COMPUTATIONAL METHODS FOR INVESTIGATING INTRADRIVER HETEROGENEITY USING VEHICLE TRAJECTORY DATA

by

Jeffrey D. Taylor

A thesis submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Civil and Environmental Engineering

The University of Utah

August 2014

The University of Utah Graduate School

STATEMENT OF THESIS APPROVAL

The thesis of **Jeffrey D. Taylor**

has been approved by the following supervisory committee members:

Richard Jon Porter , Chair **06/12/2014**
Date Approved

Xuesong Zhou , Member **06/11/2014**
Date Approved

Tatjana Jevremovic , Member **06/12/2014**
Date Approved

and by **Michael Barber** , Chair/Dean of

the Department/College/School of **Civil and Environmental Engineering**

and by David B. Kieda, Dean of The Graduate School.

ABSTRACT

Traffic simulations, which attempt to describe how individual vehicles move on road segments in a network, rely on mathematical traffic flow models developed from empirical vehicle trajectory data (position, speed, acceleration, etc.). Many of these microscopic traffic flow models are described as car-following models, which assume that a driver will respond to the actions of the driver/s or vehicle/s located in front of them (stimulus-response behavior). Model calibration can be performed using regression and/or optimization techniques, but the process is often complicated by uncertainty and variation in human behavior, which can be described as driver heterogeneity.

Driver heterogeneity is conceptually based on the idea that different drivers may have different reactions to the same stimuli (interdriver heterogeneity), and an individual driver may react differently to the same type of stimulus (intradriver heterogeneity). To capture interdriver heterogeneity, car-following model parameters must be estimated for each driver/vehicle in the dataset, which are then used to describe a probability distribution associated with those model parameters. Capturing intradriver heterogeneity requires going one step further, calculating those same model parameters over much smaller time periods (i.e., seconds, or fractions of sections) within one vehicle's trajectory. This significantly reduces the amount of data available for calibration, limiting the ability to use traditional calibration procedures.

This research introduces a new method for car-following model calibration based on the Dynamic Time Warping (DTW) algorithm. After first extending Newell's car-following model to incorporate time-dependent parameters, this thesis describes the DTW algorithm and its application for calibrating the extended microscopic simulation model by synthesizing driver trajectory data. The standard DTW algorithm is enhanced to address a number of estimation challenges in this specific application, and a numerical experiment is presented with vehicle trajectory data extracted from the Next Generation Simulation (NGSIM) project for demonstration purposes. The DTW algorithm is shown to be a reasonable method for processing large vehicle trajectory datasets and to produce reasonable results when working with high resolution vehicle trajectory data. Additionally, singularities present an interesting match solution set to potentially help identify changing driver behavior; however, they must be avoided to reduce analysis complexity.

TABLE OF CONTENTS

Chapters

LIST OF NOTATION

$X = \{x_1, x_2, \ldots, x_i, \ldots, x_N\}$, which is the 1st time series dataset

$P = \{p_1, p_2, \ldots, p_i, \ldots, p_N\}$, which is the position data associated with the 1st time series dataset

$T_1 = \{t_{1,1}, t_{1,2}, \ldots, t_{1,i}, \ldots, t_{1,N}\}$, which is the time stamp data associated with the 1st time series dataset

$Y = \{y_1, y_2, \ldots, y_j, \ldots, y_M\}$, which is the 2nd time series dataset

$Q = \{q_1, q_2, \ldots, q_j, \ldots, q_M\}$, which is the position data associated with the 2nd time series dataset

$T_2 = \{t_{2,1}, t_{2,2}, \ldots, t_{2,j}, \ldots, t_{2,M}\}$, which is the time stamp data associated with the 2nd time series dataset

i = index for time series X made up of measurements P and T_1, $(i = 1, 2, \ldots, N)$

j = index for time series Y made up of measurements Q and T_2, $(j = 1, 2, \ldots, M)$

$C(i, j)$ = the cost (in terms of similarity) of mapping/aligning point X_i to point Y_j

$D(i, j)$ = the cumulative least cost (in terms of similarity) of mapping/aligning point X_i to point Y_j, considering the cost of mapping/aligning point X_i to point Y_j as well as the costs of previous mapped points between (X_1, Y_1) and (X_i, Y_j)

k = index for time series W

L = number of coordinates in W

$W = \{w_1, w_2, \dots, w_k, \dots, w_L\}$, which is the warp path – a list of coordinates which constitutes the shortest path (or more precisely least matching cost path) through matrix D, such that $w_k = (i, j)$.

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would first like to sincerely thank my committee members, Dr. R. J. Porter, Dr. Xuesong Zhou, and Dr. Tatjana Jevremovic. They've taught me so much about what it means to do research, to be a good colleague and collaborator, and to commit myself to meeting my goals. They've kindly supported me over many years of study, and saw me through to the completion of my degree. I would also like to thank Dr. Lawrence Reaveley for encouraging me to enter the research program here at the University of Utah, and for his kind and supportive advice throughout my academic career. Together, they've opened up the world of research to me, a world with nearly infinite possibilities and worthy of a lifetime spent exploring it. Without their persistent encouragement, careful guidance, and insightful commentary and feedback, I'm not sure I would be writing this sentence.

I would like to thank Dr. Kris Koford for his swift and talented editorial work, and, more importantly, for his friendship. Kris has been a tremendous help in finalizing this thesis, but he has also always been there for me with advice and support, in good times and in bad, for which I am eternally grateful.

Lastly, I would like to thank my mother, father, brother, and grandparents for their patience and support throughout my academic career. They helped me to become the person I am today, and I could have never reached this point without them.

CHAPTER 1

INTRODUCTION

In the past, various researchers have conducted car-following studies which attempt to describe a vehicle's movements while following another vehicle. These studies often resulted in creating mathematical models designed to describe a car-following behavior or condition. Most models are described in terms of the vehicles' relative position, speed, and acceleration, in addition to model parameters requiring calibration. Commonly studied car-following models include the GM Models (e.g., Chandler et al., 1958; Gazis et al., 1961), Newell's model (Newell, 2002), Gipps' model (Gipps, 1981), and the Intelligent Driver model (Treiber et al., 2000), amongst many others. Interested readers are referred to Aghabayk et al. (2013) for a brief, comprehensive review on car-following models with additional detail provided in Kesting and Treiber (2012).

Many studies have calibrated car-following model parameters using experimental vehicle trajectory data (e.g., Kesting & Treiber, 2008; Ma & Andréasson, 2006; Ossen & Hoogendoorn, 2005) and various proposed calibration methods (e.g., Brockfeld et al., 2004; Ossen and Hoogendoorn, 2008; Yang et al., 2011). An interesting issue arises when considering how those calibrated model parameters have been described. Even the earliest experiments based on simple stimulus-response models found that model parameters varied from driver to driver (e.g., Chandler et al., 1958). This is now commonly referred to as a form of driver heterogeneity – the idea that drivers' responses to stimuli while driving may not be constant.

Within our experimental and conceptual understanding, several studies (e.g., Ossen and Hoogendoorn, 2005; Ossen and Hoogendoorn, 2007; Duret et al., 2008; Chiabaut et al., 2010) have confirmed that calibrated car-following model parameters can be different for different drivers. This is an example of interdriver heterogeneity. Generically, interdriver heterogeneity describes the idea that different drivers may have different reactions to the same stimulus. Extending this concept, other studies (e.g., Ossen et al., 2006) have found that the actions of different drivers in a group of vehicles may be better explained using multiple car-following models rather than a single model. Put simply, this indicates that different drivers may follow different driving rules or have different responses to stimuli that cannot be explained using a single model for all drivers in a group.

More recent studies have also (e.g., Wang et al., 2010; Kim et al., 2013) found evidence which suggests that a single driver's actions can be better described by using different car-following parameters and/or different car-following models to describe a single vehicle trajectory. This is an example of intradriver heterogeneity – the idea that the same driver may react differently to the same stimulus at different times or under different conditions. Most studies on this subject (e.g., Wang et al., 2010) have focused on how model parameters change between acceleration and deceleration phases in car-following, or on which car-following models are most appropriate for these phases. However, the calibration techniques applied in these efforts tend to limit the temporal resolution at which model parameters may be estimated. For example, an acceleration or deceleration phase may last several seconds, and the calibration technique may offer model parameter estimates which are applicable over several seconds. Thus far, however, few researchers have attempted to estimate how car-following model parameters might change second-by-second, or at subsecond temporal resolutions.

1.1 Problem and Approach

This thesis addresses the problem of estimating how car-following model parameters may change at high temporal resolutions, particularly in second-by-second or subsecond time frames. The proposed methodology adapts Dynamic Time Warping (DTW) for matching patterns in high-resolution vehicle trajectory data, and uses the matching results to estimate time-varying parameters for Newell's simplified car-following model. Within this framework, numerical experiments were performed in MATLAB® (The MathWorks, 2010) using Next Generation Simulation (NGSIM) datasets to evaluate model performance and feasibility for large-scale applications.

1.2 Thesis Contributions and Potential Applications

The most important contribution from this work is the Dynamic Time Warping methodology in its application of estimating model parameters for Newell's simplified car-following model. Since Newell's model assumes that a following vehicle will replicate the speed of a leader vehicle, given some time lag and distance offset, the DTW algorithm can be used to match speed data for each trajectory over time and dynamically estimate the time lag between vehicle trajectories. This approach offers very high resolution parameter estimates, which could enable further investigations to develop more advanced car-following models and traffic flow theories. Particularly interesting is the potential for car-following models with embedded state transition models to estimate how model parameters change over time and under different conditions.

Numerous potential applications present themselves following this study. For example, pattern-matching results could be used to help estimate the safety effects of distracted driving (Przybyla et al., 2012). High-resolution model parameter estimates could be used to predict driving behavior in short time ranges, which could help automated vehicles predict driver's actions in a heterogeneous driving environment (i.e., mixtures of autonomous and human drivers). Deviations from expected car-following behavior could be used to classify driver behavior (e.g., aggressive or defensive) in a method similar to that proposed by Laval and Leclercq (2011). Additionally, model parameter estimates could be translated to macroscopic traffic state estimates, potentially providing higher resolution information about traffic flow conditions.

1.3 Thesis Outline

This thesis consists of six chapters. A general introduction was provided in Chapter 1, with the need for this research demonstrated through a general discussion of car-following models and the potential usefulness of models with high time resolutions. Chapter 2 surveys the existing literature pertaining to car-following models, methods for calibrating these models, and finally, studies devoted to exploring driver heterogeneity. Chapter 3 describes the car-following model formulation used in the study, as well as the Dynamic Time Warping methodology used for estimating time-varying model parameters. Chapter 4 presents the numerical experiments and discusses their results.

Chapter 5 provides additional discussion related to some of the limitations of the proposed methodology, and Chapter 6 concludes the thesis.

CHAPTER 2

REVIEW OF LITERATURE

2.1 Driving Behavior and Microscopic Simulation Models

Driving behavior has been one of the most difficult human decision-making processes to model. A wide range of car-following models, psycho-physical models and multiphase traffic flow theories have been proposed in an attempt to capture the driving behavior at a microscopic level. In many existing simulations of driving models, the behavior of a driver is mainly determined by the relative headway, gap, speed and acceleration of the lead and surrounding vehicles. Although a number of traffic simulation models have considered multiple driver classes to accurately describe heterogeneous perception and preferences, most widely used models still assume the same behavioral characteristics under both congested and uncongested driving situations.

Several recent studies have examined driver behavior heterogeneity and its impact on microscopic simulation models. Ossen and Hoogendoorn (2005) used high resolution trajectory data from a helicopter to find optimal sensitivity and reaction time parameters for individual drivers and for multiple car-following models. A later study in Ossen and Hoogendoorn (2007) showed that driver heterogeneity could not be explained only with model parameters, but must also include model specifications. Most recently, Ossen and Hoogendoorn (2011) studied vehicle trajectories for passenger vehicles and trucks in different leader-follower scenarios. Their results showed large variations in how passenger car drivers react to different stimuli and which stimuli influence their behavior. Truck drivers maintained more consistent speeds over time. Ossen and Hoogendoorn (2011) also showed that driver behavior can change depending on the leader's vehicle type. Kim and Mahmassani (2011) calibrated multiple car-following models using Next Generation Simulation (NGSIM) trajectory data to examine the effects of considering correlation between model parameters. When testing model parameter distributions in traffic simulations, their results indicate that there was a statistically-significant improvement in model performance (e.g., reduced variation in spacing between vehicles) when using correlated parameters rather than uncorrelated parameters. Kim and Mahmassani (2011) also showed that the performance impacts associated with using correlated parameters were more pronounced for nonlinear car-following models like Gipps's model and the Intelligent Driver model. Laval and Leclercq (2010) presented a theory for modeling aggressive and timid driver behavior to partially describe traffic oscillations and their transformation into stop-and-go waves. They specifically identified traffic oscillations as a consequence of drivers' heterogeneous reactions to deceleration waves, but aggressive and timid driving behavior alone could not produce the observed traffic oscillations.

This thesis first focuses on extending Newell's simplified Linear Car-Following model (LCF) (1962, 2002), which considers the following two driving modes: (i) under uncongested conditions, vehicles are driving at free-flow speed, and (ii) under congested conditions, a following vehicle changes speeds to maintain a minimum jam spacing and a reaction time lag with respect to the leading vehicle's trajectory. Brockfeld et al. (2004)

calibrated and validated a number of well-known car-following models, and Newell's simplified LCF model showed reasonable performance with limited calibration efforts.

