Applications of Wireless Communication in Traffic Networks Using a Hierarchical Hybrid System Model

DISSERTATION

Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the
Graduate School of The Ohio State University

By

Yiting Liu, B.S., M.S.

* * * * *

The Ohio State University

2007

Dissertation Committee:

Prof. Ümit Özgüner, Adviser
Prof. Robert E. Fenton
Prof. Eylen Ekici
Prof. Andrea Serrani

Approved by

________________________________________
Adviser
Electrical and Computer Engineering Graduate Program
ABSTRACT

In intelligent transportation systems, one is faced with two crucial problems: traffic safety and traffic efficiency. Although much effort has been devoted to this area, many issues still remain unsolved, this dissertation presents a few attempts towards this topic by studying different vehicle operation control policies and traffic management strategies using a hierarchical hybrid system framework.

We first construct a hierarchical architecture of hybrid system for a single vehicle where the continuous time plant (vehicle dynamics) is supervised by the discrete-state controller, which is developed in a two-tier discrete event system. We then address the multiple vehicles cooperation problems by introducing an additional layer into the discrete event system that indicates the cooperation control tasks.

By employing the hierarchical hybrid system model, we develop microscopic traffic simulators that make it possible to analyze, test and evaluate different vehicle operation control policies and traffic management strategies.

We present an intersection collision mitigation approach, due to the practical relevance, we further design an inter-vehicle communication based three-level intersection warning system. Performance evaluation is presented using the traffic simulator which also mimic driver response and acceptability for alerts. Wireless communication based driver assistance systems are proposed to increase the traffic efficiency simultaneously decrease traffic accidents. System performance illustrates the effectiveness of wireless
communication in intelligent vehicle applications and the proof-of-concept of technology availability.
Dedicated to my parents in love and gratitude.
ACKNOWLEDGMENTS

I would like to thank, first and foremost, my adviser, Professor Ümit Özgüner, for his superb guidance and continuous support through the long process of my graduate studies. Thanks for broadening my scope and helping me achieving my goals. Professor Ümit Özgüner is an example of excellence in research and teaching. His encouragement and sage advice were crucial in the completion of this work.

I am also extremely grateful to the members of my dissertation committee, Professor Robert E. Fenton, Professor Eylem Ekici and Professor Andrea Serrani, for their constructive suggestions and technical supports in this research. For their generous assistance and comments, I would also like to acknowledge my talented colleagues including but not limited to Dr. Tankut Acarman, Dr. Keith Redmill, Dr. Xiaohuan Tan, Dr. Zhijun Tang, Dr. Lu Xu, Qi Chen, George McWilliams, Sangil Jwa, Rong Xu, Yongling Zheng and Yongjie Zhu.

I want to acknowledge with gratitude the financial support from the Center of Automotive Research and the Department of Electrical and Computer Engineering at different stages of my studies.

And finally, I would like to thank my parents, who have been a tremendous source of love, support, compassion, and strength for me over twenty-seven years. I am
especially like to thank them for their everlasting and boundless patience and encouragement. Little did they know that their six years old daughter would be in school for nearly twenty-two years and get a Ph.D degree from the Ohio State.
VITA

July 1979 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Born - Hangzhou, China

July 2001 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . B.S. in Information Science and Electronic Engineering
Zhejiang University
Hangzhou, China

June 2003 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . M.S. in Control
Department of Electrical Engineering
The Ohio State University
Columbus, OH

September 2001 - present . . . . . . . . . . . . . . . . . . . Graduate Research Associate,
Department of Electrical and Computer Engineering
The Ohio State University
Columbus, OH

PUBLICATIONS

Research Publications


FIELDS OF STUDY

Major Field: Electrical and Computer Engineering
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Dedication</td>
<td>iv</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>v</td>
</tr>
<tr>
<td>Vita</td>
<td>vii</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xiv</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xv</td>
</tr>
<tr>
<td>Chapters:</td>
<td></td>
</tr>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Problem Statement</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Contributions and Highlights of the Dissertation</td>
<td>4</td>
</tr>
<tr>
<td>1.4 Organization of the Dissertation</td>
<td>5</td>
</tr>
<tr>
<td>2. Literature Review</td>
<td>7</td>
</tr>
<tr>
<td>2.1 Hybrid System</td>
<td>7</td>
</tr>
<tr>
<td>2.2 Vehicle Control</td>
<td>10</td>
</tr>
<tr>
<td>2.2.1 Longitudinal Control</td>
<td>11</td>
</tr>
<tr>
<td>2.2.2 Lateral Control</td>
<td>12</td>
</tr>
<tr>
<td>2.3 Wireless Communication in ITS</td>
<td>14</td>
</tr>
<tr>
<td>2.4 Human Factors</td>
<td>16</td>
</tr>
<tr>
<td>2.4.1 Human Driver Model</td>
<td>16</td>
</tr>
<tr>
<td>2.4.2 Human Driving Behavior</td>
<td>21</td>
</tr>
</tbody>
</table>
Appendices:
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Behavior Layer: Discrete-State Dynamics</td>
<td>33</td>
</tr>
<tr>
<td>4.1 Vehicle Types</td>
<td>69</td>
</tr>
<tr>
<td>5.1 Summary of Simulated Scenarios</td>
<td>80</td>
</tr>
<tr>
<td>5.2 Simulation Parameters</td>
<td>98</td>
</tr>
<tr>
<td>6.1 Driver Response Motion Model</td>
<td>112</td>
</tr>
<tr>
<td>6.2 Comparison of Theoretical Maximum and Selected Values</td>
<td>115</td>
</tr>
<tr>
<td>6.3 Collision Improvement at Unsignalized Intersection</td>
<td>119</td>
</tr>
<tr>
<td>6.4 Collision Improvement at Signalized Intersection</td>
<td>120</td>
</tr>
<tr>
<td>A.1 Level of Service and Flow Rate (Unsignalized Intersection)</td>
<td>159</td>
</tr>
<tr>
<td>A.2 Level of Service and Flow Rate (Signalized Intersection)</td>
<td>160</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>19</td>
</tr>
<tr>
<td>3.1</td>
<td>30</td>
</tr>
<tr>
<td>3.2</td>
<td>31</td>
</tr>
<tr>
<td>3.3</td>
<td>40</td>
</tr>
<tr>
<td>3.4</td>
<td>42</td>
</tr>
<tr>
<td>3.5</td>
<td>52</td>
</tr>
<tr>
<td>4.1</td>
<td>57</td>
</tr>
<tr>
<td>4.2</td>
<td>60</td>
</tr>
<tr>
<td>4.3</td>
<td>62</td>
</tr>
<tr>
<td>4.4</td>
<td>71</td>
</tr>
<tr>
<td>4.5</td>
<td>72</td>
</tr>
<tr>
<td>4.6</td>
<td>74</td>
</tr>
<tr>
<td>4.7</td>
<td>74</td>
</tr>
<tr>
<td>4.8</td>
<td>75</td>
</tr>
<tr>
<td>4.9</td>
<td>77</td>
</tr>
</tbody>
</table>
5.1 Sensor Coverage for Subject Vehicle ........................................... 81
5.2 System Configuration for Intersection Access Analysis ................. 82
5.3 General Architecture of Human Driver Model ............................ 84
5.4 Intersection Area ................................................................. 88
5.5 Finite State Machine of Right Turn at Unsignalized Intersection ..... 91
5.6 Finite State Machine of Go Straight at Unsignalized Intersection ..... 93
5.7 Finite State Machine of Left Turn at Unsignalized Intersection ...... 94
5.8 Finite State Machine of Right Turn at Signalized Intersection ....... 95
5.9 Finite State Machine of Go Straight at Signalized Intersection ...... 96
5.10 Finite State Machine of Left Turn at Signalized Intersection ......... 97
5.11 Position Trajectory of A Left-turning Vehicle at An Unsignalized In-
tersection .................................................................................. 99
5.12 Velocity and Acceleration of A Left-turning Vehicle at An Unsignal-
ized Intersection ....................................................................... 100
6.1 Intersection Warning System Simulator Architecture .................. 104
6.2 Message Generator ..................................................................... 107
6.3 Route Contention ........................................................................ 109
6.4 Three-level Collision Warning System Algorithm ...................... 111
6.5 Percentage of Collisions at Unsignalized Intersection ................. 116
6.6 Percentage of Collisions at Signalized Intersection .................... 117
6.7 Collision Speed and Collisions Percentage (5200 Vehicles/hour) ... 121
6.8 Collision Speed and Collisions Percentage (7200 Vehicles/hour) ... 122
7.1 Screenshot of Simulation Scenario Constituted by Two Vehicle Groups without Inter-vehicle Communications ........................................ 130

7.2 Stop-and-go without Driver Assistance System ........................................ 131

7.3 Screenshot of Simulation Scenario Constituted by Two Vehicle Groups with Inter-vehicle Communications ........................................ 133

7.4 Stop-and-go with Driver Assistance System ........................................ 134

7.5 Screenshot of Merging Scenario ........................................ 135

7.6 Trace Files of Vehicles involved in Merging Scenario ........................................ 136

7.7 Speed Profile of Vehicles involved in Merging Scenario ........................................ 137

7.8 Ramp Merging Problem ........................................ 140

7.9 Virtual Mapping Method ........................................ 141

7.10 Merging Control Algorithm ........................................ 142

7.11 Vehicle State Diagram ........................................ 143

7.12 Typical Time-space Traces of the Merging Vehicle (red lines) and the Affected Vehicle (blue lines) in the Lane Changing Scenario ........................................ 146

7.13 Speed Changing Scenario ........................................ 147

7.14 Delay Time Improvement ........................................ 148

7.15 Effects of the IVC on Collision Avoidance ........................................ 150

A.1 Vehicle Intersection Traffic Simulator Icon ........................................ 156

A.2 Client Area of Vehicle Intersection Traffic Simulator ........................................ 157

A.3 Start of the Input Area Dialog Box ........................................ 157

A.4 The Input Area Dialog Box ........................................ 158
A.5 Starting a Vehicle Intersection Traffic Simulation . . . . . . . . . 162
A.6 Cursor for Simulation Process Controls . . . . . . . . . . . . . . . . 163
A.7 Lane label of Vehicle Intersection Traffic Simulator . . . . . . . . 165
CHAPTER 1

INTRODUCTION

1.1 Motivation

Two most important issues in transportation systems are traffic safety and efficiency problems. Traffic safety aims to reduce accidents of vehicles traveling in the traffic network. Reduction of traffic congestion, improvement in vehicle travel time and delay, etc. are the primary concern of traffic efficiency. Both of them have attracted much attention. One of the major causes of the unsafe and inefficient traffic network is the lack of instant and sufficient information (due to the drivers’ limited perception when we concentrate on effects of the drivers). Drivers usually make decisions mainly based on the information within their visual range. The prevalent warning method only rely on brake-light, which is neither efficient nor safe. With the availability of sensor and wireless communication technology, the techniques of sensor filtering, inter-vehicle communication (IVC) and road-vehicle communication (RVC) are introduced to increase perception limits (e.g. visual range) and simultaneously decrease reaction limits (e.g. driver response time). Sensors are used to detect and obtain vehicle’s and environment’s information, such as vehicle’s state information and obstacle’s position. RVC exchanges information between vehicle and
a fixed roadside entity. IVC, on the other hand, entails two or more vehicles forming an ad-hoc network. Sensor technology and wireless communication provide the interconnections between the vehicles, which at the same time also synchronize decision making, for example, cooperative driving commands, across different vehicles.

Studying the traffic network with these technologies is a challenging task as the challenges come from the traffic system itself, as well as the inputs from the sensor and wireless communication. The traffic system itself is a large scale, multi objects system [2]. Drivers/Automated vehicles not only optimize locally to satisfy their own desires, but also cooperate to resolve conflicts, e.g. to avoid collisions using the exchanged information. A system framework is in need due to the increasing complexity of traffic network, which requires complex vehicle decision making with fast response and robust performance. At the same time, the inputs from the sensor and wireless communication form a multi-objective control problem for cooperative vehicles, which has to meet a set of new control requirements and specifications where some of them are conflicting [2].

1.2 Problem Statement

In this study, we will investigate the traffic safety focusing on collision avoidance and efficiency concerning reduction in vehicle travel time and delay for traffic network system based on a hierarchical hybrid system model. The hybrid system model is able to integrate the discrete and continuous nature of the traffic network. The deployed hierarchical architecture can decompose complex control tasks into sequences of simple and easy to manage vehicle operations. Our goal is to solve traffic safety problem
while still achieve traffic efficiency. The following underlying assumptions are applied in our study:

1. Vehicle operations take place under normal conditions, e.g. within vehicle physical limits;

2. Vehicle model parameters are known, e.g. length, width, acceleration, etc.;

3. Collisions are instantaneous and perfect elastic;

4. Vehicles have equal masses;

Each vehicle here is modeled as a moving object in a 2-D space, whose state is denoted as $X^i(t) \in \mathbb{R}^n$. Let $N$ be the number of vehicles considered, we have $X(t) = \{X^i(t)\}_{i=1,...,N} \in \mathbb{R}^{N \times n}$. Generally, vehicle’s state contains information of its own position, velocity, and acceleration. In order to describe objects in motion, we should first be able to unambiguously specify their position. Let $x^i_x$ and $x^i_y$ denote vehicle $i$’s point mass position along x-axis and y-axis in Cartesian coordinates with respect to a fixed reference on the road respectively. If we introduce $\alpha^i(t)$ as the heading angle of vehicle $i$ and $v^i(t)$ is the velocity of vehicle $i$ at time $t$ respectively, the vehicle dynamics can be expressed as:

\begin{align}
\dot{x}^i_x(t) &= v^i(t) \cos \alpha^i = v^i_x(t) \\
\dot{x}^i_y(t) &= v^i(t) \sin \alpha^i = v^i_y(t) \\
\dot{v}^i(t) &= a^i(t) \\
\dot{\alpha}^i(t) &= \omega^i(t)
\end{align}

where $v^i_x$ and $v^i_y$ denote vehicle $i$’s velocity along x-axis and y-axis in the same coordinates as the position respectively. And $a^i_x$ and $a^i_y$ denote vehicle $i$’s acceleration along x-axis and y-axis in the same coordinates as the position respectively. $a^i(t)$ is the acceleration of vehicle $i$ at time $t$, and $\omega^i(t)$ is the angular velocity.
Though we model the vehicle as a point mass when describing its dynamics, its size need to be taken into consideration as we are studying the collision avoidance problem. We define a collision between vehicle $i$ and vehicle $j$ at time $t$ takes place if and only if there exists overlapping area between vehicle $i$ and vehicle $j$. Similarly a collision between vehicle $i$ and obstacle $j$ at time $t$ takes place if and only if there exists overlapping area between them.

Through our study, we want to obtain the above listed safe conditions for each vehicle in traffic network while maintain reasonable efficiency. More specifically, we want to

1. Develop generic strategies and tactics for collision avoidance/collision warning;
2. Develop generic strategies and tactics for traffic network efficiency;
3. Seek generic accident severity metrics to evaluate and quantify the damage and loss of developed collision avoidance/collision warning algorithm;
4. Apply the above theoretic results to the traffic management in different scenarios.

1.3 Contributions and Highlights of the Dissertation

The contribution of this dissertation are threefold. The first contribution is the development of a hierarchical hybrid system model for vehicles in traffic network. Such a model lends itself naturally to the description of vehicle control tasks and the vehicle dynamics in response to task based vehicle operation state.

The second contribution is the development of microscopic traffic simulators to support evaluating vehicle operation control methodologies and traffic management
strategies. The animated graphics displays in the simulators allow the user to study the overall traffic performance in the simulated area or to examine the behavior of any selected vehicle or vehicles in great detail.

The third contribution is the design of collision warning system and driver assistance systems, which help to increase the traffic efficiency while decrease the traffic accidents. The performance evaluation of the suggested systems illustrates the effectiveness of wireless communication in intelligent vehicle applications and the proof-of-concept of technology availability.

1.4 Organization of the Dissertation

The material of this dissertation is organized in eight chapters. In Chapter 2, a brief literature review in the research area related to this study is presented.

In Chapter 3, a hierarchical hybrid system model for vehicles in a traffic network is proposed. The system is designed for safety and efficiency specifications as a primary capability in addition to the usual requirements such as real-time constraints, etc.

Chapter 4 describes microscopic traffic simulators and their components in detail. The control policies and traffic management strategies, for the most part, are parametrically user defined. Users can evaluate, analyze and test the performance variations by changing parameter specifications in the traffic simulators.

The goal of Chapter 5 is to study the human driver model at the intersection, a series of finite state machines describing the human driver model are developed. The developed model, especially the driver decision making procedure, can be implemented into the microscopic simulator for mimicking human driver behavior and designing human-centered collision warning/avoidance algorithms.
In Chapter 6, inter-vehicle communication based intersection warning system is presented. The warning system performance is evaluated from both macro and micro perspectives using developed microscopic traffic simulator.

Chapter 7 proposes wireless communication based driver assistance systems, which help to increase traffic efficiency and simultaneously decrease the traffic accident.

Finally, Chapter 8 contains the concluding remarks and directions of future work.
CHAPTER 2

LITERATURE REVIEW

In this chapter, we first review the history and literature of hybrid system in Section 2.1, then review the development of vehicle control in Section 2.2. Application of wireless communication, especially in Intelligent Transportation Systems (ITS) is reviewed in Section 2.3. Research works of various types of general human driver models and human driver behaviors are presented in Section 2.4.

2.1 Hybrid System

Hybrid systems has emerged as a technique for modeling the mixture of discrete event with continuous system according to differential equations that can occur asynchronously. The initial concept of finite control of continuous state system was introduced as early as in 1970s [3]. Generally speaking, hybrid models can be divided into four classes: Automata and Transition Systems (Aut), Ordinary Differential Equations (ODE), Algebraic Approach (Alg), and Programming Languages (Prog) as proposed by Labinaz, Bayoumi and Rudie [4]. The common approaches for modeling hybrid systems are classes Aut and ODE, which distinguish models using discrete and continuous dynamic descriptions. Alg class models are used to establish an algebraic formalism for the hybrid systems. Prog class model is different from the above
three classes, where the hybrid model is implicitly defined by a programming language instead of explicitly described [4]. We will review the developments in the first two classes in the following.

Passino and Özgüner developed a hybrid system model which consists of a discrete event system (DES), a continuous time system (CTS) and an interface [5]. Various notions of reachability and stabilizability were then defined based on the introduced hybrid system model by using local controllers for the DES and general hybrid system controller for the CTS. They provided a relatively straightforward general hybrid system model and illustrated the process of modeling and analyzing several hybrid systems, e.g. a robotic system. Doğruel and Özgüner then studied controllability, reachability, stabilizability and state reduction in a discrete state system automata [6, 7], which is a key part in the hybrid system. They defined and investigated asymptotical stability and stabilizability of a set of matrices. Using Lyapunov theory, they also provided the necessary and sufficient conditions for asymptotic stability and stabilizability. Some preliminary work on the stability analysis of switched and hybrid systems was presented by Branicky, where multiple Lyapunov functions are used as tools for analyzing [8]. The case that the switched systems are indexed by an arbitrary compact sets are also discussed.

Akar, Mitra and Özgüner provided one of the early applications of the hybrid modeling framework [9]. They studied the handoff problem for cellular communication system with respect to signal-strength measurements. The system is formed by an automaton and a finite number of discrete-time systems. With the proposed algorithm, one is able to choose the continuous inputs and control the switchings within the automaton in an optimal sense.
Considering the application of the hybrid modeling in vehicle control, hybrid controller for local decision making in an autonomous vehicle is designed by Özgüner, Hatipoğlu and Redmill [10]. They developed a supervisory controller that can generate event-driven commands to continuous state low-level controllers. The continuous state system describes the motion of the vehicle for a given set of input-output pairs and is capable of automated stop, speed up, slow down, lane keeping and lane change using radar and/or vision based steering data. The discrete event system is modeled as a finite state machine with transitions are manipulated by some “events” in the continuous state system. Choi and Hong later presents a hybrid system approach to longitudinal speed and traction control of a vehicle with wheel slip constraints [11]. Using hybrid system concept, the vehicle system is divided into local subsystems and the control of each subsystem is selected in terms of control modes and operating region. The controllers are able to track a desired speed value while maintaining the safety constraints. A methodology for designing safe and efficient lane change control laws for automated vehicles is explored by Godbole, Sengupta and Hagenmeyer [12]. They designed a distributed hybrid control for automated vehicle lane changing that involved control for continuous state with mode switching for each vehicle and simple event-driven. They also showed that the optimal control for minimum time and minimum distance lane changes are different if the longitudinal and lateral tire forces are couples.

Zelinski, Koo and Sastry proposed a hybrid system design for formations of multiple autonomous vehicles [13]. They focused on the Formation Reconfiguration Planning (FRP) problem that is to determine a nominal state and input trajectories for each vehicle with given initial and final formation configuration, time and a set of
constraints. Each set of configuration parameters represents a hybrid configuration mode. When coupled with formation keeping mode, the hybrid system consists of a hybrid automation and the transition is controlled by a finite automaton. The implementation of the optimal control for this hybrid system is studied. Some research works have also been done specifically in air traffic management for multiple air vehicles, where cooperation between aircrafts is the main objective [14].

Further, many efforts have been devoted to the traffic network, most of the research on hybrid systems are the design of safe and efficient hybrid controllers for vehicle regulation on an Automated Highway System (AHS) [2, 15]. Lygeros, Godbole and Sastry used safety-guaranteed hierarchical hybrid controller to deal with the regulation of vehicles. The method also suggests in which way the inter-vehicle information can be modified to improve the system performance.

More designs and approaches of hybrid system can be found in literature [16, 17, 18].

2.2 Vehicle Control

In this section, we are reviewing the control algorithms that have been applied to a single vehicle control. In general, studies on the design of an automated vehicle system involve the solution of two decoupled control systems as proposed by Hatipoğlu and Özgüner: the longitudinal (headway) control (i.e., speed and inter-vehicle gap control), and lateral control or steering control (lane keeping) [19]. Much work has been done since 1950s, and various kinds of control algorithms and techniques have been employed there, including the conventional control theory and the modern control theory [20, 21].
2.2.1 Longitudinal Control

The longitudinal control has a longer history of research than the lateral control, as the longitudinal control has been playing an important role in the railways for a long time [20].

