



HAL
open science

Control and optimization for intelligent transportation systems in vicinity of intersections

Bing Liu

► **To cite this version:**

Bing Liu. Control and optimization for intelligent transportation systems in vicinity of intersections. Automatic. Ecole Centrale de Lille, 2016. English. NNT : 2016ECLI0008 . tel-01486731

HAL Id: tel-01486731

<https://theses.hal.science/tel-01486731v1>

Submitted on 10 Mar 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre :

| | | |
|---|---|---|
| 2 | 9 | 8 |
|---|---|---|

CENTRALE LILLE

THÈSE

présentée en vue d'obtenir le grade de

DOCTEUR

Spécialité : Automatique, génie informatique, traitement du signal et des images

par

LIU Bing

Master of Engineering of Beijing University of Aeronautics and Astronautics (BUAA)

Master de Sciences et Technologies de l'École Centrale de Lille

Doctorat délivré par Centrale Lille

**Contrôle et Optimisation des Systèmes de Transport Intelligents
dans le Voisinage des Intersections**

Soutenue le 9 septembre 2016 devant le jury :

| | | |
|------------------------|---------------------------------------|--------------------|
| M. Pierre BORNE | Ecole Centrale de Lille | Président |
| M. Nouredine ELLOUZE | Ecole Nationale d'Ingénieurs de Tunis | Rapporteur |
| M. Dumitru POPESCU | Université Polytechnique de Bucarest | Rapporteur |
| M. Ahmed EL HAJJAJI | Université de Picardie Jules Vernes | Examineur |
| M. Abdelkader EL KAMEL | Ecole Centrale de Lille | Directeur de Thèse |
| Mme. Shaoping WANG | Université de Beihang, Chine | Examineur |
| M. Jin ZHAO | Université de Guizhou, Chine | Examineur |

Thèse préparée dans le Centre de Recherche en Informatique, Signal et Automatique de Lille
CRISAL - UMR CNRS 9189 - École Centrale de Lille
École Doctorale Sciences pour l'Ingénieur - 072

Serial N° :

| | | |
|---|---|---|
| 2 | 9 | 8 |
|---|---|---|

CENTRALE LILLE

THESIS

presented to obtain the degree of

DOCTOR

Topic : Automatic control, computer engineering, signal and image processing

by

LIU Bing

Master of Engineering of Beijing University of Aeronautics and Astronautics (BUAA)

Master of Science and Technology of Ecole Centrale de Lille

Ph.D. awarded by Centrale Lille

**Control and Optimization for Intelligent Transportation Systems
in Vicinity of Intersections**

Defended on September 9, 2016 in presence of the committee :

| | | |
|-------------------------|---------------------------------------|----------------|
| Mr. Pierre BORNE | Ecole Centrale de Lille | President |
| Mr. Nouredine ELLOUZE | Ecole Nationale d'Ingénieurs de Tunis | Reviewer |
| Mr. Dumitru POPESCU | Université Polytechnique de Bucarest | Reviewer |
| Mr. Ahmed EL HAJJAJI | Université de Picardie Jules Vernes | Examiner |
| Mr. Abdelkader EL KAMEL | Ecole Centrale de Lille | PhD Supervisor |
| Mrs. Shaoping WANG | Beihang University, China | Examiner |
| Mr. Jin ZHAO | Guizhou University, China | Examiner |

Thesis prepared within the Centre de Recherche en Informatique, Signal et Automatique de Lille
CRIStAL - UMR CNRS 9189 - École Centrale de Lille
École Doctorale Sciences pour l'Ingénieur - 072

*To my parents,
to all my family,
to my professors,
and to all my friends.*

Acknowledgements

This research work has been realized at “Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIS_TAL)” in École Centrale de Lille, with the research group “Optimisation : Modèles et Applications (OPTIMA)” from September 2013 to September 2016. This work is financially supported by China Scholarship Council (CSC). Without the founding of CSC, I would never have this valuable living and studying experience in France.

First and foremost I offer my sincerest gratitude to my PhD supervisor, Prof. Abdelkader EL KAMEL, who has provided his supervision, valuable guidance, continuous encouragement as well as given me extraordinary experiences throughout my Ph.D. experience.

