldsymbol{b} - A\boldsymbol{x}$ be the residue vector associated with $\boldsymbol{x}$ and

$$t = \frac{\langle \boldsymbol{v}, \boldsymbol{b} - A\boldsymbol{x} \rangle}{\langle \boldsymbol{v}, A\boldsymbol{v} \rangle}.$$

If $\boldsymbol{m} \neq 0$ and if $\boldsymbol{v}$ and $\boldsymbol{m}$ are not orthogonal, then $\boldsymbol{x} + m\boldsymbol{v}$ gives a smaller value for g than $g(x)$, hence it is closer to $\boldsymbol{x}^*$ than $\boldsymbol{x}$.

Thus the following method is used. Let $\boldsymbol{x}^{(0)}$ be an initial approximation to $\boldsymbol{x}^*$, and also let $\boldsymbol{v}^{(1)} \neq 0$ be an initial direction. When $k = 1, 2, 3, \cdots$, we compute,

$$t_k = \frac{\langle \boldsymbol{v}^{(k)}, \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(k-1)} \rangle}{\langle \boldsymbol{v}^{(k)}, A\boldsymbol{v}^{(k)} \rangle}$$

$$\boldsymbol{x}^{(k)} = \boldsymbol{x}^{(k-1)} + t_k \boldsymbol{v}^{(k)}$$

and choose a new search direction for $\boldsymbol{v}^{(\boldsymbol{k}+\mathbf{1})}$. The objective is to make this selection so that the sequence of approximations $\{\boldsymbol{x}^{(\boldsymbol{k})}\}$ converges rapidly to $\boldsymbol{x}^{*}$. To choose the search direction, we are going to view g as a function of the components of $\boldsymbol{x} = (x_1, x_2, x_3, \cdots, x_n)^t$. Thus,

$$\begin{aligned} g(x_1, x_2, x_3, \cdots, x_n) &= \langle \boldsymbol{x}, A\boldsymbol{x} \rangle - 2\langle \boldsymbol{x}, \boldsymbol{b} \rangle \\ &= \sum_{i=1}^{N} \sum_{j=1}^{N} a_i x_i x_j - 2 \sum_{i=1}^{N} x_i b_i. \end{aligned}$$

Taking partial derivatives with respect to x_k gives us,

$$\frac{\partial g}{\partial x_k}(x) = 2 \sum_{i=1}^{N} a_{k_i} x_i - 2b_k,$$

which is the k^{th} component of the vector $2(A\boldsymbol{x} - \boldsymbol{b})$. Hence, the gradient of g is

$$\begin{aligned} \nabla g(x) &= (\frac{\partial g}{\partial x_1}(x), \frac{\partial g}{\partial x_2}(x), \cdots, \frac{\partial g}{\partial x_n}(x))^t \\ &= 2(\boldsymbol{x} - \boldsymbol{b}) \\ &= -2\boldsymbol{m}. \end{aligned}$$

Recall that the vector $\boldsymbol{m}$ is the residual vector for $\boldsymbol{x}$. The direction of the residual $\boldsymbol{m}$, is the direction given by $-\nabla g(x)$. We know this by calculus, as the direction of greatest decrease, the value of $g(x)$. We will use the method of steepest descent which is,

$$\boldsymbol{v}^{(k+1)} = \boldsymbol{m}^{(k)} = \boldsymbol{b} - A\boldsymbol{x}^{(k)}.$$

An alternative approach can be to use a set of nonzero direction vectors $\{\boldsymbol{v}^{(1)}, \boldsymbol{v}^{(2)}, \cdots, \boldsymbol{v}^{(n)}\}$ that satisfy $\langle \boldsymbol{v}^{(i)}, A\boldsymbol{v}^{(j)} \rangle = 0$ if $i \neq j$. This is called the A-orthogonality condition and the set of vectors $\{\boldsymbol{v}^{(1)}, \boldsymbol{v}^{(2)}, \cdots, \boldsymbol{v}^{(n)}\}$ is said to be A-orthogonal. This is the set of search directions gives us,

$$t_k = \frac{\langle \boldsymbol{v}^{(k)}, \boldsymbol{b} - A\boldsymbol{x}^{(k-1)} \rangle}{\langle \boldsymbol{v}^{(k)}, A\boldsymbol{v}^{(k)} \rangle} = \frac{\langle \boldsymbol{v}^{(k)}, \boldsymbol{r}^{(k-1)} \rangle}{\langle \boldsymbol{v}^{(k)}, A\boldsymbol{v}^{(k)} \rangle},$$

and $\boldsymbol{x}^{(k)} = \boldsymbol{x}^{(k-1)} + t_k \boldsymbol{v}^{(k)}$. This concludes the proof of this theorem. $\square$

Remark 4. *This theorem shows the choice of search directions provides convergence and also can be written as matrix multiplication just like in the section of discretization.*

For more details on the conjugate gradient method the, reader is referred to definition 7. We now have arrive at the end the section where lastly, we will be talking about an algorithm that we implemented in MATLAB using a black box.

(1) We first start the main function named 'optimization driver' and that sets the problem data.

(2) We now go to 'problem generator' finite matrices that we have derived in chapter 2 are constructed.

(3) Once 'problem generator' is done, we go to 'derivative checks' where it uses global variables to help solve the minimization problem.