2.2 Intradriver Heterogeneity

Several studies have examined driver heterogeneity using trajectory data in terms of driver-specific (interdriver) and/or time-varying (intradriver) car-following model parameters. However, while early studies incorporated interdriver heterogeneity, only more recent work has studied intradriver heterogeneity. For example, Ahn et al. (2004) confirmed driver-specific model parameters in Newell's car-following model (inter-driver heterogeneity), aligning with Newell's model definition, but their study did not test whether a driver's parameters remain constant over time (intradriver heterogeneity). Later, Ossen et al. (2006) used vehicle trajectory data to find optimal car-following model parameters and optimal car-following models to describe the actions of individual drivers. Their work indicated that inter-driver heterogeneity cannot be explained only by variations in car-following parameters because different drivers may follow different driving styles. Later, studies started expanding toward considering intradriver heterogeneity. Hamdar et al. (2009) noted that "heterogeneity observed in traffic dynamics may be attributed to intra-driver heterogeneity rather than inter-driver heterogeneity." This matches Kesting and Treiber's (2009) observation that "intra-driver variability accounts for a large part of the deviations between simulations and empirical observations." Wang et al. (2010) used data from Dutch motorways to identify longer response times for drivers while accelerating compared to decelerating. They found that 65 percent of drivers demonstrated different "driving styles" between the two states, where the calibrated model in the acceleration state could not accurately describe the trajectory in the deceleration state. They suggested a multiphase car-following model to capture this heterogeneity for more reliable traffic simulation. While several studies point to intradriver variability in model parameters as a significant source of error in traffic simulation, few studies have thoroughly described intradriver heterogeneity. Wang et al. (2010) directly address the intradriver heterogeneity issue, but only to provide a comparison based on "car-following phases" (acceleration and deceleration phases).

2.3 Car-Following Model Calibration

Microscopic vehicle trajectory data have been used for calibrating car-following models by numerous authors, and several methods have been used for calibrating those models. For instance, Ma and Andréasson (2006) describe calibration as a nonlinear optimization problem, where the solution approaches can be divided into gradient-based methods or derivative-free methods, which include grid-search algorithms and the genetic algorithm. Their study calibrated a GM-type model with data collected using an instrumented Volvo vehicle. Ciuffo et al. (2011) provide a review of several methodologies used for microsimulation calibration, including simultaneous perturbation stochastic approximation, simulated annealing, genetic algorithms, and the OptQuest/Multistart heuristic algorithm. Genetic algorithms were noted as the most common in their study. Hamdar et al. (2009) used a genetic algorithm to calibrate a stochastic car-following model using NGSIM data. Kesting and Trieber (2008) used a method similar to a genetic algorithm to calibrate the Intelligent Driver model (IDM) and the Velocity Difference model from vehicle-mounted radar sensors.

Ossen and Hoogendoorn (2008a) took a critical look at calibrating car-following models with real-world vehicle trajectories. They began with speed data from real-world trajectories and developed 25 leader vehicle trajectories using a model with known calibration parameters. These synthetic datasets were then injected with artificial errors, representing different types of measurement errors, and a new calibration was performed. The new parameters were then compared with the known parameters to evaluate the effects of those errors in calibration. Their study reported several effects associated with measurement errors. First, they found that measurement errors can introduce bias when estimating model parameters. Their calibration process used an objective function to quantify how well simulated and observed data (e.g., speed or distance headway) match, using different model parameters to generate the simulated results. The goal then becomes to find the parameters which minimize this performance metric (i.e., the objective function). The objective function used in their calibration process (Theils' U) became less sensitive (i.e., changing the model parameters had less of an effect on the performance metric) with measurement errors, and their optimization results did not produce the optimal model parameters used to create the synthetic calibration data. Additionally, they showed that the calibration results improved when using a simple moving average for smoothing to account for some measurement errors. Hoogendoorn et al. (2011) describe a piecewise linear approximation filtering technique, showing that speed profiles in NGSIM data can be represented well using periods of constant acceleration. A later study by Ossen and Hoogendoorn (2009) examined the degree of information contained in vehicle trajectories and the effects of measurement errors on the calibration objective function's sensitivity. They provide an in-depth review of vehicle trajectory data collection techniques, calibration and verification methods, and assess the influence of measurement errors in a guide available online (Ossen and Hoogendoorn, 2008b).

2.4 Dynamic Time Warping

The Dynamic Time Warping (DTW) algorithm is used to measure the similarity between two time-series datasets, which can be viewed as an optimization problem that minimizes the effects of shifting and distortion in time through a flexible transformation/mapping of time series (Senin, 2008). This approach can also be used to identify the optimal alignment between two time-dependent data series, and the algorithm finds the alignment with the least cumulative cost, called the warp path (Keogh and Pazzani, 2001), using a shortest path algorithm which starts at the last pair and works back to the first pair. The commonly used cost is a quantitative measure of the similarity or difference between two points.

Similar to the calibration method used by Yang et al. (2010) to find the optimal time delay between vehicle trajectories, the DTW algorithm may produce an estimate of the time delay along each vehicle trajectory, but does so for each discrete point in the time series. Once the time delay is known, simple car-following model parameters (i.e., jam spacing) can be calculated based on the best time delay estimates at each time stamp in the dataset, offering the ability to further investigate intradriver heterogeneity at higher time resolutions.

CHAPTER 3

METHODOLOGY

3.1 Overview

The goal of this research is to enable investigations into intradriver heterogeneity by estimating time-varying, car-following model parameters based on high-resolution vehicle trajectory datasets. This is accomplished by providing model parameter estimates at each discrete point in time along the vehicle trajectory, rather than providing one set of model parameter estimates for an entire trajectory or a “car-following phase.” In this research, the DTW algorithm is used to find the optimal alignment between two time-series datasets. When applied to vehicle trajectory data, the alignment is considered an estimate of the stimulus-response relationship for a driver following a leading vehicle, and it is used to infer the time-varying, car-following model parameters for a driver during the observation time period. The underlying assumption is that the stimulus-response driver behavior model can be represented in terms of time-series similarity and further estimated using the DTW algorithm. That is, time-series similarity techniques may help to identify the stimulus-response interactions observed in empirical data.

The main methodological contribution of this work comes from the combination of Newell’s model and DTW. Newell’s simplified model parameters reduce calibration process complexity, and the DTW algorithm’s pattern-matching capabilities should significantly improve the calibration solution for small datasets. With higher-resolution data, it may be possible to quantify the stochasticity present in the different model parameters and describe parameter sensitivity, providing information for incorporating time-varying parameters in car-following models which is not likely to be found in the literature. The data could also be analyzed for patterns based on different "car-following phases," such as acceleration and deceleration states, which can be used to improve multiphase models.

3.2 Car-Following Model Formulation

3.2.1 Newell's Car-Following Model

Newell's car-following model (Newell, 2002) is based upon one basic assumption: A vehicle following another vehicle (the leader vehicle) in a homogenous space replicates the trajectory of the leader vehicle, but their trajectories are separated by a time and distance offset. This relationship between vehicle trajectories is described mathematically in the following equation:

$$x_n(t+\tau_n) = x_{n-1}(t) - d_n \tag{1}$$

where

t = time index,

n = vehicle number index,

x_n = position of vehicle n,

d_n = critical jam spacing, or distance offset, for vehicle n, and

τ_n = time lag, or time offset, for vehicle n.

The left side of Eq. (1) describes the position of the following vehicle at time $(t + \tau_n)$, where n refers to the following vehicle. The right side of Eq. (1) refers to the position of the leader vehicle at time t and the distance offset between the two trajectories, where $n - 1$ refers to the vehicle preceding the following vehicle. In this generalized formulation, the model parameters d_n and τ_n, are associated with each vehicle, and may be assumed to be constant over time. This car-following model is also consistent with Newell's simplified kinematic wave model (Newell, 1993), where the Fundamental Diagram of Traffic Flow takes a triangular shape, as shown in Fig. 1. Under this model, the critical jam spacing is the inverse of the jam density ($d_n = 1/k_{jam}$), and the backward shockwave speed w is related to both car-following parameters ($w_n = d_n/\tau_n$).

Since vehicle trajectories are replicated in Newell's simplified car-following model, the speeds of both vehicles are also replicated with a certain time lag. The first derivative of Eq. (1) indicates that the speed of the leader vehicle is replicated by the speed of the following vehicle after the time offset τ_n, as derived in Eq. (2) and shown in Eq. (3). This is important because it identifies velocity time-series data as a good candidate for pattern matching, where matching the speed profile of two vehicles can help estimate the time offset.

$$\frac{dx}{dt}\left(x_n(t + \tau_n)\right) = \frac{dx}{dt}\left(x_{n-1}(t) - d_n\right) \qquad (2)$$

$$\dot{x}_n(t + \tau_n) = \dot{x}_{n-1}(t) \qquad (3)$$

Newell makes an additional simplifying assumption by using a piecewise linear approximation to describe vehicle trajectories. The visual representation in Fig. 2 shows the position of the leader vehicle in a time-space diagram with both a solid line and a dashed line, where the solid line is the linear approximation to the observed dashed line. This implicitly assumes that all acceleration is instantaneous. If the velocity of the $(n-1)th$ car and the nth car are the same, the piecewise linear approximation is reasonable, but this may not be the case where drivers' reactions to the change in speed are not homogeneous.

Newell also describes a few fundamental relationships between the model parameters which are useful for model calibration purposes. Fig. 2 shows that the spacing S_n (or distance headway) between vehicles increases after the velocity increases. Newell described the relationship between velocity and spacing using a linear model, where the slope is τ_n and the intercept is d_n, as shown in Fig. 3. This means that these parameters should be independent of velocity, and constant model parameters could be estimated using linear regression.

3.2.2 Formulation with Time-Varying Parameters

As stated before, the goal of this research is to investigate intradriver heterogeneity by estimating time-varying, car-following model parameters. That is, it is assumed that the manner by which those model parameters vary with time can be used to describe intradriver heterogeneity. More systematically, it can also be assumed that the underlying process shows a "disturbance" for tau and d because, even if these are allowed to vary with time, there will be additional variation from factors known (but unmeasured) and unknown.

Thus, this research begins with the hypothesis that car-following model parameters are not constant for each driver, but actually change over time. In order to test this hypothesis with Newell's car-following model using the proposed methodology, Newell's model needs to be re-formulated using time-varying parameters. As a result, Eq. (1) becomes Eq. (4):

$$x_n(t + \tau_{n,t}) = x_{n-1}(t) - d_{n,t} \tag{4}$$

where

$\tau_{n,t}$ = time lag, or time offset, for vehicle n at time t,

$d_{n,t}$ = critical jam spacing, or distance offset, for vehicle n at time t.

Redefining Newell's car-following model in order to use time-varying parameters introduces several potential issues. The most significant issues are related to the Newell's modeling assumptions. For example, if d and τ vary with time, the linear relationship between velocity and spacing described in Fig. 3 may no longer be valid. Additionally, Newell's approach of using a piecewise linear approximation to describe a vehicle trajectory is no longer required because the spacing between vehicles may change continuously rather than just when the speed changes significantly. Alternatively, for consistency with Newell's approach, one might assume that there is a piecewise linear approximation between each observed data point in a trajectory. Lastly, if the backward wave speed is allowed to change over time for a single vehicle, this might conflict with how Newell expected a backward wave to propagate upstream.

3.3. Dynamic Time Warping Algorithm

The Dynamic Time Warping (DTW) algorithm is a dynamic programming technique used to find the optimal alignment, or mapping, between two time-series datasets. In this application, the DTW algorithm provides the estimated optimal time offset between two vehicle trajectories during a specific time period, indicating likely points in time where the following driver reacted to a stimulus (i.e., change in speed) from the leader vehicle.

In order to better explain the inner workings of the algorithm and its components, this section first identifies the notation used to describe the algorithm. Next, the algorithm is explained step-by-step with illustrative elements which help to explain each component in the algorithm. Details are also provided for an alternative formulation where the algorithm is translated into an optimization problem which can be solved using linear programming. This section ends with an illustrative example aligning two sets of vehicle speed data time series.

3.3.1 DTW Algorithm

In its basic form, the DTW algorithm first assesses the cost for aligning each data point in one time series to all other points in the second time series, creating a cost matrix. It then begins at the first data series pair in the cost matrix, calculating the cumulative least cost for continuously moving from the first pair to the last pair in the matrix, creating a cumulative cost matrix. Lastly, the algorithm finds the alignment with the least cumulative cost, using a shortest path algorithm which starts at the last pair in the cumulative cost matrix and works back to the first pair. While the cost is a quantitative measure of the similarity or difference between two points in each time-series dataset (distance is commonly used), it can also be a flexible term which may be modified to suit the application. The algorithm is explained in further detail below with visual elements to help understand the different components.

3.3.1.1 Input Data

The DTW algorithm is based on the idea of measuring the similarity or distance between two or more elements or sets of data. This measurement is usually performed using a metric, such as the generalized norm, or L_p distance (the L_2 norm, or Euclidean distance, is commonly chosen). The cost $C(i, j)$ of mapping two points (X_i, Y_j) together is based on this measured similarity or distance calculated from the input time series datasets X and Y. In this study, the metric is based upon the input data taken from the underlying car-following model. For Newell's car-following model, the similarity metric should be based on vehicle velocity because Eq. (3) showed that the velocities of the lead and following vehicles should be the same at a time offset from each other.

Since Newell's model assumes that the velocities are the same, the acceleration should also be the same and could be a potential candidate as input data for the DTW algorithm. However, high resolution observation data typically shows that acceleration data is more volatile and subject to greater uncertainty, often due to measurement error. DTW can be sensitive to noisy input data, so velocity data were selected as the input data for the DTW algorithm. An example of velocity input time series data is shown in Fig. 4.