The longitudinal control for a single vehicle is speed control (cruising at the selected speed or keeping relative speed and relative position of the controlled vehicle with respect to backward and forward vehicles at the safe distance in the highway traffic). A longitudinal controller was designed by Hauksdottir and Fenton to achieve a velocity-invariant response, physical realizability and comfortability [22]. A nonlinear, observer-controller compensator is designed for all phases (non-emergency) of longitudinal control.

Started in late 1980s, there were many contributions to the longitudinal control that can drive a platoon maintaining a small inter-vehicle gap. Recently adaptive cruise control (ACC) system has been introduced as a technological improvement over existing cruise [23, 24, 25, 26, 27, 28]. ACC system regulates the vehicle’s speed to follow the driver’s set point if there is no preceding vehicle or obstacle insight. Besides, when a slower vehicle is captured ahead, the ACC controlled vehicle will follow the vehicle ahead with a safe distance by adjusting the relative speed. ACC controllers use sensing devices such as radar to determine and maintain safe distance among vehicles at given velocities and vehicle conditions while simultaneously maximize road capacity. More specifically, there are two approaches in ACC to form platoons: Autonomous intelligent cruise control and Cooperative intelligent cruise control. Communicating with exterior sources is not supported by autonomous intelligent cruise control, therefore, the driver needs to set the desired speed and headway.
In contrast, cooperative intelligent cruise control allows the vehicle to communicate with the other vehicles and roadside infrastructure to perform platoon maneuvers [29].

Yi and Horowitz studied the qualitative relationships between traffic flow stability and model parameter for a generalized ACC traffic flow model [25]. The concept of a semi-autonomous adaptive cruise control (SAACC) where both manually driven and adaptive cruise controlled vehicles can coexist is introduced by Rajamani and Zhu [24]. Traffic flow stability has been carefully investigated since 1990s [23, 28]. It has been proven that the most conventional control law, constant time gap, can ensure string stability [23, 24].

At present, ACC system is capable of maintaining controlled vehicle’s position relative to the leading vehicle in various traffic conditions including congested traffic and even city traffic by using stop-and-go features while maintaining a safe distance between leading and following vehicles autonomously [30].

2.2.2 Lateral Control

The lateral control of a vehicle in ITS is aiming at steering a vehicle along a planned path [21]. It is especially essential in automated lane changing operations.

The general procedure for executing the lateral control is done by the detection of a reference on a roadway indicating a planned path with sensing system, which provides the feedback control system [21]. Tracked back to 1950s and 1960s, the references in the automated driving systems were inductive cables embedded in roadways. PD and PID control algorithms are employed for lateral control. A pair of inductive coils attached to the vehicle’s rear bumper is used to detect and measure the yaw angle of the vehicle relative to the planned path [21]. A velocity-adaptive lateral controller,
which involved an inverse compensation stage for velocity-independent performance and a lead/lag stage for both lateral-position tracking accuracy and ride comfort, was designed by Fenton and Murthy [31]. The lateral reference and lateral position error is provided by a wire-follower configuration.

Recently, most of the lateral control algorithms on lane changing are based on a reference that is virtually generated [21]. Chee and Tomizuka proposed applications of the Frequency Shaped Linear Quadratic (FSLQ) control and the sliding mode control for three kinds of virtually planned paths with limitations on the lateral acceleration on lane changing [32]. FSLQ optimal controller is designed to achieve a smoother and robust control action. The sliding mode controller is implemented to improve ride comfort utilizing a filtered error signal. The lateral control presented by Jochem, Pomerleau and Thorpe is based on a virtual field of view generated from the real field of view [33]. The proposed methods utilize a simple geometric model of the road to position a virtual camera for lane transition maneuvers. Images are taken as input and consequently produce a point on the road to driver over and the output is the measure of its internal confidence in this point. Young and Özdün̈er have previously examined an optimal automatic steering control with fixed terminal constraints, which is based on an elegant time varying sliding manifold design approach [34]. The approach makes optimal control formulations with fixed time horizon and fixed terminal constraints more plausible. Kato, Tomita and Tsugawa extended the lateral control algorithm based on the approximation of a reference with a cubic curve [35]. The theoretical reasoning behind the Partners for Advanced Transit and Highways (PATH) program’s lane changing controller summarized and the experimental results obtained in San Diego, CA are presented in [36]. Hatipoğlu,
Özgüner and Redmill provided analytic approach for the systematic development of controllers that cause an autonomous vehicle to accomplish a smooth lane change maneuver. The task is accomplished by the generation of a virtual yaw reference and the utilization of a robust switching controller to generate steering commands that cause the vehicle to track that reference [37].

### 2.3 Wireless Communication in ITS

Wireless communication has a strong potential to improve traffic flow, traffic safety and driver comfort. It can be used to provide warning information to the driver, for example at intersections, or can be used to automate task like lane merging or as extension to adaptive cruise control system. Travel information can also be sent to the drivers for route decision or entertainment.

Infrared and radio waves are the main media for wireless communication in ITS. The radio waves include VHF and micro waves which are of a type of broadcasting and millimeter waves which are of a type of line-of-sight [38]. Much work has been done on the protocol for the wireless communication in ITS, but only a few has been tested by experiments. The slotted ALOHA was employed by CarTALK, where the network may consist of two vehicles [38]. The non-persistent CSMA, DOLPHIN (Dedicated Omni-purpose inter-vehicle communication Linkage Protocol for HIghway automatioN) was employed in Demo 2000 cooperative driving [39] and intersection collision warning system [40, 41].

Origin of applications of wireless communication in transportation system can be traced back to mid 1980s in Japan, two driver information service systems were developed, Advanced Mobile Traffic Information and Communication Systems by
National Police Agency and Road/Automobile Communication System by Ministry of Construction [42]. Real applications using wireless communication in ITS have recently become the subjects of substantial research in both academia and industry.

Several collision warning/collision avoidance systems using wireless communication have been proposed to improve the traffic safety [43, 44, 45, 46, 40, 41, 47]. Miller and Huang proposed a peer-to-peer adaptive collision warning system at intersections [44]. A dynamic ad hoc wireless network for information data sharing is used to support intersection collision warning algorithm. A collision avoidance system that integrates IVC and RVC was developed by Tsukamoto et al [46]. A three-level collision warning system at intersections using distance based message generator was reported by the Ohio State University [40, 41].

Applications of cooperative driving based on wireless communication have been explored to alleviate traffic congestion and increase road efficiency [48, 49, 50, 39]. Cooperative driving on multiple lanes was demonstrated in Demo 2000 in Japan [39]. With 5.8GHz DSRC inter-vehicle communication equipment, vehicles were able to perform stop and go of the platoon, merging into and splitting from a platoon, passing, and obstacle detection. A cooperative adaptive cruise control system was designed and tested by Bruin et al. [48]. Inter-vehicle communication is used to include preview information from vehicles further in front to design the proposed control system. The system enables anticipatory braking actions which can reduce the shock waves in the traffic network and in turn leads to a positive effect on traffic flows. Li and Wang studied the cooperative driving at the blind intersections to avoid traffic jams [49, 50]. A “safe driving patterns” concept is introduced to represent collision free operations of all vehicles at crossings.
2.4 Human Factors

Today’s researchers have been working toward the automatic driving technology, a manifestation of human intelligence will be then no doubt reflected by perceived benefits, or advantages to the individual. Humans not only bring a number of truly admirably skills for driving target that most automation technology can not simulate, they also help to improve driving safety and driving learning.

The focus of this section is to conduct human factors literature review for examining driver behavior, especially at intersections (both signalized and unsignalized intersections). The goal of the searches summarized here is twofold. On one hand, there is an increasing need in the real world for human-centered design and control. For example, designing vehicles, managing traffic flows, developing driver assistance systems and even planning the geometric of highway facilities and urban roads. On the other hand, we would also like to integrate a comprehensive, quantitative human driver model with automation into micro level simulation tools for traffic simulation and performance evaluation at traffic level [51, 52].

2.4.1 Human Driver Model

A. Descriptive Human Driver Model/Behavioral Model

Since 1960s, researchers started to model driver’s behavior in the field of human factor. Descriptive model is the first straightforward idea. Chandler, Herman and Montroll represented the driver-vehicle system by a proportional time controller with time delay, where the driver’s sensitivity factor is a constant [53]. The simple model proved to be useful in car following studies of local stability [54]. Mourant et al. used an eye-marker camera to record drivers’ visual search and scan patterns under
different driving conditions. Human driver behavior was organized in various driving
tasks. Some mathematical model, such response time formula were also added for
description purposes [55]. Burnham et al. extended Chandler et al.’s work and
developed two mathematical models of driver behavior in single-lane car following
situation. With standard parameters identification algorithms, optimum parameter
values were obtained [56]. Descriptive human driver model provides direct ideas of
human driver under different kinds of scenarios.

However, descriptive human driver model/behavioral model does not provide
enough information for the reproduction and prediction of human behavior. More-
over, those descriptive language of human driver’s behavioral data, like most behav-
ioral data for many real world tasks, are multimodal, continuous, noisy and even
confusing.

B. Risk-based Human Driver Model

Risk-based human driver model combined driver’s motivation, experience etc.
with risk perception, acceptation etc. together and focused on driver’s process of
thoughts. This type of model is also used for the design and development of the
driver assistance system. Hoedemaeker studied the individual driver behavior and
acceptance of ACC. Using experimental driving simulator, under certain conditions
and limitations, ACC shows itself as a promising technique [57].

Konig et al. used artificial intelligence to model the driver and his behavior. The
driver’s decision was conducted by a rule-based system. Advanced traffic management
system and advance traveler information system were evaluated within the above
suggested model [58].
Brown developed and tested two human driver models for evaluating driver response to rear-end collision avoidance systems warnings under various rear-end collision scenarios. The presented human driver models can be also used to pair design and evaluation issues to the appropriate model types. The human driver model includes a detailed attention and information-processing model that integrates driver perception of the environment with deceleration [59].

C. Cognitive Human Driver Model

The term cognitive human driver model means a model which is a simplified representation of reality. The essential quality of the cognitive human driver model focuses on human driver’s psychological activities during the driving. Researchers make a distinction between behavioral model and cognitive model, where behavioral model is a kind of descriptive models that people know what drivers will do but do not know why. Cognitive model on the other hand can help to develop understanding of driver behaviors.

i. The COSMODRIVE Cognitive Model [1]

Considering roadway human factors and behavioral safety in Europe, researchers from Finland, Netherland and France emphasized the need of cognitive models. Bellet et al. from the French Institute for Transportation Research and their Safety (INRETS), developed the COSMODRIVE (Cognitive Simulation Model of the Driver) model, which simulates human cognitive processes using computers. Figure 2.1 shows the architecture of the proposed human driver model. COSMODRIVE takes both theoretical framework which cognitive psychology and ergonomics applied and implementation where artificial intelligence is taken into consideration. The goal is
to reproduce and predict driver behavior in various situation (environment, driver experience, etc.).

![Figure 2.1: The COSMODRIVE Cognitive Model (INRETS) [1]](image)

As shown in Figure 2.1, strategic, tactical and operational modules are three basic modules. Strategic module is served as a navigator and generator of general objectives. The tactical module generates the internal representation for the road environment. The operational module has a number of autonomous units which take care of the elementary driving tasks. Other modules are added for auxiliary purpose. For example, integration of human characteristics with different driving scenarios are done by perception module.

**ii. The Cognitive Human Driver Model by PATH [52]**
PATH researchers extended and organized the COSMODRIVE framework for the purpose of driver modeling in their SmartAHS. The model allowed simultaneous simulation of vehicles controlled by drivers and semi-automated systems for comparisons between automated vehicles and human drivers. Driver’s knowledge database and the cognitive process underlying the driving activity contributes to the cognitive approaches [60]. Later the researchers continued their efforts on human driver modeling development. By developing more processing mechanisms that access the effectiveness of other driving assistance systems, the capabilities of the human driver model were refined and extended. The intention of the model is to provide a tool to evaluate and analyze the effects of ITS using throughput criteria by reproducing and predicting the information processing string developed by the driver. Besides the integration of the strategic module and the itinerary representation, perception capacity is augmented [61].

iii. Other Cognitive Human Driver Models

One of the most famous cognitive model to model human-machine interaction was proposed by Rasmussen. In his cognitive model, the information processing was organized in the hierarchical structure based on demand complexity, knowledge base, rule base and skill base [62]. Michon corresponds Rasmussen’s hierarchical model with strategic, tactical, and operational levels [63]. Boer et al. developed a human driver model based on Rasmussen and Michon’s work [64, 65]. Kuge et al. extended the model with a driver behavior recognition method using Hidden Markov Models (HMMs), which characterized and detected driving actions and placed it in the framework of a cognitive human driver model.
Salvucci applied Carnegie Mellon University’s ACT-R to predict, recognize and track driver behavior and intention. Especially he studied distraction from secondary tasks. One example is distraction from cell-phone dialing. Four key steps are conducted to develop the cognitive model architecture. Firstly, driver’s behavior was observed and a moving window was maintained for the entire multimodal stream. Secondly with the same environmental variables for the human driver, a number of actions were generated using ACT-R driver model. Thirdly, best match between the prediction and real situation were found. Consequently, the best-matching model gave predicted and observed actions, the most likely cognitive process that generated the observed action was inferred [66].

2.4.2 Human Driving Behavior

In this section, a survey on human behavior, especially in intersections area are conducted. Human behavior is important for behavioral model, where it serves as the main body. It is also essential to the cognitive and other human models, since researchers need the behavioral data to map the observed behaviors to cognitive process.

A. Understanding of the Traffic Signs

Williams et al. investigate the understanding of left-turn signal indications and auxiliary sign via a mail survey in Texas. Four different types of display configurations and 40 scenarios of feasible left-turning signal/auxiliary sign combination are studied. The survey showed that the presence of the circular green is often interpreted as a permissive rather than protected left-turn even with an auxiliary sign. Green arrow leads to a better understanding comparing with the green circle, as an indication of
Bonneson et al. performed a survey on driver understanding of protected and permitted left-turn signal displays (PPLT). The survey indicated that only 70% people correctly understood the meaning of the PPLT signal design. There was a trend toward a decreased understanding of the PPLT design with increased age and driving experience. Similarly the survey also showed a trend toward better understanding with more education. Human drivers have a better understanding of PPLT design with any of the following characteristics: modified protected indication, PPLT head centered over the opposing left-turn lane and no auxiliary sign [69, 70].

El-Shawarby et al. studied the effect of yellow-phase trigger on driver behavior at high-speed signalized intersections. The experiment was carried out to analyze for five trigger distances as drivers approach the intersection at a speed of 72 km/h (45 mph). The experiment demonstrated that the probability of stopping varies from 9% at the shortest yellow-phase trigger distance of 32 m to approximately 100% for the longest 111 m trigger distance [71].

**B. Driver’s Error and Failure at the Intersections**

Driver’s error and failure are defined by Wierville et al. as: “... the failure to achieve a sequence of mental or physical activities through a thought-out plan-of-action. For example, within the driving environment, an error is committed when a driver does not successfully stop for a red traffic light because he or she depresses the accelerator instead of the brake pedal. (p.1)” [72, 68]. Based on the above
definition, Wierville et al. concluded the following taxonomy of contributing factors which affect driving performance in the sense of driver’s error and failure in the intersection area [72, 68]:

**Inadequate knowledge, training, skill:** Driver may not have enough knowledge or misunderstanding of traffic laws, the kinematics and physics of vehicles. Driver may also do not have enough driving techniques. His incapacilities and limitations will also lead to errors/failures.

**Impairment:** Impairment includes fatigue, drowsiness, use of illegal drugs, alcohol and so on. Health related issues are also classified in this category, such as illness, lack of use or incorrect use of medication, disability, uncorrected disability and so on.

**Willful inappropriate behavior:** Willful inappropriate behavior is usually because of driver’s intention, such as purposeful violation of traffic laws, regulation, aggressive driving, and improper use of vehicle.

**Infrastructure, environment problems:** Those problems include traffic control device related problems, roadway related problems, and weather/visibility.

The University of North Carolina Highway Safety Research Center showed that the main cause for elderly drivers involved in crashes are either because of the difficulties in distinguishing target vehicles from surrounding clutter or judging the speeds of target vehicles, and/or an inability to use the acceleration capabilities of the cars they are driving [73, 70].

Campbell et al. investigated the fatal crashes due to signal and stop sign violations and found that alcohol, speeding, and inattention are the three most common
contributing factors for fatal crashes at traffic signals and stop signs. Through their efforts, no major differences were found among the crashes categories regarding the infrastructure where these fatal collisions occurred. It is also discovered that fatal crashes involving a light vehicle violating the traffic signal or stop sign occur in similar locations, no matter how many vehicles are involved in the crashes [74].

Bougler et al. at PATH also identifies the main cause of misjudgement for elderly drivers is cognitive and perception error [75, 70].

**C. Age and Gender Effects**

Through Williams et al.’s survey of understanding of traffic signs in the intersection area, the most important factors observed that influence the number of wrong answers are years of driving and age. Drivers with 11 to 20 years driving experience had the smallest numbers of incorrect response. Drivers with the lowest percentage of incorrect response are aged 26 to 35. 65 years of age and older presented the highest percentage of incorrect response [76, 68].

Staplin et al. showed that elderly drivers were over represented in intersection car crashes because of the complexities inherent at intersections. Through their experiment, elderly drivers in all cases were much slower than the younger drivers for making decision and the difference was about a half second [67, 68].

The University of North Carolina Highway Safety Research Center analyzed the older drivers behavior at intersections. The results indicate that both the “young elderly” who are aged between 65 and 74 and the “old elderly” who are aged over 75 seem to have problems at intersections. Left-turning maneuvers at signalized
intersections and turning or entering maneuvers at stop-controlled intersections are two main problems [73, 70].

Bouglar et al.’s research showed that older drivers represent a higher crash risk, and both older drivers and their passengers have a higher probability to injury in a collision. Among all the cross-path collision types for the elderly, “Left Turn Across Path-Opposite Direction” (30%) is the most common, then is “Straight Crossing Path” (28%), and “Left Turn Across Path-Lateral Direction” (20%). Also in both right and left crossing path collisions, older drivers are more often controlling the turning vehicle rather than the vehicle proceeding straight. The study again demonstrated that older drivers are cited for traffic violations in cross-path collisions more frequently than adults age 30 to 64 [75].

El-Shawarby et al. presented that drivers aged 65 and older group are more likely to stop at the onset of a yellow-phase trigger (74% compared to 66% for driver less than 65 years-old). If drivers are at the dilemma zone boundaries, the experiment indicated that younger drivers are approximately 20% more likely to attempt to run the yellow light when compared to older drivers. The point of highest uncertainty moves closer to the intersection as the driver’s age increases [71]. El-Shawarby et al.’s study also demonstrated an increase in the probability of running for male drivers when compared to female drivers. This difference increases as the trigger distance decreases [71].

D. Driver’s Characteristics

Gattis et al. analyzed intersection angles and the driver’s field of view. The authors concluded that with a 13.5-degree vision angle in some restrictive vehicles,
the 60-degree minimum intersection angle allowed by A Policy on Geometric Design of Highways and Streets (the “Green Book”) will lead to the driver’s line of sight to be obstructed by the vehicle itself and will reduce the sight distance available to the driver. The research suggested that a minimum intersection angle of 70 – 75 degrees will offer an improved line of sight [77, 70].
CHAPTER 3

HIERARCHICAL HYBRID SYSTEM MODEL

3.1 Introduction

Hybrid system, a class of dynamical systems which exhibits both continuous and
discrete dynamic behavior, has attracted much attention of researchers from various
fields. As reviewed in Section 2.1, significant efforts have been devoted to this branch
of science [5, 2]. Hybrid system has been used as mathematical model for many im-
portant applications, such as in applications of physics, e.g. bouncing ball [78, 79],
chemical reactions control [80, 79], production line control and so on [81, 79]. In
traffic network, vehicle combines mode switching dynamics with continuous evolution
according to differential and/or difference equations, which pose challenging but in-
teresting control problems. In this chapter, we present an architecture for vehicles in
traffic network, that is hierarchical and hybrid. The hybrid system can characterize
the discrete and continuous nature of the traffic network. The deployed hierarchical
architecture can decompose complex control tasks into sequences of simple and easy
to manage vehicle operations.
3.2 Single Vehicle Model

We first discuss the hierarchical hybrid system with single vehicle case, which is a basic building block in the traffic network, since multiple vehicles can be decomposed into parallel and interconnected single vehicles. Single vehicle here is defined as an isolated vehicle that is not affected by other vehicles. One example is the vehicle in free traffic flow. The general control objectives of single vehicle control are defined according to the environment and its own demands, e.g. minimizing the trip time for the desired route. Some of the problems, which should be solved by a single vehicle control, are [79],

- Accident mitigation (obstacles avoidance, etc.);
- Getting and maintaining safe speed (acceleration, cruising, deceleration, etc.);
- Vehicle trajectory tracking;
- Basic maneuvers support (lane change, etc.).

We assume that vehicles possess sensor subsystems/modules (e.g. Global Positioning System (GPS) and LIDAR for autonomous vehicles, eyes for human drivers), which provide them with the information of themselves (e.g. vehicles’ positions) and the environment (obstacles’ positions, other vehicles’ information if multiple vehicles available, etc.). These information are used by vehicle controllers for executing missions within real time bounds in the complex, uncertain and even confusing operating environment.

In this section, we consider single vehicle case, which is modeled as an hierarchical hybrid system. The basic idea is to map vehicle control tasks into a set of vehicle
operations, which are decomposed to sequences of different vehicle behaviors. The behaviors are then mapped to sequences of different vehicle actuators in the regulation level, which served as an interface between the discrete (higher level) and continuous parts (lower level). In our approach, vehicle’s objective/control task decides the mode in the operation layer. The mode will be decomposed to a sequence of finite discrete states in the behavior layer. In each discrete state, there is a set of control outputs defined and a corresponding regulator is designed via interface in the lower level with vehicle dynamics controller. Switchings between these discrete states are based on current state and the input event. The lower level control, vehicle dynamics control in the continuous time manner, such as lateral and longitudinal control is typically dynamic model based and is designed according to the discrete state in the behavior layer using the information provided via sensors.