(4) 'X new' is where the state (solution to the PDE) is being computed and it approximates solutions to fully discretized Burgers equation.

(5) 'F val' is where the discretized problem is being fully solved and it also performs the optimization.

(6) Running results from 'optimization driver' gives us nice images where we can analyze them and draw conclusions.

3.3. **Overview of Numerical Methods of Numerical Challenges.** The code was originally written by Mathias Henkenschloss [4, 5] and uses the conjugate gradient method from the previous chapter. The figures and images in the next section that the code generates are approximate solutions to the minimization of the optimal control problem that we discused throughout this chapter. Understanding Henkenschloss' code proved to be very challenging. The code calls upon MATLAB files which are exactly twenty-five files in order to compile.

Thus, in order to complete the goal of alternating the code, we use different desired states, initial conditions, and the traffic data collected through PeMS. A combination of both trial and error were used as well as simple examples by hand.

Modifications to the code are now discussed. We choose different types of functions as well as use the data that we collected through PeMS to set initial and desired states. The term Nx represents the number of spatial intervals used in the discretization, which is the spatial mesh points. The term Nt is the time steps intervals that are also used in the discretization. Viscosity, mu is a parameter that is constantly used throughout this set of code, and $Omega$ is a penalty parameter for the control.

Contributions made to the code are presented below. The following was added to the prob-gen.m file, and determines the desired state.

```
BURGERS_GLB.g            = zeros(nx-1,nt+1);

nx2                      = floor(nx-1);
%BURGERS_GLB.g(1:nx2,:)  = -BURGERS_GLB.Deltax*ones(nx2,nt+1);

%'z' represents different function that I
%The following are the desired states known as u_d.

x                         = 0:1/(nx-1):1;
%z                         = sin(2*pi*x);
z                         = 2*x+4;
%z                         = log(x+5)
%z                         = cos(4*pi*x)
%z                         = exp(x)

%The following below is data collected through PeMS
```

```
%filename                    ='uTrafficData.xlsx';
%xlRange                     ='A3:U3';
%M                           =xlsread(filename,xlRange);
Z                          =z(1:nx2);
for i=1:nx
    BURGERS_GLB.g(1:nx2,i) = -BURGERS_GLB.Deltax*Z; %change nx-1 ↩
        from nx2
end
```

The following was added to the state.m file and determines the initial state. This set of code represents the initial state of different functions, as well as, the information regarding traffic that was collected through PeMS. This code is from a state file that uses variables u and *user parameter*. What *user parameter* does is that it is a defined parameter that is used to pass problem specific information.

```
% initial time: This is where you put different functions.
  rhs          = zeros(nx-1,1);
  nx2          = floor(nx-1);
  %rhs(1:nx2) = ones(nx2,1);
  %y(:,1)      = rhs;

  %Below are functions that represent different initial states.

  x            =0:1/nx:1;
  %Below is the data collected through PeMS.

  %filename    ='uTrafficData.xlsx';
  %xlRange     ='A590:U590';
  %M           =xlsread(filename,xlRange);

  %Below are different functions.

  %z              =sin(2*pi*x);
  %z               = 2*x+4
  %z               = log(x+5)
  z               = cos(4*pi*x)
  %z               = exp(x)
  rhs(1:nx2)    =z(1:nx2);
  y(:,1)       =rhs;


% y(:,1)       = BURGERS_GLB.M\(Deltax*rhs);
```

In the following chapter we talk about Caltrans Perfomance Measurement System on how data is collected through sensors on the 101 South freeway, as well as, the images that were compiled through MATLAB.

4. Application: Traffic Control from Woodland Hills towards Universal Studios

Caltrans Performance Measurement System known as PeMS is a centralized traffic warehouse that includes data collected through automated detection. It is a system that enables system monitoring and evaluation. Over 37,000 detectors are deployed on urban freeways throughout California. The detectors that are on the freeway measures the number of vehicles (flow or volume) and how long they remain over the detector (occupancy) on a facility for each lane.

