Developing Analysis, Modeling, and Simulation Tools for Connected and Automated Vehicle Applications

Algorithm Description Document: Coordinated Merge Model

August 2020

U.S. Department of Transportation
Federal Highway Administration

Research, Development, and Technology
Turner-Fairbank Highway Research Center
6300 Georgetown Pike
McLean, VA 22101-2296

Notice

This document is disseminated under the sponsorship of the U.S. Department of Transportation (USDOT) in the interest of information exchange. The U.S. Government assumes no liability for the use of the information contained in this document.

The U.S. Government does not endorse products or manufacturers. Trademarks or manufacturers' names appear in this report only because they are considered essential to the objective of the document.

Quality Assurance Statement

The Federal Highway Administration (FHWA) provides high-quality information to serve Government, industry, and the public in a manner that promotes public understanding. Standards and policies are used to ensure and maximize the quality, objectivity, utility, and integrity of its information. FHWA periodically reviews quality issues and adjusts its programs and processes to ensure continuous quality improvement.

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle Developing Analysis, Modeling, and Simulation Tools for Connected and Automated Vehicle Applications Algorithm Description Document: Coordinated Merge Model		**5. Report Date** August 2020 **6. Performing Organization Code:**	
7. Author(s) Alireza Talebpour and Zhitong Huang		**8. Performing Organization Report No.**	
9. Performing Organization Name and Address Leidos Inc. 11251 Roger Bacon Drive Reston, VA 20190		**10. Work Unit No.** **11. Contract or Grant No.** DTFH61-12-D-00030, TO 22	
12. Sponsoring Agency Name and Address Office of Operations Research and Development Federal Highway Administration 6300 Georgetown Pike McLean, VA 22101-2296		**13. Type of Report and Period Covered** **14. Sponsoring Agency Code**	
15. Supplementary Notes The government task managers were John Halkias and Gene McHale.			

16. Abstract

This study proposes a coordinated merge (CM) model which jointly optimizes the lateral trajectory of a lane-changing vehicle and the longitudinal control of an impacted vehicle in the target lane. The objective is to minimize the disturbance caused by the merging process in the target lane. Therefore, the prediction of the impact of the lane-changing on a platoon of vehicles is shown. Towards this objective: i) first, a model predictive controller (MPC) is designed for the vehicle in the target lane to properly respond to a merge maneuver; ii) then, a set of possible lateral trajectories is specified that may be adopted by the lane-changing vehicle; iii) then, the required deceleration of the vehicle directly impacted by the lane-changing is computed based on the estimated set of the lateral trajectories; iv) finally, the trajectory that minimizes the total deceleration effort (i.e., minimizes the disturbance in the platoon of vehicles) is chosen. Analytical and simulation-based investigations are performed to assess the capability of the proposed approach in minimizing the acceleration disturbance. The simulation results show that communicating the information on the lane-changing maneuver can help connected and automated vehicles in the target lane execute more reliable and efficient maneuvers.

Another objective of this document is to help future users easily to adapt and customize this CM model in a traffic simulation tool to meet their simulation needs. To this end, this document describes the algorithms/logic of this model in detail. It also illustrates how this model was developed, calibrated, and validated. The pseudocode of this model is included in the appendix.

17. Key Words	18. Distribution Statement		
Connected and automated vehicle, traffic simulation, microsimulation, coordinated merging, speed harmonization, and market penetration rate.	No restrictions.		
19. Security Classif. (of this report) Unclassified	**20. Security Classif. (of this page)** Unclassified	**21. No. of Pages** 59	**22. Price** N/A

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized.

SI* (MODERN METRIC) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	square millimeters	mm^2
ft^2	square feet	0.093	square meters	m^2
yd^2	square yard	0.836	square meters	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	square kilometers	km^2
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	cubic meters	m^3
yd^3	cubic yards	0.765	cubic meters	m^3
		NOTE: volumes greater than 1,000 L shall be shown in m^3		
		MASS		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2,000 lb)	0.907	megagrams (or "metric ton")	Mg (or "t")
		TEMPERATURE (exact degrees)		
°F	Fahrenheit	5 (F-32)/9 or (F-32)/1.8	Celsius	°C
		ILLUMINATION		
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	candela/m^2	cd/m^2
		FORCE and PRESSURE or STRESS		
lbf	poundforce	4.45	newtons	N
lbf/in^2	poundforce per square inch	6.89	kilopascals	kPa

APPROXIMATE CONVERSIONS FROM SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	square millimeters	0.0016	square inches	in^2
m^2	square meters	10.764	square feet	ft^2
m^2	square meters	1.195	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	square kilometers	0.386	square miles	mi^2
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	cubic meters	35.314	cubic feet	ft^3
m^3	cubic meters	1.307	cubic yards	yd^3
		MASS		
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or "t")	megagrams (or "metric ton")	1.103	short tons (2,000 lb)	T
		TEMPERATURE (exact degrees)		
°C	Celsius	1.8C+32	Fahrenheit	°F
		ILLUMINATION		
lx	lux	0.0929	foot-candles	fc
cd/m^2	candela/m2	0.2919	foot-Lamberts	fl
		FORCE and PRESSURE or STRESS		
N	newtons	2.225	poundforce	lbf
kPa	kilopascals	0.145	poundforce per square inch	lbf/in^2

*SI is the symbol for International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380. (Revised March 2003)

TABLE OF CONTENTS

EXECUTIVE SUMMARY 1
CHAPTER 1. PURPOSE OF THIS MODEL 3
Purpose of this Document 3
Purpose of this Model 3
Document Overview 4
CHAPTER 2. MODEL DEVELOPMENT AND LOGIC 5
Descriptions of Model Logic 5
Interaction of Speed Harmonization and Merge Coordination 16
CHAPTER 3. MODEL CALIBRATION AND VALIATION 17
Data Set 17
Calibration Approach 19
Calibration and Validation Process 20
CHAPTER 4. BASIC INFORMATION ON MODEL IMPLEMENTATION 23
CHAPTER 5. USER CASE AND SENSITIVITY STUDY 25
Implementation of the developed model into a traffic simulation tool 25
Design of simulation experiments AND Simulation results 28
CHAPTER 6. SUMMARY AND RECOMMENDATIONS 39
REFERENCES 41
APPENDIX 43
ACKNOWLEDGEMENTS 59

LIST OF FIGURES

Figure 1. Illustration. Trajectory generation approaches: occupancy grid (left) and state lattice. . 7
Figure 2. Equation. Spacing error between the predecessor and its follower. 8
Figure 3. Equation. Acceleration of a string stable platoon. 8
Figure 4. Equations. Longitudinal dynamics equations. 9
Figure 5. Equations. Discretized longitudinal dynamics equations. 9
Figure 6. Equation. Cost function for making the safe distance. 10
Figure 7. Equation. Cost function for speed recovery. 10
Figure 8. Equation. Constraints for the safe recovery problem. 10
Figure 9. Formulas. Trajectory curve for the lane-changing maneuver. 11
Figure 10. Illustration. Sets of feasible trajectories depending on the gap choice. 12
Figure 11. Formulas. Optimization problem for finding the best lane-changing trajectory. 12
Figure 12. Equation. Total control effort. 13
Figure 13. Flowchart. Strategy for the accurate prediction of model predictive control behavior. 13
Figure 14. Illustration. Time headway adjustment: relative position comparison: no dynamic model predictive control (up) versus dynamic model predictive control (down). 14
Figure 15. Formulas. Optimization problem for case 1: maximizing distance. 15
Figure 16. Formulas. Optimization problem for case 2: new gap to make is determined. 16
Figure 17. Photo. Data collection location on Interstate 35 near Austin, Texas. 18
Figure 18. Photo. Vehicle detection and tracking in aerial images. 19
Figure 19. Illustration. Sample trajectory data collected on Interstate 35 near Austin, Texas. 19
Figure 20. Equation. Error in the gap between the lead vehicle and the target vehicle. 19
Figure 21. Equation. Definition of 〈 〉. 20
Figure 22. Flowchart. Implementation of merge coordination system. 24
Figure 23. Formula. Value function for the uncongested regime. 25
Figure 24. Formula. Value function for the congested regime. 25
Figure 25. Formula. Binary probabilistic regime selection model. 26
Figure 26. Formula. Total utility function for the choice of acceleration. 26
Figure 27. Formula. Probability density function for the evaluation of drivers' stochastic response 26
Figure 28. Formula. The intelligent driver acceleration model. 27
Figure 29. Formula. Maximum speed of automated vehicles. 27
Figure 30. Formula. Acceleration model for automated vehicles. 28
Figure 31. Formula. Safe following distance formula. 28
Figure 32. Formula. Acceleration of automated vehicles. 28
Figure 33. Diagrams. Maintaining speed: both lanes speed = 82 ft/s; (a) model predictive control-based platoon and (b) Swaroop's controller based platoon. 30
Figure 34. Diagrams. Increasing speed: current lane speed = 65 ft/s, target lane speed = 82 ft/s; (a) model predictive control-based platoon and (b) Swaroop's controller based platoon. 31
Figure 35. Diagram. Range of feasible value of α_6: constrained by maximum longitudinal acceleration. 33
Figure 36. Diagrams. Lane-changing maneuver profiles (adaptive cruise control-based platoon): (a) $\alpha_6 = 0$ (lower order polynomial), and (b) $\alpha_6 = 0.0034$ (optimal). 34
Figure 37. Diagrams. Percent of adaptive cruise control-based cars in the platoon 36

Figure 38. Graph. Effects of different variance constraints on time headway of three vehicles ahead of the lane-changing vehicles in the target lane for $h_{desired}$ = 1.0 seconds, h_{safety} = 0.6 seconds, and sum of change of h = 1.0 seconds. All the graphs. 37
Figure 39. Graph. Impact. 38
Figure 40. Graph. Effects of different variance constraints on speed variance of three vehicles ahead of the lane-changing vehicles in the target lane for $h_{desired}$ = 1.0 seconds, h_{safety} = 0.6 seconds, and sum of change of h = 1.0 seconds. All the graphs. 38

LIST OF TABLES

Table 1. Lane-changing trajectory generation methods. 7
Table 2. Car-following model calibration results for human following. 21
Table 3 . Car-following model calibration results for autonomous vehicle following. 21

LIST OF ABBREVIATIONS

ACC	adaptive cruise control
ADAS	advanced driver assistance systems
AMS	analysis, modeling, and simulation
AV	automated vehicle
CAV	connected and automated vehicle
CACC	cooperative adaptive cruise control
CV	connected vehicle
CM	coordinated merge
IDM	intelligent driver model
NGSIM	Next Generation SIMulation
TMC	traffic management center
TTI	travel time index
UAV	unmanned aerial vehicle
USDOT	United States department of transportation
V2V	vehicle-to-vehicle
V2I	vehicle-to-infrastructure

EXECUTIVE SUMMARY

It is known that merging maneuvers cause traffic disruption and result in shockwave formation and propagation. This study proposes a coordinated merge (CM) model which jointly optimizes the lateral trajectory of a lane-changing vehicle and the longitudinal control of an impacted vehicle in the target lane. The objective is to minimize the disturbance caused by the merging process in the target lane. Therefore, the prediction of the impact of the lane-changing on a platoon of vehicles is shown. Towards this objective: i) first, a model predictive controller (MPC) is designed for the vehicle in the target lane to properly respond to a merge maneuver; ii) then, a set of possible lateral trajectories is specified that may be adopted by the lane-changing vehicle; iii) then, the required deceleration of the vehicle directly impacted by the lane-changing is computed based on the estimated set of the lateral trajectories; iv) finally, the trajectory that minimizes the total deceleration effort (i.e., minimizes the disturbance in the platoon of vehicles) is chosen. Analytical and simulation-based investigations are performed to assess the capability of the proposed approach in minimizing the acceleration disturbance. The availability of Vehicle-to-Vehicle (V2V) communication is assumed. Therefore, the lane-changing vehicle and the vehicle directly impacted by this maneuver in the target lane can share their acceleration profile. The simulation results show that communicating the information on the lane-changing maneuver can help connected and automated vehicles in the target lane execute more reliable and efficient maneuvers.

Another objective of this document is to help future users easily to adapt and customize this CM model in a traffic simulation tool to meet their simulation needs. To this end, this document describes the algorithms/logic of this model in detail. It also illustrates how this model was developed, calibrated, and validated. The pseudocode of this model is included in the appendix.

This research has been performed under a FHWA project entitled “Developing Analysis, Modeling, and Simulation Tools for Connected and Automated Vehicle Applications” (contract number: DTFH6116D00030-0022). To get more information of this FHWA project, readers are encouraged to reference the final project report of this project (Lu et al, forthcoming). This report is under FHWA publication process and it will be available soon.

CHAPTER 1. PURPOSE OF THIS MODEL

PURPOSE OF THIS DOCUMENT

Connected and automated vehicle (CAV) technologies offer potentially transformative societal impacts, including significant mobility, safety, and environmental benefits. State and local agencies are interested in harnessing the potential benefits of CAVs. However, for agencies to be able to plan beneficial deployments of infrastructure-to-vehicle (I2V) and vehicle-to-vehicle (V2V) technology, it is important to be able to robustly predict the impacts of such deployments and identify which applications best address their unique transportation problems. Traffic analysis, modeling, and simulation (AMS) tools provide an efficient means to evaluate transportation improvement projects before deployment.

However, current AMS tools are not well suited for evaluating CAV applications due to their inability to represent vehicle connectivity and automated driving features. The development of a new generation of tools involves spending a lot of resources and time to develop, calibrate, and validate. Many independent researchers have developed models of CAV systems based on a divergent array of underlying assumptions. As a result, there is little consensus in the literature regarding the most likely impacts of CAV technologies.

Thus, there is a desire for a consistent set of models to produce realistic and believable predictions of CAV impacts. These models can be based on the best available data and include the most accurate possible representations of the behaviors of drivers of conventional vehicles and CAVs. Deployment concepts, strategies, and guidelines are also key for allowing State and local agencies to understand how and where to deploy CAV technologies.

To meet these goals, the Federal Highway Administration (FHWA) sponsored a project entitled "Developing Analysis, Modeling, and Simulation Tools for Connected and Automated Vehicle Applications" (contract number: DTFH6116D00030-0022). This project aimed to develop AMS models for the most prominent CAV applications and to incorporate these models into existing AMS simulation tools. Three CAV applications were developed under this project: a lane changing (LC) model for light duty CAVs, a combined application model that integrates speed harmonization (SPDHRM) and coordinated merge (CM), and an improved cooperative adaptive cruise control (CACC) model for light duty CAVs. The final project report of this project (Lu et al, forthcoming) is under FHWA publication process and it will be available soon.

This document presents the CM model in detail. The objective of this document is to provide detailed information about this model to benefit CAV simulation community. This document is expected to help future users easily adapt and customize this model in a traffic simulation tool to meet their simulation needs. To this end, this document describes the algorithms/logic of this model in detail. It also illustrates how this model was developed, calibrated, and validated. The pseudocode of this model is included in the appendix.

PURPOSE OF THIS MODEL

To illustrate the importance of considering both longitudinal and the lateral movements, this study focuses on lane-changing maneuvers at merge locations; lane-changing maneuvers can significantly impact traffic flow by creating disturbances in both initial and target lanes potentially resulting in shockwave formation and propagation. Depending on the lane-changing

trajectory and the traffic conditions in the origin and target lanes, the significance of the disturbance (i.e., shockwave magnitude and duration) can change. For instance, a sudden lane-changing maneuver in the synchronized flow regime can create a more severe shockwave compared with the same lane-changing maneuver effect in the free-flow regime. However, this study intends to show that with careful coordination between lane-changing and longitudinal movements at the merge location, one can potentially eliminate the shockwave formation or reduce its magnitude to a great extent.

In order to explore the impact of both lateral and longitudinal movements on traffic flow dynamics and shock wave formation, the researchers focus on developing a merge coordination application. Lane-changing (LC) maneuvers are considered a major source of disturbance in the traffic stream and their impact is more prominent at the merging points. The shock wave magnitude and duration resulted from an LC depends on the LC trajectory, traffic conditions in the origin and target lanes, and the response of the vehicles in the target lane to LC vehicle. The main focus of this model development effort is to illustrate that with careful coordination between the LC vehicle and other vehicles in the target lane, one can significantly reduce the shock wave magnitude and duration (and even completely prevent it in some cases).

Accordingly, the researchers propose a combined optimization approach for both lateral and longitudinal movements of the vehicles involved in the LC maneuver (both the LC vehicle and vehicles in the target lane). The proposed approach consists of: (1) designing a model predictive control (MPC) for the CF behavior of vehicles in the target lane and (2) optimizing the LC trajectory to minimize the total deceleration effort in the target lane (i.e., the amount of deceleration that occurs in the target lane to accommodate the LC vehicle). Note that the LC maneuver could have a pre-defined duration.

DOCUMENT OVERVIEW

This document will introduce a new merge coordination model capable of significantly reducing or eliminating shockwaves at merge locations. In the following sections, we introduce the model details and the logic behind each element in the model. Model calibration and validation is discussed next. This section is followed by a simulation-based analysis of the impacts of the proposed model on traffic flow dynamics. The document is concluded with a summary of the findings followed by an appendix containing the model pseudocode.

CHAPTER 2. MODEL DEVELOPMENT AND LOGIC

This section discusses the model development procedure and provides details about the overall design of the merge coordination model.

DESCRIPTIONS OF MODEL LOGIC

The merge coordination model consists of two core modules: CF and LC. In the following, the logic behind both modules is presented.

Car-Following Behavior

Interest in vehicle automation started several decades ago. From the CF behavior perspective, vehicle automation studies have mainly focused on vehicle platooning and its various derivations. Vehicle platooning and platoon control design are considered necessary to achieve a connected automated transportation systems. Moreover, the role of communications (V2V and V2I) on the performance of the platooning/CF behavior have been investigated extensively.