Besides my supervisor, I would like to thank Prof. Pierre BORNE for his kind acceptance to be the president of my PhD Committee. I would also like to express my sincere gratitude to Prof. Nouredine ELLOUZE and Prof. Dumitru POPESCU, who have kindly accepted the invitation to be reviewers of my PhD thesis, for their encouragement, insightful comments and helpful questions. My gratitude to Prof. Ahmed EL HAJJAJI, Prof. Shaoping WANG and Prof. Jin ZHAO, for their kind acceptance to take part in the jury of the PhD defense.

I am also very grateful to the staff in École Centrale de Lille. Vanessa FLEURY, Brigitte FONCEZ and Christine YVOZ have helped me in the administrative work. Many thanks go also to Patrick GALLAIS, Gilles MARGUERITE and Jacques LASUE, for their kind help and hospitality. Special thanks go to Christine VION, Martine MOUVAUX for their support in my lodgement life.

My sincere thanks also goes to Dr. Tian ZHENG, Dr. Yue YU, Dr. Daji TIAN and Dr. Chen XIA, for offering me the useful advices during my study in the laboratory as well as after their graduation.

ACKNOWLEDGEMENTS

I would like to take the opportunity to express my gratitude and to thank my fellow workmates in CRIStAL: Yihan LIU, Qi SUN, Jian ZHANG for the stimulating discussions for the hard teamwork. Also I wish to thank my friends and colleagues: Yu DU, Qi GUO, Tong LI, Xiaofeng YI, Yaqian WEI, Zhiping YE, Hongchang ZHANG, Lijie BAI, Jing BAI, Ben LI, Xiaokun DING, etc., for all the fun we have had in the past three years. All of them have given me support and encouragement in my thesis work. Special thanks to Mengnan LU, for her accompany, patience, and encouragement.

All my gratitude goes to Ms. H el ene CATSIAPIS, my French teacher, who showed us the French language and culture. She organized some interesting and unforgettable voyages in France, which inspired my knowledge and interest in the French culture, opened my appetite for art, history and cuisine/wine and enriched my life in France.

My acknowledgements to all the professors and teachers in  cole Centrale de P ekin, Beihang University. The engineer education there not only gave me solid knowledge but also made it easier for me to live in France.

A special acknowledgment should be shown to Prof. Haibin DUAN at the School of Automation Science and Electrical Engineering, Beihang University, who enlightened me at the first glance of research. I always benefit from the abilities that I obtained on his team.

Last but not least, I convey special acknowledgement to my parents, Huaqing LIU and Hui ZHAO, for supporting me to pursue this degree and to accept my absence for four years of living abroad.

Villeneuve d'Ascq, France
June, 2016

LIU Bing

Contents

| | |
|--|-------------|
| Acknowledgements | i |
| Table of Contents | iii |
| List of Figures | vii |
| List of Tables | ix |
| List of Algorithms | x |
| Abbreviations | xiii |
| General Introduction | 1 |
| 1 Introduction to ITS | 7 |
| 1.1 Background | 8 |
| 1.1.1 Motivation | 8 |
| 1.1.2 Intelligent Transportation Systems | 9 |
| 1.1.2.1 Definition of ITS | 9 |
| 1.1.2.2 Historical Research | 10 |
| 1.1.3 Self-driving cars | 12 |
| 1.2 Vehicular Ad hoc Networks | 15 |
| 1.2.1 Opportunities and challenges | 15 |
| 1.2.2 Dedicated short range communications | 18 |
| 1.2.3 Applications in VANET | 19 |
| 1.2.3.1 Vehicle-to-infrastructure applications | 19 |
| 1.2.3.2 Vehicle-to-vehicle applications | 20 |
| 1.2.3.3 An instance of VANET application | 21 |