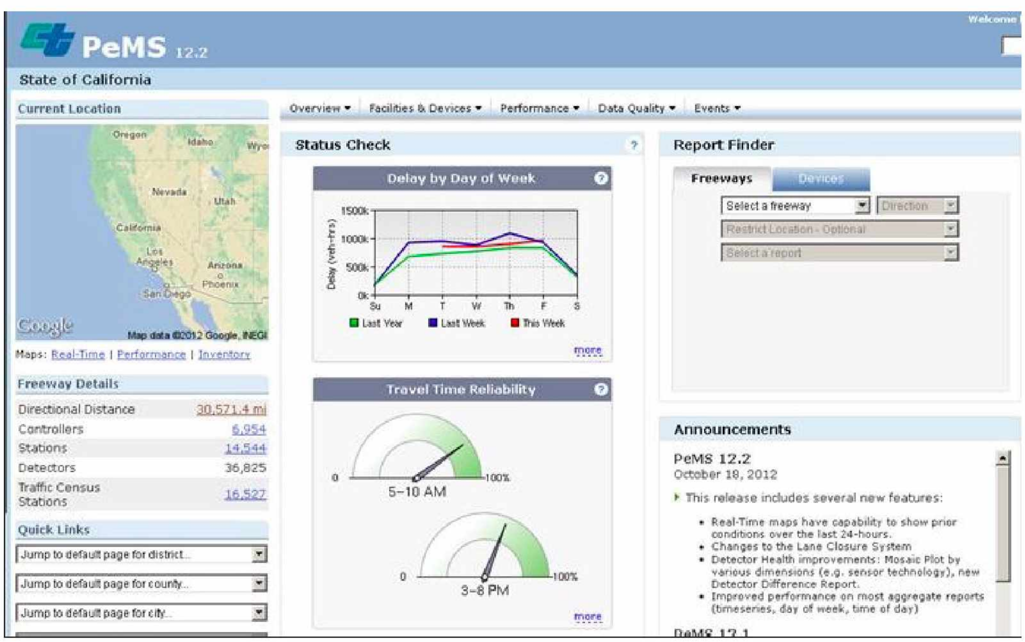

FIGURE 2. PeMS

Detectors on the road measure flow, occupancy, and sometimes speed. The vehicle detector stations are sets of detectors that converts all lanes in one direction of travel, and it monitors one type of facility. There is either on-ramp, off-ramp, or the mainline on the freeway (highway). The collected data that we got through PeMS is represented by the following,

$$\frac{\rho}{\rho_{max}} = \frac{occupancy}{100}. \tag{32}$$

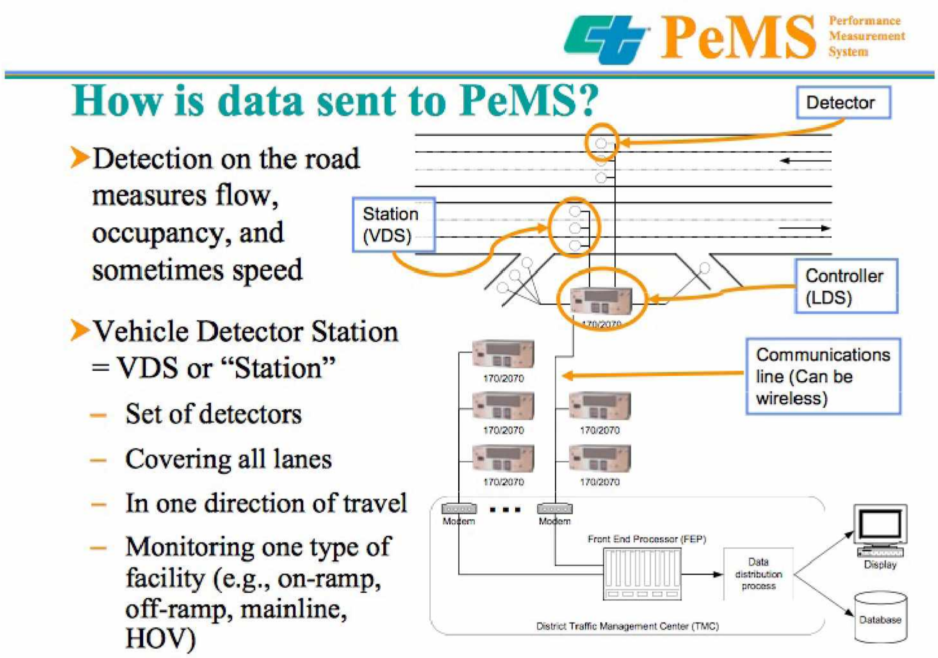

Figure 3. Detectors

PeMS allows planners, engineers, and policy makers to track system performance across most urban freeways and other facilities. Travelers can obtain the current shortest route and travel time estimates. Researchers can validate their theory and collaborate simulation models. PeMS applications are accessed over the world wide web, and custom applications can be worked directly with the PeMS database.

4.1. **Numerical Results (Images).** This section is dedicated to the figures from MATLAB files mentioned from the previous chapter.

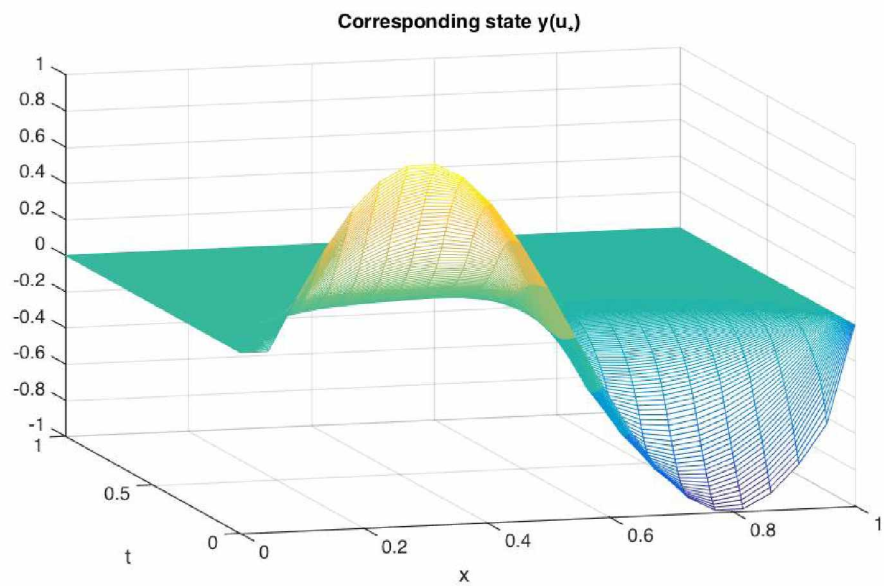

FIGURE 4. Sine Function

Figure 4 represents an approximate solution with viscosity of 0.01, where the sine wave is provided as the initial condition. Our desired state is set to be at 0. Solving the minimization problem having our initial state, desired state, and adding external forces depicts figure 4.