Several platooning strategies have been proposed in the literature. These strategies can be categorized into three distinct categories: (1) constant spacing, where the platoon controller is focused on keeping a fixed spacing between vehicles, (2) constant time headway, where the platoon controller is focused on keeping a fixed time headway between vehicles, and (3) variable time headway, where fluctuations in time headway are allowed to dampen the disturbances in the platoon. In general, the effectiveness of a platooning strategy and its associated controller have been investigated based on the concept of string stability. String stability is concerned with how the disturbance propagates through the string (Swaroop 1997). In a string stable platoon, any disturbance has to attenuate along the platoon and cannot stay for a long time. Several studies investigated string stability in ACC and CACC systems. Seiler et al. ((Seiler, Pant et al. 2004), by thinking of a platoon of vehicles as a system, describes how the disturbance caused by a vehicle propagates in the platoon. This type of research framework is mostly based on dealing with transfer functions between an initial disturbance and other states affected by the disturbance, such as position error or the acceleration of followers. Many studies have adopted a similar approach to investigate string stability. For example, Maschuw et al. (Maschuw, Keßler et al. 2008) and Kianfar et al. (Kianfar, Ali et al. 2014) investigated how a disturbance affects the acceleration of followers. These studies proposed a methodology to reduce the overshoot in followers' velocity and acceleration caused by the disturbance.

In addition to the platooning strategies that only rely on onboard sensors, several studies have explored the impact of V2V and V2I communications and the information from other vehicles and infrastructure on the string stability. In Swaroop (Swaroop 1997), information about the platoon leader and the predecessor were found important for string stability in CACC. Swaroop showed that communication with the platoon leader is the necessary condition to ensure string stability in a platoon controlled by the constant spacing policy. Seiler et al. (Seiler, Pant et al. 2004) showed that the availability of the leader information is key to ensure string stability. It also compared the platooning results with the case without leader information. Wang and Rajamani (2004) compared constant time headway and constant spacing strategies. Recently, Talebpour and Mahmassani (2016) investigated string stability in a platoon of CAVs and showed that both connectivity and automation increases the speed above which instabilities start to occur. In Talebpour et al. (2017), important concepts have been tackled to ensure connectivity in a

mixed driving environment with CAVs. The authors showed how the correlation between communication range and connected vehicle density affects the connectivity level and the string stability of the traffic flow. Arefizadeh and Talebpour (2018) proposed a new constant time headway strategy for AV platoons to prevent shock wave formation caused by speed drops.

Lane-Changing Maneuver

Many approaches have been suggested to generate LC trajectories in a safe and reliable way. Typically, adopting a function that describes the geometric representations of trajectory has been a dominant approach. Various functions have been used in the literature (see Table 1 for a summary). In order to compare several LC trajectory functions, continuity and smoothness were considered.

Circular arcs: This is a simple approach to describe an LC trajectory. By putting circular arcs together, a simple trajectory can be obtained, and the radius of curvature is determined by lateral and longitudinal displacement after an LC maneuver. Based on the constant curvature, the lateral position can be determined depending on the velocity and the corresponding longitudinal position. However, this method has certain disadvantages; for instance, additional consideration is involved when dealing with the discontinuity issue that happens from connecting two arcs. More importantly, speed adjustment during the LC maneuver can become very complicated.

Polynomial: This is the most common approach to represent an LC trajectory. The trajectory can be generated by considering position constraints that the vehicle has to pass through. The key advantages of this approach are that these curves are easy to compute and can always generate a continuous curvature. Moreover, the polynomial can be of any order, and being able to consider higher order polynomials (polynomials higher than 4 degrees) gives more flexibility in terms of designing the trajectory. Normally, using a higher order increases flexibility in terms of trajectory shaping. In Cong et al. (Cong, Sawodny et al. 2010), a fourth order polynomial was used and constraints like maximum acceleration and maximum driving force were considered. In another study, Nelson (Nelson 1989) used a fifth order polynomial to describe the lateral position of the vehicle.

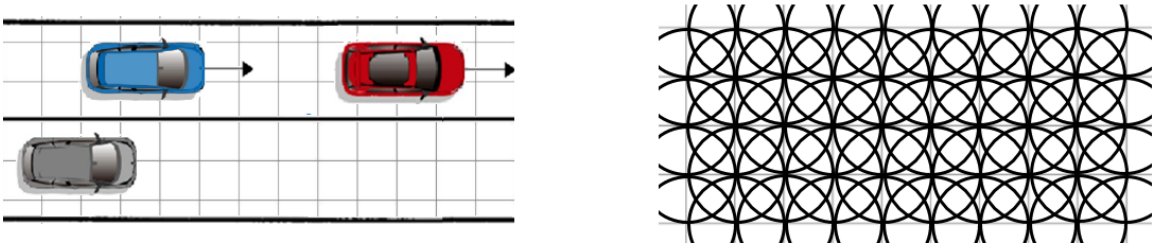
Bezier curves: Originally, Bezier curves were used in the field of computer graphics to obtain the curves. A Bezier curve is based on a concept called control points. Rastelli et al. (Rastelli, Lattarulo et al. 2014) showed that it is possible to shape the trajectory curve depending on the convex hull based on the control points. Therefore, this approach is a very intuitive way to manipulate the trajectory and it has low computational cost to generate. González et al. (González, Pérez et al. 2014) obtained the LC trajectory using a Bezier curve considering the roadway constraints.

Spline curves: In a nutshell, this method is a piecewise polynomial curve. After generating points, each sub-interval between points can be described using any type of curve, including polynomial and circular (Piazzi et al. 2002; Bacha et al. 2008).

Occupancy grid: By discretizing the search space into a grid—as shown in Figure 1—and checking the possibility that a grid is occupied by an obstacle, a feasible trajectory can be generated. Out of many graph-based approaches, this method is relatively straightforward to implement and computationally efficient. In Schroder et al. (Schroder, Gindele et al. 2008) and Kolski et al. (Kolski, Ferguson et al. 2006), for each cell, the possibility of being occupied by an obstacle and the corresponding risk were calculated to check the feasibility of a trajectory.

State lattice: This approach also starts from the reconstruction of the environment but into a different search space, called state lattice, as shown in Figure 1. Illustration. Trajectory generation approaches: occupancy grid (left) and state lattice (right).. Pivtoraiko and Kelly (Pivtoraiko and Kelly 2005) and Pivtoraiko et al. (Pivtoraiko, Knepper et al. 2009) generated the trajectory based on state lattices that guarantee the feasibility of the path.

Table 1 summarizes lane-changing trajectory generation methods descried above. For this project, the polynomial-based LC trajectory generation method was adopted. Several factors contributed to this selection. First, the approach has a low computation overhead and can use the boundary conditions to generate the polynomial that best describes the optimal trajectory. Second, by adding the sixth order term, the method can optimize trajectory curvature to minimize impacts on the target lane. These features provide the trajectory optimization process with the opportunity to predict the behavior of the MPC and to minimize the disturbance caused by the LC maneuver.



Source: FHWA.

Figure 1. Illustration. Trajectory generation approaches: occupancy grid (left) and state lattice (right).

Table 1. Lane-changing trajectory generation methods.

Methods	Description
Circular arc	Interpolation of waypoints using a circle
Polynomial	Interpolation of waypoints using polynomial curves
Bezier curve	Generating curves based on selected the Bezier control points
Spline	Generating a piecewise trajectory
Occupancy grid	Generating a feasible trajectory based on the obstacle locations in a grid
State lattice	Generating a feasible trajectory based on the obstacle locations in a state lattice

Formal Problem Statement

The main focus of this study is to develop a merge coordination application to minimize the negative impacts of the merging process on the target lane. During the merge coordination process, the lane-changing vehicle (LV) identifies the appropriate gap to initiate the merge process. The merge coordination process ends with LV joining the platoon with SV as the new follower. SV is the key vehicle in the proposed merge coordination application and will often be referenced as the platoon head (since a backward-moving shock wave starts from SV and moves backward in traffic). However, this reference does not mean that the platoon is divided into two separate platoons. On the contrary, both vehicles in front of LV and behind LV will be involved in the developed merge coordination application to ensure a smooth LC maneuver.

Merge Coordination Model Development

Car-Following Controller

The goal here is to capture the full impact of LV on the target lane. Accordingly, SV is controlled based on the MPC and the remaining vehicles in the target lane can either be HVs, controlled by IDM (Treiber, Hennecke et al. 2000), or AVs, controlled based on a well-established fix time headway car-following policy (Swaroop 1997). The overview of these modeling frameworks is presented below. More details can be found in An and Talebpour (An and Talebpour 2019).

Platoon Members: Constant Time Headway Policy

The constant time headway (CTH) policy of Swaroop (Swaroop 1997) was adopted in this section to represent the CF behavior of AVs. This model ensues a safe distance between vehicles by adjusting the vehicle speed/acceleration. As the first step, the spacing error between vehicles can be calculated in Figure 2 as follows:

$$\delta(t) = (x_s - x_l) + Tv_l + l$$

Figure 2. Equation. Spacing error between the predecessor and its follower.

where x_s, x_l, T, v_l, and l denote the follower location, predecessor location, desired time headway, predecessor velocity, and vehicle length, respectively. Once the spacing error is calculated, the equation in Figure 2 can be utilized to calculate the acceleration:

$$a(t) = \frac{1}{T}\left[v_s(t) - v_l(t) + \lambda\delta(t)\right]$$

Figure 3. Equation. Acceleration of a string stable platoon.

Where $\lambda > 0$ ensure string stability in the platoon. Note that as indicated by An and Talebpour (An and Talebpour 2019), Swaroop's model (1997) can be extremely inefficient dealing with sudden changes in time headway (e.g., case of the LC into a platoon). More details on the behavior of this model when dealing with sudden changes in time headway is presented in the next section.

Platoon Head: Model Predictive Control Approach

In order to address the issues associated with constant time headway policy, this study utilizes an MPC-based control for SV. Following the information provided by An and Talebpour (An and Talebpour 2019), the CF dynamics can be formulated as follows:

$$\dot{v}_s = a_s$$

(a) Relationship between acceleration and speed for the follower

$$\dot{v}_l = a_l$$

(b) Relationship between acceleration and speed for the leader

$$\dot{x} = (v_l - v_s) + \frac{1}{2}(a_l - a_s)t_{sampling}$$

(c) Relationship between follower location, follower speed, leader speed, follower acceleration, and leader acceleration

Figure 4. Equations. Longitudinal dynamics equations.

where v_s, v_l, a_s, and a_l denote velocity of the follower, velocity of the leader, acceleration of the follower, and acceleration of the leader, respectively. $t_{sampling}$ is the sampling duration. The equation in Figure 5 can be discretized using the Euler method for simulation purposes.

$$v_s(k+1) = v_s(k) + a_s(k)t_{sampling}$$

(a) Relationship between acceleration and speed for the follower

$$v_l(k+1) = v_l(k) + a_l(k)t_{sampling}$$

(b) Relationship between acceleration and speed for the leader

$$x(k+1) = x(k) + (v_l(k) - v_s(k))t_{sampling} + \frac{1}{2}(a_l(k) - a_s(k))t_{sampling}{}^2$$

(c) Relationship between follower location, follower speed, leader speed, follower acceleration, and leader acceleration

Figure 5. Equations. Discretized longitudinal dynamics equations.

The key difference between the constant time headway policy and the MPC design is the fact that MPC works based on the projected trajectory of LV to calculate the acceleration for SV. Assuming that LV can share its movement profile (i.e., trajectory and speed), MPC can use actual a_l to predict v_l. Accordingly, MPC can accurately predict T based on a_l and a_s. This opens up the opportunity to utilize a cost function to ensure the optimal T by generating an optimal control sequence based on a_s, i.e., $a_s(k)- a_s(k-1)$, where k is the timestep. Two cost function are defined two address (1) SV's safety requirement to slow down to prevent a collision with LV and (2) SV's desired speed requirement. The details of these cost functions are discussed below.

First Cost Function: Keeping a Safety Distance

The first cost function is concerned with creating the safe distance between SV and LV. This is the key as sudden slow down (similar to the constant time headway policy) can result in a significant shock wave. The cost function is formulated in Figure 6 as follows (An and Talebpour 2019):

$$\left\|\widehat{\boldsymbol{T}}_{\boldsymbol{current}(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{s})} - \widehat{\boldsymbol{T}}_{(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{s})}\right\|_{\boldsymbol{R_1}}^{\boldsymbol{2}} + \left\|\widehat{\boldsymbol{a}}_{\boldsymbol{s}(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{s})} - \widehat{\boldsymbol{a}}_{\boldsymbol{l}(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{s})}\right\|_{\boldsymbol{R_2}}^{\boldsymbol{2}} + \left\|\Delta\boldsymbol{a}_{(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{k})}\right\|_{\boldsymbol{R_3}}^{\boldsymbol{2}}$$

Figure 6. Equation. Cost function for making the safe distance.

where T is the desired time headway and $T_{current}$ is the current time headway. The first part of the equation tries to capture the deviation from the desired time headway. The second part of the equation ensures consistency between the acceleration behavior of SV and LV. In other words, SV tries to mimic the acceleration/deceleration behavior of SV while maintaining a desired time headway. The last part of the equation ensures comfort and prevents sudden changes in acceleration values.

Second Cost Function: Speed Recovery

The second cost function is concerned with recovering the speed after slowing down to accommodate the LC vehicle. The cost function is formulated in Figure 7 as follows (An and Talebpour 2019):

$$\left\|\widehat{\boldsymbol{v}}_{\boldsymbol{s}(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{s})} - \widehat{\boldsymbol{v}}_{\boldsymbol{0}(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{s})}\right\|_{\boldsymbol{Q_1}}^{\boldsymbol{2}} + \left\|\widehat{\boldsymbol{a}}_{\boldsymbol{s}(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{s})} - \widehat{\boldsymbol{a}}_{\boldsymbol{l}(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{s})}\right\|_{\boldsymbol{Q_2}}^{\boldsymbol{2}} + \left\|\Delta\boldsymbol{a}_{(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{k})}\right\|_{\boldsymbol{Q_3}}^{\boldsymbol{2}} + \left\|\boldsymbol{a}_{(\boldsymbol{k}+\boldsymbol{i}/\boldsymbol{k})}\right\|_{\boldsymbol{Q_3}}^{\boldsymbol{2}}$$

Figure 7. Equation. Cost function for speed recovery.

where v_0 is the desired speed and v_s is the SV velocity. The first term in this equation captures the deviation from the desired speed. The second and third terms are similar to the first cost function. The fourth term is included to capture the preference to zero acceleration (i.e., minimum possible response to LV).

Combined with the following constraints, the above cost functions can result in the optimal acceleration choice, shown in Figure 8 (An and Talebpour 2019):

$$\begin{aligned} \text{Subject to:} \quad & s_{k+1} = f(s_k, \Delta a_s(k)) && k = t, \dots, t + N_p \\ & \Delta a_{min} \le \Delta a_{s(k)} \le \Delta a_{max} && k = t, \dots, t + N_p \\ & a_{s(k)} = a_{s(k-1)} + \Delta a_{s(k)} && k = t, \dots, t + N_p \end{aligned}$$

Figure 8. Equation. Constraints for the safe recovery problem.

where $s = [v_s, v_l, T]$ and N_p denote the prediction horizon. Note that the prediction horizon is shorter for the first cost function compared to the second cost function. This ensures a quick adaptation to the LC vehicle. The first constraint in Figure 8 models the longitudinal dynamics, predicted based on control input. The second constraint limits the control input and the third constraint illustrates how acceleration value can be calculated over the prediction horizon based on the control input.

Choosing the correct model weights is the key to a successful implementation of the aforementioned MPC design. Consideration and use of different weight values depends on the scenario. R_1 captures the sensitivity to the deviation from the time headway. High values of R_1 mean that the vehicle would be sensitive to small changes in the time headway. R_2 determines how fast SV could adjust to the changes in LV's acceleration choices. The combination of R_1 and R_2 ensures naturalistic CF behavior by determining how quickly the algorithm can react to the

reduction in the time headway and how aggressively it can respond to it. Accordingly, by adjusting these weights, the desired MPC design can be reached. Note that similar argument can be made for Q_1 and Q_2.

Considering the above discussion, the following parameter values are selected in this study (An and Talebpour 2019):

- Sampling time ($t_{sampling}$): 0.2 s.
- Prediction horizon (N_p): 2 for safe distance (Figure 6) and 10 for speed recovery (Figure 7).
- Constraints on acceleration change per $t_{sampling}$: -8.20 ft/s^2 $\leq \Delta a_{s(k)} \leq$ 8.20 ft/s^2.

Lane-Changing Trajectory

As discussed previously, there are several methods to model the LC trajectory. This study utilizes a higher-order polynomial approach.

Trajectory Generation: Sixth Order Polynomial

Building on the findings of Papadimitriou and Tomizuka (2003), this study introduces a sixth order polynomial for the movement in the *x*-axis and a fifth order polynomial for the movement in the *y*-axis. Since, unlike Papadimitriou and Tomizuka (2003), the object is not to avoid objects but to minimize the negative impacts of the LC maneuver on the target lane, this study utilizes a fix LC duration, t_{LC}. The following equations in Figure 9 can be formulated (An and Talebpour 2019):

$$x(t) = \alpha_6 t^6 + \alpha_5 t^5 + \alpha_4 t^4 + \alpha_3 t^3 + \alpha_2 t^2 + \alpha_1 t$$

(a) Trajectory along the x-axis

$$y(t) = \beta_5 t^5 + \beta_4 t^4 + \beta_3 t^3 + \beta_2 t^2 + \beta_1$$

(a) Trajectory along the y-axis

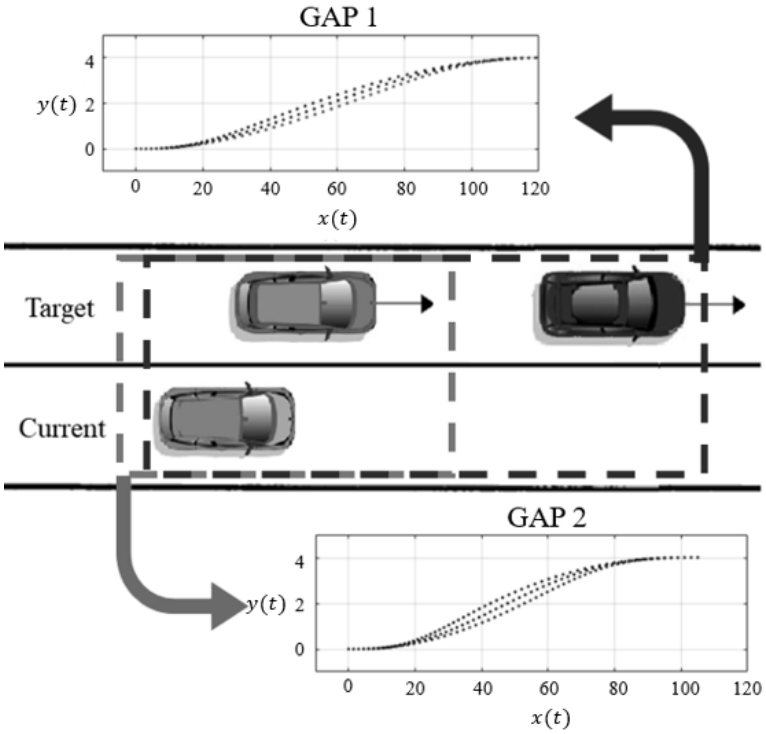
Figure 9. Formulas. Trajectory curve for the lane-changing maneuver.