Figure 3.1 shows the hierarchical hybrid system (denoted with $\mathcal{H}_S$) for single vehicle. We propose this system model to analyze and evaluate traffic network with respect to safety (e.g. collision avoidance) and efficiency (e.g. traveling time) improvement in road traffic, case studies will be presented in Chapter 5,6,7. The higher level discrete event system (DES) model (denoted with $\mathcal{D}$) is organized in a two-tier hierarchy and continuous time system (CTS) (denoted with $\mathcal{S}$), vehicle dynamics, is modeled in the standard manner by differential equations [5]. The interface is used to convert higher level command to lower level one, and simultaneously send the feedback from the lower level to the higher level. In the following sections, we will discuss each element in a top-down fashion in details.
Figure 3.1: Hierarchical Hybrid System Model for Single Vehicle
3.2.1 Operation Layer

The top level in the DES $D$ is the operation layer, in which there are two modes determined by vehicle’s objectives/control tasks.

- **Free Mode:** We may view single vehicle that is not affected by other vehicles and environment as in the free mode, for example, it can be an isolated vehicle with no leader and follower on a clear road. The objective of the free mode is to complete the desired route in a shortest period in the sense of efficiency.

- **Restricted Mode:** Though we deal with the single vehicle case, in which it will not be affected by other vehicles, the environment, for instance, road condition, may influence it. An obstacle requires immediate attention for safety. In this case we have prioritized objectives, where safety (e.g. collision avoidance) has the higher priority than the efficiency (e.g. shortest traveling time). Another mode is therefore defined as the Restricted mode.

![Figure 3.2: Mode Transitions in Operation Layer for Single Vehicle](image)

As the purpose of a single vehicle’s control may change due to the change of the environment, transitions between two modes take place depending on the
information from environment. A danger indicator (denoted as $I_d$) is developed taking on values such as 0, 1 to indicate the potential danger based on the environment event (e.g. obstacle ahead) and used to decide when to switch from one mode to another or remain in the original mode, as shown in Figure 3.2.

### 3.2.2 Behavior Layer

The lower level of the DES $D$ is behavior layer, where the behaviors are defined as a set of skills and abilities that vehicle possesses. The set of skills and abilities is converted into a set of discrete states from hybrid system point of view. We denote the discrete state at behavior layer as $S^i$, that $S^i \in \mathcal{B}$, a discrete input from the interface as $U^i$, that $U^i \in \mathcal{B}$, and a discrete output as $Y^i$, that $Y^i \in \mathcal{B}$, where $\mathcal{B} \triangleq \{A, D, C, S_t, S\}$ ($A$, $D$, $C$, $S_t$ and $S$ respectively mean Acceleration, Deceleration, Cruise (Speed Cruise), Steer and Stop) is associated with vehicle $i$ ($i = 1, 2, \ldots, N$). Note that we are considering the single vehicle case which does not necessary mean that there is only one vehicle in the system, but means the vehicle is not affected by other vehicles’ operations. Therefore, for the entire system of $N$ single vehicles, we have $S^i \in \mathcal{X}$, $Y^i \in \mathcal{Y}$, $U^i \in \mathcal{U}$ and $\mathcal{X} = \mathcal{Y} = \mathcal{U} = \mathcal{B}^N$. The discrete-state dynamics of the system are then described by

$$S^i(k+1) = F(S^i(k), U^i(k)) \quad i = 1, 2, \ldots, N$$

where the function $F : \mathcal{B} \times \mathcal{B} \rightarrow \mathcal{B}$ is defined in Table 3.1 [82].
Table 3.1: Behavior Layer: Discrete-State Dynamics

<table>
<thead>
<tr>
<th>$U^i$</th>
<th>$U^i$</th>
<th>$F(S^i, U^i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>$A, D, C, S_t$ or $S_S$</td>
<td>$A$</td>
</tr>
<tr>
<td>$D$</td>
<td>$A, D, C$, or $S_t$</td>
<td>$D$</td>
</tr>
<tr>
<td>$D$</td>
<td>$S$</td>
<td>$S$</td>
</tr>
<tr>
<td>$C$</td>
<td>$A, D, C, S_t$ or $S_S$</td>
<td>$C$</td>
</tr>
<tr>
<td>$S_t$</td>
<td>$A, D, C, S_t$, or $S$</td>
<td>$S_t$</td>
</tr>
<tr>
<td>$S$</td>
<td>$A, D, C, S_t$, or $S$</td>
<td>$S$</td>
</tr>
</tbody>
</table>

3.2.3 Interface

As the commands from operation layer propagate down, an interface which can map the discrete states into a set of continuous differential equations should be provided. On the other hand, an interface which can convert information from the continuous state space of the plant to the discrete, symbolic domain of the system is also in need. The interface between the behavior layer and the continuous time system $S$ is implemented to distinguish regions of the plant state space, according to where the trajectories lead for a given control policy [83].

The interface to the continuous time system $S$, which describes logical operations in the behavior layer can directly affect $S$, is denoted as $\psi$ [5]. The input from the behavior layer to the lower level $S$ is defined as $u^i(t)$. Consider $\psi$ from vehicle’s point of view, it can be regarded as the regulation layer, which sends the control commands from the higher level to throttle, brake and steering actuators.

The interface from the continuous time system $S$ to the behavior layer is described as $\phi$. $\phi$ is designed to convert the continuous time output of the vehicle to an asynchronous, symbolic input for the behavior layer. $\phi$ requires a trigger mechanism.
which determines when the input should be generated, and a process to determine what the input should be generated [5, 83]. In single vehicle case, the continuous state information from the lower level, such as position, velocity, and acceleration will be propagated up. With \( \phi \), the discrete state as defined in the behavior layer and belongs to \( B \) will be generated to be an input from the lower level to the higher level.

### 3.2.4 Vehicle Dynamics

Vehicle dynamics are part of the model that represent the entire continuous-time portion of the hybrid system and can be described as:

\[
\dot{X}^i(t) = \begin{pmatrix}
\dot{x}^i(t) \\
\dot{v}^i(t) \\
\dot{a}^i(t)
\end{pmatrix} = \begin{pmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & 0
\end{pmatrix} \begin{pmatrix}
x^i(t) \\
v^i(t) \\
a^i(t)
\end{pmatrix} + \begin{pmatrix}
0 \\
0 \\
1
\end{pmatrix} u^i(t)
\]

where \( x^i \) denotes the position of vehicle \( i \), \( v^i \) denotes the velocity of vehicle \( i \), and \( a^i \) denotes the acceleration of vehicle \( i \). \( u^i \) denotes the input from controller in terms of force that is possibly generated by the torque applied to wheels.

### 3.2.5 Design of Controllers

Vehicle’s control objectives are reflected by the modes in the operation layer.

- **Free Mode:** Once in free mode, we have the following cost function of vehicle \( i \), with purpose of traffic efficiency:

\[
J^i = \int_0^t \left( w_p(\tau)d^2((x^i_x(\tau), x^i_y(\tau)), R^i) + w_v(\tau)(v^i(\tau) - v^i_d(\tau))^2 \right) d\tau
\]  

(3.1)

where function \( d(p, l) \) computes the minimum distance from point \( p \) to the line defined by function \( l \), \( R^i \) describes the desired route of vehicle \( i \). \( v^i_d(t) \)
is the desired velocity at time $t$ of vehicle $i$, which is decided by the physical boundaries and constraints of vehicle $i$ and the road condition along with driver’s characteristics if applicable. $w_p(t)$ and $w_v(t)$ are the weighting parameters, the values of which represent the importance of route and speed accuracy. The first term in Equation 3.1 penalizes the errors between vehicle’s real position and desired route and the second term penalizes the difference between the vehicle’s current speed and the desired one.

The following constraints also need to be taken into accounts for the controller design,

$$a^i \leq a^i(t) \leq \bar{a}^i$$

(3.2)

$a^i$ and $\bar{a}^i$ are limited by vehicle’s own physical conditions, if comfortability of the drivers is also included in the design objective, these constraints are important.

If we consider a very simplistic point-mass model for a car, the dynamics would be:

$$\ddot{x} = u^i(t)$$

where $u^i(t)$ is the commanded input from the driver. Thus, one choice of $u^i(t)$ would be picked as 7

$$u^i(t) = k_1 d((x^i_x(t), x^i_y(t)), R^i) + k_2 (\dot{x}^i(t) - \dot{v}^i_0(t)) + k_3$$

(3.3)

where $k_1, k_2, k_3$ can be picked subconsciously by a driver or after analysis by the designer.
**Restricted Mode:** When we study the collision avoidance problem, we consider the vehicle’s size. Let’s denote $L^i$ and $W^i$ as the length and width of the vehicle $i$ respectively. We define function $g^i$ as a vector function describing the shape and position of vehicle $i$ at time $t$. For example, if vehicle $i$’s shape is defined as a quadrangle, its size and position can be described as,

$$g^i(t) = \{(x, y) | (x, y) \in Quad((x^i_x(t) + \frac{L^i}{2} \sin(\alpha^i(t)) - \frac{W^i}{2} \cos(\alpha^i(t))),$$

$$x^i_y(t) + \frac{L^i}{2} \cos(\alpha^i(t)) + \frac{W^i}{2} \sin(\alpha^i(t))),$$

$$(x^i_x(t) + \frac{L^i}{2} \sin(\alpha^i(t)) + \frac{W^i}{2} \cos(\alpha^i(t))),$$

$$x^i_y(t) + \frac{L^i}{2} \cos(\alpha^i(t)) - \frac{W^i}{2} \sin(\alpha^i(t))),$$

$$(x^i_x(t) - \frac{L^i}{2} \sin(\alpha^i(t)) + \frac{W^i}{2} \cos(\alpha^i(t))),$$

$$x^i_y(t) - \frac{L^i}{2} \cos(\alpha^i(t)) + \frac{W^i}{2} \sin(\alpha^i(t))),$$

$$x^i_y(t) - \frac{L^i}{2} \cos(\alpha^i(t)) - \frac{W^i}{2} \sin(\alpha^i(t))))\}$$ (3.4)

where $Quad(p_1, p_2, p_3, p_4)$ is the quadrangle defined by the four endpoints $p_1, p_2, p_3, p_4$.

To study the safety problem, we define the mathematical expression for collision between vehicle $i$ and vehicle $j$ at time $t$ as if and only if there exists overlapping between $g^i(t)$ and $g^j(t)$ in prediction, which can be written as:

$$C(g^i(t), g^j(t)) > 0 \quad i \neq j$$ (3.5)
where, function \( C(f(\cdot), g(\cdot)) \) calculates the overlapping area between the area defined by vector function \( f \) and \( g \). The value of the \( C(f(\cdot), g(\cdot)) \) is the value of the percentage of \( f(\cdot) \) or \( g(\cdot) \) has been overlapped, in case of no overlapping, the value will simply be 0.

Similarly if we define function \( obs^j \) as a vector function describing the shape and position of obstacle \( j \) (Let \( M \) be the numbers of obstacles, \( j = 1, 2, \ldots, M \)), the collision between vehicle \( i \) and obstacle \( j \) at time \( t \) will take place if and only if in prediction

\[
C(g^i(t), obs^j(t)) > 0 \tag{3.6}
\]

To define a strict safe condition for vehicle \( i \) in traffic network, we have the following statements,

\[
\forall t > 0, \begin{cases} 
C(g^i(t), g^j(t)) = 0 \\
C(g^i(t), obs^k(t)) = 0, \forall i, j, k, i \neq j
\end{cases} \tag{3.7}
\]

With the goal that the vehicle wants to perform safe operation on the proposed route with desired speed, the cost function for collision avoidance can then be defined as:

\[
\begin{align*}
J^i = & \int_0^t (w_p(\tau) d_2((x'_2(\tau), x'_y(\tau)), R^i) + w_v(\tau)(v(\tau) - v_i^d(\tau))^2 \\
+ & \sum_{k=1}^M w_0^k(\tau) e^{C(g(\tau), obs^k)} sgn(C(g(\tau), obs^k))) d\tau
\end{align*} \tag{3.8}
\]

where function \( sgn(x) \) is picked as

\[
sgn(x) = \begin{cases} 
1 & \text{if } x > 0 \\
0 & \text{if } x = 0 \\
-1 & \text{if } x < 0
\end{cases}
\]
From the Equation 3.6, we can find if a collision takes place, function $sgn(x)$ will generate a positive value as $C(g^i(\tau), obs^k)$ will be larger than 0, and $e^{C(g^i(\tau), obs^k)}$ will be greater than 1, which penalize the collisions. In a collision free situation, the $sgn$ function will be zero and therefore eliminate the value. Different weighting parameters $w_k^i$ are assigned to different obstacles.

In Restricted mode, the design objective of the controller is to minimize equation 3.8. One proposed controller is a lane change controller which controls the vehicle to make a lane change maneuver when

$$\frac{(v^i(t))^2}{2a} > \min d((x^i_z(t), x^i_y(t)), obs^k),$$

and emergency braking controller will be applied when

$$\frac{(v^i(t))^2}{2a} \leq \min d((x^i_z(t), x^i_y(t)), obs^k).$$

It should be noted that, although braking maneuver can always avoid possible collisions, it might not be efficient in case of static road obstacle, because vehicle will get stuck in one point forever. To avoid this situation, the first and second term in cost function Equation 3.8 forces the vehicle to move. It is quite different in the collision between vehicles case as we will discuss later.

### 3.3 Multiple Vehicle Model

In this section, we focus on the control problems of multiple vehicles which are influenced by other vehicles in the traffic network. One example is during the rush hour, vehicle’s speed highly depends on the average speed of the traffic flow. The objectives of the multiple vehicle control are classified in both traffic system manner and vehicle manner. The former one focuses on increasing the efficiency of the whole
traffic network while maintains safety requirements for each vehicle, and the later one considers only itself’s efficiency and safety. Some of problems, which should be considered by the multiple vehicle control, are [79],

- Accident mitigation (obstacles avoidance, vehicle collision, etc.);
- Getting and maintaining safe speed (acceleration, cruising, deceleration, etc.);
- Getting and maintaining safe distance between vehicles (car-following, etc.);
- Vehicle trajectory tracking;
- Basic maneuvers support (lane change, etc.).

In this section, we discuss the hybrid system for the multiple vehicles. Figure 3.3 shows the hierarchical hybrid system for two vehicles. Compared with the single vehicle case, one additional layer, coordination layer is added.

3.3.1 Coordination Layer

A new layer, coordination layer, is introduced over the behavior layer into the system architecture, which can be viewed as control tasks that the vehicle needs to accomplish due to the influences from other vehicles. Coordination layer is able to send not only commands from traffic system’s point of view, which emphasize system efficiency, but also commands from its own point of view, focusing on both safety and efficiency according to its own characteristics (e.g. overtaking the vehicle in front to reach its desired speed). Note that it is not always the case to have a command from this layer. For instance, when we consider a human driver case, there is possibility that the driver does not obey the command which is generated according to the control
Figure 3.3: Hierarchical Hybrid System Model for Two Vehicles
objectives from system aspect. A highway driver who is approaching a merge ramp may not willing to sacrifice his/her own speed to accommodate for other merging vehicles. However, from traffic management’s (coordination layer) point of view, a slow down maneuver should be done by the highway driver as it helps to increase the efficiency of the whole traffic network. Therefore, whether a command will be sent to the operation layer also depends on driver’s own characteristics and the vehicle’s state.

3.3.2 Operation Layer

There are three modes in the operation layer for the multiple vehicles case.

- **Free mode:** *Free* mode remains the same as the single vehicle case.

- **Restricted mode:** In the multiple vehicles case, both road and traffic environment need to be considered. Besides obstacle avoidance, collisions with other vehicles should also be prevented.

- **Cooperative mode:** Once a command is received from the coordination layer, the command will fill in the cooperative mode, e.g., car following command is accepted. The operation layer will now have three modes as *free*, *restricted*, and *car-following*, so the *cooperative* mode is time-variant. Consider the transitions among three modes, the danger indicator $I_D$ should be first assigned according to the collision probability of both obstacle and collisions between vehicles.

The algorithm of computing collisions between vehicle $i$ and vehicle $j$ goes as the following:
1. $P = \{(x, y)|I(R^i, R^j)\}$, $I(f, g)$ computes the intersection point of line defined by function $f$ and $g$;

2. $\forall p \in P, TTC = \min abs\left(\frac{d_p((x^i(t), x^j(t)), p)}{v^i(t)} - \frac{d_p((x^j(t), x^j(t)), p)}{v^j(t)}\right)$, $d_p(p_1, p_2)$ computes the distance between $p_1$ and $p_2$;

3. if $TTC < \gamma$, let $I_D = 1$ which means possible collision between Vehicle $i$ and Vehicle $j$, otherwise $I_D = 0$, no collision;

The main idea of the above algorithm is first to find all the intersection points on the vehicle $i$’s and vehicle $j$’s routes. If there exists one, then the estimated time for each vehicle to reach that point is calculated. The difference between the estimated time of each vehicle has to exceed certain threshold $\gamma$ to avoid possible collisions.

![Figure 3.4: Mode Transitions in Operation Layer for Multiple Vehicles](image-url)
With the danger indicator, transitions among three modes are shown in Figure 3.4. If no command is received, the cooperative mode is idle, and mode transitions will only happen between free and restricted modes, which is the same as shown in Figure 3.2.

3.3.3 Design of Controllers

In multiple vehicles case, each vehicle is modeled as a hierarchical hybrid system, and the entire system containing multiple vehicles is modeled as a set of interacting hybrid systems. It is quite difficult to design controllers for such a large-scale interconnected system. Hence, we design the decentralized controller for each vehicle with overlapping decompositions within the framework as mentioned in [82]. The whole multiple vehicles hybrid system is described as follows [82, 14]:

\[
\begin{align*}
\mathcal{H} : & \quad S(k + 1) = F(S(k), z(kT), U(k)) \quad \text{(3.10)} \\
Y(k) &= X(k) \quad \text{(3.11)} \\
Z(k) &= X(k) \quad \text{(3.12)} \\
\dot{X}(t) &= g(X(t), Z([t/T]), u(t)) \quad \text{(3.13)} \\
y(t) &= g(X(t), Z([t/T]), u(t)) \quad \text{(3.14)} \\
z(t) &= h(X(t), Z([t/T]), u(t)) \quad \text{(3.15)}
\end{align*}
\]

where, \( S \in \mathcal{B} \), \( U \in \mathcal{B} \), and \( Y \in \mathcal{B} \) are state, input, and output vectors of the discrete part of the system \( \mathcal{H} \) respectively and \( x \in \mathbb{R}^n \), \( u \in \mathbb{R}^m \), and \( y \in \mathbb{R}^l \) are state, input, and output vectors of continuous-state part of the system \( \mathcal{H} \). \( Z \in \mathcal{B} \) is the internal
output vectors of the discrete-state part of the system $\mathcal{H}$. $z \in \mathbb{R}^\lambda$ is the internal output vectors of the continuous-state part of the system $\mathcal{H}$.

we define another hybrid system as [82]:

\[
\tilde{\mathcal{H}} : \quad \tilde{S}(k+1) = \tilde{F}(\tilde{S}(k), \tilde{z}(kT), \tilde{U}(k)) \tag{3.16}
\]

\[
\tilde{Y}(k) = \tilde{S}(k) \tag{3.17}
\]

\[
\tilde{Z}(k) = \tilde{S}(k) \tag{3.18}
\]

\[
\dot{\tilde{X}}(t) = g(\tilde{X}(t), \tilde{Z}([t/T]), \tilde{u}(t)) \tag{3.19}
\]

\[
\dot{\tilde{y}}(t) = \tilde{g}(X(t), \tilde{Z}([t/T]), \tilde{u}(t)) \tag{3.20}
\]

\[
\dot{\tilde{z}}(t) = \tilde{h}(X(t), \tilde{Z}([t/T]), \tilde{u}(t)) \tag{3.21}
\]

where, $\tilde{X} \in \mathcal{B}$, $\tilde{U} \in \mathcal{B}$, and $\tilde{Y} \in \mathcal{B}$ are state, input, and output vectors of the discrete part of the system $\tilde{\mathcal{H}}$ respectively and $\tilde{x} \in \mathbb{R}^\tilde{n}$, $\tilde{u} \in \mathbb{R}^\tilde{m}$, and $\tilde{y} \in \mathbb{R}^\tilde{l}$ are state, input, and output vectors of continuous-state part of the system $\tilde{\mathcal{H}}$. $\tilde{Z} \in \tilde{Z}$ is the internal output vectors of the discrete part of the system $\tilde{\mathcal{H}}$. $\tilde{z} \in \mathbb{R}^\lambda$ is the internal output vectors of the continuous-state part of the system $\tilde{\mathcal{H}}$.

We also define the injective transformations for discrete-state portion as [82]:

\[
P : S \rightarrow \tilde{S}, \quad Q : U \rightarrow \tilde{U}, \quad R : Y \rightarrow \tilde{Y}, \quad S : Z \rightarrow \tilde{Z} \tag{3.22}
\]

and consider surjective transformations for discrete-state portion as [82]:

\[
P^\# : \tilde{S} \rightarrow S, \quad Q^\# : \tilde{U} \rightarrow U, \quad R^\# : \tilde{Y} \rightarrow Y, \quad S^\# : \tilde{Z} \rightarrow Z \tag{3.23}
\]

Note that this hybrid system is in a hierarchical fashion, continuous control objective is based on the mode in operation layer and established for discrete stated
in behavior level. Hence, the discrete state is discussed before designing continuous controllers.

We first decompose the system of $N$ vehicles into $\frac{N(N-1)}{2}$ overlapping subsystems, with only two vehicles in each subsystem.