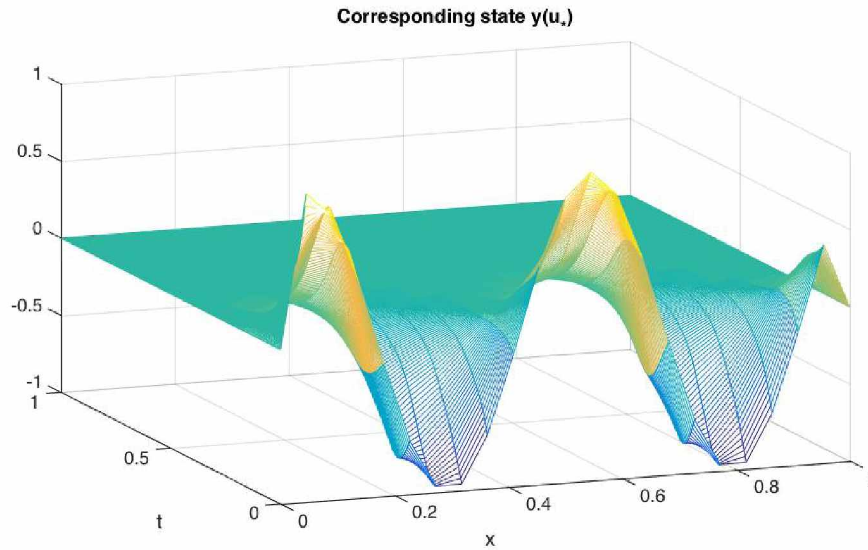

FIGURE 5. Cosine Function

Figure 5 depicts a cosine graph that is interesting. We can observe a few ripples on the image that are getting closer to 0. Figure 5 has viscocity to be set as 0.01. Our initial state is the cosine and our desired state is set to be 0, i.e. $u_0(x) = \cos(4\pi x)$ and $u_d(x,t) = 0$.

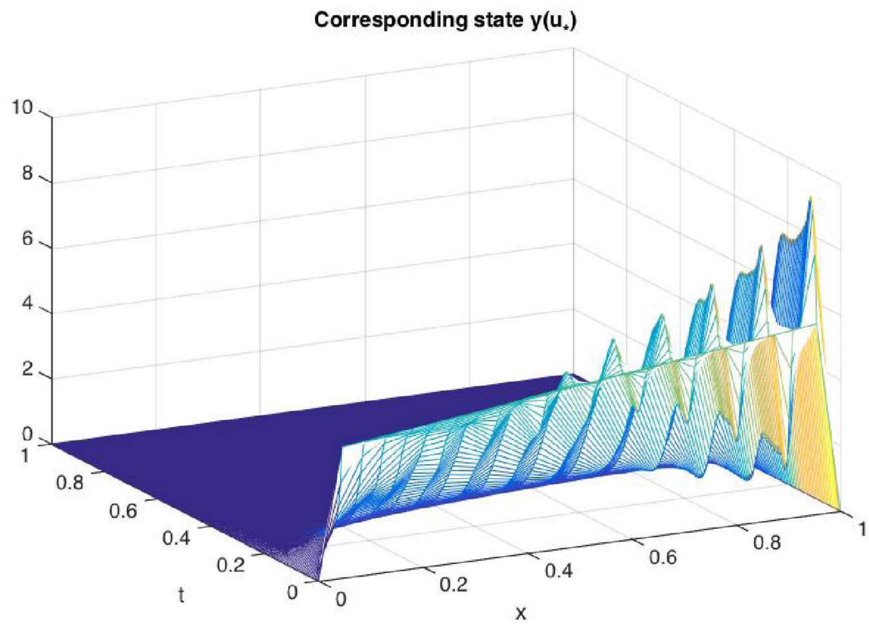

FIGURE 6. linear function

In figure 6, we observe a linear function being controlled. The initial condition is our linear function and our desired state is set to be 0. Specifically we are looking at $2x + 4$ to be the linear function having viscosity of 0.01. Having our initial state, desired state and adding external forces and solving the minimization problem we get a hold of figure 6.