CONTENTS

| | | |
|----------|--|-----------|
| 1.3 | Cooperative Adaptive Cruise Control | 23 |
| 1.3.1 | Evolution: from autonomous to cooperative | 23 |
| 1.3.2 | Development of ACC | 24 |
| 1.3.3 | Related work in CACC | 26 |
| 1.4 | Autonomous Intersection Management | 28 |
| 1.4.1 | Isolated Intersection Control | 29 |
| 1.4.2 | Motion Planning at Intersection | 30 |
| 1.5 | Conclusion | 31 |
| 2 | Decentralized Platoon Control | 33 |
| 2.1 | Introduction | 33 |
| 2.2 | Modeling | 36 |
| 2.2.1 | Vehicle kinematic model | 36 |
| 2.2.2 | Platoon model | 37 |
| 2.3 | PSO based Platoon control | 40 |
| 2.3.1 | Cost function design | 40 |
| 2.3.2 | Optimization algorithm | 42 |
| 2.4 | Simulation of algorithm | 45 |
| 2.5 | Conclusion | 48 |
| 3 | Improving Traffic Efficiency via V2X and CACC | 49 |
| 3.1 | Introduction | 49 |
| 3.2 | Problem statement | 51 |
| 3.2.1 | Scenario description | 51 |
| 3.2.2 | Design objectives | 53 |
| 3.3 | Methodology | 54 |
| 3.3.1 | Overview of CACC-VI | 54 |
| 3.3.2 | Platoon Reorganization | 56 |
| 3.3.2.1 | Trajectory planning algorithm | 56 |
| 3.3.2.2 | Space arrangement approach | 59 |
| 3.3.3 | Mixed Control Strategy | 65 |
| 3.4 | Simulation Experiments | 66 |
| 3.4.1 | Setup | 66 |
| 3.4.2 | Simulation results | 68 |
| 3.4.2.1 | Platoon reorganization | 68 |

| | | |
|----------|--|------------|
| 3.4.2.2 | Accelerating platoon | 68 |
| 3.4.2.3 | Decelerating platoon | 73 |
| 3.5 | Conclusion | 75 |
| 4 | Decentralized Autonomous Intersection Control for Light Traffic | 77 |
| 4.1 | Introduction | 78 |
| 4.2 | Intersection model and Design objectives | 80 |
| 4.2.1 | Intersection | 80 |
| 4.2.2 | Assumption and Objectives | 81 |
| 4.3 | Algorithm Design | 82 |
| 4.3.1 | Collision Definition | 82 |
| 4.3.1.1 | Cross Collision | 82 |
| 4.3.1.2 | Rear-end Collision | 83 |
| 4.3.2 | Set Partition | 84 |
| 4.3.3 | Priority Setting | 85 |
| 4.3.4 | Cost Function | 86 |
| 4.4 | UML-based Testbed Design | 88 |
| 4.4.1 | Static Design | 88 |
| 4.4.2 | Dynamic Design | 91 |
| 4.5 | Simulation | 93 |
| 4.5.1 | Parameter Settings | 93 |
| 4.5.2 | Simulation Results | 95 |
| 4.6 | Conclusion | 97 |
| 5 | Trajectory Planning for Autonomous Intersection Management | 101 |
| 5.1 | Introduction | 101 |
| 5.2 | Assumptions and Design objectives | 103 |
| 5.3 | Protocol Design | 104 |
| 5.3.1 | Overview | 104 |
| 5.3.2 | Definitions | 105 |
| 5.3.2.1 | Conflict Zone Occupation Time | 105 |
| 5.3.2.2 | Vehicle set notations | 107 |
| 5.3.2.3 | Service Priority | 107 |
| 5.3.3 | Window Searching | 110 |
| 5.3.4 | Trajectory Planning | 113 |

CONTENTS

| | | |
|---------|-------------------------------------|------------|
| 5.3.4.1 | With preceding vehicle | 114 |
| 5.3.4.2 | Without preceding vehicle | 117 |
| 5.4 | Simulation | 119 |
| 5.4.1 | Settings | 119 |
| 5.4.2 | Results | 120 |
| 5.5 | Conclusion | 124 |
| | Conclusions and Perspectives | 127 |
| | Résumé Étendu en Français | 133 |
| | References | 143 |