Figure 3.5 shows an example of decomposing three vehicles in the intersection area into three overlapping subsystems. The state, the input and the output vectors for the discrete-state part of the subsystem which has vehicle $i$ and vehicle $n$ are

$$\tilde{S}_{i,n} = \begin{pmatrix} S^i \\ S^n \end{pmatrix}, \quad \tilde{U}_{i,n} = \begin{pmatrix} U^i \\ U^n \end{pmatrix}, \quad \tilde{Y}_{i,n} = \begin{pmatrix} Y^i \\ Y^n \end{pmatrix} = \tilde{S}_{i,n}, \quad \tilde{Z}_{i,n} = \begin{pmatrix} Z^i \\ Z^n \end{pmatrix}$$

(3.24)

The transformations defined in 3.22 and 3.23 are chosen as
\[
P \left( \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_N \end{bmatrix} \right) = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_N \end{bmatrix}, \quad Q \left( \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_N \end{bmatrix} \right) = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_N \end{bmatrix}, \quad (3.25)
\]
\[ R \left( \begin{bmatrix} Y^1 \\ Y^2 \\ \vdots \\ Y^N \end{bmatrix} \right) = \begin{bmatrix} Y^1 \\ Y^2 \\ \vdots \\ Y^N \end{bmatrix} \], \quad \quad S \left( \begin{bmatrix} Z^1 \\ Z^2 \\ \vdots \\ Z^N \end{bmatrix} \right) = \begin{bmatrix} Z^1 \\ Z^2 \\ \vdots \\ Z^N \end{bmatrix} \] (3.26)
\[
P^\# = \begin{pmatrix}
\tilde{S}^{1,2} \\
\tilde{S}^{1,3} \\
\vdots \\
\tilde{S}^{1,N} \\
\vdots \\
\tilde{S}^{2,3} \\
\vdots \\
\tilde{S}^{2,N} \\
\vdots \\
\vdots \\
\tilde{S}^{N-2,N-1} \\
\tilde{S}^{N-2,N} \\
\vdots \\
\tilde{S}^{N-1,N}
\end{pmatrix}
\]

\[
= \begin{bmatrix}
\sum_{i=1}^{1} \tilde{S}_{i,2} + \sum_{j=3}^{N} \tilde{S}_{i,j} \\
\sum_{i=1}^{N-2} \tilde{S}_{i,N-1} + \sum_{j=N}^{N} \tilde{S}_{i,j} \\
\cdots \\
\sum_{i=1}^{N-1} \tilde{S}_{i,N} \\
\sum_{i=1}^{N} \tilde{S}_{i,N} \\
\end{bmatrix}, \tag{3.27}
\]

\[
Q^\# = \begin{pmatrix}
\tilde{U}^{1,2} \\
\tilde{U}^{1,3} \\
\vdots \\
\tilde{U}^{1,N} \\
\vdots \\
\tilde{U}^{2,3} \\
\vdots \\
\tilde{U}^{2,N} \\
\vdots \\
\vdots \\
\tilde{U}^{N-2,N-1} \\
\tilde{U}^{N-2,N} \\
\vdots \\
\tilde{U}^{N-1,N}
\end{pmatrix}
\]

\[
= \begin{bmatrix}
\sum_{i=1}^{1} \tilde{U}_{i,2} + \sum_{j=3}^{N} \tilde{U}_{i,j} \\
\sum_{i=1}^{N-2} \tilde{U}_{i,N-1} + \sum_{j=N}^{N} \tilde{U}_{i,j} \\
\cdots \\
\sum_{i=1}^{N-1} \tilde{U}_{i,N} \\
\sum_{i=1}^{N} \tilde{U}_{i,N} \\
\end{bmatrix}, \tag{3.28}
\]

48
\[
\begin{align*}
R^# &= \begin{bmatrix}
\tilde{Y}^{1,2} \\
\tilde{Y}^{1,3} \\
\vdots \\
\tilde{Y}^{1,N} \\
\vdots \\
\tilde{Y}^{2,3} \\
\vdots \\
\tilde{Y}^{2,N} \\
\vdots \\
\vdots \\
\tilde{Y}^{N-2,N-1} \\
\tilde{Y}^{N-2,N} \\
\vdots \\
\tilde{Y}^{N-1,N}
\end{bmatrix} \\
&= \begin{bmatrix}
\sum_{j=2}^{N} \tilde{Y}^{1,j} \\
\sum_{i=1}^{1} \tilde{Y}^{1,2} + \sum_{j=3}^{N} \tilde{Y}^{2,j} \\
\vdots \\
\sum_{i=1}^{N-2} \tilde{Y}^{i,N-1,j} + \sum_{j=N}^{N} \tilde{Y}^{N-1,j} \\
\sum_{i=1}^{N-1} \tilde{Y}^{i,N}
\end{bmatrix}, \quad (3.29)
\end{align*}
\]

\[
S^# = \begin{bmatrix}
\tilde{Z}^{1,2} \\
\tilde{Z}^{1,3} \\
\vdots \\
\tilde{Z}^{1,N} \\
\vdots \\
\tilde{Z}^{2,3} \\
\vdots \\
\tilde{Z}^{2,N} \\
\vdots \\
\vdots \\
\tilde{Z}^{N-2,N} \\
\tilde{Z}^{N-2,N} \\
\vdots \\
\tilde{Z}^{N-1,N}
\end{bmatrix} \\
= \begin{bmatrix}
\sum_{j=2}^{N} \tilde{Y}^{1,j} \\
\sum_{i=1}^{1} \tilde{Z}^{1,2} + \sum_{j=3}^{N} \tilde{Z}^{2,j} \\
\vdots \\
\sum_{i=1}^{N-2} \tilde{Z}^{i,N-1,j} + \sum_{j=N}^{N} \tilde{Z}^{N-1,j} \\
\sum_{i=1}^{N-1} \tilde{Z}^{i,N}
\end{bmatrix}, \quad (3.30)
\]

where, “+” denotes addition.

For each individual subsystem of two vehicles shown in Figure 3.3, there is a set of state, input and output vectors. In the subsystem which has vehicle \( i \) and vehicle
n, the controller for vehicle $i$ will be designed according to the mode in the operation layer which reflects the influence of vehicle $n$ if applicable.

- **Free mode**: The cost function for free mode remains the same as for the single vehicle case.

- **Restricted Mode**: The safety related cost function can be defined as the following:

\[
J^i = \int_0^t \left( \sum_{k=1}^M w^k_i(\tau) \exp(C(g^i(\tau), obs^k)) \text{sgn}(C(g^i(\tau), obs^k)) \right) d\tau \tag{3.31}
+ \sum_{n=1, n \neq i}^M w^n_{vv}(\tau) \exp(C(g^n(\tau), g^n(\tau))) \text{sgn}(C(g^n(\tau), g^n(\tau))) d\tau \tag{3.32}
\]

where $w^n_{vv}$ is the weighting parameter assigned to vehicle $n$. A proposed controller that may work in such situation though certainly not optimal is a braking controller. The motivation for a braking controller comes from the minimax problem in the game theory. An example is shown in figure 3.5, vehicle $A$ is following vehicle $B$ at the intersection. When vehicle $A$ receives the information from vehicle $C$, it calculates the collision possibility. As $I_D = 1$, vehicle $A$ switches into restricted mode. Braking controller takes over, which means that vehicle $A$ sacrifices its traveling time for vehicle $B$’s non-stopping movement. This controller is easy to design and apply, but it is not optimal. Furthermore, if vehicle $B$ applies the same controller, vehicles may run into a deadlock situation.

In restricted mode, the penalties on position and velocity are not included in order to avoid every possible collisions, moreover, the road can be cleared, if it is a problem of vehicle collisions.
Cooperative Mode: As mentioned before, the cooperative mode will be specified by the input from coordination layer. One possible case is the car-following case, which the cost function is defined as following:

\[
J_i = \int_0^t (w_p(\tau)d^2(x^i_x(\tau), x^i_y(\tau)), R^i) + w_v(t)(v^i(\tau) - v^n(\tau))^2) \\
+ w^i_n(\tau)(d_p((x^n_x(\tau), x^n_y(\tau)), (x^i_x(\tau), x^i_y(\tau))) - L)^2) d\tau \quad (3.33)
\]

where \(w^i_n\) is the weighting parameter assigned to vehicle \(n\) for safe distance, we assume vehicle \(n\) is the leading vehicle, additional constraint:

\[
d_p((x^n_x(\tau), x^n_y(\tau)), (x^i_x(\tau), x^i_y(\tau))) - L \geq 0
\]

should be satisfied.

The third term in the above formula penalizes the failure of not keeping small enough \((L)\) inter-car gap with respect to the leader.

Adaptive cruise controller is a special case in the cooperative mode, which aims at creating platoons on the highways.

For each vehicle in a \(N\) vehicle system, we need to compute out the controller for discrete state in \(N - 1\) subsystem, then we can contract back the discrete controller for this vehicle, we do the addition on \(B\), which is defined as follows

\[
A + A = A \quad A + D = D \quad A + C = C \quad A + S_t = S_t \quad A + S = S \\
D + D = D \quad D + C = D \quad D + S_t = S_t \quad D + S = S \\
C + C = C \quad C + S_t = S_t \quad C + S = S \\
S_t + S_t = S_t \quad S_t + S = S \quad S + S = S
\]  

(3.34)
Decentralized discrete controllers are designed using Equation 3.27, after the \( N - 1 \) subsystem controllers are available. Continuous controllers are then designed according to the discrete control.

Figure 3.5: Overlapping Decomposed Hybrid System of Three Vehicles
3.4 Conclusions

In this chapter, a hierarchical hybrid system model for vehicles in traffic network is presented. Single vehicle operation and multiple vehicles cooperation are modeled. The system is designed for safety and efficiency specifications as a primary capability in addition to the usual requirements such as real-time constraints, and so on. Reachability and Stabilizability need to be studied in the future.
CHAPTER 4

DEVELOPMENT OF TRAFFIC SIMULATORS

4.1 Introduction

In Chapter 3, we have developed hierarchical hybrid system model for vehicles in traffic networks. This will facilitate analysis and synthesis of control tasks for the vehicles. We now need to study the control tasks executed for the vehicles applying the proposed system model. In many situations due to the high costs and safety concern, field tests and evaluations may not be applicable. Therefore, we evaluate and analyze the vehicles and traffic network via computer simulations/animations. In this chapter, we will discuss the development of our microscopic traffic simulators\(^1\) which are designed for aiding vehicles and traffic control task studies, such as testing, verification, evaluation leading to improvement of different vehicle control methods and traffic management strategies.

Microscopic traffic simulators are simulation tools that are able to reproduce traffic conditions to a significant level of accuracy. The principal advantage of microscopic

\(^1\)Traffic simulators are predominantly divided into two classes: microscopic or macroscopic. Microscopic modeling accounts for individual vehicle in the simulation, taking into account not only vehicle characteristics, but also driver behavior and traffic conditions. Macroscopic simulation emphasizes mainly on overall flow density instead of tracking individual vehicle behavior [84, 85].
simulators is that the environment (e.g. infrastructure) is incorporated directly and the vehicle is modeled on an individual level. A number of microscopic traffic simulators have been developed in recent years. SmartPath was developed by PATH researchers in early 1990s to simulate an Automated Highway System (AHS). It is used to understand how the AHS would perform in terms of traffic flow, highway capacity and other user-defined performance measures [86, 87]. However, because SmartPath focuses on the intelligent vehicles on highways, it is unable to simulate all kinds of traffic phenomena in a complete traffic network of highway and urban area [88]. MITSIM (Microscopic Traffic Simulator), designed for modeling traffic networks with advanced traffic control, route guidance and surveillance systems, is provided by researchers from Massachusetts Institute of Technology. Although MIT-SIM models individual vehicle movements using car following, lane changing and traffic signal responding logic with detailed traffic networks, it is not based on vehicle dynamics [89, 88]. Federal Highway Administration (FHWA) designed a general traffic simulation framework called CORSIM based on two microscopic traffic simulators for freeways (FRESIM) and urban networks (NETSIM) [85]. VATSIM, a vehicle and traffic simulator developed in the Ohio State University, is designed for modeling automated vehicles with different sensors and controllers and traffic networks with real-time traffic control and route guidance [90, 88]. Because VATSIM emphasizes the intelligent vehicles which do not have a human driver model, it has not been able to evaluate driver assistance systems.

In our study, we have been investigating different aspects of wireless communication among vehicles, especially implications regarding safety and traffic efficiency based on the hierarchical hybrid system model proposed in Chapter 3. An initial
aspect of this investigation was based on intersection collision avoidance systems. We have considered specific hardware for evaluating such an application in which an extensive simulator, Intersection Warning Simulator (IWS) was developed to cover the physical layer model, MAC layer protocol model and transportation aspects of the situation at an intersection. The Transportation Module is important since it provides information on “which vehicle is where” when each one attempts to communicate with another. As shown in Figure 4.1, the complete architecture of the simulator includes

- Physical Layer Simulator,
- MAC Layer Simulator,
- Vehicle Traffic Simulator,
- Driver/Automated Vehicle Simulator.

Later, in an effort to impart more flexibility to the intersection collision analysis, we extracted the Vehicle Traffic Simulator and the Driver/Automated Vehicle Simulator to a Vehicle Intersection Traffic Simulator (VITS), which enables the vehicle to interact with each other via different sensors besides wireless communication. Single Street Traffic Simulator (SSTS) is developed to simulate the stop-and-go scenarios in a single street that connects multiple intersections. It can be used along with IWS and VITS to provide a full picture of the urban area. Moreover, it is used to evaluate the wireless communication based driver assistance system in the stop-and-go scenario. The Physical Layer Simulator and the MAC Layer Simulator that developed in previous research where they were used specifically for intersection safety studies are now be substituted into a fixed delay and statistical probability, since we shall
Figure 4.1: Unified Microscopic Traffic Simulator Architecture
only evaluate the effect of the loop closure delay on safety without considering details of how the delay is produced. IWS, VITS and SSTS focus on traffic safety problems in the urban area, where the control for a vehicle is designed according to the cost function in safety mode in the operation layer based on the propose hybrid system model. As mentioned in Chapter 3, controller of a vehicle in a two vehicle subsystem are first designed using overlapping decompositions, then we contract back the discrete controller for this vehicle using Equation 3.27 and Equation 3.34.

Merging Vehicle Traffic Simulator (MVTS) is developed to facilitate the need of testing traffic management strategies of efficient merging on a highway ramp. Wireless communication serves as a means of updating vehicle’s state information on the highway, thus there will be no need to include the Physical Layer or MAC Layer, and the portion indicated on Figure 4.1 will simply be replaced by a fixed delay. Cooperation layer in the hybrid system model is activated for all the vehicles on the highway in order to test various cooperative merging algorithms from traffic management aspect.

4.2 Simulation Framework

In this study, we developed four simulators, IWS, VITS, SSTS and MVTS as mentioned in Section 4.1. In spite of their specific functionalities, both simulators have the same simulation framework for the transportation module as shown in Figure 4.1, which includes three parts: the simulator, vehicle characteristics input, and scenario input as shown in Figure 4.2. All the simulators are implemented as modular simulation systems with an object oriented flavor, which consist of three elements: road network, vehicle management and traffic management.
4.2.1 The Road Network

The Road Network Module as shown in Figure 4.2 includes the part of the program related to the physical information about the road environment, such as signalized/unsignalized intersection, number of lanes, and speed limit. Signalized intersections in the simulators are controlled under two-phase traffic lights with user defined cycling time and split rate. Unsignalized intersections on the other hand are controlled by stop signs. Both one-way stop sign and three/four way stop sign are included. The simulated roads are created in the client area using user defined geometrical data with line segments and arcs. Note that the simulators represents the traffic network at the lane level.

4.2.2 The Traffic Management

The traffic management module as shown in Figure 4.2 includes two functions: vehicle generation and accident determination.

- **Vehicle Generation Function**: In the simulators, a vehicle is generated with the following properties: time of departure, types, driver’s characteristics, origin and destination. The inter-departure time between two vehicles is randomly drawn from a distribution which is determined by traffic flow rate. For example, in low traffic flow case, vehicles are generated according to a Poisson process with the mean equal to the traffic flow rate (vehicles/hour).

- **Accident Determination Function**: All the vehicles in the simulators are modeled as a rectangle with the mass point in the center. Accident between two vehicles will be turned into the problem of checking whether there is overlapping between two rectangles using Equation 3.5. If yes, the speeds of the two vehicles
will be set to zero and the positions remain the same. If the user wants to continue the simulation, two vehicles will then disappear by assuming that the collided vehicles will be cleared in the real world [88, 90]. Otherwise, the simulation will stop with accident message presented.

![Figure 4.2: Vehicle Traffic Simulator Framework](image)

**4.2.3 The Vehicle Management-Hybrid System Model**

The Vehicle Management Module is used to determine the maneuvering behavior of all vehicles, taking interactions between vehicles into account. Such maneuvers are dependent on vehicle’s dynamics, environment condition, and human driver’s characteristics if applicable. Hierarchical hybrid system model is used to model the vehicle management.
A. Vehicle Dynamics- Continuous Time System Model

The model of vehicle dynamics, corresponding to the continuous time system model in the hybrid system model as proposed in the Chapter 3, is a highly idealized model of all vehicle behavior under successful velocity control and normal road driving conditions. Two decoupled systems are considered: the longitudinal control system and the lateral control system, as shown in Figure 4.3. The longitudinal model of the vehicle is taken as a unity gain first order linear system with a time constant of 10 msec coupled with acceleration limits of $0.2g \ m^2/sec$ and deceleration limits (braking limit) of $0.31g \ m^2/sec$ [88, 90]. The acceleration limits are hard physical limits. The deceleration limits are the same as braking limits, which are determined by coefficient of skidding friction and physical constraint [91]. A general equation for the braking distance can therefore be written as:

$$D_b = \frac{v^2}{30(f \pm G)} \quad (4.1)$$

where $v$ is the velocity, $f$ is the coefficient of friction and $G$ is the grade of the incline. For zero grade, the deceleration limits will be:

$$a_{decel} = g \times f = 0.31g \quad (4.2)$$

For the lateral model, it is assumed that vehicle yaw rate is linearly proportional to the steering angle, where the gain is a function of the mass and dimensions of the vehicle, the approximate velocity, and the effective traction of the tires. The lateral dynamics of the vehicle are turned into a single integration computing yaw angle (orientation angle in ground coordinates) form the yaw rate [88, 90].
Figure 4.3: Continuous Time System Model-Vehicle Dynamics
B. Vehicle States - Discrete Event System Model

A set of maneuver state models, corresponding to different modes in the DES in the hybrid system model are defined to simulate vehicles activities. Car following state, lane changing state, and merging state are in the coordination layer while turning state and free driving state are the substates of free mode in the operation layer. Based on different control tasks and management strategies, the control functions may change for the same maneuver.

i. Car Following State

The car following state calculates the acceleration and braking of a vehicle with respect to the leading vehicle while maintaining specified spacing. For example, when the leading vehicle is a slower vehicle, the following vehicle will decelerate to reach a specified following distance (primarily safety distance) and match with the speed of the leading vehicle. When the leading vehicle is a faster one, the following vehicle will accelerate to achieve the same speed within the acceleration limits. In this model, control action is executed by longitudinal control which determines the acceleration and braking, and lateral control which keeps the vehicle in the same lane as the leader. Various control policies can be applied to the car following model. One example is using adaptive cruise control for the longitudinal control as described in Equation 3.33.

ii. Lane Changing State

The lane changing state is used to steer a vehicle to its adjacent lane. The lane changing model will be applied to the vehicle only when some safety requirements are fulfilled, e.g. the gap distance on the target lane is acceptable. The gap distance acceptance depends on the speed of the vehicle, the average speed on the target
lane, and the driver’s characteristics. Lane changing maneuver is accomplished by longitudinal control which maintains a longitudinal speed and lateral control which leads the vehicle traveling a specified distance (a full lane width) along the lateral axis with respect to its body orientation within a finite time period and aligns itself with the adjacent lane at the end of the maneuver [37]. The control problem relies in how to generate the appropriate steering signal to cause the vehicle to accomplish the above-described task.

iii. Merging State

Highway merging state is responsible for applying different merging strategies to the vehicle near the ramp. Beyond ramp metering control for the merging vehicle, it may also control vehicles on the highway according to traffic system control tasks which are defined by users.

iv. Turning State

The turning state is applied to the vehicles in the intersection area who need to make left/right turns. The turning model in the simulators is implemented in three steps:

1. Define the type of the turning (left/right),

2. Select the desired lane,

3. Execute the turning when safety constraints are satisfied.

The type of the turning can either be user defined or system probabilistically decided. The desired lane is selected after the type of the turning is decided based on several criteria including prevailing traffic condition, lane use privilege, driver’s desired speed, and so on. Safety requirements need to be fulfilled before the execution of the
turning to achieve safe vehicle operation. Due to the lack of available information of the environment (e.g. poor visibility prevents the recognition of other vehicles) for the turning vehicle, sometimes part of safety requirements are ignored.

In all the simulators, turning trajectory for the vehicle is an arc to reflect the reality with the following longitudinal and lateral vehicle model:

\[
\begin{align*}
\dot{x} &= v \sin \theta \\
\dot{y} &= v \cos \theta
\end{align*}
\]  

(4.3)

where \( v \) is the current speed, \( \theta \) is the orientation, and \( x \) and \( y \) are the current vehicle position in the absolute ground reference frame in the simulators.

v. Free Driving State

Free driving state in the simulators is implemented with two functions: lane keeping and achieving vehicle’s desired velocity within the speed limit range. Lane keeping is accomplished by the lateral controller. Different control laws are available for reaching the desired velocity. In the simulators, mostly we use the following simple control algorithm [88, 90], which can be replaced according to the users’ needs.

\[
v = \begin{cases} 
0.8 & (v_d - v) > 8.0 \\
-1.6 & (v_d - v) < -16.0 \\
0.1(v_d - v) & \text{else} 
\end{cases}
\]  

(4.4)

where \( v_d \) is the desired velocity within the speed limit range. In Equation 4.4, we constraint the acceleration and the deceleration of the vehicle in the range of \([-1.6, 0.8]\), which is a subset of the physical acceleration limits as mentioned in Section 4.2.3. The acceleration and the deceleration, viewed as normal acceleration and deceleration, are chosen in the concern of passenger’s comfort.

Switchings among the models are controlled by user defined control tasks and traffic management strategies, which is usually event-triggered (event-based).
C. Sensors

All the simulators are able to simulate both autonomous vehicles and human driver vehicles. Though sensors of these two kinds of vehicles are quite different, the functionalities of the sensors are almost the same with distinctions in capabilities. For example, LIDAR and eyes are visual sensors for autonomous vehicles and human driver vehicles respectively, who have the same responsibility detecting other vehicles or obstacles in the range, however the range is different for LIDAR and eyes. Therefore, we assumed all the vehicles in the simulators are equipped with the following sensors with adjustable parameters.