By fixing the LC duration, one can fix the initial and final boundary conditions (i.e., location, speed, and acceleration values before the start of the LC maneuver and at the end of it). Therefore, changing a_6 results in change in the curvature of the LC trajectory; thus, different impact on the SV is expected for different values of a_6.

Trajectory Optimization

The overall schematic for identifying the optimum trajectory is depicted in Figure 10. For the two vehicles in the target lane, the LC vehicle in the current lane evaluates two sets of possible trajectories for final boundary conditions considering a desired time headway after the LC maneuver and the desired maximum lateral acceleration. The time headway can be chosen for individual vehicles or in accordance with the platooning strategy. Since t_{LC} can directly impact the outcome of the LC maneuver, a carefully selected value is suggested. Large values of t_{LC} can make LC infeasible/unrealistic and small values of it can result in significant shock wave formation. Accordingly, this study utilizes the maximum acceleration values (in both *x* and *y* directions) to calculate the minimum t_{LC}. The largest value of t_{LC} is selected among all possible

candidates to ensure both feasibility of the LC maneuver and the minimum impact on the target lane.



Source: FHWA.

Figure 10. Illustration. Sets of feasible trajectories depending on the gap choice.

In this research, based on the findings of Toledo and Zohar (Toledo and Zohar 2007), the LC duration was fixed to 6 s. The following optimization system is formulated in Figure 11 to identify the best LC trajectory (An and Talebpour 2019):

$$
\begin{aligned}
& Z = Min(S) \\
\text{Subject to:} \quad & a_x(t) = a_x^{max} && t = 0, \dots, T \\
& a_y(t) = a_y^{max} && t = 0, \dots, T \\
& a_x(t) = \frac{d^2x(t)}{dt^2} && t = 0, \dots, T \\
& a_y(t) = \frac{d^2y(t)}{dt^2} && t = 0, \dots, T
\end{aligned}
$$

Figure 11. Formulas. Optimization problem for finding the best lane-changing trajectory.

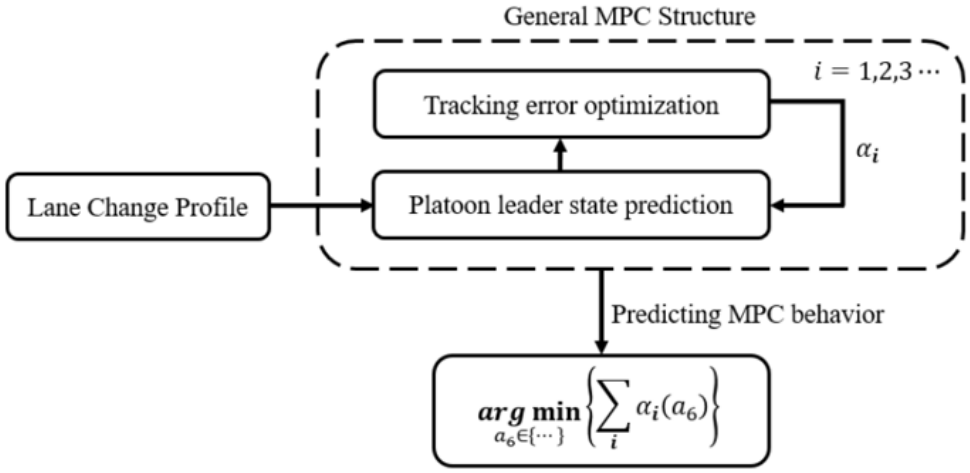
where S is the total deceleration effort of SV, a_x^{max} is the maximum longitudinal acceleration, and a_y^{max} is the maximum lateral acceleration. The equation shown in Figure 12 is based on An and Talebpour (An and Talebpour 2019):

$$S = \sum_{k=1}^{K} a_k$$

Figure 12. Equation. Total control effort.

Where $k = 1$ denotes the time when SV starts to respond to LV by reducing its speed, and $k = K$ denotes the time when SV starts the speed recovery process. Note that calculating S is the core of this optimization process.

Assuming that SV shares the MPC design with LV through V2V communications, LV can predict the SV acceleration profile for any given LC trajectory. Accordingly, the total deceleration effort can be evaluated for different values of a_6, as shown in Figure 13. Note that in this research, the constraint on the maximum longitudinal acceleration can only affect the range of feasible a_6. Then, multiple sets of possible trajectories are obtained by considering each vehicle in the target lane as the future leader. From there, LV chooses the optimal trajectory based on the effect of each trajectory on the target lane calculated using the objective function.



Source: FHWA.
MPC = model predictive control.

Figure 13. Flowchart. Strategy for the accurate prediction of model predictive control behavior.

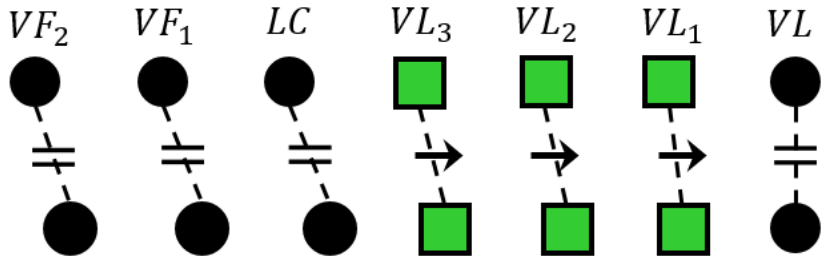
The key element in this optimization system is the methodology to predict the states of each vehicle since each LC trajectory has a different impact on the target lane and imposes a unique disturbance to the system. A methodology was developed to evaluate the impact of each trajectory on the changes in vehicles' speed in the target lane, thus capturing the impact of LC on total travel time. After generating a trajectory for a given set of available gaps, the model predicts the acceleration profile of the platoon head according to the designed MPC. Note that it is assumed that the LC vehicle has information about the MPC design.

Gap Generation

In addition to designing an MPC controller to mitigate the negative impacts of LC at the merge locations, one can generate enough gap for the LC vehicle by introducing a forward moving shockwave. Prior to controlling cars ahead of an LV, a set of time headway (h) values could be determined for the vehicles. Time headway values will be smaller than or equal to the original

time headway (h_0) that platoon members follow before the system starts to work. The change of relative position of platoon members in the platoon is illustrated in Figure 14. In this figure, circles represent vehicles, and two platoons are compared. The platoon above is when platoon members keep their constant h_0, and the one below is when some of platoon members (green rectangles) reduce their h. In other words, the system makes vehicles move forward relative to their original position in a platoon. Green rectangles are cars controlled to achieve their new h, which is smaller than or equal to h_0, and we can see that their resulting position are relatively ahead of their assumed position with the constant time headway, h_0. Consequently, LV is assigned a space to join in the target lane as a new member of the platoon. Automatically, LV will have less impact on the vehicle following it.

In this project, two formulations were considered to generate the h distribution: one approach is to maximize the generated gap, and the other approach is to provide LV the necessary gap based on the final position of the LV at the end of the LC maneuver. Note that it is assumed that the final position of the LV will be communicated to the vehicles.



Source: FHWA.

Figure 14. Illustration. Time headway adjustment: relative position comparison: no dynamic model predictive control (up) versus dynamic model predictive control (down).

Case 1: Maximizing Distance

This case maximizes a new space given the particular number of vehicles, N, to control and the constraint on their h distribution. The strategy to distribute time headways of all N vehicles based on the time headway variances. The following optimization in Figure 15 is developed:

$$Z = Max\left(\sum_{i=1}^{N}\left(T_0 - T_{current,i}\right)\right)$$

(a) Objective function

Subject to: $Var\left(T_{current,1}, T_{current,2}, \dots, T_{current,N}\right) \leq \max\ \_var_1$

(b) Constraint 1: Maximum variance considering all time headways

$Var(T_{current,i-1} - T_{current,i}) \leq \max\ \_var_2 \qquad \forall i = 0, 1, \dots, N$

(c) Constraint 2: Maximum variance between consecutive time headways

$T_{current,i} \geq T_{safety} \qquad \forall i = 0, 1, \dots, N$

(d) Constraint 3: minimum safe time headway

$T_{current,i} \geq T_{current,i+1} \qquad \forall i = 0, 1, \dots, N$

(e) Constraint 4: Relationship between consecutive time headways

Figure 15. Formulas. Optimization problem for case 1: maximizing distance.

where $T_{current,i}$ denotes the headway of the *i*th vehicle. In detail, checking the Var($T_{current,1}$, $T_{current,2}$, … , $T_{current,N}$) is based on the idea that large variance of vehicles' speed can result in shock wave formation and unsafe driving instances. Therefore, by having a constraint on the variance of $T_{current,i}$, the set of $T_{current,i}$ is obtained that does not cause a variance larger than a particular number. As long as the variance of a set does not exceed the safe limit, $T_{current,i}$s are acceptable.

Case 2: New Gap to Make Is Determined

This case assumes that LV defines the size of the necessary gap in advance. In such a case, all the *N* vehicles can be controlled focusing on making the gap just larger than a particular size instead of maximizing it. Here, it is assumed that the platoon can receive the information on the expected gap from LV before the execution of the LC maneuver. The following optimization in Figure 16 is formulated:

$$Z = Min\left(\sum_{i=1}^{N}\left(T_0 - T_{current,i}\right)\right)$$

(a) Objective function

Subject to: $Var\left(T_{current,1}, T_{current,2}, \dots, T_{current,N}\right) \leq \max\ \_var_1$

(b) Constraint 1: Maximum variance considering all time headways

$$Var(T_{current,i-1} - T_{current,i}) \leq \max\ \_var_2 \qquad \forall i = 0, 1, \dots, N$$

(c) Constraint 2: Maximum variance between consecutive time headways

$$T_{current,i} \geq T_{safety} \qquad \forall i = 0, 1, \dots, N$$

(d) Constraint 3: minimum safe time headway

$$T_{current,i} \geq T_{current,i+!} \qquad \forall i = 0, 1, \dots, N$$

(e) Constraint 4: Relationship between consecutive time headways

$$\sum_{i=1}^{N}\left(T_0 - T_{current,i}\right) \geq Required\ Gap\ by\ LV$$

(f) Constraint 5: Relationship between time headways and required gap for lane-changing

Figure 16. Formulas. Optimization problem for case 2: new gap to make is determined.

Interaction of Speed Harmonization and Merge Coordination

The inter-operation of the SPDHRM and the merge coordination systems can happen at two levels:

1. Both systems are active but independent. Accordingly, the SPDHRM system can have an improved performance due to less severe shock waves at the merge location due to implementation of the merge coordination system.
2. Both systems are exchanging information. Accordingly, in addition to the benefits that the SPDHRM can have from an active merge coordination system, the merge coordination can benefit from an active SPDHRM. The SPDHRM system can create the right gap for the merging vehicles by reducing/eliminating shockwaves.

CHAPTER 3. MODEL CALIBRATION AND VALIATION

In order to calibrate and validate the joint application calibration, two key features were developed and integrated into an open source microscopic simulation platform developed by researchers previously:

- SPDHRM: A set of novel SPDHRM algorithms were developed that utilize machine learning to predict the onset of congestion and to activate the SPDHRM in a highway segment. These algorithms also utilize various methods of communicating the updated speed limits to the CVs (automated or human-driven) and non-CVs (automated or human-driven).
- Merge coordination: A couple of algorithms were developed to enable merge coordination in connected and non-connected driving environments. These algorithms aim to prevent shock wave formation in the target lane, even at very small time headways.

In addition to these models, the simulation platform utilizes several already calibrated and validated CF and LC models for non-connected HVs, connected HVs, CAVs, and non-connected AVs (Talebpour and Mahmassani 2016).

Since most of the models used were already calibrated and validated based on the NGSIM US–101 Dataset (2007), the focus of this calibration and validation effort was on the calibration of CF models of HVs to capture the effects of interacting with AVs on driver behavior. The validation effort ensures the accuracy of the calibration process.

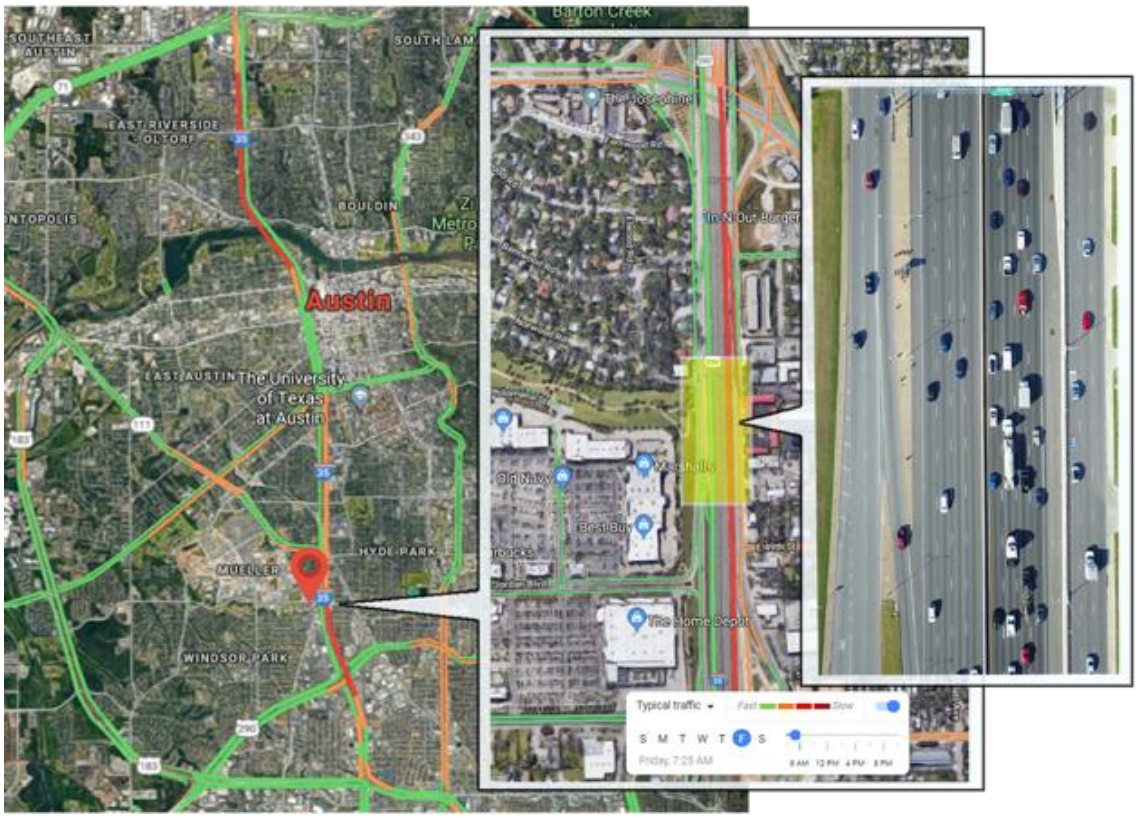
Data Set

Vehicle trajectories are a cornerstone of modern traffic flow theory with applications in driver behavior studies and AV research. Unfortunately, the existing vehicle trajectory data sets are limited, mostly due to the high cost of data collection and preparation. Moreover, with the arrival of ADAS and AVs, there is a potential to see changes in human driving behavior when interacting with these technologies. As a result, there is a desire for new vehicle trajectory data sets that cover various levels of automation. Aerial imagery using small unmanned aerial vehicles (UAV) is an economical and effective solution to collect trajectory data.

A new trajectory data set was collected on Interstate 35 (I–35) in Austin, Texas, (see Figure 17) to address the shortcomings of the existing vehicle trajectory data sets. A platoon of three SAE Level 1 AVs with ACC technology was circulating in the traffic stream during the data collection. Two UAVs (e.g., drones) were used for the aerial videography of the traffic stream. The trajectories of the vehicles were extracted from the video frames recorded in bird's-eye view from a segment of the roadway (see Figure 18). In every video frame, the location of the vehicles was estimated for a fixed coordinate system and reference point on the ground. Every video recording was converted to a sequence of images (i.e., frames) separated at a constant rate over time (e.g., 25 frames per second). Tracking the location of any vehicle over the sequence of images enabled extracting the vehicle's trajectory over time.

The vehicle trajectory extraction is performed in four steps: image stabilization, vehicle detection, vehicle tracking, and trajectory construction. In the image stabilization step, all the images are transformed to match a reference field of view. Then the vehicles are detected in

every frame and tracked over the sequence of images. Finally, the vehicles' location and trajectories are constructed by converting the image coordinates to the adopted reference coordinates on the ground. Figure 19 shows a sample of collected vehicle trajectory data.