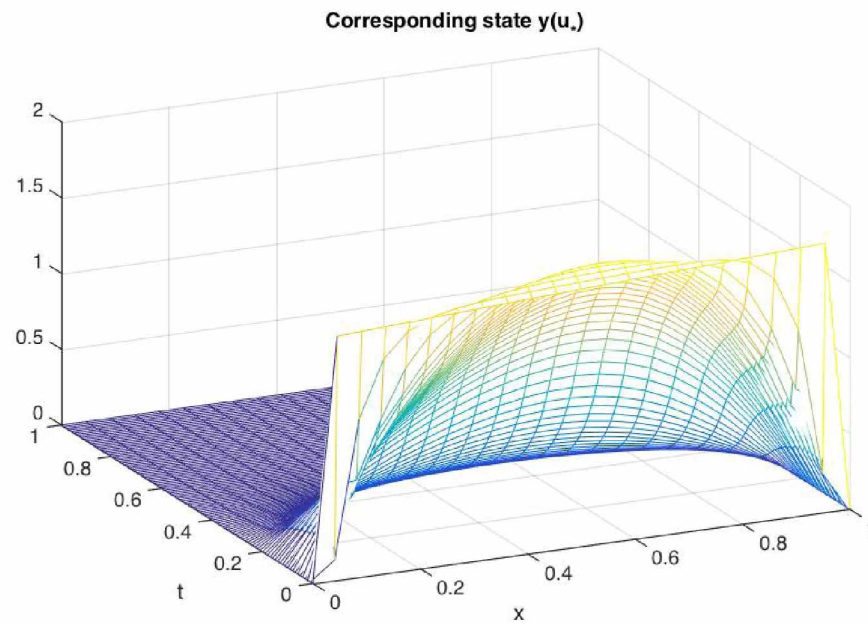

Figure 7. Logarithmic Function

Figure 7 is a logarithmic function that was implemented to the code. In this simulation viscosity is set to be 0.50. Our initial condition is the logarithmic function and our desired state is at 0. The highest point that this function has is at 1. Substituting our initial state, desired state and adding external force to the minimization problem we achieve image 7.

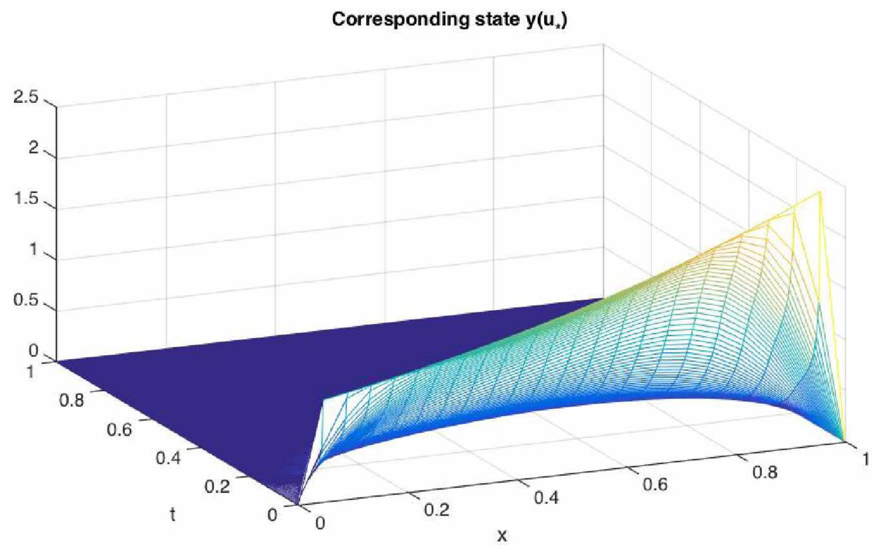

Figure 8. Exponential Function

Figure 8 is an exponential function that is being controlled. Once it reaches its maximum value it starts to decrease and go to 0. The initial state is the exponential function and our desired state is set to be 0. The viscosity of this function is set at 0.01, which is pretty low.

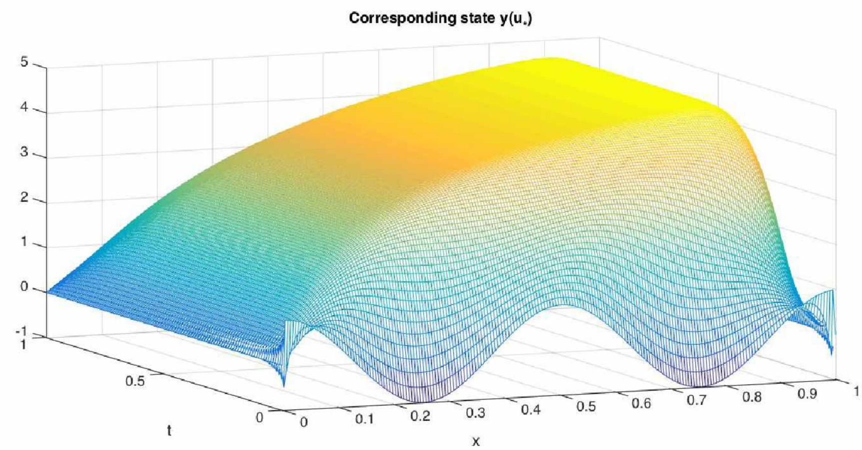

FIGURE 9. Linear Function with Different Desired State

In figure 9 our intial state is set to be $\cos(4\pi x)$, and our desired state is set to be a linear function, $2x + 4$. The viscosity is set to be 0.1. This is an approximation to the minimization optimal control problem that we mentioned in the previous chapters.