3.3.1.2 Cost Matrix

The first step in the DTW algorithm, after assembling the input data, is to construct the cost matrix. The cost matrix $C(i,j)$ is an $N \times M$ matrix which stores all pair-wise distances between X and Y. The cost in each cell of the cost matrix is calculated using Eq. 5. The cost matrix is created using Algorithm 1. An example of a cost matrix is shown in Fig. 5.

$$C(i,j) = \sqrt{\left(x_i - y_j\right)^2} = \left|x_i - y_j\right| \tag{5}$$

3.3.1.3 Cumulative Cost Matrix

The next step is to calculate the cumulative cost matrix $D(i,j)$, which is an $N \times M$ matrix which stores the cumulative least cost required to arrive at any location in the matrix by following a specified search pattern from $(1,1)$ to (N,M). The most common search pattern allows the algorithm to check costs in the next cell vertically, horizontally, or diagonally away from the current cell in the matrix. The cumulative least cost in each cell of the matrix is calculated using Eq. (6), which also identifies the search direction.

$$D(i,j) = \ C(i,j) + \min\bigl(D(i-1,j-1), D(i-1,j), D(i,j-1)\bigr) \ \ i \in N, \ \ j \in M \tag{6}$$

The cumulative cost matrix is created using Algorithm 2. An example of a cumulative cost matrix is shown in Fig. 6.

Algorithm 1:

```
    For i = 1 to N
        For j = 1 to M
            $C(i,j) = |x_i - y_j|$
        EndFor
    EndFor
```

Algorithm 2:

```
For i = 1 to N
    For j = 1 to M
        If i = 1 and j = 1
            $D(i,j) = C(i,j)$
        ElseIf i = 1
            $D(i,j) = D(i,j-1) + C(i,j)$
        ElseIf j = 1
            $D(i,j) = D(i-1,j) + C(i,j)$
        Else
            $D(i,j) = \min\big(D(i-1,j-1), D(i-1,j), D(i,j-1)\big) + C(i,j)$
        EndIf
    EndFor
EndFor
```

3.3.1.4 Warp Path and Constraints

The last step in the algorithm is to find the optimal alignment by calculating the warp path W through the cumulative cost matrix. The warp path is the shortest path from (N, M) to $(1,1)$ through the cumulative cost matrix, following a specific search pattern. Similar to the process for constructing the cumulative cost matrix, the warp path search pattern typically allows searching the next cell vertically, horizontally, and diagonally away from the current cell in the warp path.

Additionally, the warp path must satisfy the following three constraints:

- Boundaries: The start and end points of the datasets must be the start and end points of the warp path. $w_1 = (1, 1)$ $w_L = (N, M)$
- Continuity: The warp path cannot step forward more than one time index in any direction at one time.

$$w_k = (i, j) \quad w_{k-1} = (i', j') \qquad i - i' \leq 1 \qquad j - j' \leq 1 \tag{7}$$

- Monotonicity: The warp path must continuously step forward from beginning to end; the algorithm cannot step backward.

$$w_k = (i, j) \quad w_{k-1} = (i', j') \qquad i - i' \geq 0 \qquad j - j' \geq 0 \tag{8}$$

The warp path is created using Algorithm 3. An example of the warp path, and its relation to the cumulative cost matrix, is shown in Fig. 7.

Once the warp path is assembled, the car-following model parameters are

Algorithm 3:

```
Initialize i = N; j = M; k = 1
While i ≥ 1 and j ≥ 1
    $k = k + 1$
    If i = 1 and j = 1
        Break
    ElseIf i = 1
        $j = j - 1$
    ElseIf j = 1
        $i = i - 1$
    Else
        If $D(i-1, j) = \min\big(D(i-1, j-1), D(i-1, j), D(i, j-1)\big)$
            $i = i - 1$
        ElseIf $D(i, j-1) = \min\big(D(i-1, j-1), D(i-1, j), D(i, j-1)\big)$
            $j = j - 1$
        Else
            $i = i - 1; j = j - 1$
        EndIf
        $W_k = (i, j)$
    EndIf
EndWhile
```

calculated based upon the matching coordinates for each time step in the warp path using Eqs. (9-10).

$$\tau_n(k) = t_{2,j} - t_{1,i} \tag{9}$$

$$d_n(k) = p_i - q_j \tag{10}$$

The DTW algorithm allows one-to-many matching in each time series, so there may be more than one parameter estimate for each time index in the follower driver time series dataset. As a result, some additional filtering is necessary to remove duplicate estimates.

3.3.2 Illustrative Example for Vehicle Trajectory Data

To assist the reader in understanding how the algorithm works and how it is applied to vehicle trajectory data, this section describes an illustrative example in detail. The input data used in the algorithm, summarized in Table 1, consists of velocity data for two vehicles. Their trajectories are represented visually in a time-space diagram in Fig. 8a, and the velocities are plotted in Fig. 8b.

Following Algorithm 1, the cost matrix is first calculated based upon the input data. In this case, $X = \{25, 25, 25, 5, 5, 30, 30, 30\}$ and $Y = \{25, 25, 25, 25, 7, 7, 7, 30\}$, and each cell in the cost matrix is calculated as the difference between each pair of data points between the two time-series datasets. Since the cost $C(i,j) = |x_i - y_j|$, the cost in the first cell is $C(1,1) = |x_1 - y_1| = |25 - 25| = 0$, and the cost at $(4,5)$ is $C(4,5) =$

$|x_4 - y_5| = |5 - 7| = 2$. The complete cost matrix for this illustrative example is shown in Fig. 9.

The second step is to follow Algorithm 2 and calculate the cumulative cost matrix using the cost matrix calculated in the previous step. The cumulative cost matrix could be thought of as a network, where the objective is to travel from $(1,1)$ to $(8,8)$ by passing through the cells in the matrix. Each step along this path has a cost, and the search pattern restricts the "traveler's" movement to only the next adjacent cell in the matrix horizontally, vertically, or diagonally. The least cost required to arrive at a cell in the matrix is the accumulation of the costs in the previously-used cells. In this example, the cost $D(1,1)$ at the start of the matrix is 0 because $C(1,1) = 0$. At location $(1,5)$ in the cumulative cost matrix, the cost $C(1,5) = 18$ and the search pattern only allows horizontal movement at this boundary since no backward movement is allowed, so the cumulative least cost to reach this point is $D(1,5) = D(1,4) + C(1,5) = 18$. At location $(4,5)$, the search pattern allows searching in all three directions (horizontally, vertically, or diagonally). As a result, the cumulative least cost to arrive at location $(4,5)$ is $D(4,5) = \min\big(D(4,4), D(3,4), D(3,5)\big) + C(4,5)$. The minimum cost in the adjacent cells is at location $(3,4)$, so the cumulative least cost at location $(4,5)$ is $D(4,5) = [D(3,4) + C(4,5)] = [0 + 2] = 2$. The complete cumulative cost matrix for this illustrative example is shown in Fig.10.

The last step in the DTW algorithm is to follow Algorithm 3 and find the warp path through the cumulative cost matrix. Similar to calculating the cumulative least cost matrix, it is convenient to think of the warp path as the shortest path through the matrix, where the matrix is actually a network composed of cells. Again, a search pattern restricts the “traveler’s” movement to only the next adjacent cell in the matrix horizontally, vertically, or diagonally. However, rather than moving from the beginning to the end of the matrix, the warp path finds the shortest path from $(8, 8)$ to $(1, 1)$ by following the cumulative least cost. The next step through the matrix is selected based on the least cost at arriving at the potential next step.

Consider the warp path shown in Fig. 11. Starting at location $(8, 8)$, the algorithm checks the cumulative costs at the potential next steps $(8, 7)$, $(7, 7)$, and $(7, 8)$. The least cumulative cost amongst the available cells is at location $(7, 8)$ because $D(7, 8) = 6$, so the algorithm adds location $(7, 8)$ to the warp path and moves to this new location. This procedure continues until the algorithm reaches location $(1,1)$ at the beginning of the cumulative cost matrix.

The warp path indicates the optimal alignment between points over time. For example, it matches point x_5 in the leader trajectory with points y_6 and y_7 in the follower trajectory. Using this information, the velocity data points are matched in Fig. 12. Additionally, this matching can be translated into the time-space domain in Fig. 13. This alignment information is then used to estimate the time offset $\tau_{n,t}$ and distance offset $d_{n,t}$ following the Eqs. (9-10). For example, at time $t = 4$ in the follower’s time-series (vehicle $n = 1$), $\tau_{1,4} = (4 - 3) = 1$ time unit, and $d_{1,4} = p_4 - q_3$.

3.4 DTW Algorithm Modifications

While the DTW algorithm was designed to match time series data, the matching results for vehicle trajectory data may not always be consistent with our understanding of car-following behavior. For example, without modifications, the DTW algorithm could return negative time offsets, which might indicate that the driver reacted to a change in the leading vehicle's trajectory before it occurred. Also, similar to other calibration methods, the DTW algorithm may have difficulty in attempting to match trajectory datasets containing time periods with little variation (e.g., trajectories with constant speed). This section describes several algorithm modifications/adjustments to help address these issues, primarily related to changing the cost function and adding constraints.

3.4.1 Constraints and Costs

As mentioned above, the DTW algorithm (without modifications) can return matching results which produce negative car-following model parameters. Alternatively, the algorithm may also return extreme estimates with very large values for $\tau_{n,t}$ and/or $d_{n,t}$. The logical solution to these issues would be to add constraints to the algorithm to prevent unreasonable matching results – essentially adding an upper and lower bound on $\tau_{n,t}$ and $d_{n,t}$. Setting boundary conditions for these parameters is similar to the "windowing" methods described in the literature (e.g., Sakoe-Chiba bands by Sakoe and Chiba, 1978). However, an upper bound condition may artificially prevent the algorithm from identifying correct matches, dependent upon the leader-follower relationship. For example, an upper bound placed on an actual, abnormally long following distance may exclude several matches for a leader-follower pair. In the proposed methodology, a more conservative approach is implemented by only considering the lower bound constraint and implementing the lower bound constraint as a soft constraint by modifying the cost function. In this way, the new cost function penalizes unacceptable matching pairs ($\tau_{n,t} \leq 0, d_{n,t} \ll 0$) between the trajectory time series datasets, as shown in Eq. 9.

$$C(i,j) = \begin{cases} Penalty \times \left|x_i - y_j\right| & \tau_{n,t} \leq 0, d_{n,t} \leq 0 \\ \left|x_i - y_j\right| & Otherwise \end{cases} \tag{9}$$

This soft constraint encourages the algorithm to make theoretically-acceptable matches, but also allows unacceptable matches when necessary to guarantee a continuous path for both trajectories. Additionally, the penalty is applied as a scaling factor to the calculated cost so that the similarity information at these locations is not lost. This is important because there may be situations in which the penalty is applied to a block of cells in the cost matrix. In one such situation, if the penalty simply replaced the cost in the cell, there would be no obvious best choice for the warp path through the matrix. This formulation for the cost function could be further adjusted to implement an upper bound limit, and the penalty could be adjusted as necessary (a reasonable Penalty value might ten times greater than the maximum speed difference in the datasets).

When building the warp path, another potential issue arises in which the cumulative cost in the adjacent cells may be equal. When this occurs, the algorithm may not have an obvious best choice for the next step in the warp path. With a well-defined cost function and relatively precise measurements, there should be very few pairs of time-series data which produce equal costs in close proximity in a matrix. In this application, however, this “equal-cost” situation may arise when the speeds are nearly constant for both vehicles over a short period of time. To resolve this issue, a prespecified warp path step direction (diagonal step is preferred) is specified to help guide the algorithm through the cumulative cost matrix. While this may be the simplest option, it may create additional issues when the same situation arises consecutively because the shortest path is unknown beyond the current location in the cumulative cost matrix. In this way, using alternative (robust) optimization model formulations may produce more reliable warp paths.

3.4.2 Incorporating Prior Information

One of the challenges in calibrating Newell's model with the DTW algorithm is time-series segments with nearly constant velocities. When the velocities are nearly constant over a period of several time units, the cost of matching multiple points is nearly the same. As a result, the cost matrix calculated in Algorithm 1 could produce a region in the matrix with little or no variation in cost. Essentially, the algorithm does not have enough information to reasonably quantify the similarity in speed in these situations. This often causes the DTW algorithm to produce unrealistic matching results and reduces the quality of the parameter estimates. To prevent these conditions an alternative cost matrix function, incorporating additional information, is proposed in the following section.

3.4.2.1 Enhanced DTW Cost Formulation

The proposed enhancements to the DTW cost function incorporate prior estimates for Newell's model parameters (i.e., the time lag, critical spacing, and backward wave speed) in the cost calculation. This cost function formulation, which takes a similar form to those used in Chen et al. (2005), is described in the following equation:

$$C(i,j) = \left|x_i - y_j\right| + \alpha\left|t_{2,j} - t_{1,i} - \hat{\tau}_{n,t}\right| + \beta\left|p_i - q_j - \hat{d}_{n,t}\right| + r\left|\frac{p_i - q_j}{t_{2,j} - t_{1,i}} - \widehat{w}_{n,t}\right| \tag{10}$$

where:

α = weight on time lag similarity,

β = weight on critical spacing similarity,

r = weight on backward wave speed similarity,

$\hat{\tau}_{n,t}$ = prior estimate for the time lag sample mean for vehicle *n* and time *t* ,

$\hat{d}_{n,t}$ = prior estimate for the critical spacing sample mean for vehicle *n* and time *t*, and

$\widehat{w}_{n,t}$ = prior estimate for the backward wave speed sample mean for vehicle *n* and time *t*.

The implicit assumption in this approach is that an average model parameter value for an individual driver is representative of the sample mean for that estimated parameter's distribution over time. As a result, an estimated parameter value at a given time instance should be expected to be similar to that average parameter value. In this way, the cost function is adapted to consider the similarity of each model parameter in terms of a prior estimate of its sample mean. This additional information generally helps to better describe similarity in the cost matrix, offering more guidance to the algorithm when speed data alone are not sufficient to produce reasonable pattern matching results.