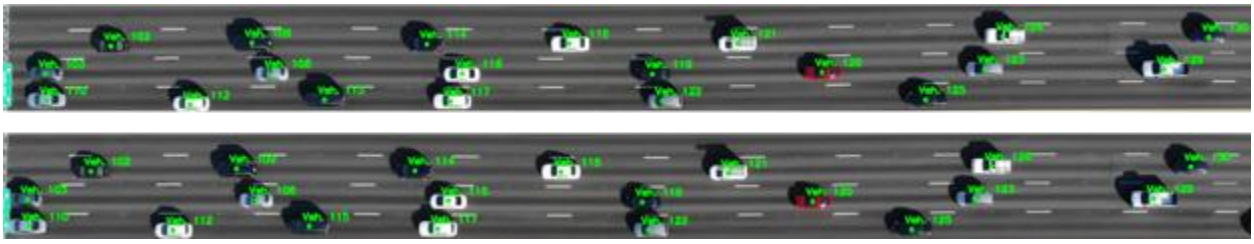


Original Photo: © 2019 Google Earth™.

Figure 17. Photo. Data collection location on Interstate 35 near Austin, Texas.

(a) Bird's Eye View

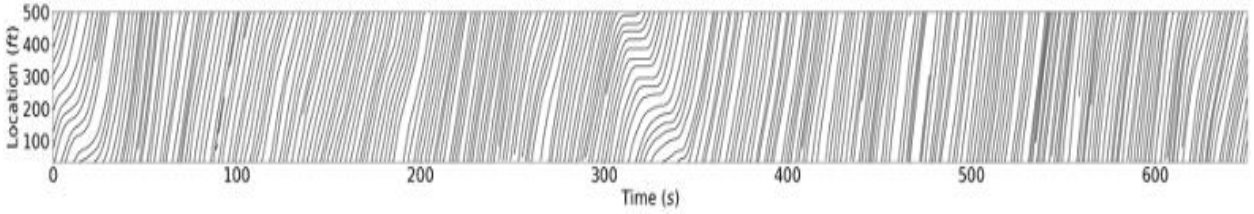
(b) Vehicle Detection Using Convolutional Neural Network



(c) Vehicle Tracking

Source: FHWA.

Figure 18. Photo. Vehicle detection and tracking in aerial images.



Source: FHWA.

Figure 19. Illustration. Sample trajectory data collected on Interstate 35 near Austin, Texas.

Calibration Approach

This study adopts the genetic algorithm calibration approach introduced by Hamdar et al. (2009). The approach relies on comparing the driving behavior in the data set with the simulated behavior based on a set of model parameters. For CF models, the error is calculated based on the error in the gap between the lead vehicle and the target vehicle, as shown in Figure 20 and Figure 21:

$$F_{mix}[s^{sim}] = \sqrt{\frac{1}{|s^{data}|}\left\langle\frac{(s^{sim} - s^{data})^2}{|s^{data}|}\right\rangle}$$

Figure 20. Equation. Error in the gap between the lead vehicle and the target vehicle.

$$\langle z \rangle = \frac{1}{\Delta T}\int_0^{\Delta T} z(t)dt$$

Figure 21. Equation. Definition of ⟨ ⟩.

where,, $s^{sim}(t) = x_l^{data}(t) - x^{sim}(t)$, and $s^{data}(t) = x_l^{data}(t) - x^{data}(t)$. For LC models, the same process is followed with one key difference: the error in the gap between the LC vehicle and both the new leader and new follower is considered.

Once the error function is defined, the genetic algorithm heuristic can be implemented as follows:

- The parameters of a CF/LC model are initialized to random numbers. Each set of these parameters is called a chromosome and the total of N_{GA} chromosomes will be created.
- The fitness of each chromosome is determined using the aforementioned error function.
- Except for the chromosome with the lowest error value, every other chromosome will be evolved through crossover and mutation (see Hamdar 2009 for the definition of crossover and mutation in genetic algorithm).
- The process is terminated once a minimum error threshold is achieved by the best chromosome. The parameters of that chromosome will form the calibration results.

Following the procedure outlined above, an initial set of 100 parents will be initiated. These parents will produce 900 children at each iteration and the top 99 children will join the best of the parents to move to the next iteration. The calibration process stops once the error is below 5 percent or less than 0.1 percent improvement in error is observed for more than 20 consecutive iterations.

Calibration and Validation Process

The behavioral parameters of drivers in microscopic simulation models are expected to be correlated. Kim and Mahmassani (Kim and Mahmassani 2011) presented a methodology to capture this correlation across the parameters of each driver. They showed that sampling from the empirical data while accounting for the correlation between the parameters of each sample (individual drivers) is the best method for capturing heterogeneity in microscopic simulation models. The same method was utilized in this study for the calibration of CF and LC models.

In order to calibrate and validate the model, each vehicle trajectory in the data set was divided into calibration and validation sets. The calibration set had about four times more data points than the validation set, with all selected randomly from the data points in the vehicle trajectory data set. The model was first calibrated using the data in the calibration set utilizing the same error function presented in the previous section. The calibrated model parameters were then used to simulate the data in the validation set and the results (gap between vehicles) were compared. The outcome of this calibration and validation process is a set of CF/LC parameters for each individual vehicle trajectory in the data set.

As discussed previously, the models utilized in the adopted simulation platform have gone through a similar calibration and validation process based on NGSIM US–101 data set (FHWA 2007). Accordingly, this study only focuses on the cases when a human driver interacts with an AV in the data set. The focus was on a human driver following an AV. Note that Rahmati et al.

(Rahmati, Khajeh Hosseini et al. 2019) showed that there is potential to see significant changes in driver behavior in this case.

A Note on Calibrating and Validating Car-Following and Lane-Changing Models for Autonomous Vehicles

Regarding CF models of AVs, the adopted simulation framework utilizes the ACC/CACC models that were calibrated based on empirical data (Van Arem, Van Driel et al. 2006). Accordingly, the CF behavior of AVs will not be calibrated again in this study.

Regarding LC models of AVs, these models are designed based on the capabilities and characteristics of an automated vehicle. Accordingly, any LC trajectory generated by the models can be followed in the real world.

Calibration and Validation Results

This section presents the calibration results for the CF model of human drivers. The selected CF model is the Prospect theory model developed by Hamdar et al. (Hamdar, Treiber et al. 2008) and extended by Talebpour et al. (Talebpour, Mahmassani et al. 2011).

Table 2 and Table 3 show the calibration results for the Austin data along with the data collected by Rahmati et al. (Rahmati, Khajeh Hosseini et al. 2019). The details of the model are presented in the next section. The model consists of three core parameters that were calibrated in this study. w_m and γ capture drivers' different preferences when dealing with acceleration and deceleration. In other words, drivers put more negative weight on deceleration. w_c is the crash weighting factor and higher values of this parameter represent more cautious drivers.

Conducting ANOVA test (Wilcox 1996) between values in Table 2 and Table 3 showed that two of the key model parameters show no statistically significant difference (i.e., w_m and γ), indicating that drivers' utility in response to acceleration and deceleration were the same for both cases of following an AV and following another human driver.

Table 2. Car-following model calibration results for human following.

Parameters	Mean	Standard Deviation
w_m	0.268	0.443
w_c	115200.00	21432.52
γ	0.71	0.62

Table 3 . Car-following model calibration results for autonomous vehicle following.

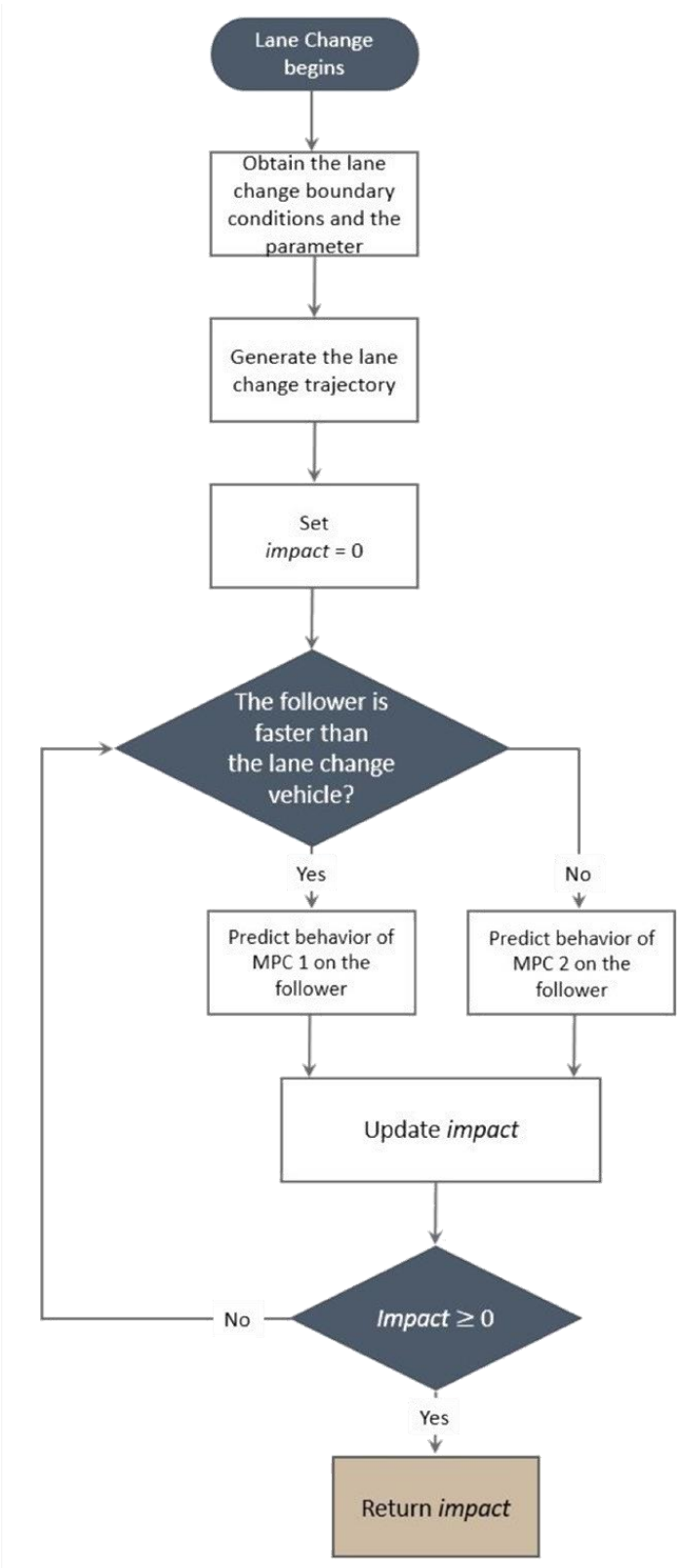
Parameters	Mean	Standard Deviation
w_m	0.271	0.365
w_c	81432.83	2870.79
γ	0.69	0.61

Unlike w_m and γ parameters, the crash weighing factor w_c showed statistically significant difference between the two cases. Following an AV, human drivers' behavior resulted in much less w_c compared with the case of following another human driver (w_c is 36 percent less for the AV following case than the human following case). Such a significant difference shows that human drivers are more comfortable following the AV compared with another human driver and they feel safer. Such an observation on w_c can also be interpreted from the risk-taking perspective. The smooth behavior of the AV encourages more risk-taking behaviors by the following human driver, resulting in lower values of w_c.

CHAPTER 4. BASIC INFORMATION ON MODEL IMPLEMENTATION

Figure 22 illustrates the flowchart for the implementation of the merge coordination algorithm. The integration with any microscopic traffic simulation is possible. The model input is vehicle location, speed, and acceleration (both merging vehicles and other vehicles in the target lane). The output, however, can be different based on the capabilities of the microscopic simulation tool. Since the model operates in a 2D environment (i.e., both lateral and longitudinal movements are considered), the implementation in most commercially available microscopic simulation tools requires an adjustment to the model outcome. In other words, instead of outputting both lateral and longitudinal movements, the model should be adjusted to output longitudinal location and lane number.

In addition to the implementation of the lane-changing maneuver, the existing implementation is capable of outputting the impact on shockwave formation. This part can be removed, if found to be unnecessary for the study, without impacting the merging/lane-changing algorithm.



Source: FHWA

Figure 22. Flowchart. Implementation of merge coordination system.

CHAPTER 5. USER CASE AND SENSITIVITY STUDY

IMPLEMENTATION OF THE DEVELOPED MODEL INTO A TRAFFIC SIMULATION TOOL

The merge coordination system was implemented in Northwestern University's microscopic simulation tool. The framework was incorporated into a microsimulation tool developed in Python® (Python Software Foundation, n.d.). The microsimulation tool is a special-purpose platform for simulating mixed traffic conditions on freeways with the possibility of including connected vehicles and autonomous vehicles in the system.

In the simulation platform, distinct car-following models are defined to specify the behavior of each agent: 1) Manually driven vehicles (regular vehicles); 2) Connected vehicles; and 3) Automated vehicles.

In the microsimulation platform, manually driven vehicles use the acceleration model first developed by Hamdar et al. (Hamdar, Treiber et al. 2008) and extended by Talebpour et al. (Talebpour, Mahmassani et al. 2011). The model was formulated based on Kahneman and Tversky's prospect theory. Two value functions, one for modeling driver behavior in congested regimes and the other for modeling driver behavior in uncongested regimes, were introduced. The following formula shows the value function for the uncongested regime:

$$U_{PT}^{UC}(a_n) = \frac{\left[w_m + (1 - w_m)\left(\tanh\left(\frac{a_n}{a_0}\right) + 1\right)\right]}{2}\left[\frac{\left(\frac{a_n}{a_0}\right)}{\left(1 + \left(\frac{a_n}{a_0}\right)\right)^{\left(\frac{g-1}{2}\right)}}\right]$$

Figure 23. Formula. Value function for the uncongested regime.

Where U_{PT}^{UC} denotes the value function for the uncongested traffic conditions. $y > 0$ and w_m are parameters to be estimated and calibrated and $a_0 = 1\ m/s^2$ is used to normalize the acceleration. On the other hand, the following formula shows the value function for the congested regime:

$$U_{PT}^{C}(a_n) = \frac{\left[w'_m + (1 - w'_m)\left(\tanh\left(\frac{a_n}{a_0}\right) + 1\right)\right]}{2}\left(\frac{a_n}{a_0}\right)^{g}$$

Figure 24. Formula. Value function for the congested regime.

Where U_{PT}^{C} denotes the value function for the congested traffic conditions. $y' > 0$ and w'_m are parameters to be estimated and calibrated. At each evaluation time step, the driver evaluates the gain from a candidate acceleration selected from a feasible set of values. The surrounding traffic condition is taken into consideration by the driver throughout the acceleration evaluation process. The driver utilizes the following binary probabilistic regime selection model to evaluate each acceleration value:

$$U_{PT}(a_n) = P(C).U_{PT}^{C} + P(UC).U_{PT}^{UC}$$

Figure 25. Formula. Binary probabilistic regime selection model.

Where U_{PT}, $P(C)$, and $P(UC)$ denote the expected value function, the probabilities of driving in a congested traffic condition, and the probability of driving in uncongested traffic conditions, respectively. After calculating the expected value function, the total utility function for acceleration could be written as follows:

$$U(a_n) = (1 - p_{n,i})U_{PT}(a_n) - p_{n,i} w_c k(v, \Delta v)$$

Figure 26. Formula. Total utility function for the choice of acceleration.

Where $p_{n,i}$ is the crash probability. Finally, the following probability density function is used to evaluate the stochastic response of the drivers:

$$f(a_n) = \begin{cases} \dfrac{e^{b_{PT}U(a_n)}}{\int_{a_{\min}}^{a_{\max}} e^{b_{PT}U(a')}\,da'} & a_{\min} < a_n < a_{\max} \\ 0 & Otherwise \end{cases}$$

Figure 27. Formula. Probability density function for the evaluation of drivers' stochastic response.

Where β_{PT} is the sensitivity of choice to the utility $U(a_n)$.

Connected vehicles are capable of exchanging information with other vehicles and infrastructure-based equipment. The information is exchanges through the vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications networks. As a result, the driver receives information about other connected vehicles as well as updated information containing TMC decisions (e.g., real-time changes in speed limit). The drivers' behavior may change based on the information conveyed to the driver. The reliability and the frequency of the information received by the driver plays a significant role in the drivers' behavior and on the overall performance of the traffic network.

An active V2V communication network allows the drivers to be aware of other drivers' behavior, the driving environment, road condition, and weather condition. As a result, the driving behavior could be modeled using a deterministic acceleration modeling framework. The simulation tool utilizes the Intelligent Driver Model (IDM) to model this connected environment. Because the IDM can capture various congestion dynamics and provides greater realism than most of the deterministic acceleration modeling frameworks.

The acceleration model specified by the IDM entails the vehicle's current speed, the ratio of the current spacing to the desired spacing, the difference between the leading and the following vehicles' velocities, and subjective parameters such as desired acceleration, desired gap size, and comfortable deceleration.

$$a_{IDM}^{n}(s_n, v_n, \Delta v_n) = \bar{a}_n \left[1 - \left(\frac{v_n}{v_0^n}\right)^{\delta_n} - \left(\frac{s^*(v_n, \Delta v_n)}{s_n}\right)^2\right]$$

$$s^*(v_n, \Delta v_n) = s_0^n + T_n v_n + \frac{v_n \Delta v_n}{2\sqrt{\bar{a}_n \bar{b}_n}}$$

Figure 28. Formula. The intelligent driver acceleration model.

Where δ_n is the free acceleration exponent; T_n is the desired time gap; a_n is the maximum acceleration; b_n is the desired deceleration; s_0^n is the jam distance; and v_0^n is the desired speed. These parameters need to be calibrated to better capture the behavior of connected vehicles.

If the V2V communication network is inactive, the driving behavior of connected vehicles would be similar to that of isolated-manually driven vehicles. In the presence of V2I communications, the TMC decisions, such as the speed limits in the case of speed harmonization, could be transferred to the drivers. However, their reaction times would still be like regular drivers.

Automated vehicles can continuously monitor other vehicles in their vicinity, which results in a deterministic behavior in interacting with other drivers. Furthermore, they can quickly react to any perturbations in the driving environment. Therefore, the car-following behavior of automated vehicles could be specified by a deterministic modeling framework. Talebpour and Mahmassani (Talebpour and Mahmassani 2016) developed a car-following model for automated vehicles based on the previous simulation studies by Van Arem et al. (Van Arem, Van Driel et al. 2006) and Reece and Shafer (Reece, Shafer et al. 1993). They simulated similar individual sensors installed on the automated vehicles to generate the input data for the acceleration model.