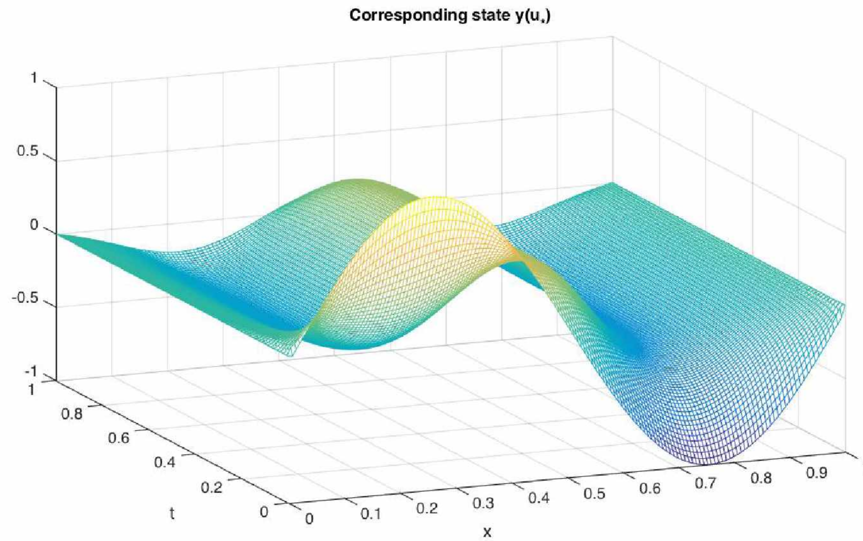

FIGURE 10. Sine Function with Different Desired State

In figure 10 we have the desired state to be $\cos(4\pi x)$ and the initial state to be $\sin(2\pi x)$. The viscosity is set to be 0.1, in this case it appears favorable to select $\mu = 0.1$. Figure 10 represents an approximation to the minimization optimal control problem that was mentioned in the previous chapter.

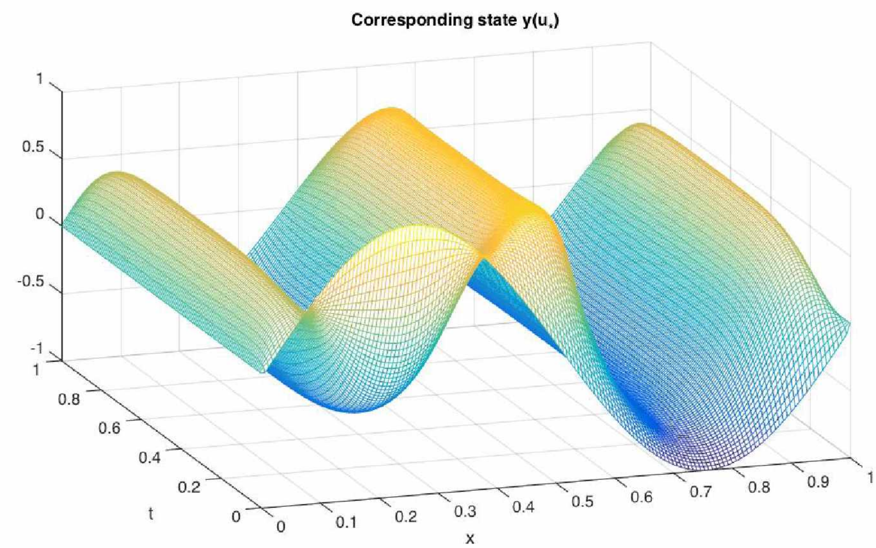

FIGURE 11. Sine Function with Different Desired State and Viscosity

In figure 11 we have the desired state to be the same as figure 10. The intial state is the same as figure 10, which is $\sin(2\pi x)$. The difference between figure 10 and 11 is the viscosity. Recall in figure 10 the viscosity was set to be 0.1, in figure 11 the viscosity is set to be 0.01. In this image we do notice that control was achieved more closely! Figure 11 represents an aproximation to the minimization problem that was mentioned in the previous chapter.

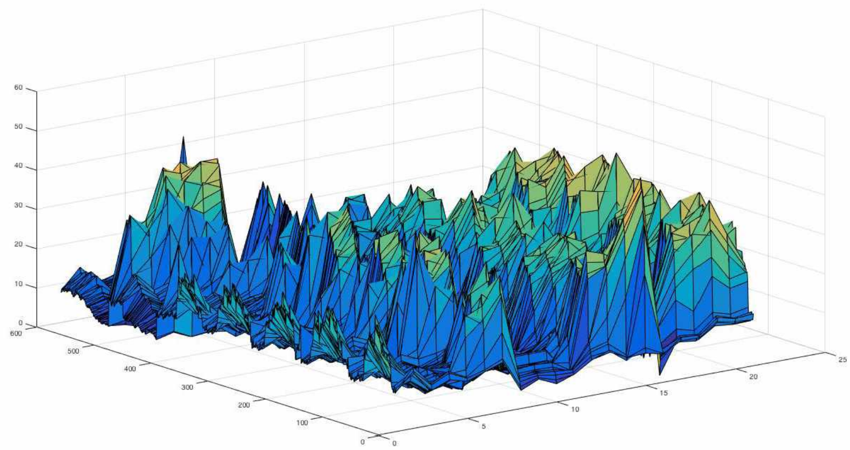

FIGURE 12. Raw Data

The following images are related to traffic. We collected information from Caltrans Performance Management System (PeMS). Once we have the information we implement that information to the code in MATLAB. Again our information is from January 11, 2016 to January 17, 2016. The specific time that we are looking at is from $6am - 1pm$. We are on the 101 South freeway from Woodland Hills to Universal Studios.