- DGPS which offers the exact real-time position data,
- Forward-looking ranging device that can detect the distance to the closest vehicle ahead and estimate its speed [88, 90],
- Forward-looking traffic light sensor, which can detect the color of traffic light, stop signs at the intersection (traffic light state) [88, 90],
- Side-looking sensor that can detect the distance to the closest vehicles in the adjacent lane and estimate their speed [88, 90],
- Navigation System which offers digital map (not available for some types of vehicles).

All the sensors are represented in the capabilities instead of simulating their operations in the simulators.
D. The Human Driver Model

It is desired to integrate a comprehensive, quantitative human driver model with automation into microscopic simulator for traffic simulation and performance evaluation at traffic level, because, humans not only bring a number of truly admirably skills for driving targets that most artificial intelligent technology can not accurately simulate, they also help to improve driving safety and driving learning. Consequently, human driver behaviors need to be carefully studied and implemented [51, 52]. In the simulators, human driver model is implemented using two modules, which are Driver Reaction Module and Driver Decision Making Module, for three kinds of personalities, namely conservative, normal, and aggressive.

Desired speeds are chosen according to both the simulated personalities and speed limit, one possible choice is:

\[
V = \begin{cases} 
\text{SpeedLimit} - 5 \text{ m/s} & \text{Conservative} \\
\text{SpeedLimit} \text{ m/s} & \text{Normal} \\
\text{SpeedLimit} + 5 \text{ m/s} & \text{Aggressive}
\end{cases}
\]

(4.5)

The Driver Reaction Module determines both driver reaction time and driver’s reaction. The driver reaction time is referred as the time taken for the driver to respond to the outside changes. The Driver Reaction Module also decides the value of the reaction time based on the triggering event and the response. For an autonomous vehicle, the driver reaction time is actually the device response time. The driver’s reaction is simulated as a direct response to the outside changes, the event triggered by the outside changes. For instance, a collision warning message is generated to vehicle \(i\), the Driver Reaction Module simulates that it takes driver of vehicle \(i\) 2.54 second to apply an emergency braking. The Driver Decision Making Module is a key
module in the simulations of human driver vehicles. The decision is made based on the perceptive information, its itinerary from task planning module, the driver’s own characteristics along with vehicle’s current state. One example is the Driver Decision Making Module decides when to make a left turn based on current vehicle state and environment.

### 4.2.4 The Scenario Input

The Scenario Input Module not only provides the Road Network Module with the geometric data of the simulated area (e.g. road layout, lane numbers), it also served as data source for the traffic management for vehicle generation (e.g. traffic flow, vehicle’s origin and destination).

### 4.2.5 The Vehicle Characteristics Input

The Vehicle Characteristic Input Module includes the user-specified information for the simulated vehicles. Vehicle types, driver’s characteristics and the parameters of equipped sensors are defined.

Four types of vehicles are implemented in the simulators as listed in Table 4.1. Vehicle size is assigned according to its type, which is an important parameter for accident determination. Although the physical constraints for acceleration and deceleration are the same for all the vehicle, the choice for normal acceleration and deceleration is different which is defined by its type.

Compared with other types of vehicles, motorcycles are small-size, low-height, lightweight and performance oriented motorized vehicles. [92] Besides, motorcycles also possess an additional distinction, which may cause changes in the traffic network
<table>
<thead>
<tr>
<th>Type</th>
<th>Width(m)</th>
<th>Length(m)</th>
<th>Acceleration</th>
<th>Deceleration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passenger Car</td>
<td>2.13</td>
<td>5.79</td>
<td>0.2g</td>
<td>0.31g</td>
</tr>
<tr>
<td>Single-unit Truck</td>
<td>2.59</td>
<td>9.14</td>
<td>0.2g</td>
<td>0.31g</td>
</tr>
<tr>
<td>Single-unit Bus</td>
<td>2.59</td>
<td>12.19</td>
<td>0.2g</td>
<td>0.31g</td>
</tr>
<tr>
<td>Motorcycle</td>
<td>0.76</td>
<td>2.13</td>
<td>0.2g</td>
<td>0.31g</td>
</tr>
</tbody>
</table>

Table 4.1: Vehicle Types

behavior. Throughout the observations and exposure of the data, some differences are summarized as follows [93]:

1. Because of its small size, a motorcycle can be easily hidden by objects, such as other vehicles, bushes, fences, bridges, etc.,

2. The view of the motorcycle is limited by glare or obstructed by other vehicles,

3. Intersections are the most likely place for the motorcycle accident, with the other vehicle violating the motorcycle right-of-way, and often violating traffic controls,

4. The motorcycle’s signal may not be real. Turning signals on a motorcycle usually are not self-canceling, thus some riders, (especially beginners) sometimes forget to turn them off after a turn or lane change,

5. Motorcycles are generally cheaper than other vehicles, therefore, most of the motorcycles do not install navigation systems, though we assume they do have DGPS systems,

6. More stopping distance should be provided for motorcycles. Generally speaking, stopping distance for motorcycles is nearly the same as the one of cars, but
slippery pavement makes quick stopping difficult. Also carrying a passenger complicates a motorcyclist’s task. Balance is more difficult.

Note that a vehicle’s type not only determines its physical dimensions and properties (i.e. maximum acceleration/deceleration rates, turning radius, etc.), but also dictates its behavior in traffic which has significant impact for traffic studies.

Driver’s characteristics are used to distinguish not only different kinds of drivers, but also autonomous vehicles and human driver vehicles.

User defined sensor ranges and angles, and penetration rate are essential for analyzing, evaluating, and testing various sensors.

All the simulators mainly use a time-based simulation approach in modeling the movements of the vehicles, that is vehicle’s states are updated at specified frequencies (time-based) or when certain events (e.g. collisions) occur (event-based). The simulators are coded in Visual C++ using object-oriented design. Graphical user interfaces (GUI) are implemented for animated 2-D output of the simulation runs.

4.3 Intersection Warning System Simulator

Intersection Warning System (IWS) simulator (GUI is shown in Figure 4.4) which is capable of evaluating different warning systems and communication protocols was developed based on Vehicle Traffic Simulator (VTS) and a Wireless Simulator (WS). The IWS simulator is developed by OSU in the second phase of a project with OKI Electric Industry Co. In the first phase of this project, WS was developed to evaluate the performance of communication protocols in an intersection environment [40, 94, 41]. In the second phase, ITS components (vehicle intersection traffic simulator, collision warning system, and driver behavior model) and wireless communication
components are merged together. In addition, physical layer model of WS is improved to simulate intersections with or without buildings and model is tuned to fit the field test results collected in Columbus, OH. Finally, an optional repeater is included at the intersection in order to disseminate the messages to the shadowed portions of the road segments. The performance of the proposed IWS in an urban intersection environment was studied under a variety of traffic flow and other relevant conditions defined by users via the presetting dialog 4.5. Human factors such as driver behavior and response time are also modeled and analyzed for the performance evaluation. VTS is presented with a real-time online wireless medium simulator (WS). The WS enables a feedback path to a collision warning system simulator that uses packet delivery status to alert and affect driver behavior.

Figure 4.4: Intersection Warning System Simulator Graphic User Interface
Figure 4.5: Intersection Warning System Simulator Presetting
4.4 Vehicle Intersection Traffic Simulator

The Vehicle Intersection Traffic Simulator (VITS) is extracted from IWS, which provides a real-time environment where various types of sensors, control policies, and traffic management strategies can be analyzed, tested and evaluated under different traffic conditions. The simulator is able to interact with the sensors and controls on a real-time basis. Further, in addition to realistic representation of traffic behavior at an intersection, it is also able to model different types of human drivers. The simulator is based on a time-step simulation scheme, which updates the location and speed of each vehicle periodically. The input user data are defined via presetting dialog as shown in Figure 4.6. Besides standard traffic data which includes scenario input and vehicle characteristics input, users are able to input specified intersection control data, such as different turn ratios. Moreover, sensors range is also defined by users to simulate different types of sensors. As presented in Figure 4.7, the movement of each individually characterized vehicle is displayed on the monitor screen as simulation progresses. Human driver model is implemented and each vehicle makes its own decisions regarding to turning depending on its current location, destination and surrounding traffic conditions. VITS can also be used to replace the VTS and WS in the IWS to test sensor based collision warning algorithms.

4.5 Single Street Traffic Simulator

Single Street Traffic Simulator (SSTS) as shown in Figure 4.8 simulates traffic activities on a road segment (e.g. single street) that connects multiple intersections. The area of interest is actually the portion observed on the GUI. Traffic is generated to provide input to the road segments in the GUI screen, and disappears as it leaves
Figure 4.6: Vehicle Intersection Traffic Simulator Presetting

Figure 4.7: Vehicle Intersection Traffic Simulator Graphic User Interface
view. The simulator defines the total distance along the horizontal (x) and vertical (y) axes. Inbound vehicles are not considered within the plotted area until they cross this virtual border. In addition, information of the vehicle’s position is available only when the vehicle is within this area. Therefore, inbound vehicle information is generated once it crosses this virtual border and outbound vehicle information ceases to be generated once it passes this limit. The simulator is designed in such a way that user is able to make stop and go to the vehicles by intervening through a mouse’s click.

Figure 4.8: Single Street Traffic Simulator Graphic User Interface
SSTS is used to evaluate wireless communication based stop and go driver assistance systems in the urban area. Combining with IWS, VITS, a full picture of urban area traffic network is presented, where safety issue is the main focus.

4.6 Merging Vehicle Traffic Simulator

Merging Vehicle Traffic Simulator simulates traffic activities on a three-lane highway as shown in Figure 4.9. The simulator defines the total distance along the horizontal (x) and vertical (y) axes. Similarly as SSTS, inbound vehicles are not considered within the plotted area until they cross this virtual border. In addition, information of the vehicle’s position is available only when the vehicle is within this area. Therefore, inbound vehicle information is generated once it crosses this virtual border and outbound vehicle information ceases to be generated once it passes this limit. Merging Vehicle Traffic Simulator is used to evaluate inter-vehicle communication based merging strategies in highway which are designed to achieve traffic efficiency.

4.7 Conclusions

Traffic simulators are key components in the traffic studies. In our study, four different simulators covering urban area and highway network are developed. They are responsible for providing realistic, microscopic level traffic flow information for vehicle simulations in the whole traffic network. In doing so, they account for such elements as road layout, vehicle throughput, vehicle types, and route selection. The control policies and traffic management strategies, for the most part, are parametrically user defined, so users can evaluate, analyze and test the performance variations
Figure 4.9: Merging Vehicle Traffic Simulator Graphic User Interface
by changing parameter specifications. The animated graphics display in the simula-
tors allows the user to study the overall traffic performance in the simulated area or
to examine the behavior of any selected vehicle or vehicles in great detail.
CHAPTER 5

CASE STUDY AND SIMULATION RESULTS - HUMAN DRIVER MODEL AT INTERSECTIONS

5.1 Introduction

Some specific applications of the general hierarchical hybrid system to a traffic network will be presented in detail in this and the following chapters. The applications will cover both urban and highway environment. Urban environment includes intersections (both signalized and unsignalized intersection) and street that connects multiple intersections, where we will study scenarios with and without wireless communication based driver assistance system. We will focus on the highway merging/lane changing scenario in highway environment. Chapter 5 will discuss the human driver decision making model at the intersection which served as the basis for further intersection collision avoidance discussion. Collision warning system using wireless communication in both unsignalized and signalized intersection scenarios will be studied in Chapter 6. In each scenario, we will test the wireless communication based collision warning algorithms with various initial conditions (e.g. different initial positions, velocity, driver characteristics, etc.) using IWS. Chapter 7 will discuss the stop-and-go driver assistance system in urban street scenario and highway merging
driver assistance system in highway merging/lane scenario. Both driver assistance systems are wireless communication based. SSTS will be used to evaluate the performance of the stop-and-go driver assistance system in a semi-autonomous vehicle. In this scenario, we assume that slowing down or emergency braking to avoid collision is controlled by the automated driving system in a predefined sensing range; and starting or acceleration is executed by the human driver. Human driven vehicle’s performance will be presented as a comparison. MVTS will be used to test the highway merging driver assistance system using the proposed merging algorithm in highway merging scenario with respect to the average delay time criteria. All the discussed scenarios are summarized in Table 5.1.

<table>
<thead>
<tr>
<th>Environment</th>
<th>Urban</th>
<th>Highway</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inter-</td>
<td>Signalized</td>
<td>With communication</td>
</tr>
<tr>
<td>section</td>
<td></td>
<td>Without communication</td>
</tr>
<tr>
<td>Urban</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Signalized</td>
<td>With communication</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Without communication</td>
<td></td>
</tr>
<tr>
<td>Unsignalized</td>
<td>With communication</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Without communication</td>
<td></td>
</tr>
<tr>
<td>Street</td>
<td>Stop-and-go</td>
<td>With communication</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.1: Summary of Simulated Scenarios

Since traffic safety is the most critical issue, we will first study the intersection accident mitigation problem. As mentioned in Chapter 3, vehicle control systems
usually have several communicating subsystems which are able to interact between themselves and the road environment with the aid of sensors or wireless communication. A typical scenario is that a Subject Vehicle (SV, the yellow vehicle in Figure 5.1) attempts to cross the path of an oncoming vehicle (Principal Other Vehicle, POV, the red vehicle in Figure 5.1) in the intersection area. In order to prevent and mitigate possible collision, the SV needs to collect information of the POV’s states to predicate the POV’s future movement and then makes its own decision.

![Sensor Coverage for Subject Vehicle](image)

Figure 5.1: Sensor Coverage for Subject Vehicle
Figure 5.2: System Configuration for Intersection Access Analysis
Figure 5.2 shows the system configuration for the intersection access analysis. Human driver decision making is applied in the configuration for both the predication of the POVs future movement and the reaction of the SV. Consequently, human driver behavior need to be carefully studied. In the following, we construct a human driver model and examine the driver behavior at both signalized and unsignalized intersection.

5.2 Human Driver Model at Intersections

In our studies, we first present a general architecture of the human driver model as shown in Figure 5.3. This model is based on the structure of COSMODRIVE [1, 52, 60, 61], a well-known framework developed for the application of a driver cognitive model, as mentioned in Section 2.4.1. The model consists of seven modules, which can be divided into two groups from external and internal view of the driver. Environment module is the external group, while all other modules are in the internal group.

- **Environment Module**: The environment module represents the outside world of the vehicle. It takes

  - traffic environment (traffic flow, traffic composition, etc.)
  - road environment (speed limit, lane numbers, etc.)
  - weather/visibility (sunny/rainy/windy, good/poor, etc.)

into accounts. Additional sensors, such as camera/LIDAR may also provide the driver with necessary visual data of the environment. Once a driver-assistance system is available, the system assistant message will also be served as a complement to the environment (e.g. beep collision warnings).
Figure 5.3: General Architecture of Human Driver Model
• **Perception Module:** The perception module represents the visual sense. If audio is available, it will also take it into consideration. The data generated by the perception module includes estimation of velocity, direction, distance between vehicles in range and angle. In a microscopic traffic simulator, when we are considering the changes of visibility for the drivers, we can simply adjust the parameters of “Range” and “Angle” according to the situation in the perception module. In real world, “Range” and “Angle” are based on the driving environment, for example, crowded urban area and blizzard weather leads to short range and small angle.

• **Task Planning Module:** The task panning module has the same function as route planning. It provides the decision making module with information on which direction the vehicle is going, such as N→S, S→E.

• **Driver Characteristics Module:** The essential function of driver characteristics module is to predict human driver’s psychological activities based on his/her knowledge, which contains both driving knowledge based and traffic rule based information, and his/her driving skill, which indicates his/her ability of driving (novice/expert). The driver characteristics module changes as the subject driver changes.

• **Decision Making Module:** The decision making module is the most important part of the driver model. It acts as a higher level control for the vehicle. Tasking planning module provides strategy, while the decision making module develops tactics. The decision is made based on the perceptive information, its itinerary from task planning module, the driver’s own characteristics along
with vehicle’s current state. Details of decision making module will be discussed in Section 5.3 using finite state machine.

- **Implementation Module:** The implementation module is responsible for the two dimensional control of the vehicle based on the information it receives from decision making module. A microscopic vehicle simulator simulates the kinematics of the vehicle if this part is considered.

- **Emergency Management Module:** The emergency management module deal with unexpected/irregular emergency, such as other vehicle’s traffic violation, obstacle avoidance. Generally a hard brake decision will be made in case of emergency.

### 5.3 Driver Decision Making

In this section we will expand the driver decision making module and develop a series of finite state machines to represent the driver decisions.

#### 5.3.1 Intersection Area

The intersection area is redefined as shown in Fig. 5.4. The vehicle enters the intersection when $d$ is less than $L_1$, in other words, the intersection area is $2L_1 \times 2L_1$. $d$ is the distance from the mass point of the vehicle to the center $o$ of the intersection. $L_1$ is determined by the environment (traffic flow on the lane, speed limit, etc.). $L_1$ has to be long enough to enable the vehicle to change to the correct lane before it is $L_2$ away from the center. One choice of $L_1$ is presented as in Figure 5.3.

$L_2$ is the distance indicating from where the vehicle should decelerate to its turning speed and prepare for turning. $L_2$ is also determined by the environment (speed limit,
road condition, etc.) and the vehicle deceleration limit. \( L_2 \) needs to be large enough to ensure not only the desired turning speed but also a safe stop without entering the conflict area (the red dashed rectangle in case of emergency in Figure 5.4). A larger \( L_2 \) is not efficient as the slow-down procedure will affect all the following vehicles, while a smaller \( L_2 \) may not meet the safety requirement. One choice of \( L_2 \) is presented as in Fig. 5.3. An analytical expression can be developed as

\[
L_1 = F(f, v) = L_2 + \alpha(f) \cdot T_L \cdot v \tag{5.1}
\]

\[
L_2 = F(v, a, a_{\text{max}}) \tag{5.2}
\]

\[
= \frac{v^2 - (\beta v)^2}{2a} + \frac{2(\beta v)^2}{2a_{\text{max}}} + S + S_m \tag{5.3}
\]

where \( \alpha(f) \) is a coefficient function indicating the severity of the traffic flow (low, medium, heavy traffic) based on the traffic flow rate and the LOS (level of service) of the intersection. \( T_L \) is the time required for completing a lane change. \( v \) can be either the speed limit or current average velocity. \( \beta \) is the road condition coefficient for determining the desired turning speed, \( a \) is the regular deceleration rate, and \( a_{\text{max}} \) is the deceleration limit and \( S_m \) is the safety margin to with respect to the average waiting before turning. Note that \( L_1 \) is a constant for all the vehicles that entering the intersection within a short period if the environment does not change much, but \( L_2 \) may be varied from vehicle to vehicle.

5.3.2 Pre-conditions

No matter which direction the vehicle will turn to (left, right or keep straight), there must exist no other vehicle in the conflict area. Furthermore, the car-following condition should always be simultaneously satisfied, i.e. the minimum safety distance
Figure 5.4: Intersection Area
between the vehicle and the last vehicle on the target lane has to be retained. We define condition 1 (C1) to be:

C1: Conflict area is cleared and minimum safety distance is kept.

5.3.3 Priority

Priority is introduced to establish efficiency and solve deadlock problems that may occur at the intersection. Priority is assigned to vehicles based on their position, anticipated movement, time, etc.

- **Unsignalized case (Stop Sign):** Priority levels are designated (highest to lowest): Go Straight (without stop sign), Right Turn (without stop sign), Left Turn (without stop sign), Go Straight (with stop sign), Right Turn (with stop sign), Left Turn (with stop sign)

- **Signalized case:** Priority levels are designated (highest to lowest): Go Straight, Right Turn, Left Turn

If the vehicles are of the equal priority according to the above criterion, the earlier arrival has higher priority; If vehicles happen to arrive at the same time, then the rightmost vehicle deserves the highest priority; If four vehicles with the same priority from four directions arriving at the same time and one is another’s rightmost vehicle, some window mechanics have to be introduced, random amount of time is added to each vehicle and the one with smallest time has the highest priority.
5.3.4 Finite State Machines for Driver Decision Making at Intersections

Here we discuss driver decision making based on different scenarios, intersection (unsignalized/signalized intersection) and driving goals (left turning/go straight/right turning). At this stage, we do not consider the skill-based (expert/novice) driver characteristics but conservative, obeying traffic rule drivers. For simplicity, we define Go 1: Go Straight; Go 2: Make the Right Turn; Go 3: Make the Left Turn; Stop 1: Decelerate to its desired velocity and prepare for turning; Stop 2: Decelerate to stop; Stop 3: Emergency stop.

Unsignalized Intersection

- **Right Turn:** The process flow diagram of vehicle that performs a right turning at an unsignalized intersection is plotted in Figure 5.5. Considering a vehicle entering the intersection from point A as shown in Figure 5.4, it has to make sure that no coming vehicles from B enter its target lane during the right turning procedure. Therefore, we define condition 2 (C2) to be:

  C2: No go-straight-vehicle from B in $D_2$ meters. The calculation of $D_2$ highly depends on the speed limit, traffic flow rate, current velocity, the main idea of computing this kind of limits is that vehicles will not arrive in the target lane within a short period.

  We also assume that there are stop signs in at least one direction. Obstacle emergency service will be checked during the entire Traveling. Traffic sign violation will be also checked when the distance between the vehicle and the center of the intersection is $L_2$ less.
Figure 5.5: Finite State Machine of Right Turn at Unsignalized Intersection
• **Go Straight:** Figure 5.6 shows the process flow diagram of the vehicle that goes straight at the intersection. In this case, both left turning vehicles from B and right turning vehicles from D bring the potential danger to the traveling vehicle. Thus, we define condition 3 (C3) as:

C3: No right-turning vehicle will be on the target lane in $D_3$ meters.

C4: No left-turning vehicle will be on the target lane in $D_4$ meters.

The methods for calculation $D_3$ and $D_4$ are similarly as $D_2$.