Considering the sensor range and limitations in accuracy, the automated vehicles must be ready to react to any situation outside of their sensing range once it is detected (e.g., a vehicle at a complete stop right outside of the sensors detection range). Furthermore, if a leader is spotted, the speed of the automated vehicle should be adjusted in a way that allows it to stop if the leader decides to decelerate with its maximum deceleration rate and reach a full stop. Considering different situations that require the immediate reaction of the automated vehicle, the maximum safe speed can be calculated using the following equations:

$$\Delta x_n = (x_{n-1} - x_n - l_{n-1}) + v_n \tau_n + \frac{v_{n-1}^2}{2a_{n-1}^{decc}}$$

$$\Delta x = min(SensorDetectionRange, \Delta x)$$

$$v_{max} = \sqrt{-2a_i^{decc} \Delta x}$$

Figure 29. Formula. Maximum speed of automated vehicles.

Where n and n-1 represent the automated vehicle and its leader, respectively; x_n is the position of vehicle n; l_n is the length of vehicle n; v_n is the speed of vehicle n; τ_n is the reaction time of vehicle n; and a_n^{decc} is the maximum deceleration of vehicle n.

Besides the safety constraint, the following formula, adopted from the model proposed by Van Arem et al. (Van Arem, Van Driel et al. 2006), updates the acceleration of the automated vehicle at every decision point:

$$a_n^d(t) = k_a a_{n-1}(t-\tau) + k_v(v_{n-1}(t-\tau) - v_n(t-\tau)) + k_d(s_n(t-\tau) - s_{ref})$$

Figure 30. Formula. Acceleration model for automated vehicles.

Where a_n^d is the acceleration of vehicle n; and k_a, k_v, and k_d are model parameters that need to be calibrated. s_n is the spacing and s_{ref} is the maximum between the minimum distance (s_{min}), following distance based on the reaction time (s_{system}), and safe following distance (s_{safe}). In this study, the minimum distance is set at 2.0 m and s_{safe} is calculated according to the following formula:

$$s_{safe} = \frac{v_{n-1}^2}{2}\left(\frac{1}{a_n^{decc}} - \frac{1}{a_{n-1}^{decc}}\right)$$

Figure 31. Formula. Safe following distance formula.

Finally, the acceleration of the automated vehicle can be calculated using the following equation:

$$a_n(t) = min\left(a_n^d(t), k\left(v_{max} - v_n(t)\right)\right)$$

Figure 32. Formula. Acceleration of automated vehicles.

Where k is a model parameter. Van Arem et al. (Van Arem, Van Driel et al. 2006) suggested using the following values for the model parameters: $k = 1$, $k_a = 1$, $k_v = 0.58$, and $k_d = 0.1$.

DESIGN OF SIMULATION EXPERIMENTS AND SIMULATION RESULTS

In the following sections, the performance of the proposed joint optimization of LC trajectory, CF controller, and gap generation mechanisms are evaluated.

Car-Following Control

Several cases of highway driving are simulated. The behavior of a platoon is analyzed that consists of a combination of AVs and HVs. It is assumed that the vehicles in the ramp have lower or equal speed limit (different scenarios were tested). The focus of this study is on merge coordination, thus a form of mandatory LC will happen. The merge process starts by SV accepting LV as the new predecessor. Accordingly, SV starts to slow down to accommodate LV and once the LV has safely entered the target lane, SV starts to speed up to the desired speed.

The first set of experiments is focused on evaluating the impact of different weights in the cost functions (Figure 6 and Figure 7) on the performance of the merge coordination application. Note that the weights used in this set of experiments are not optimized weights and they were selected to illustrate the impact of weight selection on the merge coordination process. In fact, the performance of the model is expected to significantly improve by selecting optimized values.

For a general description of how the weight values were selected for simulation in this study, R_2 is related to the initial conditions such as the velocity difference between SV and LV and the headway between SV and the old leader in the target lane. Considering how polynomial-based trajectory is generated, those two factors are related to the condition used to generate the LC

trajectory and velocity profile. To find the proper value for R_2 in a scenario, some values for R_2 based on several generic simulations are found first. Afterward, they are interpolated to obtain an estimated proper R_2 depending on the initial conditions (i.e., $v_s - v_l$ and d). This does not reveal accurate values for the weight, but it can still be an appropriate estimate depending on $v_s - v_l$.

In the remainder of this section, different values of R_2 were selected for each experiment. The remaining weights used for the simulations are:

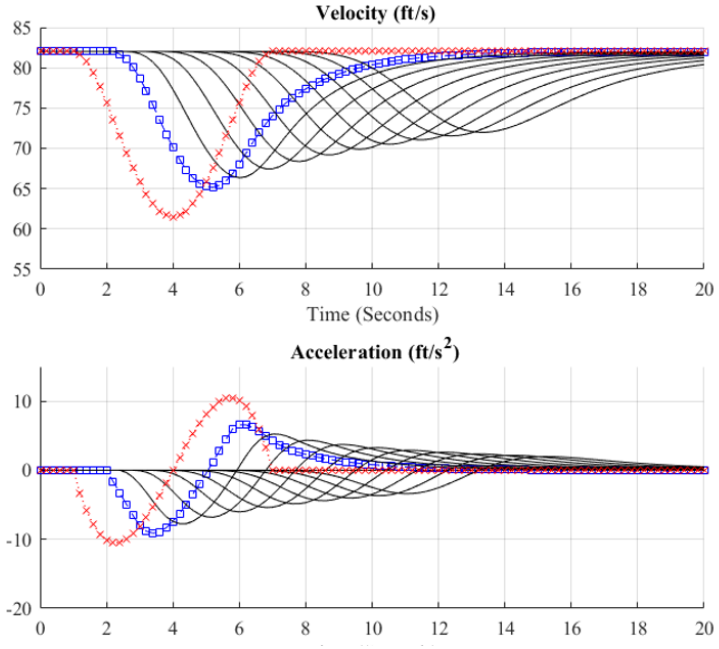
- Cost function of Figure 6: $R_1 = 100$, $R_3 = 500$.
- Cost function of Figure 7: $Q_1 = 70$, $Q_2 = 50$, $Q_3 = 800$, and $Q_4 = 200$.

In order to fully investigate the performance of the proposed model, speed and acceleration profiles of all vehicles in the simulation are plotted. The model performance was assessed against the performance of Swaroop's (1997) control logic. Two generic scenarios are tested:

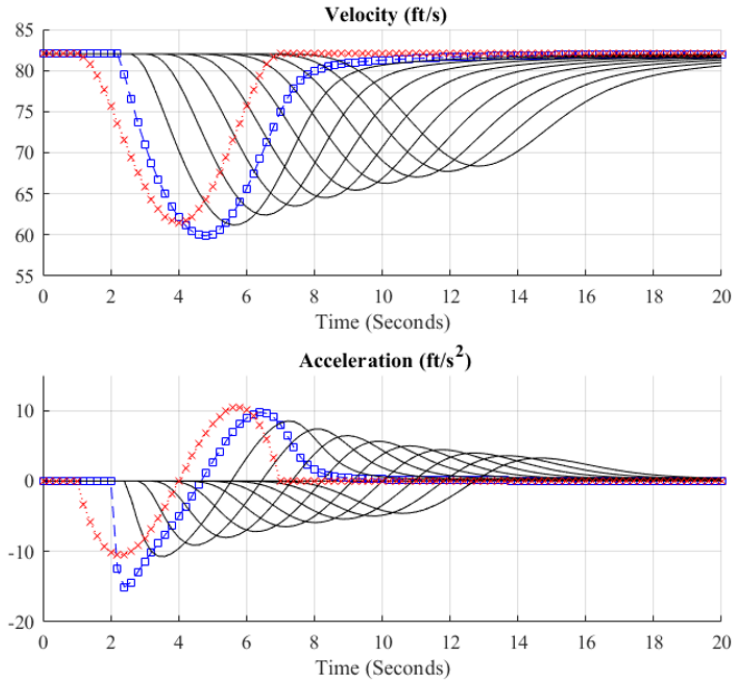
- Maintaining speed: both lanes have the same speed (v_s and $v_t = 82$ ft/s).
- Increasing speed: current lane speed ($v_s = 65$ ft/s) and target lane speed ($v_t = 82$ ft/s).

Two key factors were considered in the assessment: (1) driver/passenger comfort (i.e., changes in the acceleration value) and (2) safety. Figure 33 illustrates the simulation results for the maintaining speed scenario. The red line (also indicated by x) represents LV, and the blue line (also indicated by square) represents SV hereafter in all figures. This figure indicates that Swaroop's (1997) model results in significant and rapid change in the acceleration value due to the lane-changing maneuver. However, in Figure 34, with the same simulation setup, the proposed MPC resulted in a smooth change in acceleration and velocity.

The improvement in the performance of the merging process is even more evident in the increasing speed scenario. While Swaroop's model (1997) result in an even more significant deceleration and reduction in speed (this happens since SV is dealing with a much slower LV), the proposed MPC results in behavior similar to the maintaining speed scenario. This is mainly due to the fact the LV has information about SV's decision logic and can optimize its trajectory.



(a) Model predictive control-based platoon

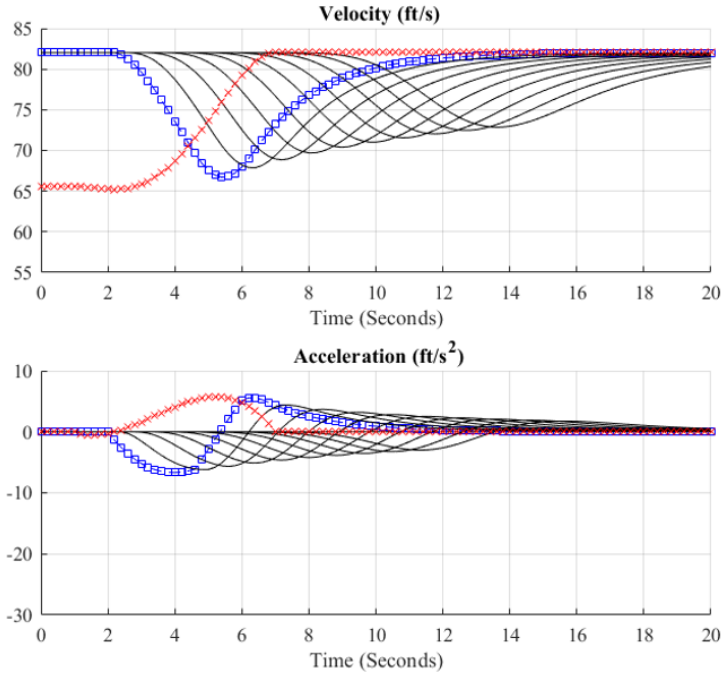


(b) Swaroop's controller based platoon

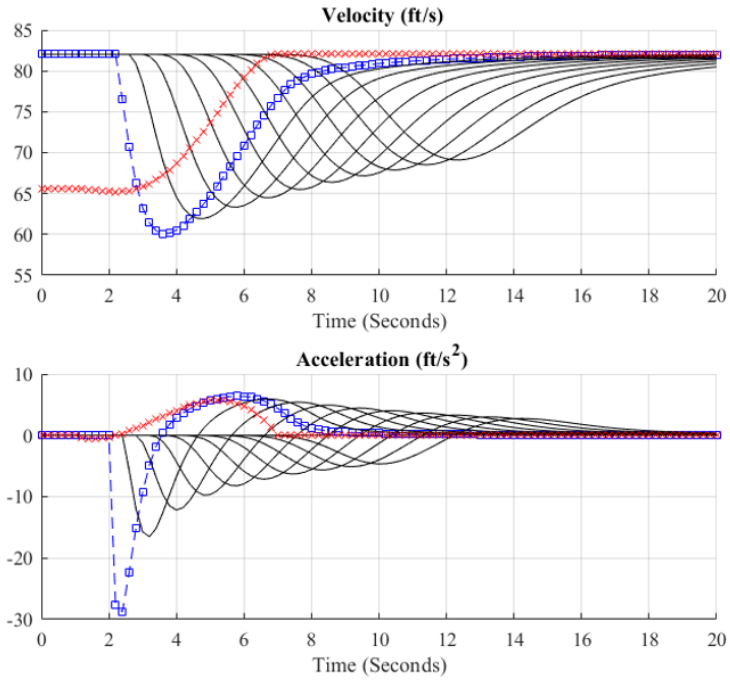
Source: FHWA.

Note that 1) Swaroop's controller results in unrealistic jerk; and 2) x marks, squares, and solid lines in the figure illustrate the lane-changing vehicle, immediate follower in the target lane, and upstream vehicles, respectively.

Figure 33. Diagrams. Maintaining speed: both lanes speed = 82 ft/s; (a) model predictive control-based platoon and (b) Swaroop's controller based platoon.



(a) Model predictive control-based platoon



(b) Swaroop's controller based platoon

Source: FHWA.

Note that 1) Swaroop's controller results in unrealistic jerk; and 2) x marks, squares, and solid lines in the figure illustrate the lane-changing vehicle, immediate follower in the target lane, and upstream vehicles, respectively.

Figure 34. Diagrams. Increasing speed: current lane speed = 65 ft/s, target lane speed = 82 ft/s; (a) model predictive control-based platoon and (b) Swaroop's controller based platoon.

Performance

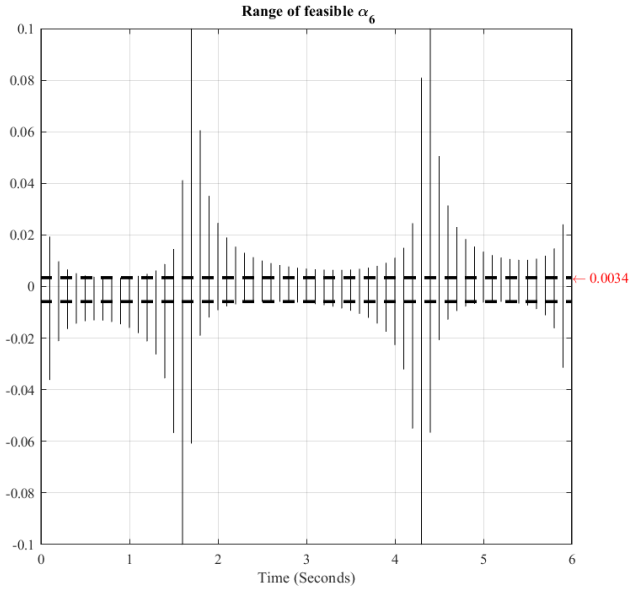
The previous section showed that the proposed MPC can easily outperform a typical constant time headway AV model. In this section, we further explore the impact of trajectory selection on the outcome of the proposed merge coordination model. The following simulation parameters are selected:

- Lane-changing duration: T = 6 s.
- Current lane speed (v_s) = 65 ft/s, target lane speed (v_l) = 74 ft/s.
- Distance between LV and the target leader: d = 0.2, $v_l \times d$ = 14.8 ft.

Note that these parameters are chosen carefully to show the impact of adding the sixth order term to the LC trajectory. In order to conduct the trajectory optimization, first a range of feasible a_6 is obtained based on the physical constraints. Note that since a_6 only impacts the movement along the x-axis, the constraints along the y-axis are not considered in this simulation. Note that in a real-world setting, such an assumption can result in infeasible trajectories to follow even by fixing the initial and final conditions as well as the LC duration. However, for simplicity, this section assumes that only longitudinal location of LV impacts the behavior of SV.

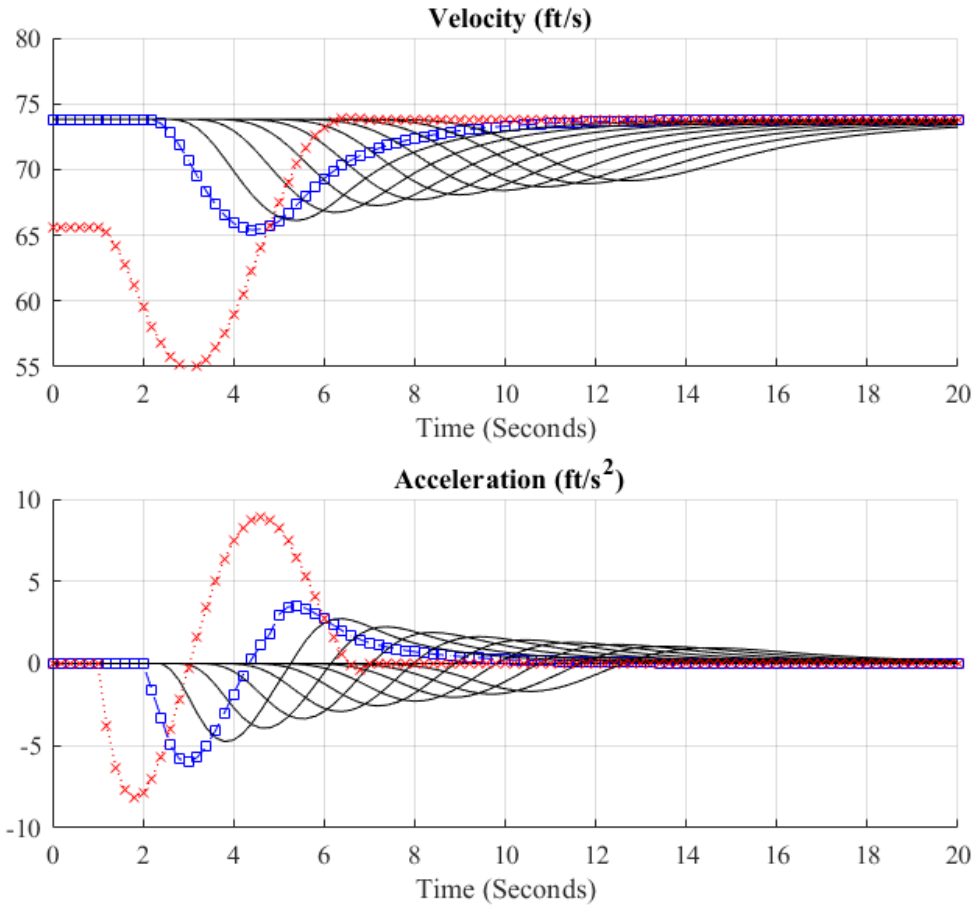
Figure 35 shows the feasible values of a_6 during the LC. Accordingly, as long as a_6 remains between the two dotted lines, the trajectory will be feasible to execute. Note that in this simulation setup, the longitudinal acceleration values were confined between 8.2 ft/s^2 and -13.0 ft/s^2. Accordingly, the optimization process searches for the optimal a_6 between -0.0058 and 0.0034.