List of Figures

| | | |
|------|---|----|
| 1.1 | Conceptual Image of Intelligent Transportation System | 9 |
| 1.2 | Instance for road ITS system layout | 11 |
| 1.3 | Stanley at Grand Challenge 2005 | 12 |
| 1.4 | Self-driving cars | 14 |
| 1.5 | Sensors on Google's self-driving car (Source www.aventurine.com) | 14 |
| 1.6 | Vehicular Ad hoc Networks (Source johndayautomotiveelectronics.com) | 16 |
| 1.7 | DSRC spectrum band and channels in the U.S. (Jiang & Delgrossi (2008)) | 18 |
| 1.8 | An instance of VANET-based application development (Source: www.mobility.siemens.com) | 22 |
| 1.9 | Structure of longitudinal control system | 25 |
| 1.10 | Vehicle platoon in GCDC 2011 | 27 |
| 2.1 | Platoon illustration | 37 |
| 2.2 | Spacing error in the platoon | 46 |
| 2.3 | Velocity error in the platoon | 47 |
| 2.4 | Acceleration error in the platoon | 47 |
| 2.5 | Velocities of vehicles | 48 |
| 3.1 | Scenario illustration | 52 |
| 3.2 | Overview of CACC-VI | 55 |
| 3.3 | An example for space arrangement approach | 65 |
| 3.4 | Motion of accelerating platoon G'_2 | 71 |
| 3.5 | Tracking errors in the platoon G'_2 | 72 |
| 3.6 | Control inputs of vehicles in the platoon G'_2 | 73 |
| 3.7 | Motion of decelerating platoon G'_3 | 74 |

LIST OF FIGURES

| | | |
|------|--|-----|
| 4.1 | 6-lane 4-leg intersection | 81 |
| 4.2 | Class diagram of the testbed | 88 |
| 4.3 | Simulation using Matlab and SUMO | 90 |
| 4.4 | Sequence diagram of the testbed | 91 |
| 4.5 | Statechart diagram of a vehicle object | 92 |
| 4.6 | Statechart diagram of a vehicle in tuning mode | 92 |
| 4.7 | Evacuation time comparison | 96 |
| 4.8 | Delay in different scenarios | 98 |
| 4.9 | Throughput comparison | 99 |
| 4.10 | Screen shots at $t = 70s$ using traffic lights and DAIC | 99 |
| 5.1 | Overview of TP-AIM | 106 |
| 5.2 | Phases for 4 serving groups | 108 |
| 5.3 | Intersection illustration | 109 |
| 5.4 | Conflict zone occupancy | 110 |
| 5.5 | Window Searching for a vehicle going straight | 111 |
| 5.6 | Planning trajectory using Dynamic programming | 115 |
| 5.7 | Evacuation time comparison | 121 |
| 5.8 | Throughput comparison at $t = 98s$ | 121 |
| 5.9 | Delay Comparison | 122 |
| 1 | La voiture sans conducteur “Bubble” de Google | 135 |
| 2 | Réseaux ad hoc des véhicules au voisinage d’une intersection | 136 |
| 3 | La topologie de communication dans la caravane | 137 |
| 4 | L’aperçu de l’algorithme CACC-VI | 138 |
| 5 | Le diagramme de TP-AIM | 140 |

List of Tables

| | | |
|-----|--|-----|
| 2.1 | Vehicles' parameters setting. | 45 |
| 3.1 | Upstream message | 60 |
| 3.2 | Downstream message | 60 |
| 3.3 | Vehicles' parameters setting. | 67 |
| 3.4 | Platoon reorganization of CACC-VI algorithm. | 69 |
| 4.1 | Different scenarios in the simulation. | 95 |
| 4.2 | Traffic flow composition. | 95 |
| 5.1 | Arriving rate in different scenarios. | 120 |
| 5.2 | Attempt times and computation time. | 124 |

LIST OF TABLES

List of Algorithms

| | | |
|---|---|-----|
| 1 | Pseudo Code of PSO | 44 |
| 2 | Vehicle's response to a broad-casted message | 60 |
| 3 | Vehicle's response to an upstream message | 62 |
| 4 | Vehicle's response to a downstream message | 63 |
| 5 | Vehicle V_i 's strategy in DAIC at each time step | 94 |
| 6 | Trajectory planning in a segmentation using DP | 118 |