Incorporating additional information in the DTW cost function requires considering some complex issues about the fundamental nature of the algorithm. The cost function is used to quantify similarity. However, a similarity measure for one source of information may be different from a similarity measure for another source of information. In order to properly account for variations in similarity between sources of information, it is necessary to normalize the similarity measures in the cost function. This is accomplished here using the weights α, β, and r. The process of normalization is generally represented in statistics by $\frac{X-\mu}{\sigma}$. In this case, where the prior estimate is an average value, the weight would be the inverse of the standard deviation for the estimate parameter distribution.

3.4.2.2 Sources of Prior Estimates and Weights

One of the difficulties in applying this proposed alternative cost function is in selecting values to use as the prior estimates and weights. The most obvious potential source of model parameter prior estimates is a textbook reference or the results of car-following study (the source used in numerical experiments in Chapter 4). Alternatively, a prior estimate could be provided by using a common calibration technique to estimate a model parameter value for a single vehicle's trajectory, or a group of trajectories. Estimating the weights, however, is more difficult due to their definitions. Since the prior estimate is assumed to be an estimate of the sample mean, it is easy to assume that a calibrated model parameter approximates this sample mean, but there are no obvious solutions to estimating the standard deviation for the sample distribution. One option is to use a trial-and-error approach (as is utilized in the numerical experiments). A more complicated approach might involve using an iterative process to estimate and adjust the weights (standard deviations) until the calculated and estimated sample standard deviations are nearly equal. This advanced technique is recommended for consideration in future research.

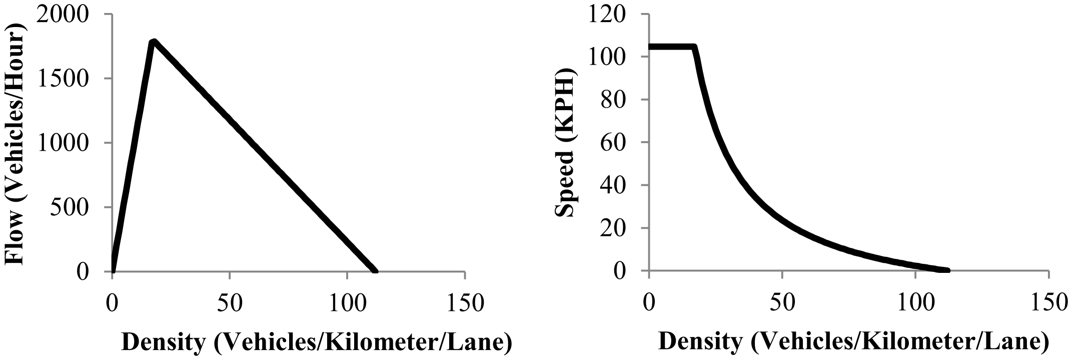


Figure 1: Flow-Density and Speed-Density diagrams associated with Newell's car-following model. Typical values: w = 19 KPH and k_{jam} = 112 vehicles/km/lane, which leads to $\overline{d_n}$ = 9 meters, $\overline{\tau_n}$ = 1.7 seconds.

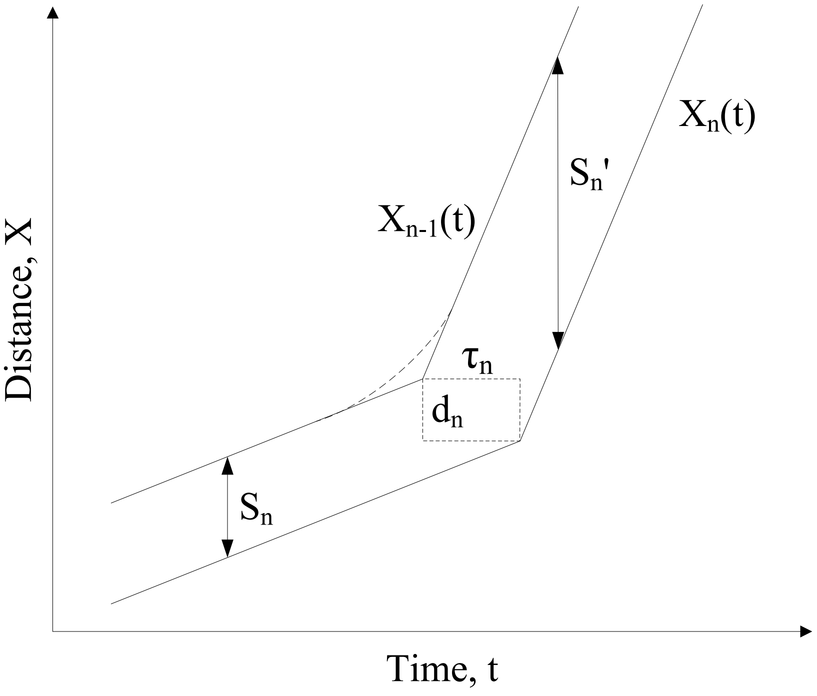


Figure 2: Visual representation of Newell's car-following model with piecewise linear approximation (adapted from Newell, 2002).

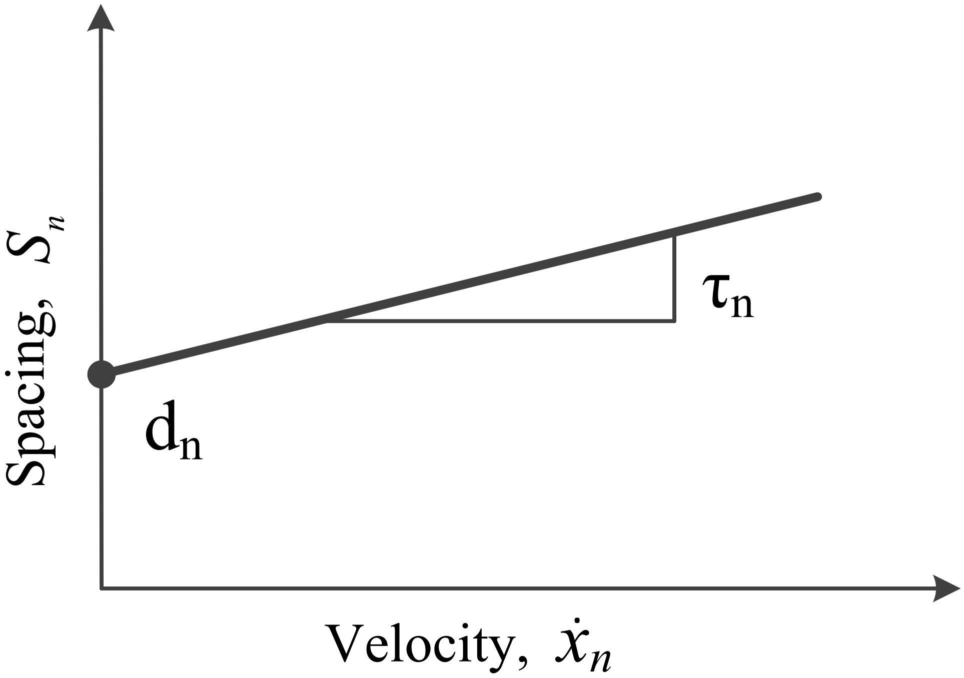
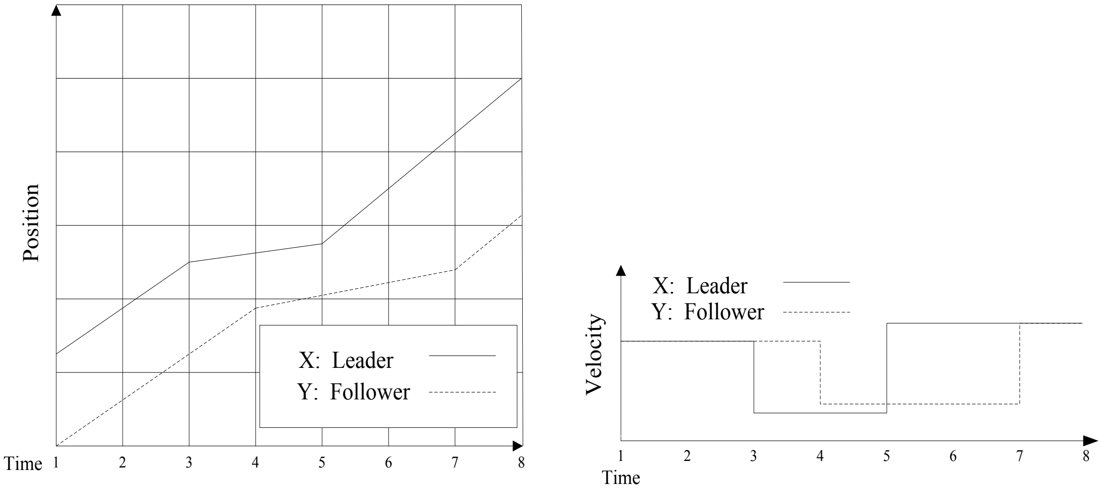


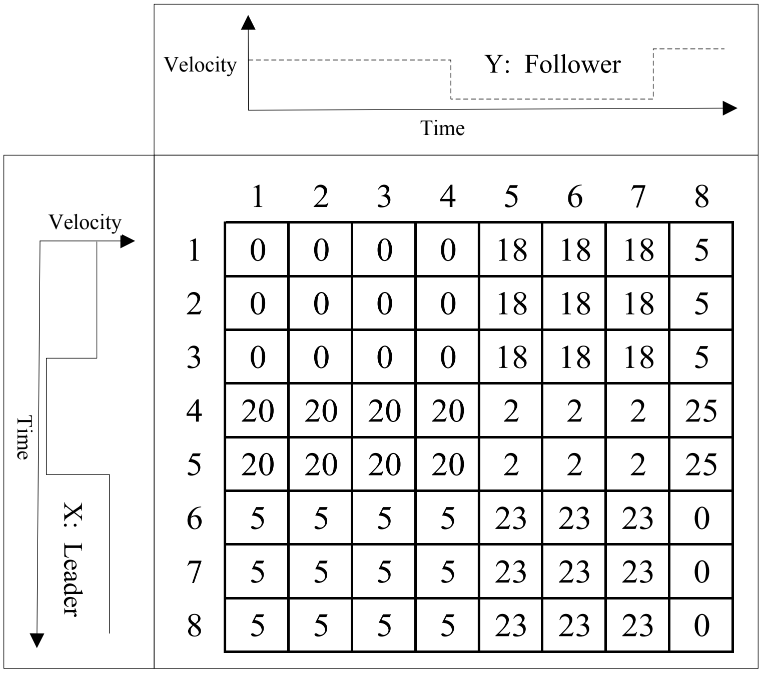
Figure 3: Velocity-spacing relationship in Newell's model (adapted from Newell, 2002).



(a) Raw Vehicle Trajectories

(b) Vehicle Velocity Time Series

Figure 4: Example of position speed data for explaining DTW input data. (a) Raw vehicle trajectories, and (b) their corresponding velocity time series datasets.



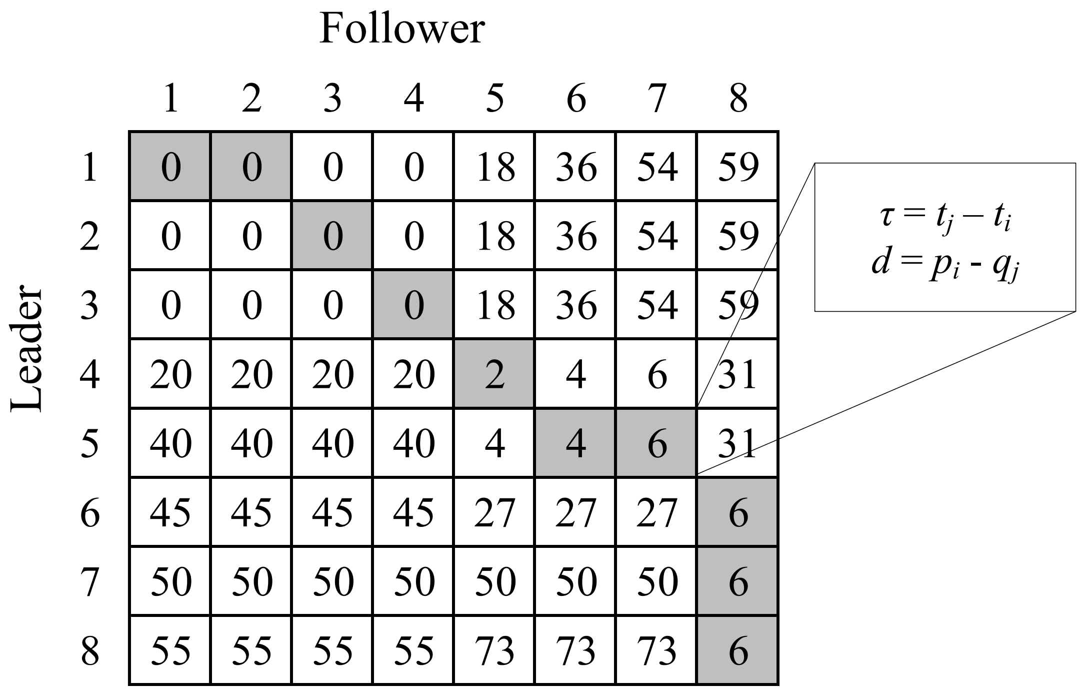
	1	2	3	4	5	6	7	8
1	0	0	0	0	18	18	18	5
2	0	0	0	0	18	18	18	5
3	0	0	0	0	18	18	18	5
4	20	20	20	20	2	2	2	25
5	20	20	20	20	2	2	2	25
6	5	5	5	5	23	23	23	0
7	5	5	5	5	23	23	23	0
8	5	5	5	5	23	23	23	0

Figure 5: Visual representation of the cost matrix.