As discussed previously, the optimization approach focuses on minimizing the total control/deceleration effort. Figure 36 illustrates how the optimization results in less deviation from the desired speed in the target lane. Based on this figure, even with a_6 = 0.0, the proposed MPC design works well and can significantly limit the shock wave magnitude. However, by adding a non-zero a_6 (a_6 = 0.0034), the model is capable of even further improving the merge process and reducing its impact on the target lane.

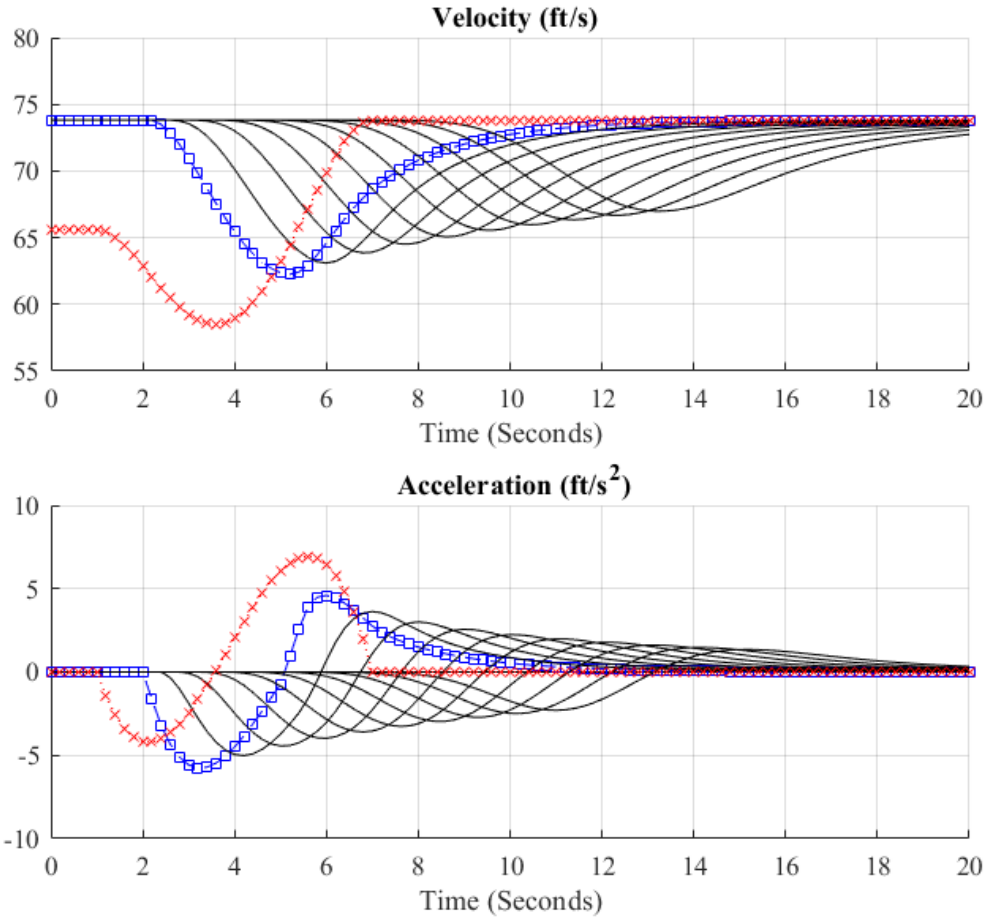


Source: FHWA.

Figure 35. Diagram. Range of feasible value of α_6: constrained by maximum longitudinal acceleration.



(a) $\alpha_6 = 0$ (lower order polynomial)



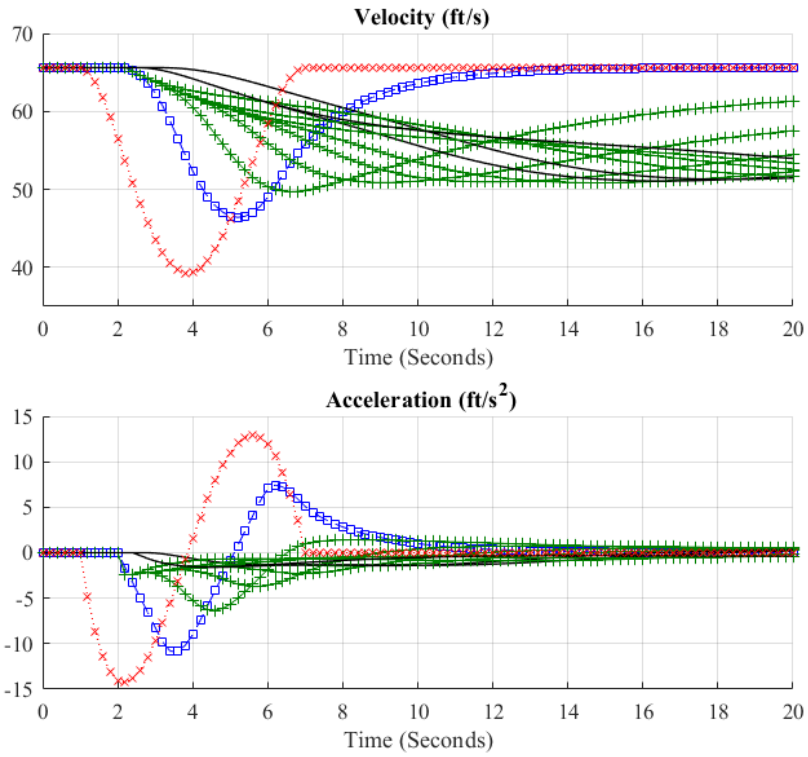
(b) α_6 = 0.0034 (optimal)

Source: FHWA.

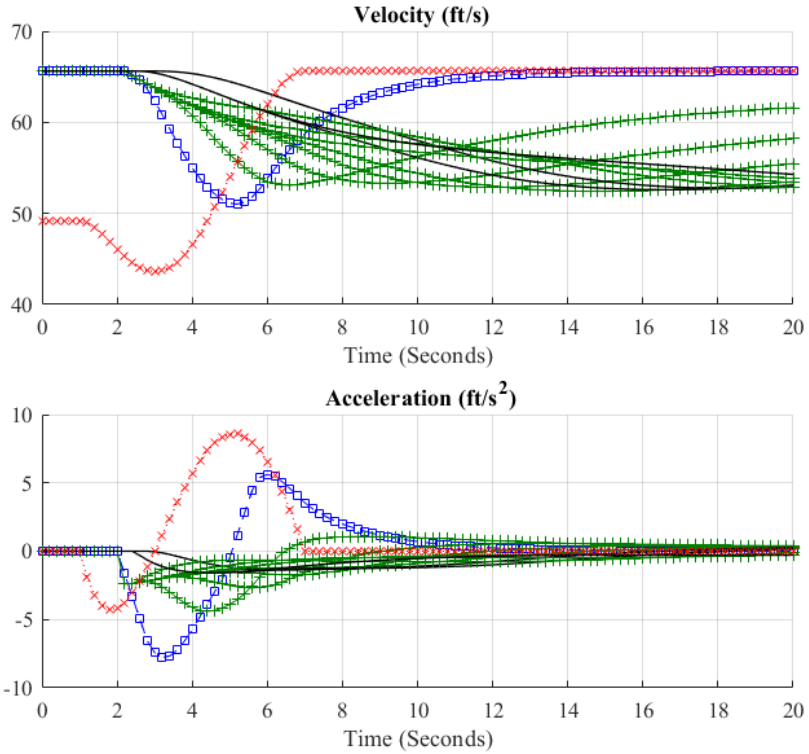
Note: x marks, squares, and solid lines in the figure illustrate the lane-changing vehicle, immediate follower in the target lane, and upstream vehicles, respectively.

Figure 36. Diagrams. Lane-changing maneuver profiles (adaptive cruise control-based platoon): (a) α_6 = 0 (lower order polynomial), and (b) α_6 = 0.0034 (optimal).

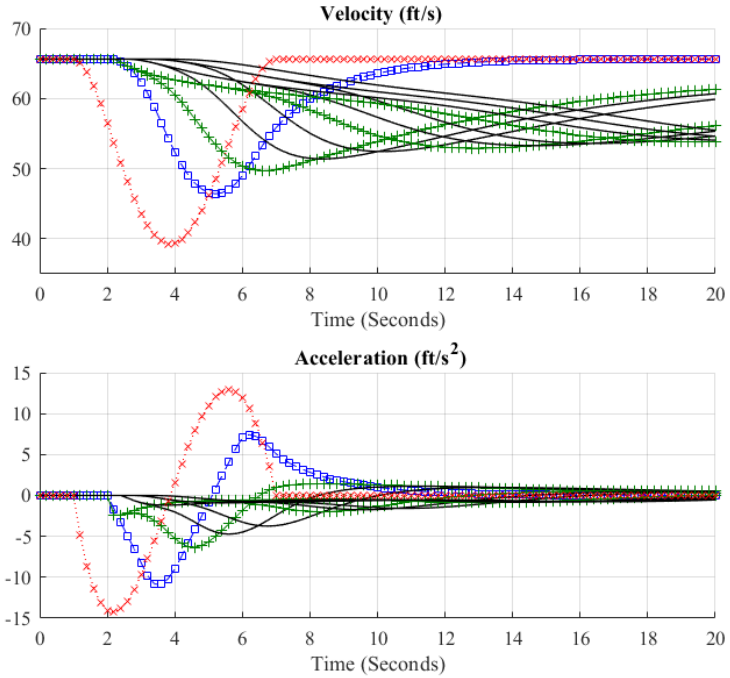
Note that the existence of V2V communications is necessary in the performance of the proposed model. Without V2V communications, LV cannot accurately predict the movements of SV and can only be reactive (based on the onboard sensor inputs). Finally, in order to investigate if minimizing total control effort can reduce shock wave magnitude in mixed driving environment, the same set of simulation scenarios were performed for a platoon of AVs and HVs. Platoon members were considered based on IDM only, ACC only, and ACC and IDM combined. Figure 37 illustrates the simulation results for various market penetration rates of AVs (i.e., 30 percent and 70 percent). The results indicate the capability of the presented approach in containing the shock wave due to the LC maneuver in mixed driving environment. However, as the market penetration rate of AV increases, the shock wave magnitude and duration decreases.



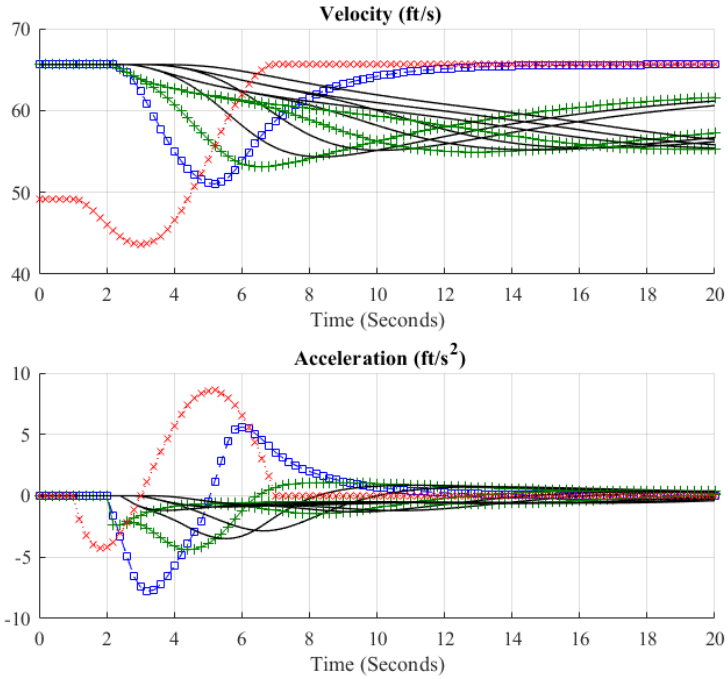
(a) 30 percent of adaptive cruise control-based cars in the platoon: maintaining speed



(b) 30 percent of adaptive cruise control-based cars in the platoon: increasing speed



(c) 70 percent of adaptive cruise control-based cars in the platoon: maintaining speed



(d) 70 percent of adaptive cruise control-based cars in the platoon: increasing speed
Source: FHWA.
Note: x marks, squares, plus marks, and solid lines illustrate the lane-changing vehicle, immediate follower in the target lane, and human-driven upstream vehicles, and ACC upstream vehicles, respectively.

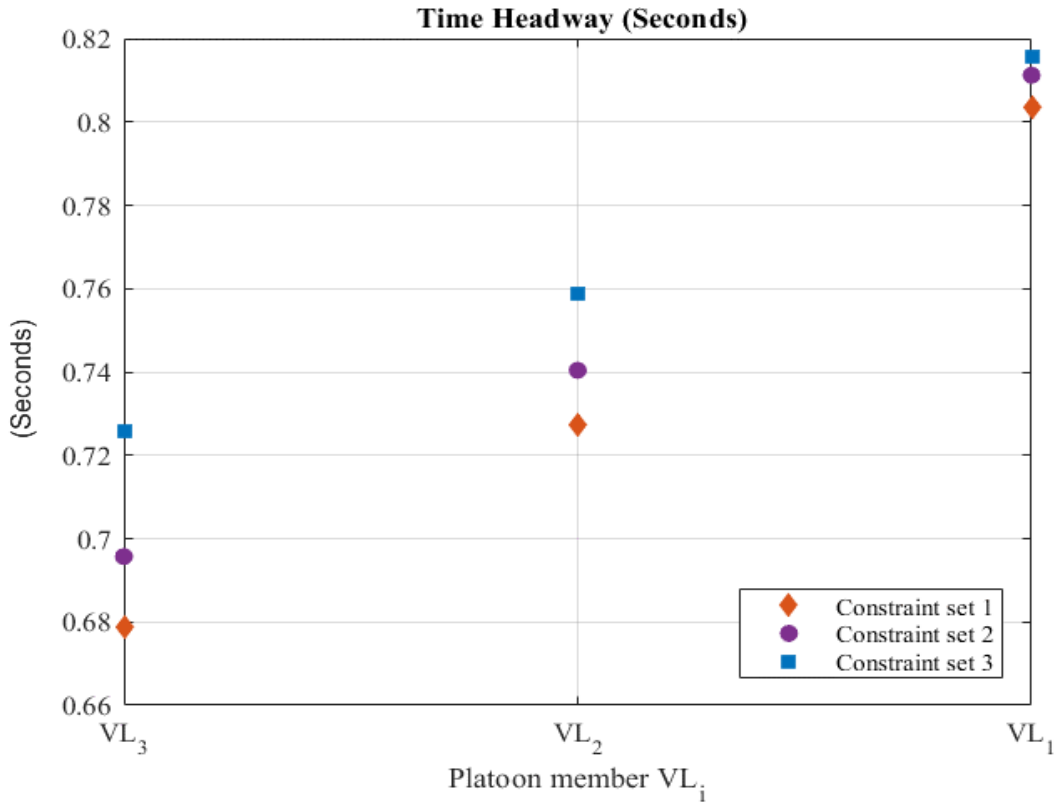
Figure 37. Diagrams. Percent of adaptive cruise control-based cars in the platoon

Gap Generation

Figure 38, Figure 39, and Figure 40 show how the value of variance constraints can change the platoon behavior. Three constraint sets were utilized for this purpose:

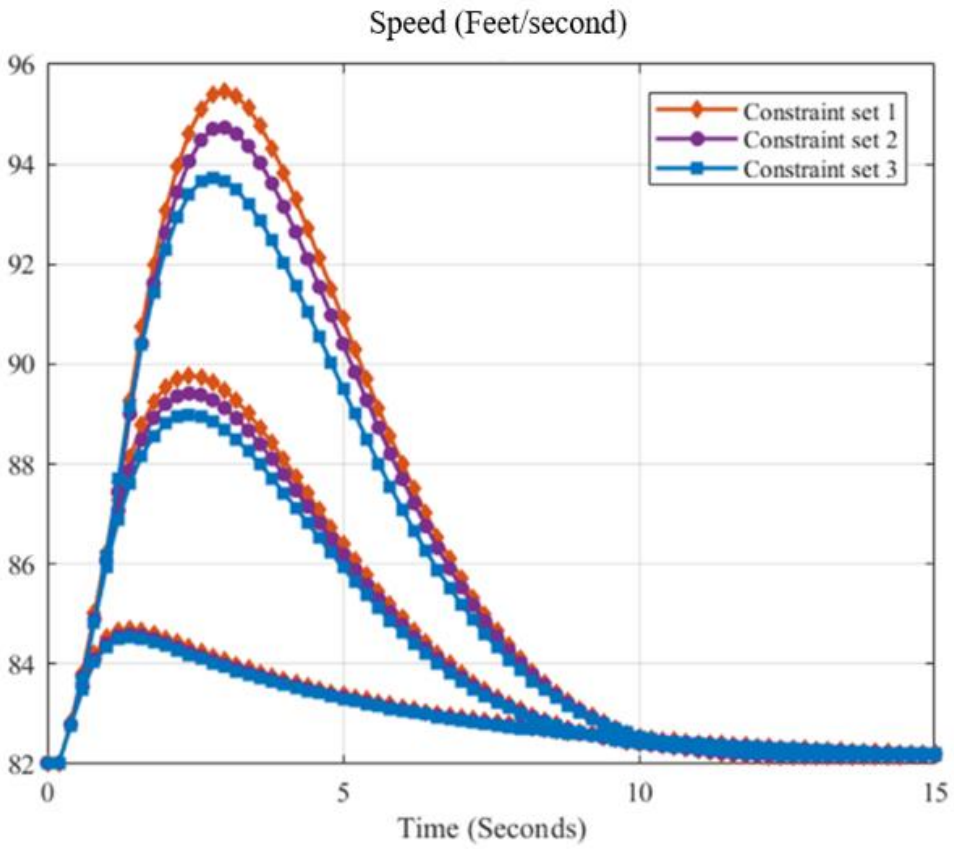
- Constraint set 1: $max\_var_1 = 0.02$ and $max\_var_2 = 0.007$.
- Constraint set 2: $max\_var_1 = 0.0018$ and $max\_var_2 = 0.0065$.
- Constraint set 3: $max\_var_1 = 0.0015$ and $max\_var_2 = 0.0063$.