LIST OF ALGORITHMS

Abbreviations

AIM - Autonomous Intersection Management

CACC - Cooperative Adaptive Cruise Control

CACC-VI - Cooperative Adaptive Cruise Control in Vicinity of Intersection

CZOT - Conflict Zone Occupancy Time

DAIC - Decentralized Autonomous Intersection Control

DP - Dynamic Programming

DSRC - Dedicated Short-Range Communications

DP - Dynamic Programming

EBW - Electronic braking warning

ITS - Intelligent Transportation Systems

IVW - Intersection Violation Warning

LCW - Lane Change Warning

OOT - Overlapped Occupancy Time

OTW - On-coming Traffic Warning

PSO - Particular Swarm Optimization

SA - Speed Advisory

TIS - Traffic Information Systems

ABBREVIATIONS

TP-AIM - Trajectory Planning based Autonomous Intersection Management

UML - Unified Modeling Language

VANETs - Vehicular Ad hoc Networks

V2V - Vehicle-to-Vehicle

V2I - Vehicle-to-Infrastructure

V2P - Vehicle-to-Pedestrian

V2C - Vehicle-to-Cloud

General Introduction

Scope of the thesis

This thesis is dedicated to research the application of intelligent control theory in the future road transportation systems. With the development of human society, the demand for transportation is much stronger than any other period in history. More flexible and more comfortable, private cars are preferred by many people. Besides, the development of automobile industry reduces the cost to own a car, thus car ownership has been growing rapidly worldwide, especially in major cities. However, the increase of car number makes our society to suffer from traffic congestions, exhaust pollution and accidents. These negative effects force people to find ways out. In this context, the concept of “Intelligent Transportation Systems (ITS)” is proposed. Scientists and engineers have been working for decades to apply multidisciplinary technologies to transportation, in order to make it closer to our vision, such as safer, more efficient, more effort saving, and environmentally friendly.

One thinking is (semi-)autonomous systems. The main idea is to use autonomous applications to assist/replace human operation and decision. Advanced driver assistance systems (ADAS) are designed to assist drivers by alerting them when danger (e.g. lane keeping, forward collision warning), acquiring more information for decision-making (e.g. route plan, congestion avoidance) and liberating them from repetitive and trick maneuvers (e.g. adaptive cruise control, automatic parking). In semi-automatic systems, driving process still needs the involvement of human driver: the driver should set some parameters in the system, and he/she can decide to follow the advisory assistance or not. Recently, with the improvement of sensing technology and artificial intelligence, enterprises and institutes have been committed to the research and development of autonomous driving.

GENERAL INTRODUCTION

In certain scenarios (e.g. highways and main roads), with the help of accurate sensors and highly precise map, hands-off and feet-off driving experience would be achieved. Elimination of human error will make the road transportation much safer, and better space utilization will improve the usage of road capacity. However, autonomous cars still need driver's anticipation in certain scenarios with complicated traffic situation or limited information. The inner layout of autonomous cars would not be much different from current cars, because steering wheel and pedals are still necessary. The next step of autonomous driving is driver-less driving, in which the car is totally driven by itself. The seat dedicated for driver would disappear and people on board would focus on their own staff. The car-sharing economy behind driver-less cars would be enormous: in the future, people would prefer calling for a driver-less car when needed to owning a private car. Thus congestions and pollutions could be relieved.

Another thinking is cooperative systems. Obviously, the current road transportation notifications are designed for human drivers, such as traffic lights, turning lights and road side signs. The current autonomous vehicles are equipped with cameras dedicated to detect these signs. However, notifications designed for humans is not efficient enough for autonomous vehicles, because the usage of camera is limited by range and visibility, and algorithms should be implemented to recognize these signs. If the interaction between vehicles and environment is enabled, the notifications can be delivered via V2X communications, thus vehicles can be noticed in larger distance even beyond the sight, and the original information is more accurate than the information detected by sensors. When the penetration rate of driver-less cars is high enough, it would not be necessary to have physical traffic lights and signs. The virtual personal traffic sign can be communicated to individual vehicles by the traffic manager. In cooperative systems, an individual does not have to acquire the information all by its own sensors, but with the help of other individuals via communication. Therefore, autonomous intelligence can be extended into cooperative intelligence.