Follower

Leader	1	2	3	4	5	6	7	8
1	0	0	0	0	18	36	54	59
2	0	0	0	0	18	36	54	59
3	0	0	0	0	18	36	54	59
4	20	20	20	20	2	4	6	31
5	40	40	40	40	4	4	6	31
6	45	45	45	45	27	27	27	6
7	50	50	50	50	50	50	50	6
8	55	55	55	55	73	73	73	6

Figure 6: Visual representation of the cumulative cost matrix.



	1	2	3	4	5	6	7	8
1	0	0	0	0	18	36	54	59
2	0	0	0	0	18	36	54	59
3	0	0	0	0	18	36	54	59
4	20	20	20	20	2	4	6	31
5	40	40	40	40	4	4	6	31
6	45	45	45	45	27	27	27	6
7	50	50	50	50	50	50	50	6
8	55	55	55	55	73	73	73	6

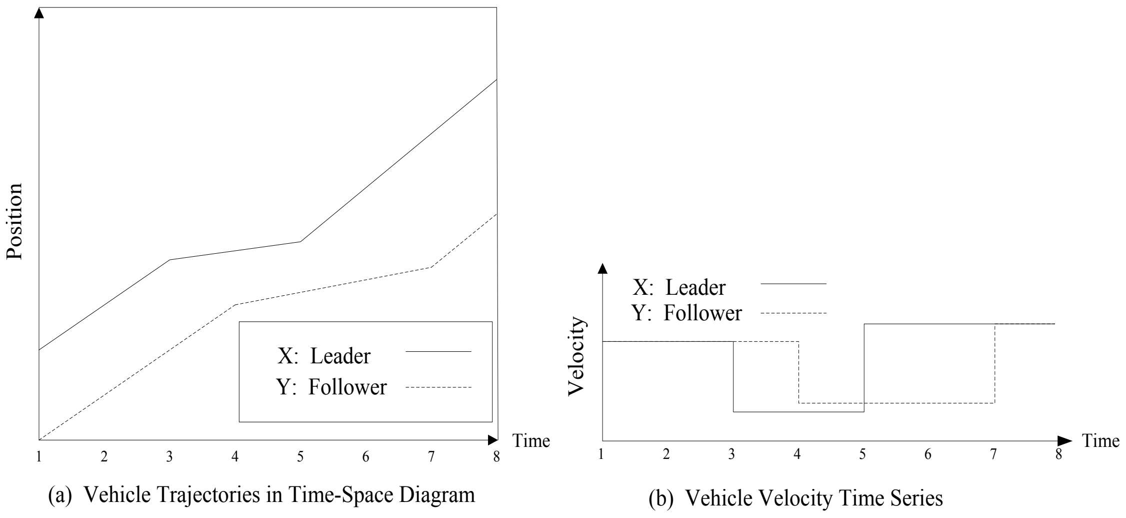
$\tau = t_j - t_i$

$d = p_i - q_j$

Figure 7: Visual representation of the warp path (highlighted in grey) in the cumulative cost matrix.

Table 1: Velocity time-series data for illustrative example.

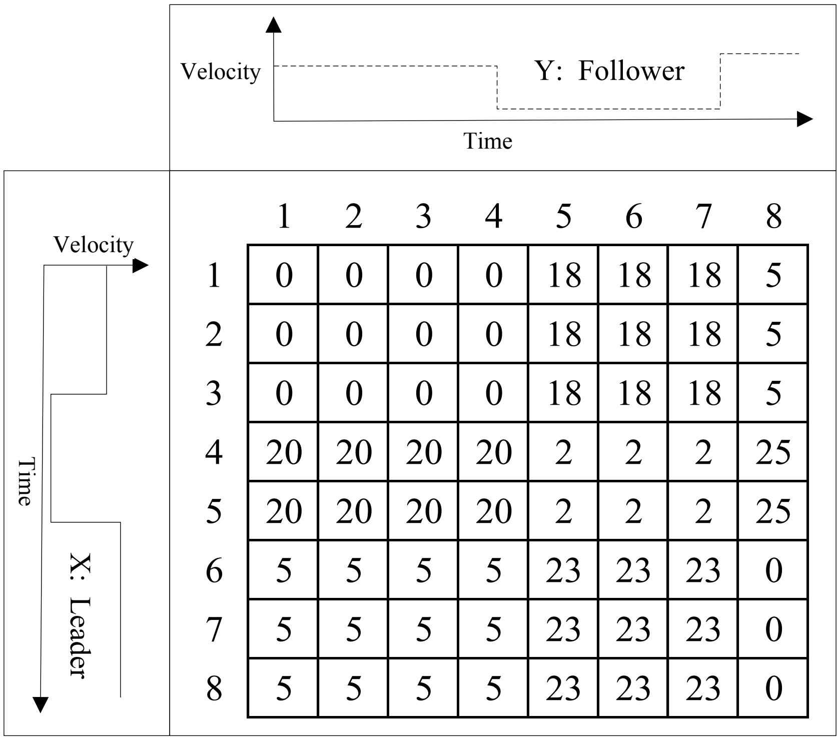
Time	1	2	3	4	5	6	7	8
X (Leader Velocity)	25	25	25	5	5	30	30	30
Y (Follower Velocity)	25	25	25	25	7	7	7	30



(a) Vehicle Trajectories in Time-Space Diagram

(b) Vehicle Velocity Time Series

Figure 8: Vehicle trajectory data for illustrative example, corresponding to the velocity data in Table 1.



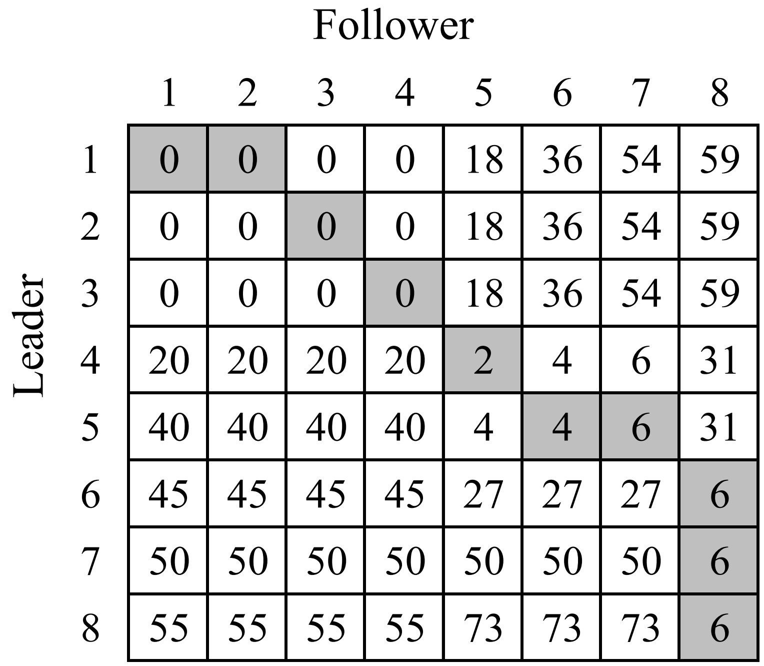
	1	2	3	4	5	6	7	8
1	0	0	0	0	18	18	18	5
2	0	0	0	0	18	18	18	5
3	0	0	0	0	18	18	18	5
4	20	20	20	20	2	2	2	25
5	20	20	20	20	2	2	2	25
6	5	5	5	5	23	23	23	0
7	5	5	5	5	23	23	23	0
8	5	5	5	5	23	23	23	0

Figure 9: Cost matrix for the illustrative example, with visual orientation of the leader and follower velocity data along the dimensions of the matrix.

Follower

Leader	1	2	3	4	5	6	7	8
1	0	0	0	0	18	36	54	59
2	0	0	0	0	18	36	54	59
3	0	0	0	0	18	36	54	59
4	20	20	20	20	2	4	6	31
5	40	40	40	40	4	4	6	31
6	45	45	45	45	27	27	27	6
7	50	50	50	50	50	50	50	6
8	55	55	55	55	73	73	73	6

Figure 10: Cumulative cost matrix for illustrative example.



Follower

Leader	1	2	3	4	5	6	7	8
1	**0**	**0**	0	0	18	36	54	59
2	0	0	**0**	0	18	36	54	59
3	0	0	0	**0**	18	36	54	59
4	20	20	20	20	**2**	4	6	31
5	40	40	40	40	4	**4**	**6**	31
6	45	45	45	45	27	27	27	**6**
7	50	50	50	50	50	50	50	**6**
8	55	55	55	55	73	73	73	**6**

Figure 11: Warp path (highlighted in grey) through the cumulative cost matrix for the illustrative example.

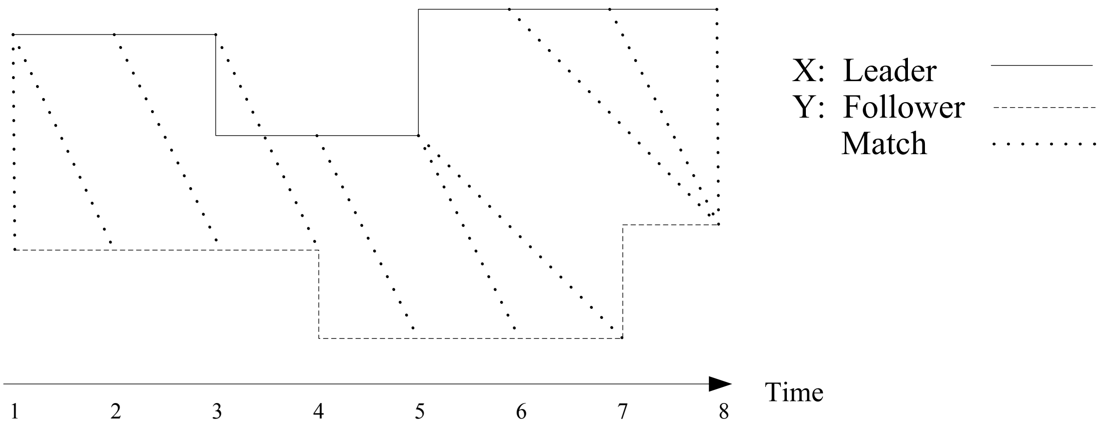


Figure 12: Matching/alignment between vehicle velocity data for illustrative example.

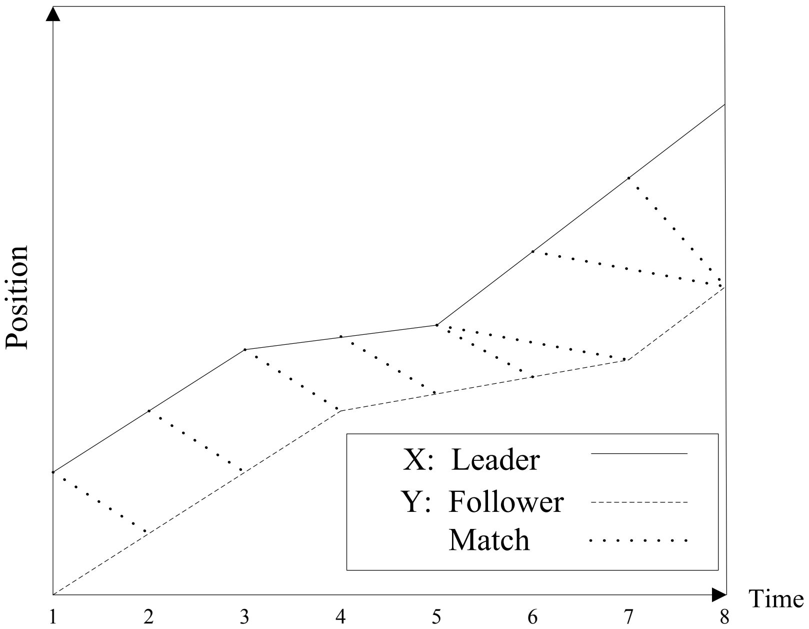


Figure 13: Matching/alignment between vehicle trajectory data for illustrative example.

CHAPTER 4

NUMERICAL EXPERIMENTS WITH NGSIM DATA

Starting with data from the NGSIM project for I-80 in California (NGSIM, 2006), we use the DTW algorithm to extract the optimal match points and analyze individual drivers' car-following parameters as they change over time. For this numerical experiment, we have extracted a series of vehicles from Lane 4, including both trucks and passenger cars. Data for these vehicles are available at 0.1 second resolution, and the dataset was trimmed to approximately 60 seconds so that data was available for all vehicles for each time index. The DTW algorithm was applied using the calculated acceleration to develop the cost matrix, with a lower bound constraint applied when calculating the cost matrix (i.e., artificial cost = 100). The methodology was applied with and without reduced input data. All DTW calculations and visualizations were performed with MATLAB.

4.1 Analysis Results with Data Reduction

First, the DTW algorithm is applied to a time series which has undergone data reduction. This was performed manually, where best judgment was used to form a piecewise linear approximation for each vehicle trajectory. The algorithm produced nine match points for the first following vehicle (the truck, highlighted in the middle red line in Fig. 14), and ten match points for the second following vehicle (the vehicle following the truck). The figure only shows six and seven plotted matches for the first and second following vehicles, respectively, because only acceptable solutions are plotted (i.e., $\tau > 0$, $d > 0$). Complete results for the matches are shown in the time series in Fig. 15.

The matching solution results, and especially in the congested region between $t = 5150$ and $t = 5400$, appear to show consistent backward wave speeds in multiple locations along the trajectory. Additionally, the wave speed also appears to change in the deceleration and acceleration regions, showing a slight trend toward decreasing before congestion and increasing after congestion. This presents the possibility that situation-dependent car-following parameters may exist, but does not conclusively prove or disprove their existence. Further study on a larger scale is required to investigate these characteristics. Further enhancements to the DTW algorithm which could also improve solution quality (which are not applied here) are discussed in the next section.