These figures indicate that more restrictive set of constraints results in lower overall variance as well as lower maximum variance. Considering that having smaller speed variance is more desirable for safety, this observation indicates that the desired safety level can be achieved by setting up the correct set of constraints. Moreover, more strict constraints result in closer time headways to the original time headway, which results in small gap overall. Accordingly, choosing the right set of constraints is key to ensure both safety and efficiency of the LC maneuver.



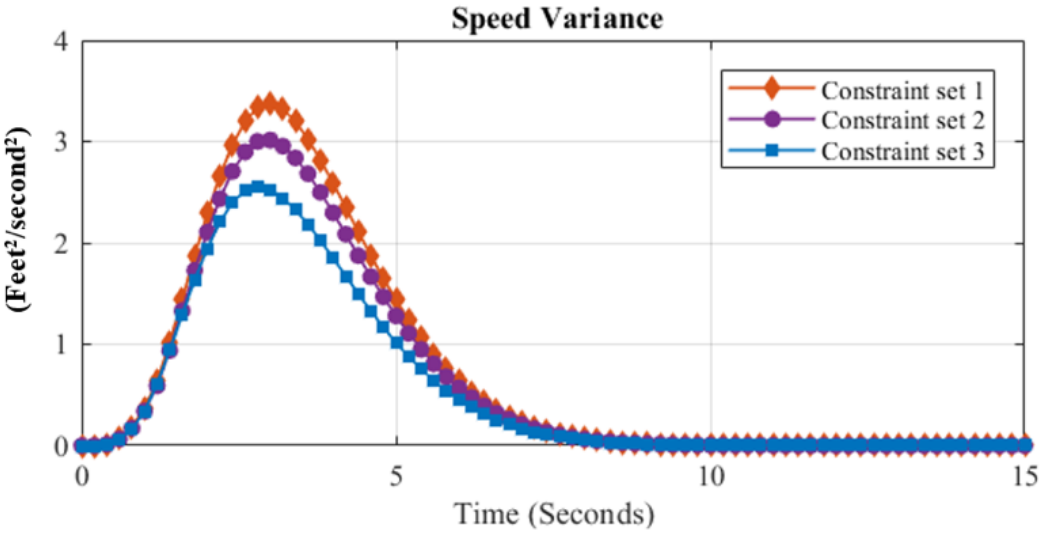
Source: FHWA.

Figure 38. Graph. Effects of different variance constraints on time headway of three vehicles ahead of the lane-changing vehicles in the target lane for $h_{desired}$ = 1.0 seconds, h_{safety} = 0.6 seconds, and sum of change of h = 1.0 seconds. All the graphs.



Source: FHWA.

Figure 39. Graph. Impact.



Source: FHWA.

Figure 40. Graph. Effects of different variance constraints on speed variance of three vehicles ahead of the lane-changing vehicles in the target lane for $h_{desired}$ = 1.0 seconds, h_{safety} = 0.6 seconds, and sum of change of h = 1.0 seconds. All the graphs.

CHAPTER 6. SUMMARY AND RECOMMENDATIONS

This study proposed a novel merge coordination system based on controlling both the lane-changing trajectory and car-following behavior of the vehicles in the target lane. The model was implemented in Northwestern University's microscopic traffic simulation tools and the simulation results indicated the effectiveness of the proposed methodology in reducing the shockwave magnitude and duration. Multiple sensitivity analysis experiments were also conducted for both fully autonomous and mixed driving environments. The model was successful in mitigating shockwave formation and propagation in almost all cases.

The proposed methodology, however, depends significantly on reliable communications. Vehicle to everything (V2X) communications do not necessarily exist at all times due to signal interference and information loss. An updated model should be developed to consider this lack of information availability. Such a model is critical to ensure a reliable merge coordination system in the real-world.

REFERENCES

(2007). Next Generation Simulation: US101 Freeway Dataset Federal Highway Administration, Washington, D.C.

An, G. and A. Talebpour (2019). Lane-Changing Trajectory Optimization to Minimize Traffic Flow Disturbance in a Connected Automated Driving Environment. 2019 IEEE Intelligent Transportation Systems Conference (ITSC), IEEE.

Cong, Y., O. Sawodny, H. Chen, J. Zimmermann and A. Lutz (2010). Motion planning for an autonomous vehicle driving on motorways by using flatness properties. 2010 IEEE International Conference on Control Applications, IEEE.

González, D., J. Pérez, R. Lattarulo, V. Milanés and F. Nashashibi (2014). Continuous curvature planning with obstacle avoidance capabilities in urban scenarios. 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), IEEE.

Hamdar, S. H., M. Treiber, H. S. Mahmassani and A. J. T. r. r. Kesting (2008). "Modeling driver behavior as sequential risk-taking task." **2088**(1): 208-217.

Lu, X., Liu, H., Li, X., Li, Q., Mahmassani, H., Talebpour, A., Hosseini, M., Huang, Z., Hale, D. K., and Shladover, S. E. (Forthcoming). Developing Analysis, Modeling, and Simulation Tools for Connected and Automated Vehicle Applications. FHWA.

Kianfar, R., M. Ali, P. Falcone and J. Fredriksson (2014). Combined longitudinal and lateral control design for string stable vehicle platooning within a designated lane. 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), IEEE.

Kim, J. and H. S. Mahmassani (2011). "Correlated parameters in driving behavior models: Car-following example and implications for traffic microsimulation." Transportation research record **2249**(1): 62-77.

Kolski, S., D. Ferguson, M. Bellino and R. Siegwart (2006). Autonomous driving in structured and unstructured environments. 2006 IEEE Intelligent Vehicles Symposium, IEEE.

Maschuw, J. P., G. C. Keßler and D. Abel (2008). "LMI-based control of vehicle platoons for robust longitudinal guidance." IFAC Proceedings Volumes **41**(2): 12111-12116.

Nelson, W. (1989). Continuous-curvature paths for autonomous vehicles. Proceedings, 1989 International Conference on Robotics and Automation, IEEE.

Pivtoraiko, M. and A. Kelly (2005). Efficient constrained path planning via search in state lattices. International Symposium on Artificial Intelligence, Robotics, and Automation in Space, Munich Germany.

Pivtoraiko, M., R. A. Knepper and A. Kelly (2009). "Differentially constrained mobile robot motion planning in state lattices." Journal of Field Robotics **26**(3): 308-333.

Rahmati, Y., M. Khajeh Hosseini, A. Talebpour, B. Swain and C. Nelson (2019). "Influence of Autonomous Vehicles on Car-Following Behavior of Human Drivers." Transportation Research Record: 0361198119862628.

Rastelli, J. P., R. Lattarulo and F. Nashashibi (2014). Dynamic trajectory generation using continuous-curvature algorithms for door to door assistance vehicles. 2014 IEEE Intelligent Vehicles Symposium Proceedings, IEEE.

Reece, D. A., S. A. J. T. R. P. A. P. Shafer and Practice (1993). "A computational model of driving for autonomous vehicles." **27**(1): 23-50.

Schroder, J., T. Gindele, D. Jagszent and R. Dillmann (2008). Path planning for cognitive vehicles using risk maps. 2008 IEEE Intelligent Vehicles Symposium, IEEE.

Seiler, P., A. Pant and K. Hedrick (2004). "Disturbance propagation in vehicle strings." IEEE Transactions on automatic control **49**(10): 1835-1842.

Swaroop, D. (1997). "String stability of interconnected systems: An application to platooning in automated highway systems."

Talebpour, A. and H. S. Mahmassani (2016). "Influence of connected and autonomous vehicles on traffic flow stability and throughput." Transportation Research Part C: Emerging Technologies **71**: 143-163.

Talebpour, A., H. S. Mahmassani and S. H. J. T. r. r. Hamdar (2011). "Multiregime sequential risk-taking model of car-following behavior: specification, calibration, and sensitivity analysis." **2260**(1): 60-66.

Talebpour, A. and H. S. J. T. R. P. C. E. T. Mahmassani (2016). "Influence of connected and autonomous vehicles on traffic flow stability and throughput." **71**: 143-163.

Toledo, T. and D. Zohar (2007). "Modeling duration of lane changes." Transportation Research Record **1999**(1): 71-78.

Treiber, M., A. Hennecke and D. Helbing (2000). "Congested traffic states in empirical observations and microscopic simulations." Physical review E **62**(2): 1805.

Van Arem, B., C. J. Van Driel and R. J. I. T. o. i. t. s. Visser (2006). "The impact of cooperative adaptive cruise control on traffic-flow characteristics." **7**(4): 429-436.

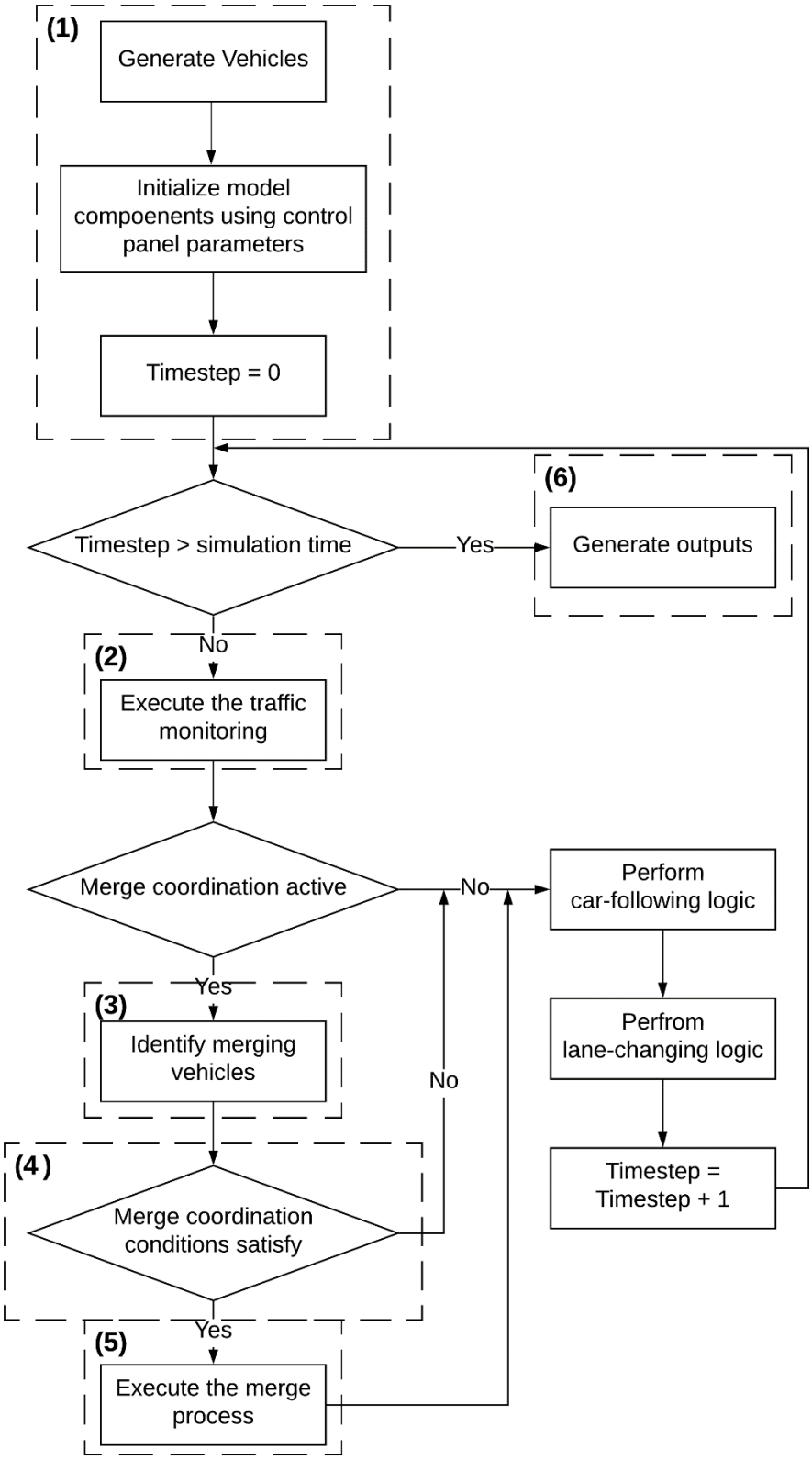
Wilcox, R. R. (1996). Statistics for the social sciences, Academic Press.

APPENDIX

The following structure is used to predict the impact of a lane-changing maneuver on a platoon when we assume that the communications between vehicles such as V2X communication is available. In detail, the lane-changing vehicle can read information on the controller of its follower in the platoon. With this information, the lane-changing vehicle can predict the followers' behavior in response to a pre-generated lane-changing trajectory using an iterative process.

A.1. Main Algorithm

This function controls the implementation process of the simulation logic. The logic is presented in Figure A1. The details of the setup and algorithm performance are presented in the following sections.



Source: FHWA

Figure A1. Flowchart. Main algorithm.

main

Syntax

main

Description

This function contains four components. The first component corresponds to the "Freeway Segment" element of the framework. It includes the inputs, outputs, and the driving logic of the tool. The driving logic contains the car-following and lane-changing models that specify the interaction among vehicles. The "Traffic Monitoring" module of the framework relates to the second element of the algorithm. The "Identify Merging Vehicles" module of the framework relates to the third element of the algorithm. Checking for the existence of the possibility of conducting a coordinated merge is the fourth module of the algorithm. "Executing the merge process" is the fifth element of the algorithm and is considered the core of this effort. The last element of the algorithm generates output for the simulation.

Input Arguments

Importing the generated vehicles array and the control panel. The generated vehicles array contains the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated). The following lists the module inputs that are specified in the control panel: activation of merge coordination and advisory speed limit.

Output Arguments

The vehicle trajectories are the outputs that are written into a csv file.

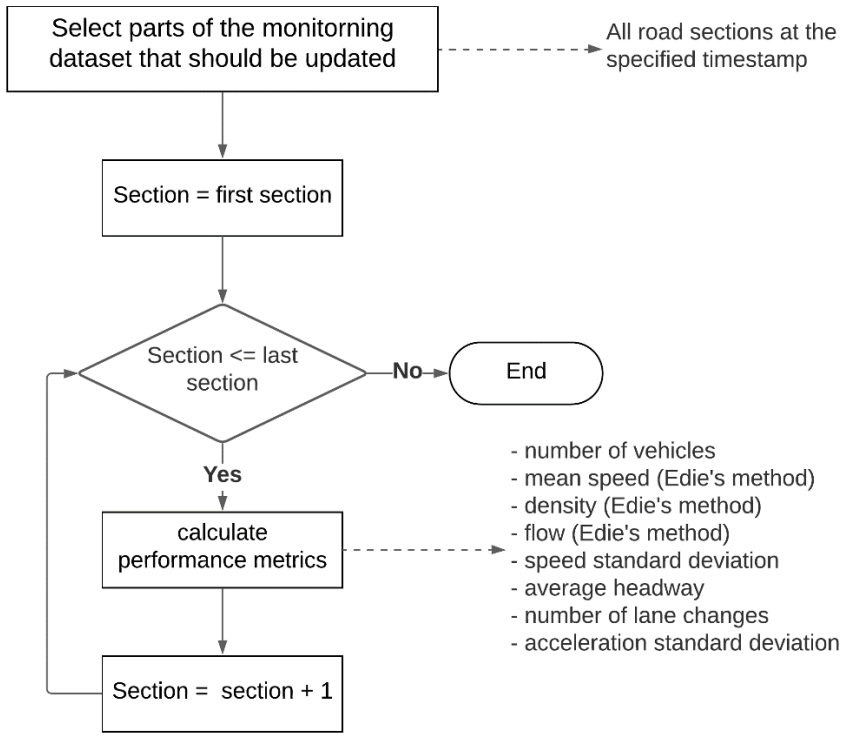
Pseudo-code

- Load generated vehicles
- Load vehicle trajectories
- Load car-following models
- Load initial traffic monitoring array for CVs and CAVs using init_tm_array function
- Load independent traffic monitoring array for all vehicles (this would be used for evaluating the system performance) using init_tm_array function
- If merge coordination is active:
    - Identify merging vehicles using identify_merging_veh function
    - Check the feasibility of the merge coordination using feasibility_check function
    - Execute the merge process using coordinated_merge function
- Conduct car-following and lane-changing maneuvers for the remaining vehicles+
- Generate the visualized results of the simulation

A.2. Traffic Monitoring Module

The traffic monitoring module, as shown in Figure A2, calculates the performance metrics for all sections of the road segment at each monitoring timestep. The performance metrics are

calculated using the connected vehicles and connected automated vehicles in order to be used in the congestion prediction and speed control modules. Simultaneously, the performance metrics are calculated for all the vehicles in the system in order to evaluate the accuracy of the prediction models. As a result, for the purpose of evaluating and implementing the speed harmonization strategies, only the information of connected and automated vehicles is analyzed.



Source: FHWA

Figure A2. Flowchart. Traffic monitoring algorithm.

init_tm_array

Syntax

init_tm_array()

Description

This function creates the traffic monitoring array.

Input Arguments

None

Output Arguments

tm_array: the initial traffic monitoring array

Pseudo-code

- Calculate the number of sections
- Calculate the number of monitoring timesteps
- Calculate the number of rows for the array = sections * monitoring timestep
- Return tm_array

update_tm_array_edie

Syntax

update_tm_array_edie(in_tm_array, in_veh_traj, t)

Description

This function updates the traffic monitoring array at the timestep t using CVs and CAVs.