• **Left Turn:** Figure 5.7 shows the process flow diagram of the vehicle that performs a left turning at the intersection. Left turning is more complicated, as not only the vehicles from B and D, but also the vehicles from C have to be considered. Thus, we define condition 5 (C5) as

C5: No go-straight vehicle will come from C in $D_5$ meters.

We also define condition 6 (C6) as

C6: No right-turning vehicle from C will be on the target lane in $D_6$ meters.

**Signalized Intersection**

• **Right Turn:** The process flow diagram of vehicle that performs a right turning at signalized intersection is shown in Figure 5.8.

• **Go Straight:** Figure 5.9 shows the process flow diagram of the vehicle that goes straight at the signalized intersection. Besides traffic violation and obstacle emergency, yellow phase makes the situation more complicated. During the yellow phase, a vehicle may neither be able to complete the passing the intersection nor make a safe stop (vehicle is in the dilemma zone [95]). In Figure 5.9,
Figure 5.6: Finite State Machine of Go Straight at Unsignalized Intersection
Figure 5.7: Finite State Machine of Left Turn at Unsignalized Intersection
Figure 5.8: Finite State Machine of Right Turn at Signalized Intersection
conservative driver is considered which the above case may never happen as the driver will try to stop at very beginning if C1 is not satisfied during the yellow phase. However in case that the vehicle is in the dilemma zone, go-straight vehicle will try to accelerate and pass the intersection, as it should maintain a high speed.

Figure 5.9: Finite State Machine of Go Straight at Signalized Intersection

- **Left Turn:** Figure 5.10 shows the process flow diagram of the vehicle that performs a left turning at the intersection. Considering a left-turning vehicle
that is in the dilemma zone, as its speed is already reduced, we assume that it will try to make an emergency stop.

Figure 5.10: Finite State Machine of Left Turn at Signalized Intersection
Table 5.2: Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed Limit</td>
<td>35 miles/hour</td>
</tr>
<tr>
<td>Traffic Flow</td>
<td>3600 vehicles/hour</td>
</tr>
<tr>
<td>Deceleration Limit</td>
<td>$-0.31g$</td>
</tr>
<tr>
<td>Acceleration Limit</td>
<td>$0.2g$</td>
</tr>
<tr>
<td>Vehicle length</td>
<td>$9.14m$</td>
</tr>
<tr>
<td>Vehicle width</td>
<td>$2.59m$</td>
</tr>
</tbody>
</table>

5.4 Simulation Results

The driver decision making finite state machines that were discussed in Section 5.3 are applied for simulating a left turn scenario at an unsignalized intersection using a microscopic simulator. The simulated intersection area is the same as Figure 5.4.

In this study, visibility is assumed to be good, the driver is able to estimate the speed and position of other vehicles in 50 $m$ range with an angle of $-rac{\pi}{2}$ to $\frac{\pi}{2}$ radians. The lane change model developed in [96] has been picked as the lateral lane change model. A simple single time-constant, delayed driver response model as mentioned in [97] is assumed as the driver time delay model. The subject vehicle moves from south to west.

Figure 5.11 shows the position trajectory of the left-turning vehicle starting from its entering into intersection to its arrival of the target lane. Figure 5.12 shows the velocity and acceleration on the $x$ and $y$ axis of the subject vehicle. The first drop from 0 in the velocity along the $x$-axis indicates the lane change of the subject vehicle. From time point of 13 sec to 24 sec, both the velocity along the $x$-axis and the $y$-axis are zero, which indicates the subject vehicle is waiting for clearance of the conflict.
area. Only after the proposed condition C1, C5, and C6 are satisfied, the subject vehicle speeds up from zero and makes the left-turning. The velocity along the y-axis first increases and then decreases, as the vehicle makes a turn until the vehicle successfully arrives the target lane, the velocity along the y-axis becomes zero with acceleration of zero. Simultaneously, the velocity along x-axis increases. Note that trajectories of both velocity and acceleration in Fig. 5.12 is smooth. Also note that the acceleration is far below the physical limits which indicates that comfortability is also obtained. The simulation also shows that safety is ensured if the decision making strategy is applied by autonomous vehicle.

![Vehicle Trajectory (Left-turning Unsignalized)](image)

Figure 5.11: Position Trajectory of A Left-turning Vehicle at An Unsignalized Intersection
Figure 5.12: Velocity and Acceleration of A Left-turning Vehicle at An Unsignalized Intersection
5.5 Conclusions

A general architecture of driver model at intersections is presented. Details on driver decision making in various scenarios are discussed with finite state machines presented. The developed model, especially the driver decision making procedure, is implemented into a microscopic simulator for simulating a typical left-turning scenario at an unsignalized intersection. Simulation results show safe vehicle operations and mimicked human driver behavior with reasonable velocity and acceleration within certain bounds and constraints. The proposed architecture of driver model may be used in autonomous vehicle for decision making, in microscopic simulator for human driver response modeling, and in human-centered design for collision warning system. It should be mentioned that this study is based on rather simple assumptions for driver characteristics module, more study with complicated assumptions will be conducted in the future since it is crucial to the driver model.
CHAPTER 6

CASE STUDY AND SIMULATION RESULTS - INTER-VEHICLE COMMUNICATION BASED INTERSECTION WARNING SYSTEM

With recent advanced communication technologies, it is practical for researchers to design and develop wireless communication based collision warning system to further improve vehicle safety. IWS simulator, as mentioned in Chapter 4 allows the test and evaluation of different wireless protocols and collision warning systems.

6.1 Introduction

Each year intersection collisions account for nearly 2 million accidents and 6,700 fatalities in the United States [98]. To avoid intersection collisions, which constitutes approximately 26% of all vehicle collisions, vehicle location and direction information need to be exchanged before vehicles approach the intersection [99]. In this chapter, we present an Intersection Warning System (IWS) that utilizes a wireless network in which all nodes implement DOLPHIN [100], a non-persistent, random-backoff, CSMA-based medium access control protocol. In this IWS, a distance-based message generator decides when vehicles will transmit messages. By advertising the necessary information, such as position, velocity, and acceleration, vehicles are able to detect
potential collision hazards. The messages are sent either directly between vehicles over wireless modems or relayed by roadside wireless repeaters.

The performance of the proposed IWS in an urban intersection environment was studied under a variety of traffic flow and other relevant conditions. Human factors such as driver behavior and response time are also modeled and analyzed for the performance evaluation. A Vehicle Traffic Simulator (VTS) is presented with a real-time online wireless medium simulator (WS). The WS enables a feedback path to a collision warning system simulator that uses packet delivery status to alert and influence driver behavior. A high-level representation of the Simulator of IWS is presented in Figure 6.1. Our simulator architecture includes VTS, online WS, message generator, collision warning system, and driver behavior model. In this study, we will focus on the performance evaluation of the IWS.

The remainder of this chapter is organized as follows. A short description on VTS and WS is given in Section 6.2. Section 6.3 presents the distance-based message generator that connects the WS and collision warning system. Section 6.4 discusses the design of the collision warning system module. The driver behavior module is outlined in Section 6.5. In Section 6.6, performance evaluation of IWS is discussed. Finally, Section 6.7 concludes the chapter.

6.2 VTS and WS

6.2.1 Vehicle Traffic Simulator (VTS)

VTS is an important simulator utilized in this study. This simulator is responsible for providing realistic, microscopic-level traffic flow information to the wireless simulators (offline and online). In doing so, it has to account for such elements as intersection
Figure 6.1: Intersection Warning System Simulator Architecture
layout, vehicle throughput, vehicle types, and route selection. It is composed of road management, vehicle management and signal management components [84, 40].

**A. Road Management**

This module is responsible for handling the specific details of the urban intersection, including dimensions, maximum speed, number of lanes, building configuration, and the presence of a traffic signal or wireless packet repeater. This module provides significant flexibility for specifying the intersection layout and characteristics.

**B. Vehicle Management**

All vehicles in VTS must comply with the physical behavior guidelines specified by this module. These guidelines may vary by vehicle type, but include maximum acceleration/deceleration rates, minimum distances between vehicles, and effects due to vehicle dimensions while turning. Furthermore, this module is responsible for determining the route that vehicles will follow through the intersection, affecting additional vehicle behavior such as lane changes and turns.

**C. Signal Management**

This module is responsible for the traffic signal (if present). In addition to controlling the amount of time the light will stay at a given stage (green, yellow, red), it also determines if vehicles can turn right when the light is red.

**6.2.2 Wireless Simulator (WS)**

For our simulation study, two distinct wireless simulators are used. The offline WS [94] provides an accurate representation of packet transmission behavior, focused on packet loss probability, based on protocol and physical specifications. The
simulator handles a variety of configurable conditions, which include Media Access Control (MAC) protocol, building presence, repeater presence, initial data transmission, transmission interval, maximum number of retransmissions, and retransmission interval.

VTS uses millisecond-level intervals to model the vehicle movements while wireless communication needs to be simulated at microsecond level. Due to significant differences in operation time scales of the two simulators, an online WS is introduced, which performs a statistical approximation of wireless packet transmission. The online simulator relies on the data gathered from the offline wireless simulator [94] under a variety of simulation scenarios to estimate appropriate packet collision rates and packet transmission latencies as shown in Figure 6.1. It also accounts for the physical layer characteristics to perform actual determination of signal path loss and frame error rate.

6.3 Message Generator

The distance-based message generator is part of IWS and is responsible for determining when vehicles will transmit messages. As shown in Figure 6.2, vehicles transmit on a distance-based pattern that is determined by an initial transmission distance and a transmission interval.

6.3.1 Initial Transmission Distance

This value specifies the distance from the intersection at which vehicles will perform the initial data transmission. It also determines how soon a vehicle entering the intersection will begin broadcasting its presence, direction, and speed to other
vehicles. It is set to the largest possible value within the intersection area, which also maximizes the opportunity of earlier notifications for collision warnings.

6.3.2 Transmission Interval

Vehicles broadcast messages periodically based on this distance-based transmission interval value. The smaller the interval, the larger is the number of transmissions a specific vehicle performs while crossing the intersection area.

![Figure 6.2: Message Generator](image)

The message generator keeps track of vehicle positions and when they move across a specific transmission border (the horizontal dashed line in Figure 6.2), it triggers the message generation and transmission for these values. This distance based method
is different from the traditional time-based method, in which messages are transmitted at constant time intervals. The distance-based method is adopted to eliminate redundant message broadcasts in slow moving and stationary traffic conditions.

The message generator is flexible enough to handle different initial transmission distances and transmission intervals. Transmission intervals play an important role in collisions reduction. Section 6.6 will discuss the effects of different transmission intervals.

6.4 Collision Warning System

The collision warning system computes the probability of route contention for a given vehicle based on received messages and expected destinations of vehicles. Based on the immediacy of a collision, it will issue warning messages to the driver. Collision detection is a complex task given that there are a multitude of variables involved, including vehicle speed, acceleration/deceleration rate, and direction. A good collision warning system should reduce the collisions while minimize the driver’s attention load, for example, by not generating excessive warning signals and not distracting the driver \[101, 102, 44, 103\]. A three-level warning system is used to achieve the above requirements as described in Section 6.4.2.

6.4.1 Route Contention Computation

Route contention is defined as the possibility of the subject vehicle and principal other vehicle reaching their expected intersection point at the same time instant. \[44\] As shown in Figure 6.3, Vehicle 1 is the subject vehicle which is on its way from north to south and both Vehicle 2 and Vehicle 3 are principal other vehicles (a vehicle
which can potentially collide with the subject vehicle). Vehicle 1 and Vehicle 3 may have the route contention while Vehicle 1 and Vehicle 2 may not. To compute route contention, the following communication data are needed: direction, turning signal, position, current speed, and deceleration/acceleration.

Figure 6.3: Route Contention

For each vehicle, we need to compute the route contention between a subject vehicle and all other principal vehicles. To reduce the computation complexity, direction
TTC: Time-To-Collision (TTC) is used to compute route contention between vehicle pairs. TTC has proven to be an effective measure for rating the severity of conflicts, which is defined as the time required for two vehicles to collide if they continue at their present speed and on the same path. In principle, the lower the TTC, the higher is the possibility of a collision [43, 104, 105].

If a cross point on the routes of two vehicles exists and $abs \left( \frac{R_1 - l_1}{v_1} - \frac{R_2 - l_2}{v_2} \right) < \frac{l_1 + l_2}{R}$ is satisfied, TTC can be calculated as:

$$TTC = \min \left( \frac{R_1 - l_1}{v_1}, \frac{R_2 - l_2}{v_2} \right)$$ (6.1)

$R$: relative distance.

$\dot{R}$: relative velocity.

$l_i$: Vehicle i’s length along the route.

$v_i$: velocity of vehicle i.

### 6.4.2 Three Level Collision Warning Algorithm

**Time to Avoidance** (TTA) is introduced for the collision warning algorithm design [44]. Assuming that vehicles must come to a full stop to avoid the collisions,

$$TTA = \min \left( \frac{v_1}{\mu g}, \frac{v_2}{\mu g} \right)$$ (6.2)

$\mu$: Friction Coefficient.

$g$: Gravitational acceleration.

The three-level collision warning system algorithm is illustrated in Figure 6.4. The multi-level warning system is composed of three levels, namely level 1 (elevated), level 2 (high) and level 3 (severe).
**Driver Response Time:** Time to the initial driver action. Initial driver action is defined as the first action the subject performs after the incurring vehicle initiated movement, i.e., either begins to release the accelerator pedal or begins to steer as part of this measure. The data used in this study is normally distributed with the mean value of 2.23s and the standard deviation of 0.1s and is truncated to the range 1.93s - 2.53s [106].

![Diagram of Three-level Collision Warning System Algorithm](image)

Figure 6.4: Three-level Collision Warning System Algorithm

### 6.5 Driver Behavior Model

Modeling driver behavior is a complex but necessary responsibility within VTS since not all drivers behave and react the same way. This module defines a set of driver types and distributes each type among the vehicles. The driver model determines the
driver’s acceleration, deceleration, and response to warnings. For this purpose, a classification of driver’s characteristics is needed. The drivers are grouped into three categories based on their desired speed \([88], [90]\), according to Equation 4.5.

Though driver response time is independent of driver’s characteristics, driver response motion is assumed to depend on driver’s characteristics. Drivers may release initial accelerator only, decelerate, or brake, depending on their own characteristics and the warning level \([107, 44, 106]\). The response motion is summarized in Table 6.1.

<table>
<thead>
<tr>
<th></th>
<th>Level 1 (Elevated)</th>
<th>Level 2 (High)</th>
<th>Level 3 (Severe)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggressive</td>
<td>N/A</td>
<td>Initial Accelerator</td>
<td>Decelerate</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Release</td>
<td></td>
</tr>
<tr>
<td>Normal</td>
<td>Initial Accelerator</td>
<td>Decelerate</td>
<td>Brake</td>
</tr>
<tr>
<td></td>
<td>Release</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conservative</td>
<td>Decelerate</td>
<td>Brake</td>
<td>Brake</td>
</tr>
</tbody>
</table>

Table 6.1: Driver Response Motion Model

### 6.6 Performance Evaluation of the IWS

As shown in Figure 4.4, VTS and online WS including the driver behavior model are used to simulate the IWS at urban intersection and evaluate its performance.

#### 6.6.1 Simulation Environment

The urban intersection layout is depicted in Figure 6.2. It is a four-way four-lane crossing that each intersection lag has two lanes, where traffic can travel in any direction on the appropriate lanes. Depending on the specific scenario being analyzed, a traffic signal is present in the middle of the intersection to direct all incoming traffic.
In the case where a traffic signal is not present, the intersection becomes a four-way stop, where all vehicles must stop at before proceeding. Intersection parameters and properties are discussed below. Two additional elements, a wireless repeater and buildings, are also incorporated into the simulator. These two elements mostly have effects on packet delivery. The analysis of their effects on the IWS is out of the scope of this study.

A. Common Parameters

For this investigation, certain environment and simulation properties were constant for all scenarios.

Intersection area: This defines the total distance along the horizontal (x) and vertical (y) axes, from one edge of the intersection area to the other. Inbound vehicles are not considered within the intersection area until they have crossed this virtual border. Furthermore, positional vehicle information is available only when the vehicle is within this area. Therefore, inbound vehicle information is generated once the vehicle crosses this virtual border and outbound vehicle information ceases to be generated once it passes this limit. The intersection area is defined as 215 m by 215 m.

Vehicle types: There are four vehicle types in our simulation: passenger vehicles (cars), buses, trucks, and motorcycles as shown in Table 4.1. A vehicle’s type determines not only its physical dimensions and properties, but also dictates its behavior in traffic and has significant impact on wireless transmission behavior, details can be found in Section 4.2.5.
Vehicle distribution: This parameter specifies the average share of a given type of vehicle within the intersection and throughout the simulation. Each vehicle is assumed to have a 25% uniform distribution in all simulations.

Speed limit: This is the maximum speed at which vehicles travel, although motorcycles are allowed to travel at a higher speed than the speed limit. Given that driver behavior and intersection conditions ultimately dictate vehicle speed, this value serves as a general guideline for average vehicle speed. However, based on the message transmission behavior described previously, actual vehicle speed directly determines the frequency at which information is transmitted. The speed limit is set to 45 mph (20 m/s) for passenger vehicles, trucks, buses and 57.5 mph (30 m/s) for motorcycles.

Initial transmission distance: It is selected as the largest possible value within the intersection area, which also maximizes the opportunity of advanced notifications for collision warnings. The initial transmission distance is set to 100 m.

B. Variable Parameters

The IWS is analyzed under a variety of different conditions.

Intersection vehicle throughput: Vehicle throughput is the target average number of vehicles that enter and leave the intersection area once the simulation reaches a steady state. Even though traffic patterns are intentionally non-deterministic, a concerted effort is made to maintain the overall throughput at the specified level. As expected, there is a transitional period for the traffic pattern to reach that desired vehicle throughput. Determining the appropriate set of vehicle throughput values is essential to evaluate performance across a representative range of traffic conditions.
Based on the level of service (LOS) [91] of the intersection and for scenarios without traffic signals, the range of the simulated vehicle throughput values are 1200, 1800, 3600, 5200, 7200, 12000 vehicles/hour as shown in Table 6.2. For signalized intersections, though the same range is chosen, the saturated and unsaturated conditions are different as shown in the same table. In our studies, we evaluate not only the free-flow operations in which vehicles are completely unimpeded in their ability to maneuver (LOS A), the reasonably free-flow operations where free-flow speeds are sustained (LOS B), and the vehicle operations are restricted (LOS C), but also high traffic throughput in which the traffic condition is quite poor (LOS D and LOS E) to show the trends of the wireless communication effectiveness.

<table>
<thead>
<tr>
<th>LOS</th>
<th>Unsignalized Theoretic</th>
<th>Simulated Theoretic</th>
<th>Signalized Theoretic</th>
<th>Simulated Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>≤ 2592</td>
<td>1200</td>
<td>≤ 4013</td>
<td>1200</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1800</td>
<td>(unsaturated)</td>
<td>1800</td>
</tr>
<tr>
<td>B</td>
<td>≤ 4147</td>
<td>3600</td>
<td></td>
<td>3600</td>
</tr>
<tr>
<td>C</td>
<td>≤ 6260</td>
<td>5200</td>
<td>saturated</td>
<td>5200</td>
</tr>
<tr>
<td>D</td>
<td>≤ 8294</td>
<td>7200</td>
<td></td>
<td>7200</td>
</tr>
<tr>
<td>E</td>
<td>≤ 8662</td>
<td></td>
<td></td>
<td>12000</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>12000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.2: Comparison of Theoretical Maximum and Selected Values

**Traffic signal presence:** The traffic signal controls the flow of vehicle traffic within the intersection. Its presence is optional.

**Number of vehicles:** This value determines the total number of vehicles that will pass through the intersection during the simulation period. Effectively, this parameter dictates the length (in time) of the simulation. Simulation length is an
important factor, as sufficient time must be spent under stable flow conditions to provide a representative sample for a given intersection vehicle throughput. Two values are used in the study. Vehicle number of 400 is used on all cases except the highest throughput scenarios (7200 and 12000 vehicles/hour), in which case, a larger number of vehicles, 800, is used to reach a stable traffic flow for a significant period.

Transmission interval: The simulated transmission intervals are 5, 10, 20, and 50 m.

6.6.2 Simulation Results

![Collisions Percentage versus Traffic Flow](image)

Figure 6.5: Percentage of Collisions at Unsignalized Intersection
Figure 6.6: Percentage of Collisions at Signalized Intersection
C. Percentage of Collisions

One important metrics used to evaluate the performance of the IWS is the vehicle collision percentage. From the macro point of view, total collision number of the system is counted and collision percentage is computed as: 

\[ CP = \frac{\text{Collided Vehicles}}{\text{total number of vehicles}}. \]

First, the collision percentage under various flow rates at an intersection without IWS are simulated. In case of IWS, the transmission intervals used are 5, 10, 20, and 50 m. Simulations are performed for both signalized and unsignalized intersections. Figure 6.5 and Figure 6.6 depict the collision percentage vs. flow rate at a unsignalized intersection and a signalized intersection, respectively.

i. IWS and transmission interval effect

Figure 6.5 and Figure 6.6 clearly show that the IWS significantly reduce the vehicle collision percentage. Lowering the transmission interval can avoid collisions more effectively. However, further reducing transmission interval does not improve the IWS, as increasing the vehicle path information update frequency does not affect the decisions of the driver behavior module.

ii. Effect of traffic flow

As shown in Figure 6.5 and Figure 6.6, the collision percentage increases initially with increased flow rate. At higher flow rates (lower LOS), the free traffic flow no longer exists, and congestion starts to appear at the intersection, which will cause the actual intersection throughput to saturate. As a consequence, an almost constant collision percentage is observed at higher flow rates.
The performance improvement by IWS can be characterized by the collision avoidance percentage, which can be calculated as:

\[ I = \frac{CP_{wo} - CP_{w}}{CP_{wo}} \]  

(6.3)

where

- \( I \): Collision Improvement.
- \( CP_{w/o} \): Collision Percentage with/without IWS.

Table 6.3 and Table 6.4 list the calculated performance improvement for each transmission interval. The results show that the flow rate also affect the performance improvement due to the IWS. When flow rate is high, i.e., LOS is low, more collisions can be avoided. Because the IWS acts more effectively as an additional traffic signal to regulate the traffic flow in case of low LOS.