Examining the complete solution set in Fig. 15, we observe multiple solutions which are not within the boundary constraints ($\tau > 0$, $d > 0$) for the car-following model. The wave speeds at the end points (w = infinity) are ignored because the matching points create vertical lines with $\tau = 0$. We also observe that singularities are located near points of extreme results in the time-series plots, but their influence at these locations is not clear from the simple analysis provided in this numerical experiment.

4.2 Results without Data Reduction

The datasets are approximately 60 times larger without using the data reduction algorithm (approximately 600 data points for each trajectory). This produces a very different matching pattern compared to the reduced data matching solution. The match solution in Fig. 16 is focused upon an area within the plot shown in Fig. 14 so that the reader may inspect the solution quality. The changing slope is observed again here, indicating a changing wave speed. This plot presents a much stronger case for situation dependent parameters, where the reaction time increases greatly for the second following vehicle after congestion begins and increases after congestion ends. It appears that the matching solutions for both drivers align well with each other in some regions of the plot. However, singularities once again introduce an element of uncertainty in the matching solution. This uncertainty limits our ability to draw conclusions at this state in the research.

4.3 Example of Time Series Results for Car-Following Parameters

Translating the DTW matching results, following the procedures described in Chapter 3, produces car-following parameter estimates which can be represented as a time-series. An example is shown in Fig. 17 using a vehicle from the I-80 NGSIM dataset. Fig. 17 shows the estimated time lag, critical spacing, and backward wave speed (shown in blue, red, and green, respectively) at each time interval over the duration of the observed vehicle trajectory. Overlaid on top of this time-series data, the purple and gold lines show the space-time trajectory of the leader and follower, respectively. The DTW matching results in this case used prior estimates for the parameters to help reduce the effect of singularities in the experiment results.

The experimental results show a decrease in the time lag and critical spacing during a deceleration period at the beginning of the time series. After a few seconds, these parameters appear to recover back to some near-steady-state condition for each parameter (which is similar to the prior estimate values). In this case, the backward wave speed does not vary much over the duration of the time series, potentially indicating that the weight on this parameter is biasing the matching results to minimize its variation.

When applied to the entire I-80 dataset, these high-resolution car-following parameter estimates can be aggregated to estimate distributions for these parameters. Experimental results are shown in Fig. 18. for the time lag, critical spacing, and backward wave speed.

The time lag and critical spacing appear to have broad distributions with long tails at one end of the distribution. The frequency of parameter estimates in the tail could be attributed to unrealistic or inappropriate matching results (e.g., matching vehicles when they are too far apart), but they could also be related to different vehicle types in the dataset (e.g., trucks with long following distances). Further analysis by vehicle type would help to identify the source of these outlier parameter estimates. Additionally, the backward wave speed distribution appears to show a very tight distribution. This most likely can be attributed to an overestimated weight applied to the backward wave speed in the prior information formulation. That is, the higher weight on the backward wave term biases the matching results to minimize the deviation from the prior estimate for this parameter, thus providing an estimated distribution with a much narrower distribution.

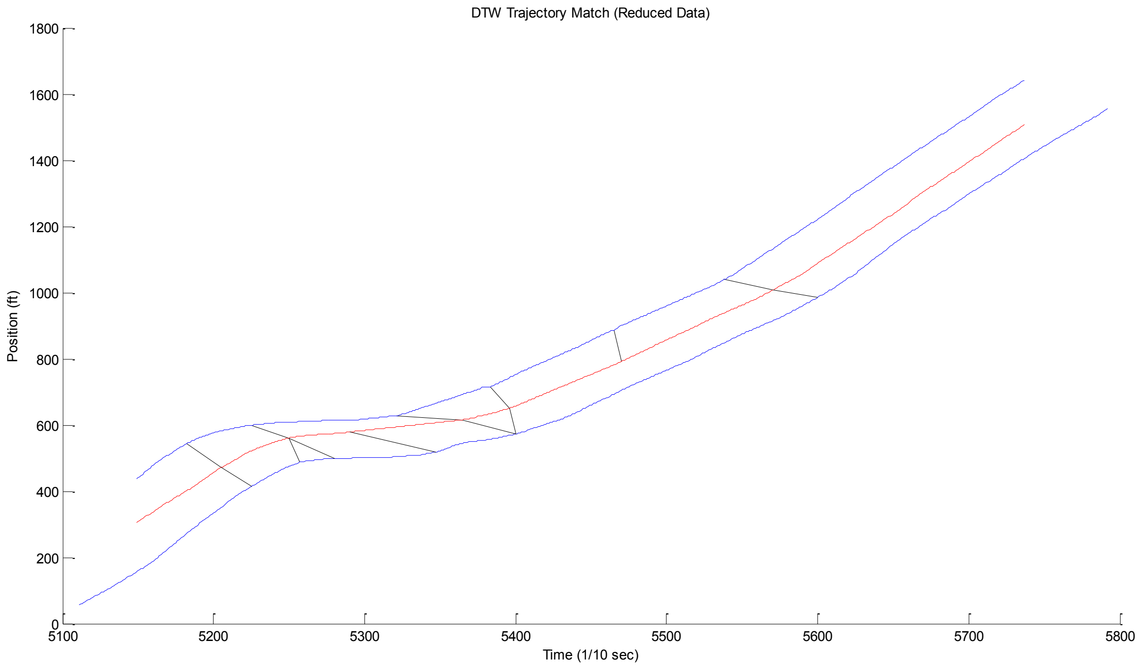


Figure 14: Dynamic Time Warping trajectory match for reduced data.

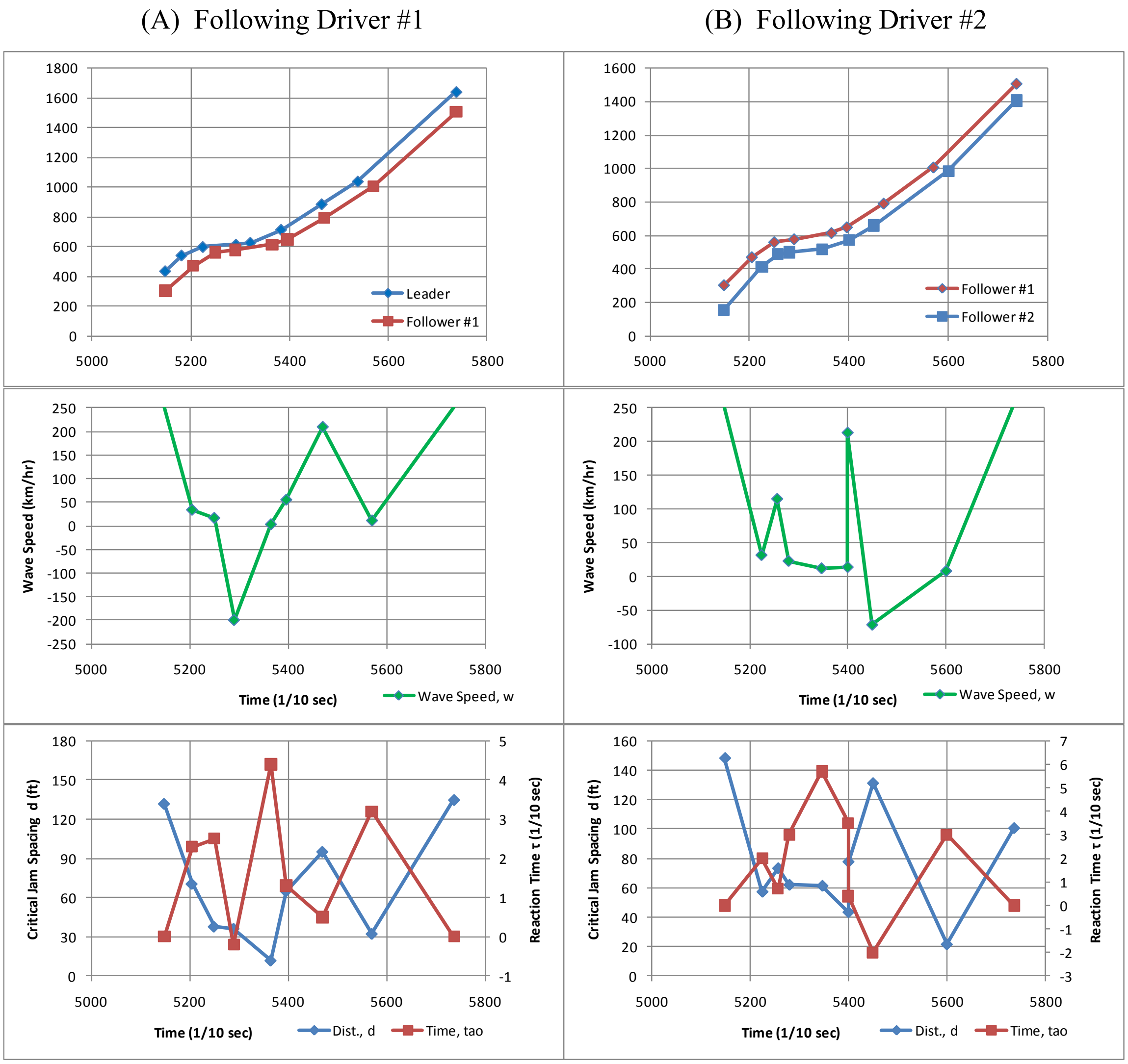


Figure 15: Time-series plots for car-following parameters d, τ, and w. The position of the leader and follower is plotted alongside the wave speed, time lag, and critical spacing parameter estimates for Following Driver 1 (A) and Following Driver 2 (B).

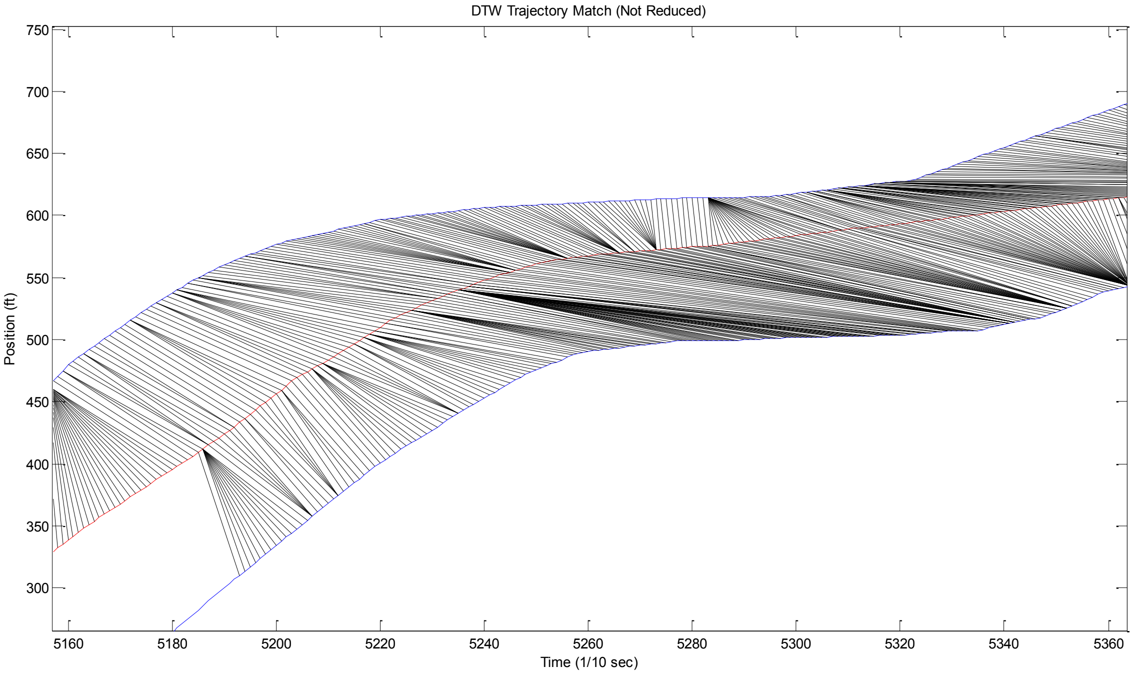
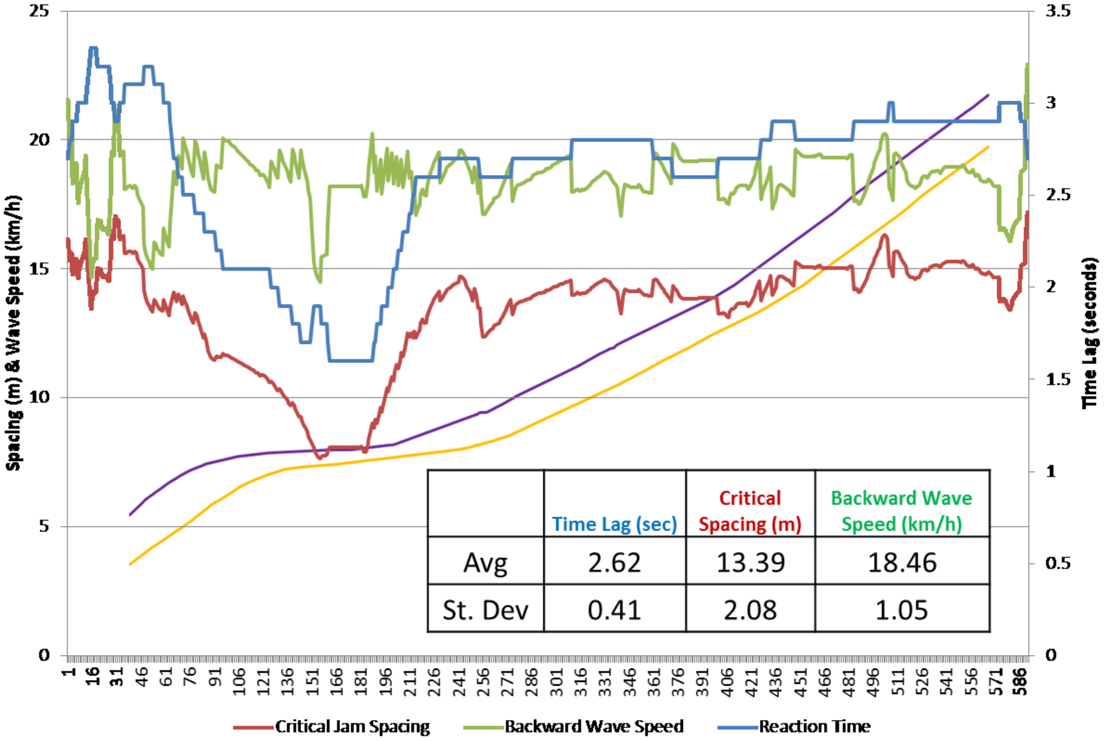


Figure 16: DTW trajectory match for unreduced data.