Figure 12 is the raw data that we collected from Caltrans Performance Management System. This represents the amount of cars (density) at a given time where there was traffic. The steep peaks represents the amount of traffic there was at specific moment of day and time. Implementing these values into the MATLAB, renders an interesting surface.

Observing figure 12, we can see many peaks that are either going up or down. The peaks that are dramatically increasing means there is traffic on

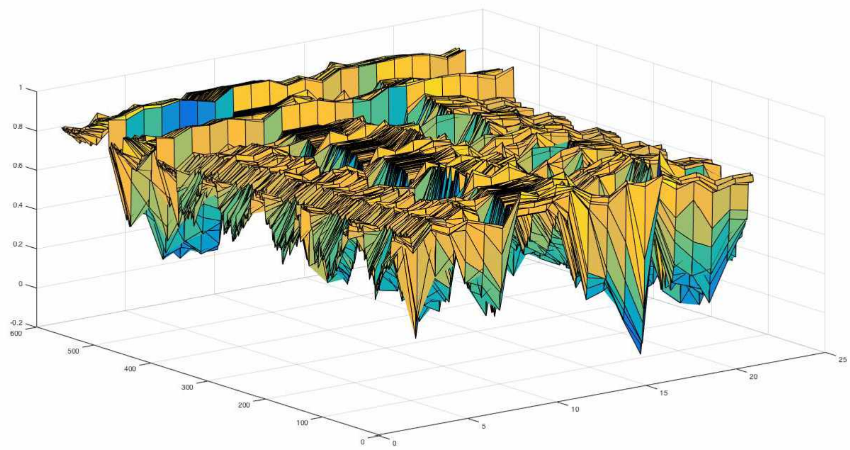

FIGURE 13. u Data

the 101 South freeway, and the reason why is because the occupancy is very high.

Figure 13 looks familiar to figure 12; however, the difference between these images is that image 13 is facing downwards.

Recall in the previous chapter, how we have to do a change of variables (see equation (32)) to our original data and convert it to "u" data. The change of variable that we used is the following,

$$u(x,t) = 1 - 2\frac{\rho}{\rho_{max}}. \tag{33}$$

Observing figure 13 we can see many peaks that are either going up or down. There are a couple of peaks that are dramatically decreasing, which means that at a specific day and time on the 101 South freeway there was traffic.

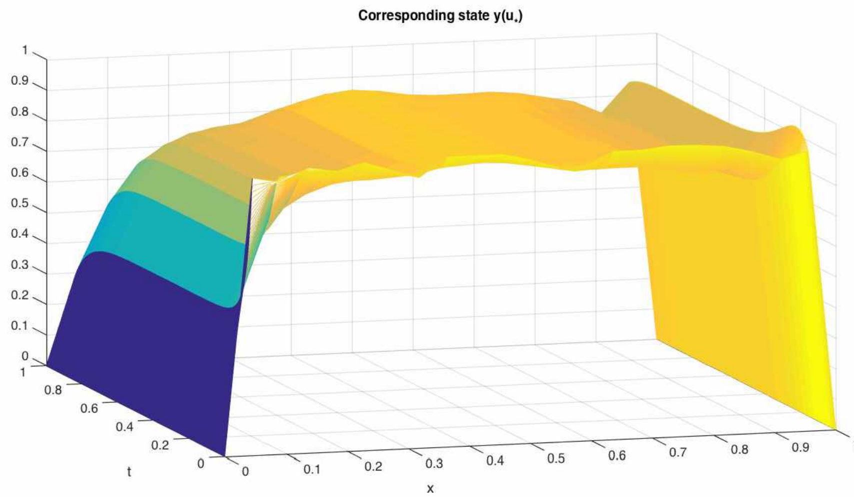

FIGURE 14. Specific State 1

Describing this mathematically means that the density of vehicles at a given day and time was very high were the lowest peaks occur.

Figure 14 represents a specific state of day and time on the 101 South freeway as an initial condition to the optimization problem. We notice on this figure that there are not so many peaks that are dramatically decreasing, than figure 16. We can observe that the density of vehicles on image 14 was very good.

We pick an initial state and a desired state as well as introducing external forces. Once we pick our states we substitute those values into the minimization problem. Once the minimization problem is solved we can conclude traffic conditions. As we see figure 14 we realize that traffic conditions are improved, which means good traffic flow. Mathematically adding external forces to the 101 South will alleviate traffic conditions.