<table>
<thead>
<tr>
<th>Traffic Flow</th>
<th>5m</th>
<th>10m</th>
<th>20m</th>
<th>50m</th>
</tr>
</thead>
<tbody>
<tr>
<td>1800</td>
<td>20%</td>
<td>40%</td>
<td>55%</td>
<td>40%</td>
</tr>
<tr>
<td>3600</td>
<td>36%</td>
<td>23%</td>
<td>43%</td>
<td>26%</td>
</tr>
<tr>
<td>5200</td>
<td>62%</td>
<td>43%</td>
<td>25%</td>
<td>10%</td>
</tr>
<tr>
<td>7200</td>
<td>65%</td>
<td>65%</td>
<td>53%</td>
<td>31%</td>
</tr>
<tr>
<td>12000</td>
<td>66%</td>
<td>69%</td>
<td>57%</td>
<td>39%</td>
</tr>
</tbody>
</table>

Table 6.3: Collision Improvement at Unsignalized Intersection

iii. Traffic Signal effect

In both unsignalized and signalized cases, IWS lowers the collision percentage. In case of signalized intersection, the collision percentage reach its saturated state at a lower traffic flow rate. Since the traffic signal can regulate the flow, it reduces the collision percentage at high flow rate. However, the improvement due to IWS with
traffic light is not as much as without traffic light, as can be revealed by seen Table 6.3 and Table 6.4.

**D. Collision Speed**

The performance of the IWS can also be evaluated by collision speed. From the micro point of view, each driver is concerned more on the severity of the collision if it cannot be avoided. A lower collision speed is desirable, as it implies the a lower severity of collisions.

Figures 6.7 and 6.8 depict the average collision speed for flow rates of 5200 and 7200 vehicles/hour, respectively. The collision percentages are also plotted on the same figures. Figures 6.7 and 6.8 clearly show that, by using IWS and decreasing the transmission interval, the collision speed can be reduced, which is similar to the trend observed in collision percentage. The comparison between unsignalized and signalized intersection show the presence of traffic light can reduce the collision speed, however, the improvement due to IWS with traffic light is lower than without traffic light, which is also observed in the improvement of collision percentage.

<table>
<thead>
<tr>
<th>Traffic Flow</th>
<th>5m</th>
<th>10m</th>
<th>20m</th>
<th>50m</th>
</tr>
</thead>
<tbody>
<tr>
<td>1200</td>
<td>36%</td>
<td>27%</td>
<td>33%</td>
<td>12%</td>
</tr>
<tr>
<td>1800</td>
<td>48%</td>
<td>41%</td>
<td>33%</td>
<td>11%</td>
</tr>
<tr>
<td>3600</td>
<td>57%</td>
<td>40%</td>
<td>32%</td>
<td>26%</td>
</tr>
<tr>
<td>5200</td>
<td>50%</td>
<td>44%</td>
<td>30%</td>
<td>27%</td>
</tr>
<tr>
<td>7200</td>
<td>58%</td>
<td>50%</td>
<td>38%</td>
<td>37%</td>
</tr>
<tr>
<td>12000</td>
<td>52%</td>
<td>49%</td>
<td>41%</td>
<td>41%</td>
</tr>
</tbody>
</table>

Table 6.4: Collision Improvement at Signalized Intersection
Figure 6.7: Collision Speed and Collisions Percentage (5200 Vehicles/hour)
Figure 6.8: Collision Speed and Collisions Percentage (7200 Vehicles/hour)
6.7 Conclusions

In this chapter, an IWS with a distance-based warning message generator is presented. Three levels of warning message are generated to alert the drivers. The performance of this IWS is studied with a VTS and a real-time online WS, which enables a feedback path to a collision warning system simulator. The driver behavior model is also included in the simulator. The IWS performance is evaluated from both macro and micro perspectives. Simulation results show that our IWS can significantly reduce both the collision percentage and collision speed. Smaller distance interval for message generation can more effectively avoid collisions and lower the collision speed.
CHAPTER 7

CASE STUDY AND SIMULATION RESULTS - DRIVER ASSISTANCE SYSTEMS

7.1 Introduction

Besides intersection warning system proposed in Chapter 6, two driver assistance systems: stop-and-go driver assistance system and highway merging driver assistance system will be presented in this chapter, where the inter-vehicle communication is served as a key technology because of its capability as shown in Chapter 5 in acquiring the data on neighboring vehicles. These data are essential for improving safety and efficiency but difficult for on-board equipment to estimate. This chapter also describes the development of two road traffic simulators, SSTS and MVTS, which are integrated analysis-evaluation systems designed for traffic safety and efficiency assessment with the primary focus on estimating the effects of the driver assistance systems. The effectiveness and validity of the present systems are demonstrated through comparison with simulated traffic data.
7.2 Stop and Go Driver Assistance System

7.2.1 Introduction

One of the major causes of the accidents on the road is determined by the perception limits of the driver. In order to increase perception limits and simultaneously help decreasing reaction limits, inter-vehicle communication (IVC) or road-to-vehicle communication (RVC) is introduced. Different communication technologies including Bluetooth, Wide Area Network (WAN) through 802.11b wireless technology are used for exchanging information among the vehicles and infrastructure [108]. A wireless network simulator using a non-persistent, random-backoff, CSMA-based medium access control protocol is also used in this section [109]. IVC entails the communication between two or more vehicles, whereas RVC involves traffic monitoring and management services i.e., exchange of road, traffic information or electronic toll collection systems while navigating. Likewise, IVC has a large set of ongoing efforts and possible applications. These are generally divided into three categories: communication-based longitudinal control, co-operative assistance systems, and information and warning functions. Co-operative assistance systems aim to improve vehicle safety by communicating time-critical information to nearby vehicles. Inter-vehicle communication to improve road safety by cooperative collision warning operation has been considered in [110]. The platoon stability for autonomous intelligent cruise control using both front and back information for vehicle following maneuvers has been considered in [111]. In this section, we present stop and go driver assistance system with and without wireless communication. We propose the simulation scenarios based on the assumption that slowing down or emergency braking to avoid collision is controlled
by the automated driving system in a predefined sensing range; and starting or acceleration is executed by the human driver. We consider a totally isolated car (without cooperation) and we also consider the possibility of cooperation through wireless with the cars ahead. We use constant time gap control as cruise control policy and we simulate collision avoidance and merging scenarios to illustrate the effectiveness of wireless communication in intelligent vehicle applications and the proof-of-concept of technology availability.

7.2.2 Single Street Traffic Simulator and its Components

A. Driver Model

In developing and simulating the stop and go driver assistance system we assumed a simple driver model,

$$\frac{\delta_s}{\delta_d} = e^{-\tau_s} \frac{1}{T_I s + 1}$$  

(7.1)

where $\delta_s$ is the driver model steering wheel angle, $\delta_d$ is the desired steer angle, $\tau$ is the magnitude of pure time delay and $T_I$ is the lead time constant. These driver model parameters based on average driver behavior are in the range $0.4 \leq \tau \leq 0.5$ seconds and $0.11 \leq T_I \leq 0.29$ seconds for lane tracking with the full-size car [97]. Furthermore, the driver’s perception and reaction time play a significant role in all problems of active safety. Reaction time is sometimes known as “perception-reaction” time or “perception-response” time. For longitudinal motion, where we are interested in a leading driver’s input in applying a throttle signal to accelerate the car, a simple driver model such as the one in Equation 7.1 will also be used. However, the two parameters for longitudinal motion will necessarily be different. Deceleration and
stopping for time-critical situations are going to be accomplished by the stop-and-go driver assistance system.

**B. Adaptive Cruise Control**

The most conventional ACC algorithm is constant time gap control, which is given by,

\[ \ddot{x}_{des} = -\frac{1}{h}(\dot{\varepsilon}^i + \lambda \delta^i) \]  \hspace{1cm} (7.2)

\[ \varepsilon^i = x^i - x^{i-1} + L \]  \hspace{1cm} (7.3)

\[ \delta^i = x^i - x^{i-1} + L + hx^i \]  \hspace{1cm} (7.4)

where \( L \) denotes desired constant spacing between vehicles in a group (typically 2–10 m), \( \varepsilon^i \) denotes spacing errors, \( h \) is used for constant time gap (CTG) and it is usually between 1 and 2 seconds, \( \lambda \) is the slope of the sliding surface. This adaptive cruise control policy uses the relative speed and it contains an extra term to fulfill time headway control. It has been proven that this control law can ensure string stability in a group of vehicles [23]. In this work, a typical ACC controlled car-following case with desired constant time headway of 1 seconds, constant spacing between vehicles in a platoon of 5 meters, and convergence rate of 0.2 are chosen for headway control simulation scenarios in urban traffic.

**C. SSTS**

A three-lane roadway, as shown in Figure 4.8, is implemented in the simulator. It defines the total distance along the horizontal (x) and vertical (y) axes. Inbound vehicles are not considered within the plotted area until they have crossed this virtual
border. In addition, information of the vehicle position is available only when the vehicle is within this area. Therefore, inbound vehicle information is generated once it crosses this virtual border and outbound vehicle information ceases to be generated once it passes this limit. The simulator is designed in such a way that user is able to make stop and go to the vehicles by intervening through a mouse’s click. Numbers of vehicle groups are simulated in the simulator to evaluate the group effects.

Four types of vehicles are implemented in this simulator according to Table 4.1. We use simplified longitudinal model of the vehicle, a first order linear system with a time constant of 10 seconds and with a unity gain coupled with an acceleration limit of $2\text{m/s}^2 (0.2g)$ and a deceleration limit of $3.1\text{m/s}^2 (0.31g)$.

\[
\dot{V} = \begin{cases} 
2 & (V_C - V) > 20.0 \\
-3.1 & (V_C - V) < -31.0 \\
0.1(V_C - V) & \text{else} 
\end{cases} 
\]  

(7.5)

where $V_C$ is commanded velocity and $V$ is current velocity. In car-following mode, the vehicle model reaches a specified following distance $D$ and it matches the speed of leading vehicle according to $V_C = \max(V_{lead} + 0.1(D_{\text{ahead}} - D), V_{\text{desired}})$. In case the leader is slower and the leader is faster, $V_C = V_{lead} - V$ is used. We assume that slowing down and stopping is accomplished by the automated system.

### 7.2.3 Simulation Scenarios

Stop and go in urban area scenario, constituted by 12 vehicles which are divided in two groups, is simulated in this study. The effect of inter-vehicle communication is considered from active safety point view.
A. Vehicle Groups Without Inter-vehicle Communications

Car following without inter-vehicle communication simulation scenario implementation using traffic simulator is given in Figure 7.1. In the simulation of stop and go scenario in urban area, we assume all the vehicles have an initial speed of 25 mph, their time gaps are 1.5 seconds and safety distance $L$ is 5 m, which is equal to sensing range. A stationary obstacle is placed into the lane of the coming vehicles. In case that no communications exist in two groups, we consider isolated vehicles only, it means that there is no communications between vehicles; the vehicle can only see the vehicle ahead when the leader is in its detection range. All the preceding vehicles has only the information of the distance and velocity of the vehicle directly ahead and the vehicle can only see the vehicle ahead when the leader is in its detection range. ACC may well perform if enough CTG is kept between the following and preceding vehicle.

In Figure 7.2, the spacing between the preceding and leading vehicles is given. When the stationary obstacle is sensed by the leading vehicle in the first group, all the following vehicles in the sensing range of the front vehicle decelerate to avoid collision. Spacing between vehicles becomes very tight just barely to avoid the collision. Also two groups merge into one. CTG for each vehicle is enough to safely stop to avoid the stationary obstacle at low speed operation. The longitudinal velocities of the vehicles in two groups are decreasing by following the front vehicle’s reaction. Since communication between two groups or among the vehicles does not exist, the speed profile of the vehicles in the groups is activated only by sensing the leading vehicle’s reaction. As CTG has been adequate for stop and go or collision avoidance by emergency braking, the emergency deceleration limit $0.31 g \text{ m/s}^2$ has
Figure 7.1: Screenshot of Simulation Scenario Constituted by Two Vehicle Groups without Inter-vehicle Communications
Figure 7.2: Stop-and-go without Driver Assistance System
been satisfactory to stop the leader vehicle of the second group in 5 meters’ sensing distance.

**B. Vehicle Groups With Inter-vehicle Communications**

Vehicle groups’ stop-and-go scenario with inter-vehicle communication is simulated and a screen shot of the considered vehicle group’s scenario is given in Figure 7.3. In this scenario, all the vehicles are assumed to be capable of communicating and all of them are within the communication range of the leading vehicle of the first group. Vehicles are able to cooperate with their capability of communicating. After the leading vehicle of the first group senses an obstacle ahead and starts to decelerate to avoid collision, it sends out the warning message. All the vehicles are informed about the possible hazard and situation at the same time regardless of the packets transmission time. The desired CTG are well maintained between vehicles due to communication and early reaction capabilities as shown in the trace files of vehicles in Figure 7.4. Since message is received by all the vehicles in both of the groups, the vehicles start braking simultaneously in order to avoid possible obstacle and collision by stopping while maintaining safe following distance or assigned constant time gap profile. Since CTG is well maintained with early reaction of the following vehicles in the groups, smooth deceleration and acceleration profiles are obtained.

**C. Merging Scenarios on the Highway**

In merging scenarios we assume that the vehicles have an initial speed of 55 mph on highway with a safety distance $L$ of 10 m and time headway of 2.5 sec. These values are higher than previous stop and go operations. We simulate merging scenario followed by stop and go scenario. A stationary obstacle is placed on the lane and all
Figure 7.3: Screenshot of Simulation Scenario Constituted by Two Vehicle Groups with Inter-vehicle Communications
Figure 7.4: Stop-and-go with Driver Assistance System
the vehicles in the first group and the merging vehicles have the communication ability. Safety distance and time headway are chosen to be larger than urban traffic scenarios because of the high speed on highway.

![Figure 7.5: Screenshot of Merging Scenario](image)

As the vehicle is taking on-ramp on highway to merge into the lane, it broadcast its presence and ask for merging space. The vehicle on the highway in the back neighborhood slows down to preserve safe space to the merging vehicle as shown in Figure 7.5.

Merging scenario is followed by stop and go scenario. A stationary obstacle is detected by the leader of the first platoon and possible collision warning message is issued to the other vehicles in the first group. Since the vehicles of the second group do not receive warning message, as in the urban traffic scenarios, the late reaction
Figure 7.6: Trace Files of Vehicles involved in Merging Scenario
of the leading vehicles in the second group may result in possible collision with the last vehicle of the first group. Trace files show that enough safety distance between vehicles may not be maintained due to the lack of communication with the other vehicles and high speed on the highway, Figure 7.6. Two groups are merged into one and the vehicles in the first group are able to keep the prespecified safe CTG, but those in the second group are not able to preserve safe CTG, as shown in Figure 7.7.

![Figure 7.7: Speed Profile of Vehicles involved in Merging Scenario](image)

### 7.2.4 Conclusion

In this study, we present stop and go driver assistance system with and without wireless communication. We consider a totally isolated car (without cooperation) and
the possibility of cooperation through wireless communication with the cars ahead. We use constant time gap control as cruise control policy assuring stability of vehicles’ string and we simulate collision avoidance and merging scenarios to illustrate the effectiveness of wireless communication in intelligent vehicle applications and the proof-of-concept of technology availability.

7.3 Merging/Lane Changing Driver Assistance System

7.3.1 Introduction

An important operation in automated highway systems is merging control. A good merging control system can avoid possible collisions and minimize the traffic jam caused by the merging vehicles. As early as in 1960s, merging control under manual driving has been studied extensively by many researchers [112]. Many theoretical and simulation studies were conducted and some of them progressed to field testing of designed controllers [113, 114, 115]. Recently, extensive research has been reported with the goal of designing efficient and collision free merging control for adaptive cruise control vehicles [116, 28]. With the aid of wireless communication, researchers are able to design vehicle control algorithm for integrating cooperative driving with ACC systems [116, 117, 118, 100]. Even with no automation, a warning to the driver could help safety. Simulation results showed that with the inter-vehicle communication, intersection collision hazards can be lowered [40]. The Vehicle Safety Communication Consortium (VSCC) proposed communication requirements and information for these high-priority safety applications [119]. At the Ohio State University, we have been investigating different aspects of wireless communication among vehicles, especially implications regarding safety. An initial aspect of this investigation was based on
intersection collision avoidance systems [40]. We have developed and demonstrated such a system with OKI Electric Co. [40, 94, 120, 41]. An extensive simulator covering the physical layer model, MAC layer protocol model and transportation aspects of the situation at an intersection was also developed [120, 41]. The transportation aspects are important since such a model provides information on “which vehicle is where” when each one attempts to communicate with another. In this section, we consider two other applications where vehicle to vehicle communication is important for safety. These are “merging” and “lane change”. We introduce both a vehicle merging algorithm and provide simulation studies to investigate the implications of using communication between the vehicles, based on a system like the one utilized in our intersection studies. The performance of the proposed algorithm was studied under a variety of traffic flow and other relevant conditions, which can later be served as a driver assistance system. The simulation results clearly show that, the proposed algorithm can significantly reduce the delay and avoid the collisions causing by ramp merging.

7.3.2 Finite State Machine for Merging Control

A. Typical Merging Problem

Consider the three vehicles shown in Figure 7.8 that will be directly affected when the ramp vehicle performs a merge maneuver. Vehicle $M$ needs to maintain a certain safety distance between vehicle $F$ and Vehicle $L$, so that it can perform a collision free merge maneuver. Conditions are such that there is no enough gap for Vehicle $m$ to merge in between Vehicle $L$ and Vehicle $F$ at the time Vehicle $m$ decides to merge. Vehicle $M$ therefore needs to slow down or stop to avoid collisions and waits
for enough gaps. If there is enough spacing ahead of Vehicle $L$, or in other lanes. Vehicles $m$ actually makes an unnecessary wait, which increases the time taken for it to merge into the main lanes. A merge control algorithm is thereby needed to solve this problem by better utilizing the space around the merging spot. To implement the proposed merge control algorithm, inter-vehicle communication is required to exchange information such as position and speed between vehicles.

![Figure 7.8: Ramp Merging Problem](image)

**B. Merging control algorithm**

With the aid of inter-vehicle communication, Vehicle $m$ broadcasts its position and speed when it arrives at the ramp. A virtual mapping method [117, 100] is employed to map the merging vehicle into the main lane. Therefore, the critical issue in the merging control algorithm is to generate enough spacing between Vehicle $F$ and Virtual Vehicle $M$ as well as Virtual Vehicle $M$ and Vehicle $L$.

As stated in Section 7.3.2, there is no enough spacing between Vehicle $F$ and Vehicle $L$ for Vehicle $M$ to cut in, but there is an open space ahead of Vehicle $L$. 
The main idea for this algorithm is to utilize the open spot ahead of vehicle $L$ within the communication range. Using IVC, vehicles may exchange information and make cooperative maneuver to move the open spot backward and accommodate vehicle $M$.

Figure 7.9: Virtual Mapping Method

Figure 7.10 shows the detailed merging control algorithm. Upon entering the system, the merging vehicle broadcast its GPS coordinates, Time stamp, Vehicle speed, Vehicle acceleration, Vehicle heading, Vehicle size and GPS antenna offset [40]. Vehicles in the communication range send their own position, speed, headway, and vehicle size to the merging vehicle. After receiving the information, the merging vehicle virtually maps itself into the main lanes and looks for the closest open space at the time it reaches the merging point. It also calculates whether there is any road contention between itself with its closest follower. It will then broadcast the position of the closest open spacing which is ahead of itself, the closest follower’s ID if there is a potential road contention. Based on these information, the vehicles on the main lanes compute whether they are affected and determine what to do. Those vehicles which are ahead of the merging vehicle and behind the closet open spacing will change
lane and/or speed up, depending on whether the open spot is in the rightmost lane. The open space is then effectively moved backward and matched to the position of the merging vehicle. If there is a potential road contention between the closest follower and the merging vehicle, the closest follower will slow down until the spacing is clear. It should be mentioned that the merging vehicle doesn’t look for the closest open space which is behind itself, because it is not wise to have a number of vehicles on the main lanes slow down to move the open space forward. It also needs to be noted that all the affected leaders will match its speed to the merging vehicle until the merging vehicle finishes its lane change. Figure 7.11 shows the vehicle state transitions using the above algorithms.

Figure 7.10: Merging Control Algorithm
Figure 7.11: Vehicle State Diagram
7.3.3 Merging Vehicle Traffic Simulator

A. The Ramp Layout

A parallel ramp entrance as shown in Figure 4.9 is implemented in the simulator. It defines the total distance along the horizontal (x) and vertical (y) axes, from one edge of the ramp area to the other normal highway. Inbound vehicles are not considered within the plotted area until they have crossed this virtual border. In addition, vehicle positional information is available only when the vehicle is within this area. Therefore, inbound vehicle information is generated once it crosses this virtual border and outbound vehicle information ceases to be generated once it passes this limit. The speed limit on the simulated highway is 60 miles/hour, the acceleration length is assigned to be 375 m in accordance with [121]. A total length of 600 m is simulated.

B. The Vehicle Management

Four types of vehicles are implemented in this simulator according to Table 4.1. Longitudinal model of the vehicle is mentioned in Section 7.2.2 and the lateral model of the vehicle can be found in details in [37].

7.3.4 Simulation Results

A. Assumptions and Parameters

The merging algorithm is evaluated using DrivCom Simulator under the following assumptions and parameters:

- Communication Range: All the vehicles have the same communication range of 200m.
• **Traffic flow on the highway:** Traffic flow on the highway is the average number of vehicles that enter and leave the highway area once the simulation reaches a steady state. Even though traffic patterns are intentionally non-deterministic, a concerted effort is made to maintain the overall throughput at the specified level. As expected, there is a transitional period or the traffic pattern to reach the targeted vehicle throughput. Determining the appropriate set of vehicle throughput values is essential to evaluate performance across a representative range of traffic conditions. The range of the simulated vehicle throughput values are 1200, 1800, 2400, 3200 vehicles/hour.

• **Traffic flow on the ramp:** To avoid interaction between vehicles on the ramp, we assume free flow equals to 600 vehicles/hour. All the vehicles are generated according to Poisson arrival process.