The research presented in this thesis focuses on the development of applications to improve the safety and efficiency for intelligent transportation systems in context of autonomous vehicles and V2X communications. Thus, this research is in the scope of cooperative systems. Control strategy are designed to define the way in which the vehicles interact with each other and with infrastructures.

Main contributions

The main contributions of the thesis are summarized as follows:

- A PSO (particle swarm optimization)-based platoon control algorithm is proposed for following vehicles in the platoon. Constraints like control input, engine power and maximum speed are considered when design the cost function. It is an attempt of applying bionic optimization algorithm in vehicular platoon control. The parameters are regulated to response to potential real-time usage in the future. This algorithm could be used in highways and main city road scenarios.
- Cooperative Adaptive Cruise Control in Vicinity of Intersections (CACC-VI) is designed to improve the intersection's throughput by taking advantage of the "opportunity space" on road. The vehicular platoons are re-organized to accelerating platoon and decelerating platoon based on vehicles' dynamic abilities. Different message types and structures are defined and vehicles' behaviors when receiving these messages are also described. CACC-VI would be practical for urban roads and intersections.
- For intersections without crosswalk, there are no other individuals but vehicles. In this case, except traffic lights, there can have some other management methods with the help of V2X communications. The algorithm named Decentralized Autonomous Intersection Control (DAIC) is designed for light traffic situations, to control vehicles to get through the intersection with no collision and less delay. Before arriving at the intersection, a vehicle demands the space-time occupancy situation from the intersection manager, and based on the motion of the preceding vehicle, it will adjust its own velocity to a collision-free value and it should maintain this velocity until fully getting through the intersection. Basically, DAIC is a enumerative negotiation-based method.

A simulation testbed to test the intersection control algorithm is designed in UML. The structure of the testbed is expressed by a class diagram. And the running process is summarized in sequence diagram and statechart diagram. Then the testbed is realized by the combination of Matlab and a traffic simulator SUMO.

- Based on some ideas in DAIC, we have tried to design an autonomous intersection management algorithm called Trajectory Planning based Autonomous Intersection Management (TP-AIM) for all traffic densities. The main difference between TP-AIM and DAIC is that in TP-AIM, a vehicle's trajectory before entering the intersection is fully planned, and it can make a stop and restart again. Based on the occupancy information from the manager, the vehicle searches for safe entering windows. And according to these windows and the future motion of the preceding vehicle, a dynamic programming based method is used to plan the vehicle's trajectory. The related communication procedure is also described. Compared to traditional and adaptive traffic lights, the efficiency of TP-AIM is greatly improved. The principle of TP-AIM could be used in different types of intersections even in heavy traffic conditions.

Outline of the thesis

This thesis is divided into 5 chapters:

In **Chapter 1**, the development of intelligent transportation systems (ITS) is reviewed. At first, the motivation why we need to add intelligence to the current transportation systems is explained; then historical researches all over the world are presented; as a booming research and development domain, the current situation of autonomous vehicles is introduced. After that, the communication in ITS is presented, and applications constructed based on V2X communications are also described. The development of the two cooperative applications that we studied in this thesis: cooperative adaptive cruise control and autonomous intersection management, is reviewed in details.

In **Chapter 2**, a vehicle platoon control algorithm is designed. The vehicle model and platoon model we use in the chapter is described. Then objective of platoon control is expressed by minimizing three tracking errors: spacing error, velocity error and acceleration error. A particle swarm optimization algorithm is applied to find a near-optimal control input for each following vehicle. The heterogeneity of vehicles and passenger comfort are also considered in the design of objective function.