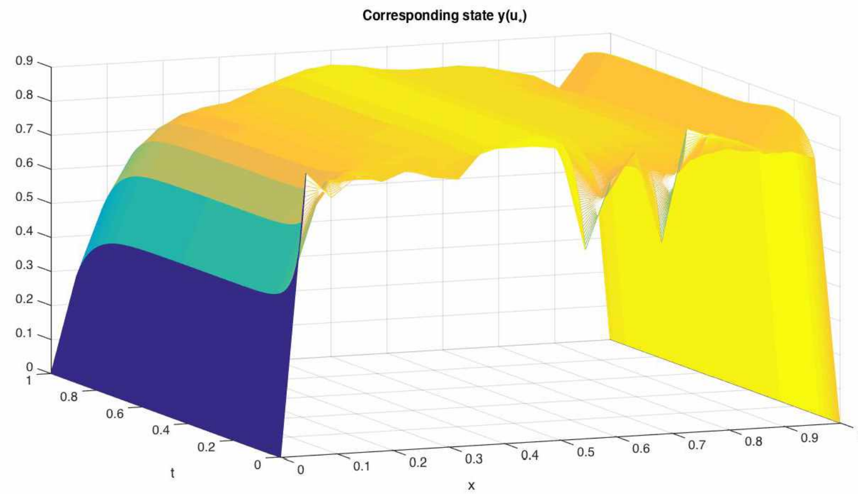

FIGURE 15. Specific State 2

Figure 15 represents a specific state of day and time on the 101 freeway. We observe, that there are a couple of low peaks that are decreasing. However, these peaks are relatively low. This means that a given day and time the density of cars was not so high. Furthermore, there was traffic at those given time, but not like image 16.

We pick an initial state and a desired state and introduce external forces to solve the minimization problem that was stated in the previous chapters. By solving the minimization problem we have figure 15 which we can conclude certain situations.

We can observe that there are some values that are trying to go to -1. What that means is that there is traffic at that given time. Even though we have these peaks on this image we can observe that traffic conditions are alleviated by adding external forces to the freeway.

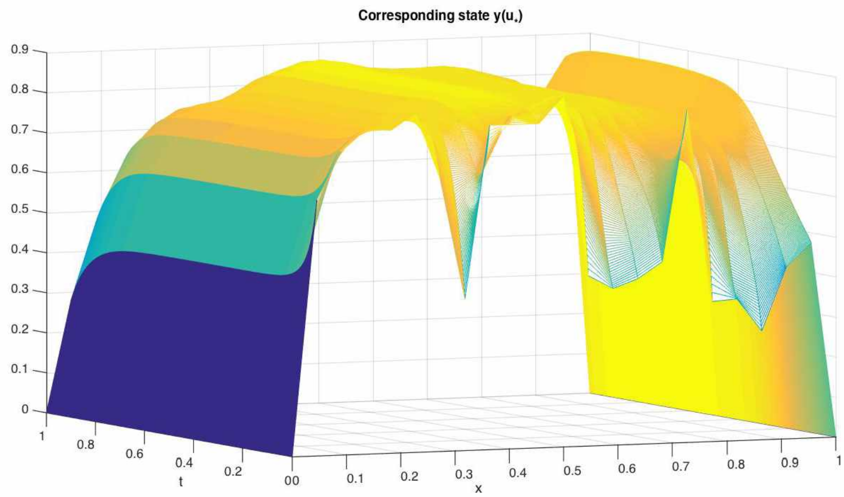

FIGURE 16. Specific State 3

Figure 16 shows an initial condition taken from a specific day and time on the 101 south freeway. We notice that there are a couple of peaks that are pointing downward. This means that at a time and day the traffic was really bad. The density of vehicles on the 101 South freeway were high. Compared to the other peaks that are not as low, the density of cars would be high, meaning there is traffic.

We pick an initial state and a desired state which solves the minimization problem by introducing external forces. Once that is done, we have figure 16 which we can observe that traffic conditions were improved by adding external forces to the 101 freeway.

5. Conclusion and Future Work

This thesis concludes with an overview of what has been accomplished and future goals that can be later pursued.

Everything now a days is being analyzed through an app. Whether its looking for great deals at Target, or browsing to see what movies are out using Fandango. Future work consist of developing an app for traffic flow on cell phones. If someone can make an app specifically for traffic flow, it could be great and beneficial for the people that have to commute to work, school, or have a long road trip plan.

We can also consider other applications where Burgers equation is used as a model for gas dynamics, fluid flow and other applications. Moreover, working with an engineer that can design and interpret external forces in a physical way to monitor traffic flow would be interesting to work with.

There are many partial differential equations out there. It will be interesting if future students can work with optimization problems adapted to other PDEs. It would be interesting to see what numerical solutions they come to and the images they would render using MATLAB.

References

[1] Nick Desautels. *Numerical Methods for Burgers Equation.* California State University Channel Islands (Master's Thesis).

[2] Burden R., and Faires J., Numerical Anaylsis. 9th ed. Boston, MA: *Brooks/Cole,* Centering Learning, 2011. Print.

[3] Burkard, J., "MATLAB Source Codes." MATLAB Source Codes. N.P., n. d. Web. Jan 2016.

[4] Heinkenschloss, M., "MATLAB Codes for Implicitly Constrained Optimization Problems." MATLAB Codes for Implicitly Constrained Optimization Problems. Rice University. 2008. Web. July 2016.,

[5] Heinkenschloss, M., Numerical Solution of Implicitly Constrained Optimization Problems.

[6] Kreyszig, E., Introductory Functional Analysis with Applications. Willey Classics Library ed. New York: Wiley, 1989. Print.

[7] LeVeque, R., Numerical Methods for Conservation Laws. Basel: Birkhuser Verlag, 1992. Print.

[8] Nocedal J., and Wright S., Numerical Optimization. 2nd ed. New York: Springer, 2006. Print.

[9] Stewart, J., Calculus: Early Transcendentals. Belmont, CA: Thomson Books/-Cole, 2008. Print.