Input Arguments

in_tm_array: the input traffic monitoring array with the following attributes: monitoring timestep, section number, number of vehicles in the section, mean speed, density, flow, speed standard deviation, mean headway, number of lane changes, acceleration standard deviation, speed standard deviation in the downstream section, mean speed in the downstream section, mean headway in the downstream section, congestion status, main line volume, ramp volume, SPDHRM type choice.

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

t: timestep

Output Arguments

in_tm_array : The updated traffic monitoring array.

Pseudo-code

- Find the current monitoring timestep = t / monitoring timestep
- Find the section indices that require to be updated.
- For i in the range of section indices:
    - Filter the CVs and CAVs that are in the current monitoring timestep, section i, and on the main lanes.
    - If number of filtered vehicles in the section I at the current monitoring timestep is more than zero:
        - Update number of vehicles
        - Calculate Edie's mean speed

        - Calculate Edie's density
        - Calculate flow
        - Calculate speed standard deviation
        - Calculate mean headway
        - Calculate number of lane changes
        - Calculate acceleration standard deviation
        - Calculate speed standard deviation in the downstream section
        - Calculate mean speed in the downstream section
        - Calculate mean headway in the downstream section
    - If it is the last section:
        - Set the attributes of the last section equal to the attributes of the second to last section

update_tm_array_edie_all

Syntax

update_tm_array_edie_all(in_tm_array, in_veh_traj, t)

Description

This function updates the traffic monitoring array at the timestep t using all vehicles in the system. The output of the function is used to evaluate the accuracy of the congestion prediction models and the effectiveness of the speed control module.

Input Arguments

in_tm_array: the input traffic monitoring array with the following attributes: monitoring timestep, section number, number of vehicles in the section, mean speed, density, flow, speed standard deviation, mean headway, number of lane changes, acceleration standard deviation, speed standard deviation in the downstream section, mean speed in the downstream section, mean headway in the downstream section, congestion status, main line volume, ramp volume, SPDHRM type choice.

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

t: timestep

Output Arguments

in_tm_array : The updated traffic monitoring array.

Pseudo-code

- Find the current monitoring timestep = t / monitoring timestep
- Find the section indices that require to be updated.
- For i in the range of section indices:
    - Filter the vehicles that are in the current monitoring timestep, section i, and on the main lanes.
    - If number of filtered vehicles in the section I at the current monitoring timestep is more than zero:
        - Update number of vehicles
        - Calculate Edie's mean speed
        - Calculate Edie's density
        - Calculate flow
        - Calculate speed standard deviation
        - Calculate mean headway
        - Calculate number of lane changes
        - Calculate acceleration standard deviation
        - Calculate speed standard deviation in the downstream section
        - Calculate mean speed in the downstream section
        - Calculate mean headway in the downstream section
    - If it is the last section:
        - Set the attributes of the last section equal to the attributes of the second to last section

A.3. Identify Merging Vehicles

This module checks for the existence of merging vehicles in the simulation at every timestep.

identify_merging_veh

Syntax

identify_merging_veh (in_veh_traj, t)

Description

This function checks the location of vehicles to identify the merging vehicles at timestep t. The output of the function is used to evaluate the possibility of merge coordination.

Input Arguments

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane

follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

t: timestep

Output Arguments

active_merge_array: The array containing the vehicle IDs of the merging vehicles.

Pseudo-code

- For i in range of indices for the in_veh_traj array:
    - If vehicle i is on a ramp and at the merge location add vehicle i to the active_merge_array
- Return active_merge_array

A.4. Merge Coordination Conditions Satisfy

This module checks for the possibility of executing a coordinate merge. Two factors are considered:

1. The merging vehicle and the immediate follower in the target lane are both CAVs and can communicate via V2V communications. The immediate follower should be also in the sensor range of the merging vehicle.
2. The merging vehicle and at least three immediate leaders in the target lane are CAVs. This condition applies to the gap generation algorithm.

feasibility_check

Syntax

feasibility_check (in_veh_traj, active_merge_array, t)

Description

This function checks the aforementioned conditions in the simulation at timestep t. The output of the function is used to execute the merge coordination.

Input Arguments

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

active_merge_array: the array containing the vehicle ID of merging vehicles.

t: timestep

Output Arguments

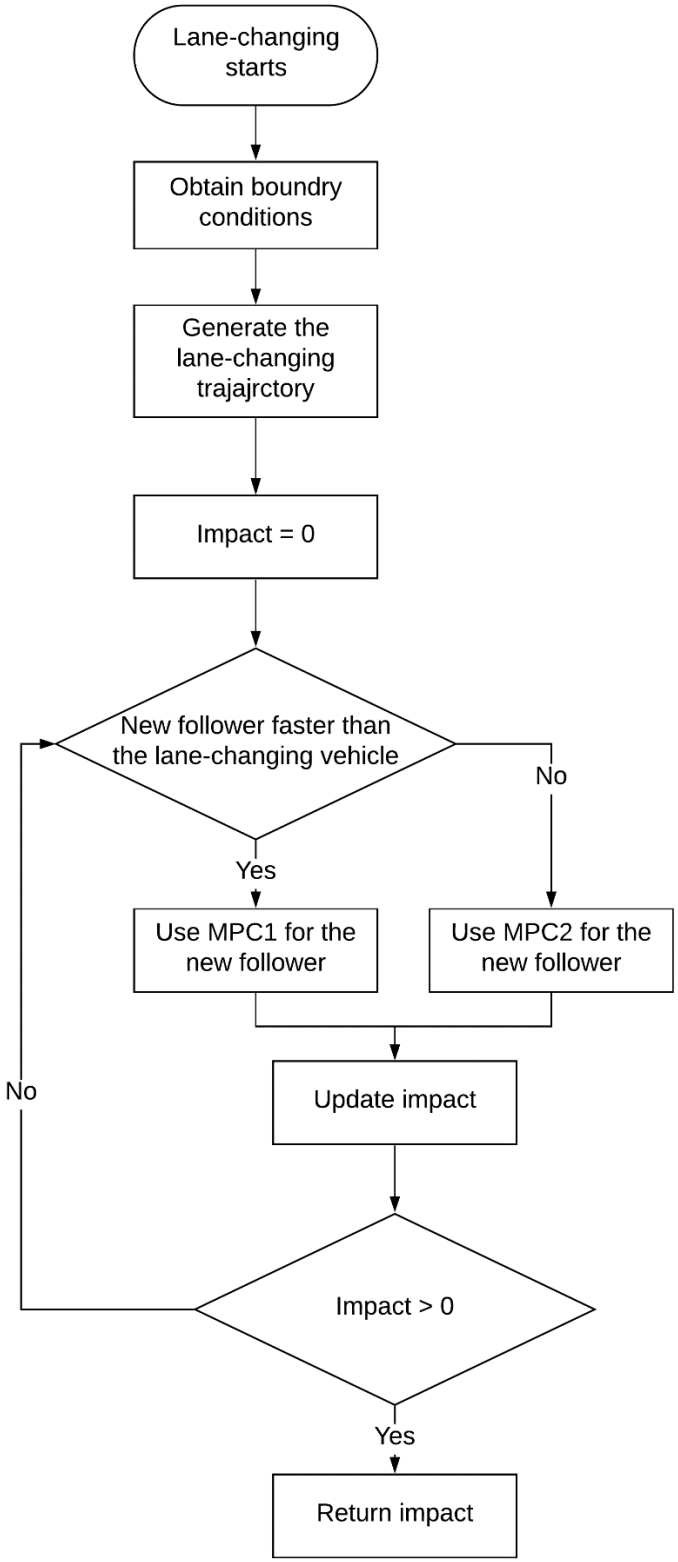
merge_feasibility_array: The array containing the vehicle IDs of the merging vehicles that can be considered for merge coordination.

Pseudo-code

- For i in range of indices for the active_merge_array array:
    - If vehicle i and its immediate follower in the target lane are both CAVs and they are in the sensor range of each other,
        - If vehicle i and three immediate leaders in the target lane are CAVs, add (vehicle ID, 1, 1) to the merge_feasibility_array
        - Else add (vehicle ID, 1, 0) to the merge_feasibility_array
    - Else:
        - If vehicle i and three immediate leaders in the target lane are CAVs, add (vehicle ID, 0, 1) to the merge_feasibility_array
        - Else add (vehicle ID, 0, 0) to the merge_feasibility_array
- Return merge_feasibility_array

A.5. Merge Coordination Module

This module covers the implementation of the merge coordination system. Note that this module contains several functions that will be discussed in detail. The overall setup of this module is Figure A3.



Source: FHWA

Figure A3. Flowchart. Implementation of merge coordination system.

merge_coordination

Syntax

merge_coordination(in_veh_traj, _alpha, merge_feasibility_array, t)

Description

This function is the core of this effort and implements the merge coordination algorithm. It also includes several subroutines that can be called during the merge coordination process. These subroutines are: obtain_boundry_conditions, generate_trajectory, MPC1, and MPC2.

Input Arguments

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

_alpha: lane-changing trajectory parameter to test: Sixth order coefficient of X-axis polynomial.

merge_feasibility_array: The array containing the vehicle IDs of the merging vehicles that can be considered for merge coordination.

t: timestep

Output Arguments

Total acceleration/deceleration applied to the follower of a lane-changing vehicle: *Impact*

Pseudo Code

- If (Lane change begins)
    - Get the trajectory boundary conditions: location, velocity, and acceleration of a lane-changing vehicle in X-axis and Y-axis.
    - Get the trajectory parameter to test: Sixth order coefficient of X-axis polynomial.
    - Generate the lane change trajectory: X-axis polynomial and Y-axis polynomial.
    - Set *Impact* = 0, which represents the total deceleration applied to the potential follower of a lane-changing vehicle during a lane-changing maneuver.
    - While (Keep iterating until the termination condition is met):
        - Compare current velocities of the lane-changing vehicle and its potential follower after a lane-changing maneuver.
        - If (Velocity of the follower > Velocity of lane change car)
            - Predict the behavior of the first Model Predictive controller on the follower, which is designed to slow down for the safety, in response to the pre-generated trajectory: Computes optimal acceleration change.
        - Else:

- Predict the behavior of the second Model Predictive controller on the follower, which is designed to recover its velocity, in response to the pre-generated trajectory: Computes optimal acceleration change.

- *Impact* += the optimal acceleration changes of the follower.
- If (*Impact* ≥ _0)
  - Return *Impact*
  - Terminates While loop

obtain_boundry_conditions

Syntax

obtain_boundry_conditions (in_veh_traj, _alpha, merge_feasibility_array, t)

Description

This function calculates the lane-changing trajectory boundary conditions: location, velocity, and acceleration of the Lane-changing vehicle at the beginning and the end of the maneuver in both X-axis and Y-axis.

Input Arguments

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

_alpha: lane-changing trajectory parameter to test: Sixth order coefficient of X-axis polynomial.

merge_feasibility_array: The array containing the vehicle IDs of the merging vehicles that can be considered for merge coordination.

t: timestep

Output Arguments

location, velocity, and acceleration of the Lane-changing vehicle at the beginning and the end of the maneuver in both X-axis and Y-axis: boundry_conditions

Pseudo Code

- Identify the immediate leader of the lane-changing vehicle in the target lane.
- Based on constant acceleration policy, calculate its future location and speed at the end of the lane-changing maneuver. Note that this study assumes a fixed 6 seconds lane-changing duration.
- Return current location, speed, and acceleration of the lane-changing vehicle at the start of the maneuver as well as desired location, speed, and acceleration of the lane-changing vehicle at the end of the maneuver.

generate_trajectory

Syntax

generate_trajectory (_alpha, boundry_conditions, t)

Description

Considering the sixth order polynomial assumption for generating the lane-changing trajectory, this function generates a feasible trajectory to meet the boundary conditions based on _alpha.

Input Arguments

_alpha: lane-changing trajectory parameter to test: Sixth order coefficient of X-axis polynomial.

boundry_conditions: location, velocity, and acceleration of the Lane-changing vehicle at the beginning and the end of the maneuver in both X-axis and Y-axis.

t: timestep

Output Arguments

location, velocity, and acceleration of the Lane-changing vehicle during the lane-changing maneuver: (_X, _Y)

Pseudo Code

- Calculate location, velocity, and acceleration in the X-axis during the 6 seconds lane-changing maneuvers using Figure 9 of this report.
- Calculate location, velocity, and acceleration in the Y-axis during the 6 seconds lane-changing maneuvers using Figure 9 of this report.
- Return location, velocity, and acceleration in both X and Y axes.

MPC1

Syntax

MPC1 (in_veh_traj, (_X, _Y), t)

Description

Considering the sixth order polynomial assumption for generating the lane-changing trajectory, this function generates a feasible trajectory to meet the boundary conditions based on _alpha.

Input Arguments

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

(_X, _Y): location, velocity, and acceleration of the Lane-changing vehicle during the lane-

t: timestep

Output Arguments

Location, velocity, and acceleration of the immediate follower in the target lane: (x, x_dot, x_ddot)

Pseudo Code

- Calculate location, velocity, and acceleration of the immediate follower in the target lane using Figure 6.
- Return (x, x_dot, x_ddot)

MPC2

Syntax

MPC2 (in_veh_traj, (_X, _Y), t)

Description

Considering the sixth order polynomial assumption for generating the lane-changing trajectory, this function generates a feasible trajectory to meet the boundary conditions based on _alpha.

Input Arguments

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

(_X, _Y): location, velocity, and acceleration of the Lane-changing vehicle during the lane-

t: timestep

Output Arguments

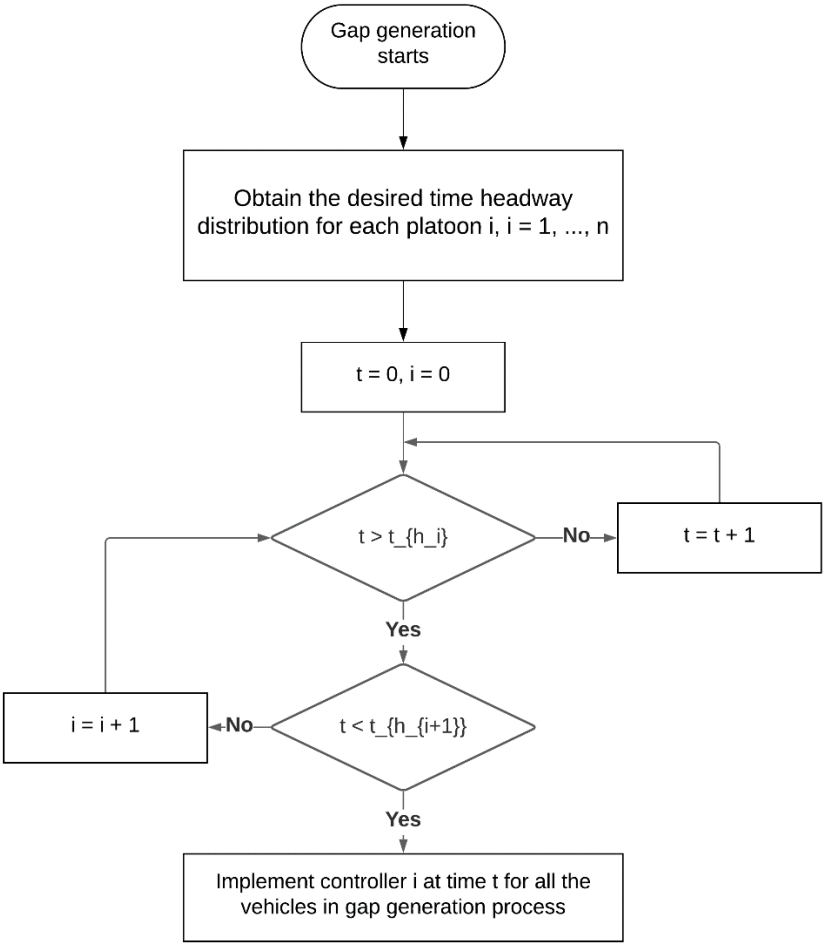
Location, velocity, and acceleration of the immediate follower in the target lane: (x, x_dot, x_ddot)

Pseudo Code

- Calculate location, velocity, and acceleration of the immediate follower in the target lane using Figure 7.
- Return (x, x_dot, x_ddot)

A.6. Gap Generation Module

This module covers the implementation of the gap generation algorithm. This algorithm is complementary to the proposed merge coordination system and aims to facilitate the merge process by generating the required gap for the lane-changing vehicle. The overall setup of this module is Figure A4.



Source: FHWA

Figure A4. Flowchart. Implementation of gap generation system.

gap_generation

Syntax

Gap_generation(in_veh_traj, merge_feasibility_array, t)

Description

This function creates the additional gap for the lane-changing vehicles to facilitate the merge process.

Input Arguments

in_veh_traj: the input vehicle trajectory array. This array possesses the following information: vehicle initial position (lane and location in the lane), initial speed, initial acceleration, gap and speed difference between the vehicle and the leader, desired speed, relative position of the vehicle in the traffic (leader's ID, follower's ID, potential left lane leader's ID, potential left lane follower's ID, potential right lane leader's ID, potential right lane follower's ID), maximum deceleration, target exit location, and vehicle class (regular, connected, or automated).

merge_feasibility_array: The array containing the vehicle IDs of the merging vehicles that can be considered for merge coordination.

t: timestep

Output Arguments

Location, speed, and acceleration of the new leaders of a lane-changing vehicle: *Impact*

Pseudo Code

- If Lane change begins
    - By solving equations in Figure 15 or Figure 16, obtain the desired time headway distribution for the n CAVs in the leading position of merging vehicles in the target lane.
    - For t in the duration of the gap generation (predefined):
        - If t < t_h_i (see Figure 15 or Figure 16 for a definition)
            - Implement controller i for the vehicles i, i+1, i+2, …

              (The controller is utilized to reduce the time gap and can be found in Figures 6 and 7).

ACKNOWLEDGEMENTS

The original map in Figure 25 has been modified and is the copyright property of Google® Maps™ and can be accessed from https://www.google.com/maps/.(#) The map overlays show the data collection location on Interstate 35 near Austin, TX.