In **Chapter 3**, the CACC-VI algorithm is expressed in details. The algorithm is designed to improve the throughput of an intersection during the current green light while considering the heterogeneous characteristics and capabilities of vehicles and their fuel consumption. The main idea is to reorganize the platoons to take advantage of the redundant road capacity. A vehicle can choose to accelerate to join in the preceding platoon or decelerate to depart from the current one. Velocity planning algorithm and space arrangement approach are developed to accomplish the reorganization. Control strategies are designed for vehicles in different cases.

In **Chapter 4**, the DAIC algorithm is developed. The intersection layout we studied is described. The conception of conflict zone occupancy time (CZOT) is defined to represent the space-time conflict between vehicles. In order to reduce computation cost, priorities are assigned to vehicles according to their types, routes and distances to the intersection, so that they can make decisions individually in turn. An objective function is designed considering cross collision between vehicles on different lanes and rear-end collision between vehicles on the same lane. The objective of a vehicle is to regulate its velocity until it arrives at a collision-free value, so that in light traffic conditions, vehicles can get through the intersection without come to a stop. The testbed to generate randomized traffic flow and to implement our algorithm is also introduced in this chapter. The static design and dynamic design of the testbed are expressed in several functional diagrams in UML. Different criteria are defined for the evaluation.

In **Chapter 5**, the TP-AIM algorithm is introduced. The first part is window searching: based on the CZOT information stored at the intersection manager, vehicles should search for possible safe entering windows. The second part is trajectory planning: a segmented dynamic programming algorithm is designed to plan the vehicle's trajectory with reduced computation time based on the entering window and the motion of preceding vehicle. Backup mechanism in case the trajectory planning fails is also designed in this chapter to improve the robustness. The algorithm is tested by randomized heterogeneous traffic flows with different densities, and comparisons are conducted with other intersection management protocols.

GENERAL INTRODUCTION

Chapter 1

Introduction to ITS

Contents

| | | |
|------------|--|-----------|
| 1.1 | Background | 8 |
| 1.1.1 | Motivation | 8 |
| 1.1.2 | Intelligent Transportation Systems | 9 |
| 1.1.3 | Self-driving cars | 12 |
| 1.2 | Vehicular Ad hoc Networks | 15 |
| 1.2.1 | Opportunities and challenges | 15 |
| 1.2.2 | Dedicated short range communications | 18 |
| 1.2.3 | Applications in VANET | 19 |
| 1.3 | Cooperative Adaptive Cruise Control | 23 |
| 1.3.1 | Evolution: from autonomous to cooperative | 23 |
| 1.3.2 | Development of ACC | 24 |
| 1.3.3 | Related work in CACC | 26 |
| 1.4 | Autonomous Intersection Management | 28 |
| 1.4.1 | Isolated Intersection Control | 29 |
| 1.4.2 | Motion Planning at Intersection | 30 |
| 1.5 | Conclusion | 31 |

1.1 Background

1.1.1 Motivation

Global economy is highly dependent on the safety and efficiency of transportation systems to move passengers and goods between multiple origins and destinations. Arriving at one place safely in a certain period is the basic requirement of transportation. However, today's road transportation is far from perfect. For a variety of reasons, the current transportation systems cannot satisfy the numerous mobility requirements. Some of the most important problems of current road transport are (Rodrigue (2013)):

- *Accidents and safety.* Increasing traffic have produced growing number of accidents and fatalities. Nearly 1.3 million people die in road crashes each year, on average 3,287 deaths a day, and 20-50 million are injured or disabled. The majority of accidents are caused by incorrect driving behaviors, such as violate regulations, speeding, fatigue driving and drunken driving.
- *Congestion.* Congestion is a very common transport problem in urban agglomerations. It is usually due to the lag between infrastructure construction and the increasing vehicle ownership. Another reasons can be referred to improper traffic light signal, inappropriate road construction and accidents.
- *Environment impacts.* Air pollution and noise pollution are the by-products of road transportation systems, especially in metropolis where vehicles are considerably gathered. Smog brought by vehicles, industries and heating facilities is hurting people's health. The exhaust from incomplete combustion when the vehicle is in congestion is even more pollutant.
- *Loss of public space.* In order to deal with congestion and parking difficulties due to the increasing amount of vehicles, streets are widen and parking areas are built, which seizes the space for public activities like markets, parades and community interactions.