• **Velocity:** The desired velocity used in this study is normally distributed with the mean value of speed limit equals to 26.67 m/s (60 miles/hour) and the standard deviation of 2.67 m/s and is truncated to the range 24 – 29.34 m/s.

**B. Simulation Results**

i. **Typical scenarios**

• **Lane changing:** In the lane changing scenario, the affected vehicle changes its lane to accommodate the merging vehicle. Figure 7.12 shows typical time-space traces of the merging vehicle (red lines) and the affected vehicle (blue lines) in this scenario. X-axis and y-axis are parallel, and perpendicular to the lane direction, respectively. The dash lines, which are the projection of the solid lines on the x-y plane, represent the space traces of those two vehicles. This
figure shows that after receiving the broadcast from the merging vehicle and computing the emerging road contention, the affected vehicle changes to the left lane, where the vacancy is. The merging vehicle then merges into the main road safely. Note, the vehicle speeds are kept constant according to merging vehicle in this scenario.

![Figure 7.12: Typical Time-space Traces of the Merging Vehicle (red lines) and the Affected Vehicle (blue lines) in the Lane Changing Scenario](image)

- **Speed changing scenario**: In speed changing scenario, the affected vehicles will change their speed to accommodate the merging vehicle. Figure 7.13 shows
a typical speed changing scenario, in which two affected vehicles move in platoon with the spacing less than the threshold, while the ideal merging point for the merging vehicle is in between these two affected vehicles. At time point of 24.76 second, both the two affected vehicles receive the broadcast from the merging vehicle and compute out the road contention. Since there is no vacancy on the left lane, the leading vehicle starts to accelerate while the following vehicle starts to decelerate to accommodate the merging vehicle as shown in Figure 7.13 (a), (b). After the spacing between these two vehicles exceed the threshold, both the two vehicles change back to their ideal speed. Figure 7.13 (c) clearly shows the vehicle spacing change in the platoon.

![Figure 7.13: Speed Changing Scenario](image-url)
ii. Delay time

Without inter-vehicle communication, merging vehicles have to slow down or even stop in case of the merging point is occupied by another vehicle, therefore it takes longer time for vehicles to merge into the right lane, hence the delay occurs. Simulation was run to study the effect of IVC on the delay due to the occupied merging point. Traffic throughput of 1200, 1800, 2400, and 3200 vehicles/hour in the right lane are simulated. The delay of the merging vehicle is calculated by subtracting the time required for vehicles to merge in case of no traffic in the right lane from the time obtained by simulation. For each throughput, 100 merging vehicles are simulated. The average delays under various conditions are plotted in Figure 7.14. The

![Figure 7.14: Delay Time Improvement](image-url)
data show that as traffic throughput increase from 1200 to 3200 vehicles/hour, the average delay is increased from 0.3 s to 1.5 s when no IVC is employed, because the probability of occupied merging point is increased. However, the delay is kept almost constant around 0 s when IVC is employed. The simulations clearly demonstrate that by employing IVC and this proposed merging algorithm, the vehicles which have road contention with the merging vehicles will try to change their speed or move to the left lane to leave the merging point open, therefore, the delay due to increased traffic throughput is minimized.

iii. Collision avoidance

Collisions may happen during the lane merge when no IVC is available. For example, one possible scenario is that one vehicle merge into the main lane at a relative low speed, while another vehicle with a higher speed on the main lane doesn’t have enough time and distance to decelerate. In this case, collision occurs. With the IVC, since the vehicle on the main lane received the broadcast of the merging vehicle, it will decelerate in advance so that collision can be avoided.

Figure 7.15 shows the simulation results. No collision occurs for the entire 4 traffic throughput with the IVC. However, the collision percentage increases from 0% to 1.5% without communication. The results clearly demonstrate that IVC can significantly avoid collisions.

7.3.5 Conclusions

In this section, we proposed a new merging control algorithm for vehicles with inter-vehicle communication, which can be later served as a driver assistance system. Available spacing inside the communication range for merging vehicle can be utilized
The effect of IVC on collision avoidance

Figure 7.15: Effects of the IVC on Collision Avoidance
by the vehicle’s cooperative driving. A merging vehicle traffic simulator has been
developed for evaluating the proposed algorithm. Simulation results show that the
algorithm can significantly reduce average delay and avoid collision. The proposed
merging control algorithm can be readily applied in general lane change scenario.
CHAPTER 8

CONCLUSIONS

Traffic safety and efficiency assessments have been in substantiated needs. This work describes the development of a hierarchical hybrid system model for testing vehicle safety and traffic management control policies as a primary capability in addition to the usual requirements such as real-time constraints, and so on. The model takes both single vehicle operation and multiple vehicles cooperations into account. By employing the hierarchial hybrid system model, we develop the vehicle traffic simulators for modeling traffic networks with advanced vehicle control and traffic management. The main components of the simulators architecture have been presented and discussed in details in this dissertation. The simulators are appropriate for analyzing, testing and evaluating vehicle operation control policies and traffic management strategies, such as merging algorithms. In addition, it supports simulation of various human driver models and decision makings, which mimics human driver’s response in various scenario and may be used as basic knowledge for automotive vehicle decision making process.

A typical safety case, intersection accident mitigation problem is analyzed and studied. Human driver model, including the driver decision making, is proposed for both simulation and designing the warning system. An intersection warning system
with a distance-based warning message generator is presented, which generates three levels of warning message to alert the drivers. The performance of this intersection warning system was evaluated with the vehicle traffic simulator and a real-time online wireless simulator, which enables a feedback path to a collision warning system simulator. Our current simulation results suggest that wireless communication based driver assistance system can significantly increase traffic efficiency while simultaneously decreasing accidents rate.

Future work may include enriching the behavior state switching logic to accommodate faults and automated driving for the hierarchical hybrid system model. In the traffic simulators development, more extensive testing and validation for a variety of driver behavior and traffic conditions are needed. The introduction and deployment for the driver assistance systems into the real life need to address the problem originated from the coexistence of IVC equipped and non-equipped vehicles.
APPENDIX A

VEHICLE INTERSECTION TRAFFIC SIMULATOR
USER MANUAL

A.1 Introduction

This document describes how to use the Vehicle Intersection Traffic Simulator (VITS) software, version 1.0. The Vehicle Intersection Traffic Simulator V1.0 is a real-time traffic simulator developed using Visual C++ language and environment. It is a derivative of the VTS simulator used in

VITS is a framework for simulation, evaluation and analysis of an urban intersection. It considers traffic flow, turns and possible accidents. A preliminary version of decision-making for turning is included. Real time simulation is presented graphically in the client area and trace files are generated after each simulation run for further analysis.

A.1.1 System Design Specifications:

- User Friendly Graphical User Interface (GUI);
- Mouse Driven;
- Easy Install/Uninstall;
• Adjustable Simulation Speed;

• Real-time Traffic Movement;

• Facilitate both micro- and macro- study of intersection traffic;

• Adjustable traffic flow, speed limit, possible to differentiate primary and secondary road;

• Detail simulation data exported in files for further analysis

• Simple human model is included (choices of turn decisions, etc.)

A.2 Installation

A.2.1 System Requirement:

• Operating System: Windows® XP

• CPU: PENTIUM® III 1.1 GHz

• Free Hard Disk Space: 50MB

• Memory: 256M RAM

• Peripherals: Keyboard and mouse

A.2.2 Install:

Copy all following files under the same directory: VITS.exe (file)

A.2.3 Uninstall:

VITS can be uninstalled by removing the following file: VITS.exe (file)
A.3 Start

A.3.1 Getting Started

Double click the VITS icon as shown in Figure A.1. The “Client Area Window” as shown in Figure A.2 will pop out.

![VITS Icon](VITS.png)

Figure A.1: Vehicle Intersection Traffic Simulator Icon

A.3.2 Developing Environment

1. **Client Area (Figure A.2:)** The VITS Client Area can be divided into a control area and the display area as shown in Figure A.2. In the control area, simulation parameters can be selected and the simulation process can be controlled. Real time traffic flow at the intersection will appear in the display area when simulation is initiated.

2. **Pre-Settings (Figure A.3:)**

   Click the “PreSettings” under “Data” in the main frame or the “S” on the toolbar, (Figure A.3). The Input Area Dialog Box will pop up (Figure A.4).

3. **Vehicle Intersection Traffic Simulator Input Area Dialog Box (Figure A.4):** The Input Area Dialog Box (Figure A.4) is used to specify the
Figure A.2: Client Area of Vehicle Intersection Traffic Simulator

Figure A.3: Start of the Input Area Dialog Box
Figure A.4: The Input Area Dialog Box
simulation parameters. Press “OK” when insertion or selection of data is complete. The user will be redirected to the client area.

- **Scenarios**: *Signalized Intersection/ Unsignalized Intersection (W/O Traffic Lights)*

- **Traffic Flow**: *Vehicles/hour (integer between 0 and 10800)*

  The user can set traffic flow for the North/South and the East/West streets. Table A.1 and Table A.2 show the relationship between traffic flow and the level of service for this four-lane intersection area. The user can use those tables to define a reasonable value for the traffic flow. Please note that the traffic flow in Table A.1 and Table A.2 is the total input traffic flow of the entire intersection.

<table>
<thead>
<tr>
<th>Level of Service</th>
<th>Computed Flow Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>( \leq 2592 )</td>
</tr>
<tr>
<td>B</td>
<td>( \leq 4147 )</td>
</tr>
<tr>
<td>C</td>
<td>( \leq 6260 )</td>
</tr>
<tr>
<td>D</td>
<td>( \leq 8294 )</td>
</tr>
<tr>
<td>E</td>
<td>( \leq 18662 )</td>
</tr>
<tr>
<td>F</td>
<td></td>
</tr>
</tbody>
</table>

Table A.1: Level of Service and Flow Rate (Unsignalized Intersection)

- **Speed Limit**: *Miles/hour (any number between 0 and 100)*

  Speed limit specifies the speed limit for the two roads at the intersection. Therefore, based on the different combination of choices of Traffic flow and
Table A.2: Level of Service and Flow Rate (Signalized Intersection)

<table>
<thead>
<tr>
<th>Level of Service</th>
<th>Computed Flow Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>( \leq 4013 ) (unsaturated)</td>
</tr>
<tr>
<td>C, D, E, F</td>
<td>oversaturated</td>
</tr>
</tbody>
</table>

Speed limit, VITS can simulate an intersection with primary and secondary road.

- **Vehicle Num**: Vehicle numbers (integer between 0 and 10000)
  
  Total number of vehicles that wants to be simulated

- **Turn Ratio**: Percentage of vehicles to make a right turn and left turn (Any number between 0 and 1)
  
  The sum of left turn and right turn of the N→S or the E→W streets should be always between 0 and 1.

- **Turn Decision**: Drag the slider between two end points.
  
  No decision means vehicles would like to continue its route regardless of other vehicles. Vehicles in range means vehicles will stop to avoid possible collisions with all other vehicles in the visual range. The slider indicates the percentage of vehicles in range.

- **Visual Range**: Any number between 0 and 100 m

- **Simulation Speed**: Quick/medium/slow
  
  Simulation Speed defines the real time display speed in the client area.
Buttons:

- **OK**: Close the dialog box and back to client area. The values were input to simulator.
- **Cancel**: Close the dialog box and back to client area. The values were not input to simulator
- **Reset**: Reset to its default values.
  - **Scenarios**: Unsignalized
  - **Intersection Traffic Flow**: 1800 vehicles/hour for N→S 1800 vehicles/hour for E→W
  - **Speed Limit**:
    40 miles/hour for N→S 40 miles/hour for E→W
  - **Vehicle Num**: 10 vehicles
  - **Turn Ratio**:
    N→S: Left turn: 0.2 Right turn: 0.1 E→W: Left turn: 0.2 Right turn: 0.3
  - **Turn Decision**: 100% No Decision
  - **Visual Range**: 30m
  - **Simulation Speed**: medium

4. **Starting a Vehicle Intersection Traffic Simulation (Figure A.5)**: After setting the parameters, the user can start the vehicle traffic simulation either by clicking the “start” button under the “process” in the main frame or the button on the toolbar (Figure A.5).
Figure A.5: Starting a Vehicle Intersection Traffic Simulation
5. **Other Simulation Process Controls**: Other simulation process controls are listed as the following and shown in Figure A.6.

- **Start**: Begin the simulation
- **Pause**: Pause the simulation when running. Click it again will continue the simulation. (Pause/Play)
- **Stop**: Stop the simulation and exit the entire environment

![Figure A.6: Cursor for Simulation Process Controls](image)

6. **Screenshot of Vehicle Intersection Traffic Simulator (Figure 4.4)**:

7. **To do another run with different settings**: If the simulation is still running, stop the simulation first by clicking “stop” under “process” or “stop” button in the toolbar (Figure A.6), then click “PreSettings” under “Data” in the main
frame or the “S” on the toolbar (Figure A.3). If the simulation is finished, click “PreSettings” under “Data” in the main frame or the “S” on the toolbar (Figure A.3).

### A.3.3 Output files

Five files will be generated and saved in the VITS folder after simulation for further analysis. Data is appended after one run

- vehicle.txt
- xposition.txt
- yposition.txt
- velocity.txt
- acceleration.txt

- **vehicle.txt**: Format: Label Time Source Turning Destination Type Length Width Height TurnDecision

  - *Time*: the time of vehicle’s presence in the intersection
  - *Source*: source lane (Figure A.7)
  - *Turning*: 1: turn left 2: go straight 3: turn right
  - *Destination*: destination lane (Figure A.7)
  - *Type*: driver type, 0: conservative, 1: normal, 2: aggressive
  - *Length*: vehicle length
  - *Width*: vehicle width
  - *Height*: Vehicle height
- *TurnDecision*: Vehicle’s Turn Decision

Data from different runs are separated by “#id time source dir dest type length width height turn_decision”

Figure A.7: Lane label of Vehicle Intersection Traffic Simulator

- **xposition.txt**: Format: simtime Vehicle 1 xposition Vehicle 2 xposition Vehicle 3 xposition . . . .

  999999.000000 means out of intersection area Data from different runs are separated by “ time xposition”

- **yposition.txt**: Format: simtime Vehicle 1 yposition Vehicle 2 yposition Vehicle 3 yposition . . . .
999999.000000 means out of intersection area Data from different runs are separated by “time yposition”

- **velocity.txt**: Format: simtime Vehicle 1 velocity Vehicle 2 Velocity Vehicle 3 Velocity......

999999.000000 means out of intersection area Positive velocity means the vehicle is moving along north/south and negative velocity means the vehicle is moving along west/east. Data from different runs are separated by “time velocity”

- **acceleration.txt**: Format: simtime Vehicle 1 acceleration Vehicle 2 acceleration Vehicle 3 acceleration......

999999.000000 means out of intersection area Data from different runs are separated by “time acceleration”
# APPENDIX B

## LIST OF SYMBOLS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a^i$</td>
<td>vehicle $i$’s acceleration</td>
<td>3</td>
</tr>
<tr>
<td>$a'^i$</td>
<td>acceleration lower limit of vehicle $i$</td>
<td>35</td>
</tr>
<tr>
<td>$a''^i$</td>
<td>acceleration upper limit of vehicle $i$</td>
<td>35</td>
</tr>
<tr>
<td>$a_x^i$</td>
<td>vehicle $i$’s acceleration along x-axis</td>
<td>3</td>
</tr>
<tr>
<td>$a_y^i$</td>
<td>vehicle $i$’s acceleration along y-axis</td>
<td>3</td>
</tr>
<tr>
<td>$v^i$</td>
<td>vehicle $i$’s velocity along y-axis</td>
<td>3</td>
</tr>
<tr>
<td>$v^i_d$</td>
<td>desired velocity at time $t$ of vehicle $i$</td>
<td>34</td>
</tr>
<tr>
<td>$v_x^i$</td>
<td>vehicle $i$’s velocity along x-axis</td>
<td>3</td>
</tr>
<tr>
<td>$v_y^i$</td>
<td>vehicle $i$’s velocity along y-axis</td>
<td>3</td>
</tr>
<tr>
<td>$w_o$</td>
<td>weighting parameter, importance of obstacles</td>
<td>38</td>
</tr>
<tr>
<td>$w_p$</td>
<td>weighting parameter, importance of route accuracy</td>
<td>35</td>
</tr>
<tr>
<td>$w_v$</td>
<td>weighting parameter, importance of speed accuracy</td>
<td>35</td>
</tr>
<tr>
<td>$w_{vv}$</td>
<td>weighting parameter, importance of vehicle</td>
<td>50</td>
</tr>
<tr>
<td>$x_x^i$</td>
<td>vehicle $i$’s mass point position along x-axis</td>
<td>3</td>
</tr>
<tr>
<td>$x_y^i$</td>
<td>vehicle $i$’s mass point position along y-axis</td>
<td>3</td>
</tr>
<tr>
<td>$I_d$</td>
<td>danger indicator</td>
<td>32</td>
</tr>
<tr>
<td>$L^i$</td>
<td>vehicle $i$’s length</td>
<td>36</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
</tbody>
</table>
| $R^i$  | desired route of vehicle $i$  
| $S^i$  | discrete state at behavior layer for vehicle $i$  
| $U^i$  | discrete input from the interface for vehicle $i$  
| $W^i$  | vehicle $i$’s width  
| $X^i$  | vehicle $i$’s state  
| $Y^i$  | discrete output from the DES for vehicle $i$  
| $\alpha^i$ | vehicle $i$’s heading angle  
| $\omega^i$ | vehicle $i$’s angular velocity  
| $d(p, l)$ | minimum distance from point $p$ to the line $l$  
| $d_p(p_1, p_2)$ | distance between $p_1$ and $p_2$  
| $g^i$  | shape and position of vehicle $i$  
| $\text{obs}^i$ | shape and position of obstacle $i$  
| $C(f(\cdot), g(\cdot))$ | overlapping area defined by $f$ and $g$  
| $I(f, g)$ | intersection point of line defined by function $f$ and $g$  
| $\text{Quad}(a, b, c, d)$ | quadrangle defined by the four endpoints $a, b, c,$ and $d$  

168
## APPENDIX C

### LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACC</td>
<td>Adaptive Cruise Control</td>
<td>11</td>
</tr>
<tr>
<td>AHS</td>
<td>Automated Highway System</td>
<td>10</td>
</tr>
<tr>
<td>Alg</td>
<td>Algebraic Approach</td>
<td>7</td>
</tr>
<tr>
<td>Aut</td>
<td>Automata and Transition Systems</td>
<td>7</td>
</tr>
<tr>
<td>COSMODRIVE</td>
<td>Cognitive Simulation Model of the Driver</td>
<td>18</td>
</tr>
<tr>
<td>CTG</td>
<td>Constant Time Gap</td>
<td>127</td>
</tr>
<tr>
<td>CTS</td>
<td>Continuous Time System</td>
<td>8</td>
</tr>
<tr>
<td>DES</td>
<td>Discrete Event System</td>
<td>29</td>
</tr>
<tr>
<td>FHWA</td>
<td>Federal Highway Administration</td>
<td>55</td>
</tr>
<tr>
<td>FRP</td>
<td>Formation Reconfiguration Planning</td>
<td>9</td>
</tr>
<tr>
<td>FSLQ</td>
<td>Frequency Shaped Linear Quadratic</td>
<td>13</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
<td>28</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
<td>70</td>
</tr>
<tr>
<td>HMMs</td>
<td>Hidden Markov Models</td>
<td>20</td>
</tr>
<tr>
<td>INRETS</td>
<td>Institute for Transportation Research and their Safety</td>
<td>18</td>
</tr>
<tr>
<td>ITS</td>
<td>Intelligent Transportation Systems</td>
<td>7</td>
</tr>
<tr>
<td>IVC</td>
<td>Inter-vehicle Communication</td>
<td>1</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>IWS</td>
<td>Intersection Warning System</td>
<td>70</td>
</tr>
<tr>
<td>LOS</td>
<td>Level of Service</td>
<td>87</td>
</tr>
<tr>
<td>MAC</td>
<td>Media Access Control</td>
<td>106</td>
</tr>
<tr>
<td>MITSIM</td>
<td>Microscopic Traffic Simulator</td>
<td>55</td>
</tr>
<tr>
<td>MVTS</td>
<td>Merging Vehicle Traffic Simulator</td>
<td>58</td>
</tr>
<tr>
<td>ODE</td>
<td>Ordinary Differential Equations</td>
<td>7</td>
</tr>
<tr>
<td>PATH</td>
<td>Partners for Advanced Transit and Highways</td>
<td>13</td>
</tr>
<tr>
<td>POV</td>
<td>Principal Other Vehicle</td>
<td>81</td>
</tr>
<tr>
<td>PPLT</td>
<td>Protected and Permitted Left-turn</td>
<td>22</td>
</tr>
<tr>
<td>Prog</td>
<td>Programming Languages</td>
<td>7</td>
</tr>
<tr>
<td>RVC</td>
<td>Road-vehicle Communication</td>
<td>1</td>
</tr>
<tr>
<td>TTA</td>
<td>Time to Avoidance</td>
<td>110</td>
</tr>
<tr>
<td>TTC</td>
<td>Time-To-Collision</td>
<td>110</td>
</tr>
<tr>
<td>SAACC</td>
<td>Semi-Autonomous Adaptive Cruise Control</td>
<td>12</td>
</tr>
<tr>
<td>SSTS</td>
<td>Single Street Traffic Simulator</td>
<td>56</td>
</tr>
<tr>
<td>SV</td>
<td>Subject Vehicle</td>
<td>81</td>
</tr>
<tr>
<td>VATSIM</td>
<td>Vehicle and Traffic Simulator</td>
<td>55</td>
</tr>
<tr>
<td>VITS</td>
<td>Vehicle Intersection Traffic Simulator</td>
<td>56</td>
</tr>
<tr>
<td>VSCC</td>
<td>Vehicle Safety Communication Consortium</td>
<td>138</td>
</tr>
<tr>
<td>VTS</td>
<td>Vehicle Traffic Simulator</td>
<td>103</td>
</tr>
<tr>
<td>WS</td>
<td>Wireless Simulator</td>
<td>70</td>
</tr>
</tbody>
</table>

170
BIBLIOGRAPHY


175


176