	Time Lag (sec)	Critical Spacing (m)	Backward Wave Speed (km/h)
Avg	2.62	13.39	18.46
St. Dev	0.41	2.08	1.05

Figure 17: Model parameter estimates for a single following vehicle, displayed as a time series (produced using prior information method).

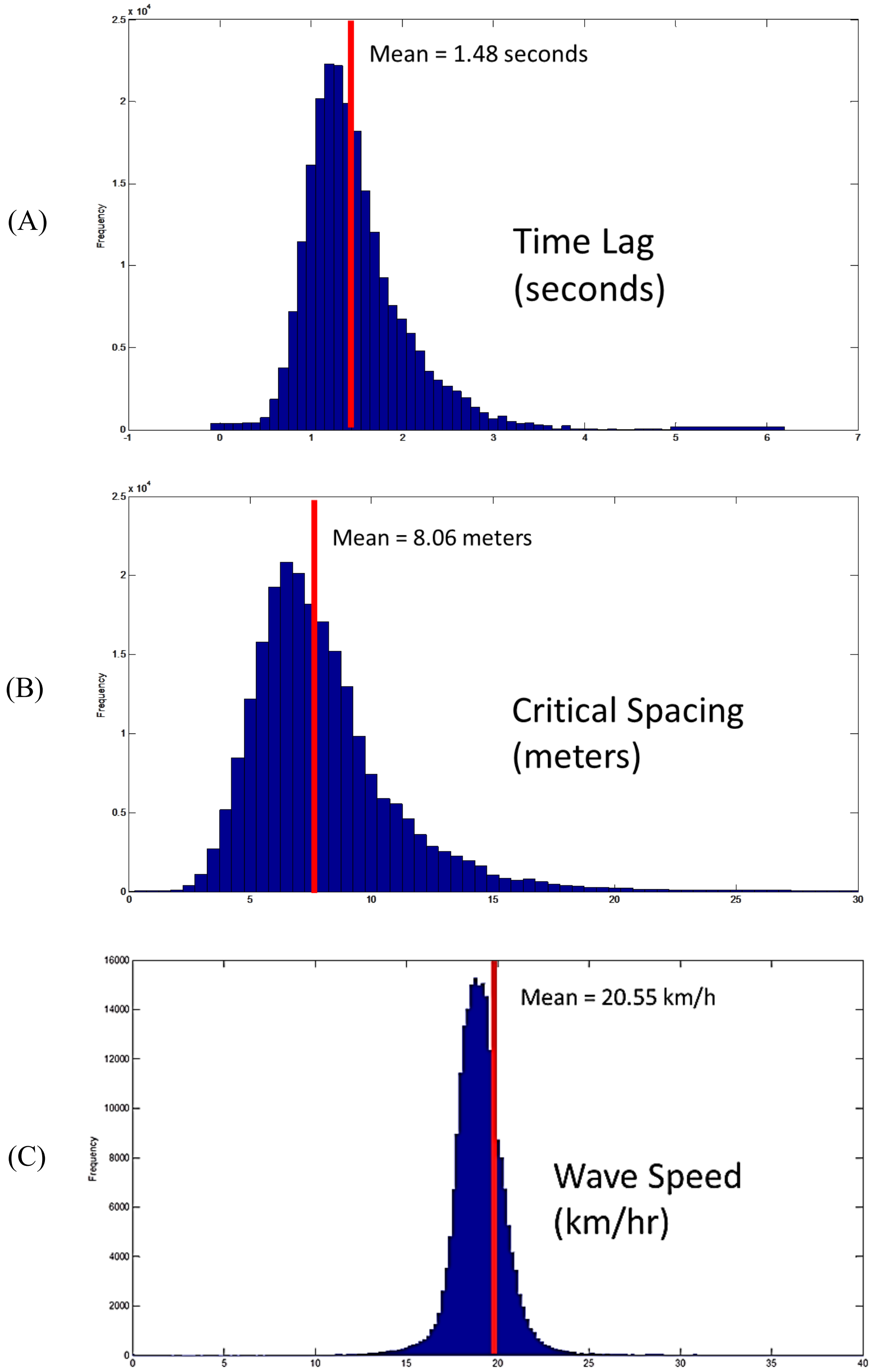


Figure 18: Estimated distributions for car-following parameters for the I-80 dataset. (A) Histogram describing the estimated time lag distribution. (B) Histogram describing the estimated critical spacing distribution. (C) Histogram describing the estimated backward wave speed distribution.

CHAPTER 5

DISCUSSION ON LIMITATIONS AND CHALLENGES

5.1 Limitations in DTW Input Data

Applying the DTW algorithm when working with vehicle trajectories requires some considerations for selecting input data, including the type of data, its time resolution, and the size of the datasets. The time-series input data for the DTW algorithm is two time-series datasets – one for the lead vehicle, and one for the following vehicle. Since the goal is to determine the driver's car-following parameters, the input data should come from the variable which forms the basis for the car-following model – velocity or acceleration. However, the algorithm is often applied with a distance measure used as the cost of aligning the datasets, where the distance is related to the difference in the two variables. This means that using velocity as the input will match the leader's velocity to the follower's velocity as it changes with time. If acceleration is chosen as the input, the match is performed based on the response to the change in velocity. From a purely data analysis standpoint, if the time series data are smoothed to the point of being composed of nearly constant velocities, matching based on velocity will result in a large number of singularities, making the results very unrealistic.

Data resolution is another issue of concern when working with DTW for vehicle trajectory matching. High resolution (0.1 seconds) vehicle trajectory data is widely available, but significantly increases the computational resources required by the DTW algorithm, especially for large datasets. It may be desirable to reduce the datasets to only the most important data points for each time series. However, more dispersed data points may result in unrealistic or undesirable matches, and data reduction further reduces the number of points available for analysis. A multiresolution approach may be necessary, where the matches are made between the reduced data points, followed by a second run through the algorithm for matching the trajectories between the reduced data points.

In many cases, the datasets may have different sizes, especially after any kind of data reduction algorithm is applied to the raw input data. The DTW algorithm can analyze datasets with different sizes, but this increases the number of singularities in the output data. While singularities may be undesirable in some cases, they may also be useful for different analyses, which will be discussed in the following section.

5.2 Singularities

Several variations of the DTW algorithm exist, each with their own unique features and components. Examples include Derivative DTW (Keogh and Pazzani, 2001), Fast DTW (Salvador and Chan, 2004), Multiscale DTW (Zinke and Mayer, 2006), and DTW with Piecewise Aggregate Approximation or PDTW (14), among many others. Similarly, many modifications have been made to this algorithm to reduce the incidence of "singularities" – a case where a large section of one time series is matched with a single point in the other time series, sometimes in undesirable or unexpected combinations. For vehicle trajectory analysis, a singularity exists when the follower's reaction is mapped to multiple actions by the leader, or multiple actions by the follower are mapped to a single action by the leader. This also tends to occur in regions with constant velocity, and when a car-following parameter changes compared to that estimated in a previous time period. Viewed as part of a warp path in a matrix, as in Fig. 19 (D), singularities occur when the path moves vertically or horizontally, rather than diagonally. Horizontal and vertical steps in the warp path indicate changes in the reaction time τ (horizontal = increases, vertical = decreases, diagonal = same).

From a theoretical standpoint, singularities offer an interesting new perspective for analysis while simultaneously complicating that analysis. The match results for singularities imply a more complicated following behavior than the underlying model, where one stimulus could result in multiple responses, and vice versa. Additionally, singularities could also be used to classify drivers, where multiple responses to a single stimulus could indicate more aggressive behavior. However, the degree to which singularities truly represent the leader-follower relationship, as opposed to artifacts of the algorithm, needs further study and analysis. Singularities have been considered undesirable in most studies using DTW, and a singularity must exist when datasets are not of equal size so that all points are matched. Additionally, a singularity may present multiple solutions at one point for the time-dependent model parameters, which raises the issue of which value to use for calibration.

As a result of these issues, we cannot conclude that a singularity accurately reflects the leader-follower response. At the same time, we can only assume that a more complicated driver behavior is not present. We can implement some algorithm enhancements to reduce the presence of singularities, but care must also be taken to ensure that the methods used to reduce singularities also do not produce undesirable singularities which may also affect solution quality. For example, Fig. 20 highlights a singularity which helps the algorithm transition from an impossibly-high w (nearly vertical slope) to more reasonable results.

5.3 Additional Enhancements for Singularity Reduction

Methods used to reduce the occurrence of singularities include, but are not limited to, windowing, slope weighting, and using different step patterns. Windowing is a process which limits the available number of matches to a single point based on a selected window width, which limits the size of a singularity. For vehicle trajectories, this method simply limits the calculated τ value for any given match to a range of reasonable values. This can also be used in conjunction with calculated d and w values for those match points to force the algorithm to always provide theoretically-acceptable matches.

Slope weighting adds coefficients to the cumulative cost terms in Eq. 6. Its implementation is a modified form of Eq. 6, which is shown in Eq. 11. The weight coefficients tend to encourage a more diagonal warp path through the cumulative cost matrix.

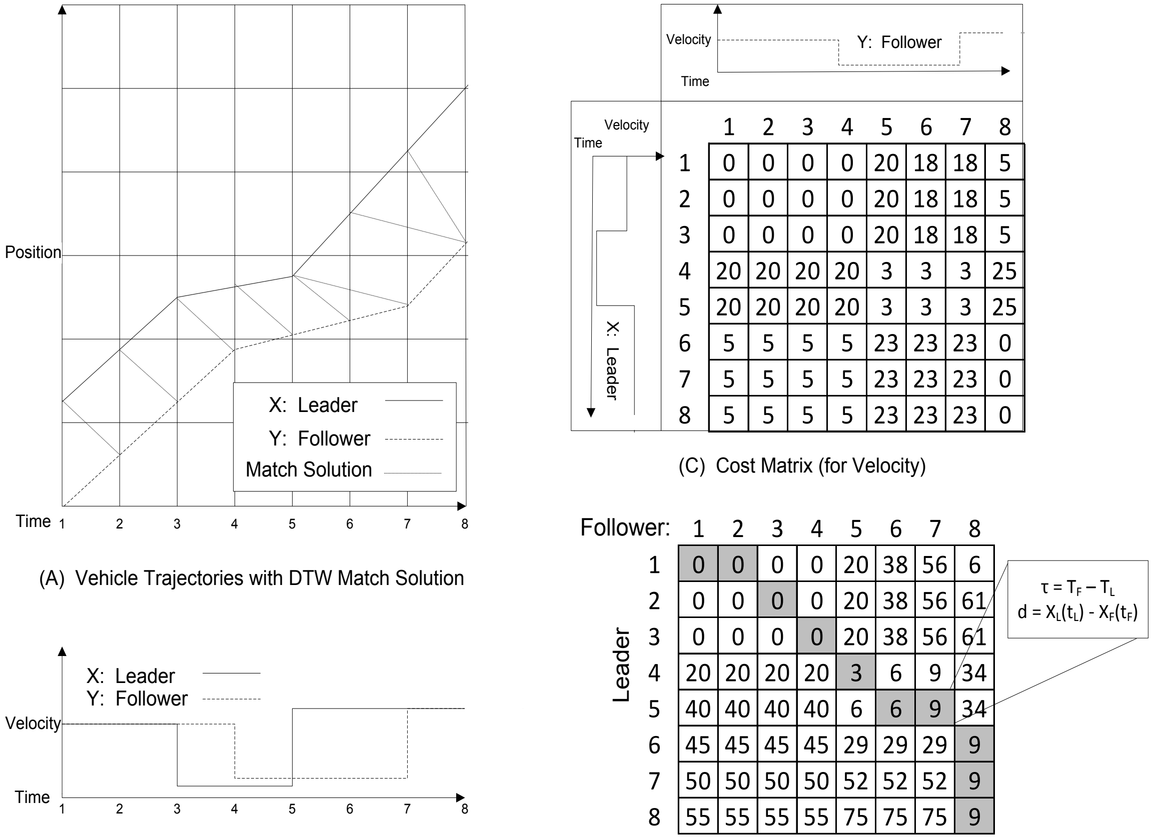
$$D(i,j) = C(i,j) + \min\big(D(i-1,j-1), \beta * D(i-1,j), \beta * D(i,j-1)\big) \qquad (11)$$

As the coefficients increase, the warp path should become more diagonal in nature. A more diagonal warp path limits the presence of singularities, but may also have implications for the resulting model parameters from that warp path. Since the algorithm must produce matches near the beginning and end of the dataset, it may require some "warm-up time" before it produces reasonable results. This transition usually requires singularities so that the reaction time changes from zero to a reasonable solution. Thus, an attempt to limit the formation of singularities may extend that "warm-up time." Additionally, if the driver's behavior changes such that the model parameters are different at that location in the trajectory, a singularity should be expected, but large slope weights may disguise that change.

Different step patterns can also be implemented in the cumulative cost calculation. This requires changing Eq. 6 so that the algorithm works with cells in the cumulative cost matrix that are more than one step away in each direction. An example of this approach is given in Eq. 12 below.

$$D(i,j) = C(i,j) + \min\big(D(i-1,j-1), D(i-2,j-1), D(i-1,j-2)\big) \qquad (12)$$

Again, this method increases the likelihood for a more diagonal warp path by forcing the path to move diagonally in addition to when it moves vertically or horizontally.



(A) Vehicle Trajectories with DTW Match Solution

(B) Vehicle Velocity Time Series

	1	2	3	4	5	6	7	8
1	0	0	0	0	20	18	18	5
2	0	0	0	0	20	18	18	5
3	0	0	0	0	20	18	18	5
4	20	20	20	20	3	3	3	25
5	20	20	20	20	3	3	3	25
6	5	5	5	5	23	23	23	0
7	5	5	5	5	23	23	23	0
8	5	5	5	5	23	23	23	0

(C) Cost Matrix (for Velocity)

Follower:	1	2	3	4	5	6	7	8
1	0	0	0	0	20	38	56	6
2	0	0	0	0	20	38	56	61
3	0	0	0	0	20	38	56	61
4	20	20	20	20	3	6	9	34
5	40	40	40	40	6	6	9	34
6	45	45	45	45	29	29	29	9
7	50	50	50	50	52	52	52	9
8	55	55	55	55	75	75	75	9

(D) Cumulative Cost Matrix with Highlighted Warp Path

Figure 19: Illustrative example of singularities as DTW results. (A) shows the vehicle trajectory data, (B) shows the velocity data representative of the trajectory data in (A), (C) shows the cost matrix calculated using the velocity data in (B), and (D) shows the warp path, with singularites represented by consecutive shaded cells in a column or row.

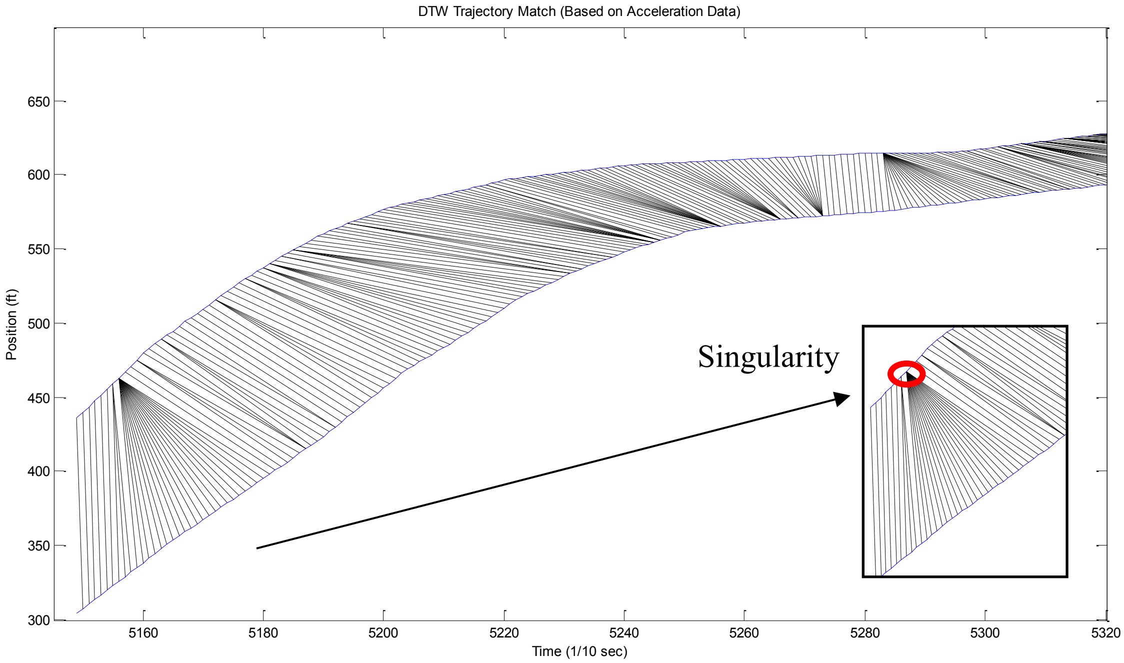


Figure 20: Example output for DTW vehicle trajectory match with highlighted singularity.

CHAPTER 6

CONCLUSIONS AND FUTURE STUDY

This paper describes a method for using the Dynamic Time Warping algorithm to calibrate an extension of Newell's car-following model incorporating time-dependent car-following parameters. The unique capabilities of the DTW algorithm may provide an efficient method for observing driver heterogeneity in car-following behavior, as well as the driver's heterogeneous situation-dependent behavior within a trip. Although the algorithm was made to analyze time-series data, several modification techniques are described to address specific challenges in this application and the algorithm solution quality for analyzing vehicle trajectories. A brief numerical experiment is presented with vehicle trajectory data extracted from the Next Generation Simulation (NGSIM) project, demonstrating the algorithm's ability to process large vehicle trajectory datasets, but significant data reduction and more algorithm modification may be necessary to produce more reasonable results. Additionally, singularities present an interesting match solution set to potentially help identify changing driver behavior, but they must be avoided to reduce analysis complexity and solution uncertainty. Future research could focus on algorithm enhancements with different traffic data sources (e.g., an extended version of Newell's three detector model by Deng et al., 2013), parameter validation methods, comparisons with alternative calibration methods, evaluating potential applications with other car-following models, and large-scale vehicle trajectory analysis to potentially explore situation-dependent driver behavior.

REFERENCES

Aghabayk, K., Forouzideh, N., & Young, W. (2013). Exploring a local linear model tree approach to car-following. *Computer-Aided Civil and Infrastructure Engineering, 28*(8), 581-593.

Ahn, S., Cassidy, M. J., & Laval, J. (2004). Verification of a simplified car-following theory. *Transportation Research Part B: Methodological, 38*(5), 431-440.

Brockfeld, E., Kühne, R., & Wagner, P. (2004). Calibration and validation of microscopic traffic flow models. *Transportation Research Record: Journal of the Transportation Research Board, 1876*, 62-70.

Chandler, R. E., Herman, R., & Montroll, E. W. (1958). Traffic dynamics: Studies in car following. *Operations Research, 6*(2), 165-184.

Chen, L., Ozsu, M. T., & Oria, V. (2005). Robust and fast similarity search for moving object trajectories. Paper presented at the *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, Baltimore, Maryland.

Chiabaut, N., Leclercq, L., & Buisson, C. (2010). From heterogeneous drivers to macroscopic patterns in congestion. *Transportation Research Part B: Methodological, 44*(2), 299-308.

Chu, S., Keogh, E., Hart, D., & Pazzani, M. (2002). Iterative deepening dynamic time warping for time series. Paper presented at the *Second SIAM International Conference on Data Mining*, Arlington, Virginia.

Ciuffo, B., Punzo, V., & Quaglietta, E. (2011). Kriging meta-modelling to verify traffic micro-simulation calibration methods. Paper presented at the *90th Annual Meeting of the Transportation Research Board*, Washington, D.C.

Deng, W., Lei, H., & Zhou, X. (2013). Traffic state estimation and uncertainty quantification based on heterogeneous data sources: A three detector approach. *Transportation Research Part B: Methodological, 57*, 132-157.

Duret, A., Buisson, C., & Chiabaut, N. (2008). Estimating individual speed-spacing relationship and assessing ability of Newell's car-following model to reproduce trajectories. *Transportation Research Record: Journal of the Transportation Research Board, 2088*, 188-197.

Gazis, D. C., Herman, R., & Rothery, R. W. (1961). Nonlinear follow-the-leader models of traffic flow. *Operations Research, 9*(4), 545-567.

Gipps, P. G. (1981). A behavioural car-following model for computer simulation. *Transportation Research Part B: Methodological, 15*(2), 105-111.

Hamdar, S. H., Treiber, M., & Mahmassani, H. S. (2009). Genetic algorithm calibration for a stochastic car-following model using trajectory data: Exploration and model properties. Paper presented at the *88th Annual Meeting of the Transportation Research Board*, Washington, D.C.

Hoogendoorn, S. P., Hoogendoorn, R. G., & Daamen, W. (2011). Wiedemann revisited: A new trajectory filtering technique and its implications for car-following modeling. Paper presented at the *90th Annual Meeting of the Transportation Research Board*, Washington, D.C.

Keogh, E., & Pazzani, M. (2001). Derivative dynamic time warping. Paper presented at the *First SIAM International Conference on Data Mining (SDM'2001)*, Chicago, Illinois.

Kesting, A., & Treiber, M. (2008). Calibrating car-following models by using trajectory data: Methodological study. *Transportation Research Record: Journal of the Transportation Research Board, 2088*, 148-156.

Kesting, A., & Treiber, M. (2009). Calibration of car-following models using floating car data. In: Appert-Rolland, C., Chevoir, F., Gondret, P., Lassarre, S., Lebacque, J.-P. and Schreckenberg, M., eds. *Traffic and Granular Flow '07*, 2009. Springer Berlin Heidelberg, 117-127.

Kesting, A., & Treiber, M. (2013). *Traffic flow dynamics: Data, models and simulation*. Berlin, Germany: Springer.

Kim, I., Kim, T., & Sohn, K. (2013). Identifying driver heterogeneity in car-following based on a random coefficient model. *Transportation Research Part C: Emerging Technologies, 36*, 35-44.

Kim, J., & Mahmassani, H. (2011). Correlated parameters in driving behavior models. *Transportation Research Record: Journal of the Transportation Research Board, 2249*(-1), 62-77.

Laval, J. A., & Leclercq, L. (2010). A mechanism to describe the formation and propagation of stop-and-go waves in congested freeway traffic. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 368*(1928), 4519-4541.

Ma, X., & Andréasson, I. (2006). Estimation of driver reaction time from car-following data: Application in evaluation of General Motor-type model. *Transportation Research Record: Journal of the Transportation Research Board, 1965*, 130-141.

MATLAB. (2010). Natick, Massachusetts: The MathWorks, Inc.

Newell, G. F. (1962). Theories of instability in dense highway traffic. *Journal of the Operations Research Society of Japan, 5*(1), 9-54.

Newell, G. F. (1993). A simplified theory of kinematic waves in highway traffic, part I: General theory. *Transportation Research Part B: Methodological, 27*(4), 281-287.

Newell, G. F. (2002). A simplified car-following theory: A lower order model. *Transportation Research Part B: Methodological, 36*(3), 195-205.

Ossen, S., & Hoogendoorn, S. (2005). Car-following behavior analysis from microscopic trajectory data. *Transportation Research Record: Journal of the Transportation Research Board, 1934*, 13-21.

Ossen, S., Hoogendoorn, S., & Gorte, B. (2006). Interdriver differences in car-following: A vehicle trajectory-based study. *Transportation Research Record: Journal of the Transportation Research Board, 1965*, 121-129.

Ossen, S., & Hoogendoorn, S. (2007). Driver heterogeneity in car following and its impact on modeling traffic dynamics. *Transportation Research Record: Journal of the Transportation Research Board, 1999*, 95-103.

Ossen, S., & Hoogendoorn, S. (2008a). Validity of trajectory-based calibration approach of car-following models in presence of measurement errors. *Transportation Research Record: Journal of the Transportation Research Board, 2088*, 117-125.

Ossen, S., & Hoogendoorn, S. (2008b). *Calibrating car-following models using microscopic trajectory data: A critical analysis of both microscopic trajectory data collection methods and calibration studies based on these data* (Version 1.1 ed., pp. 109). Delft, Netherlands: Delft University of Technology.

Ossen, S., & Hoogendoorn, S. (2009). Reliability of parameter values estimated using trajectory observations. *Transportation Research Record: Journal of the Transportation Research Board, 2124*, 36-44.

Ossen, S., & Hoogendoorn, S. (2011). Heterogeneity in car-following behavior: Theory and empirics. *Transportation Research Part C: Emerging Technologies, 19*(2), 182-195.

Przybyla, J., Taylor, J., Jupe, J., & Zhou, X. (2012, 16-19 Sept. 2012). Simplified, data-driven, errorable car-following model to predict the safety effects of distracted driving. Paper presented at the *15th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2012, Anchorage, Alaska.

Sakoe, H., & Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on, 26*(1), 43-49.

Salvadore, S., & Chan, P. (2004). FastDTW: Toward accurate dynamic time warping in linear time and space. Paper presented at the *3rd Workshop on Mining Temporal and Sequential Data*, Seattle, WA, USA.

Senin, P. (2008). *Dynamic time warping algorithm review*. Information and Computer Science. University of Hawaii at Manoa. Honolulu, HI.

Treiber, M., Hennecke, A., & Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *Physical Review E, 62*(2), 1805-1824.

US Department of Transportation (USDOT). (2006). NGSIM - Next Generation Simulation. Available from http://ngsim-community.org.

Wang, H., Wang, W., Chen, J., & Jing, M. (2010). Using trajectory data to analyze intradriver heterogeneity in car-following. *Transportation Research Record: Journal of the Transportation Research Board, 2188*(-1), 85-95.

Yang, H., Gan, Q., & Jin, W.-L. (2011). Calibration of a family of car-following models with retarded linear regression methods. Paper presented at the *90th Annual Meeting of the Transportation Research Board*, Washington DC.

Zinke, A., & Mayer, D. (2006). Iterative Multi Scale Dynamic Time Warping. Technical Report CG-2006-1, *Universität Bonn*.