Facing all these drawbacks of current road transportation systems, ameliorations in both system level and individual level need to be investigated.

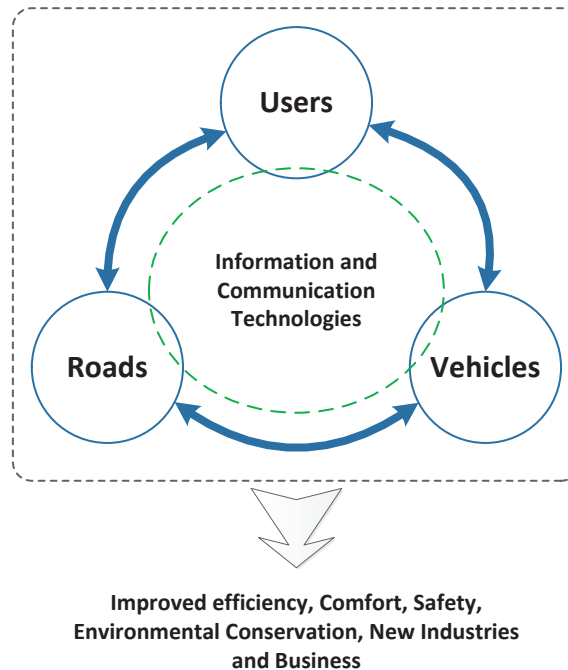


Figure 1.1: Conceptual Image of Intelligent Transportation System

1.1.2 Intelligent Transportation Systems

1.1.2.1 Definition of ITS

Intelligent Transportation Systems (ITS) can be broadly defined as the application of advanced technologies, such as communications, sensors, and computing, to the transportation systems, in order to supply real-time information to improve safety, efficiency and comforts, at the same time reduce the impact to environment (Perng *et al.* (2011)). The development of ITS depends on results from research activities in many different areas such as electronics, control, communications, sensing, robotics, signal processing and information systems. This multidisciplinary nature increases the problem's complexity due to the requirement of knowledge transfer and cooperation among different research domains (Figueiredo *et al.* (2001)). The purpose of ITS is to take advantage of the appropriate technologies to create “more intelligent” roads, vehicles and users. A conceptual Image of ITS can be illustrated in Figure 1.1.

For instance, an architecture of road ITS for commercial vehicles is designed in (Intel (2015)). This system is used to reduce fuel consumption through fuel-

1. INTRODUCTION TO ITS

saving advice, maintain driver and vehicle safety with remote vehicle diagnostics and enable drivers to access information more conveniently. As shown in Figure (1.2), there are three layers in this system:

- *Sensing Layer*: this layer employs a vehicle terminal which enables the interaction between the driver and the vehicle. Meanwhile, it acts as a gateway for in-vehicle technologies and sensors, including microwave detection, speed sensor, RFID, camera, monitoring equipment, etc.
- *Communication Layer*: this layer ensures real-time, secure and reliable transmission from a vehicle to the service layer via different networks, such as 3G/4G, Wi-fi, Bluetooth, wired networks and optical fiber.
- *Service Layer*: in this layer, diverse applications using various technologies are implemented, such as cloud computing, data analytics, information processing and artificial intelligence. Vehicle services are supported by a cloud-based, back-end platform that has a network connection to vehicles and runs advanced data analytic applications. Different categories of services can be supplied, including collision notification, roadside rescue, remote diagnostic, positioning monitoring.

1.1.2.2 Historical Research

The development of ITS in different countries can be divided into two stages (Yan *et al.* (2012)). The first stage is mainly focused on transportation information acquisition and processing intellectualization. During the 70s the CACS (Comprehensive Automobile Traffic Control System) (Koshi (1989)) was developed in Japan, in which several technological programs were conducted to deal with the large number of traffic deaths and injuries as well as the structural ineffective traffic process. In Europe, the first formalized transportation telematics program named PROMETHEUS (Programme for European Traffic with Highest Efficiency and Unprecedented Safety) (Williams (1988)) was initiated by governments, companies and universities in 1986. While in 1988, DRIVE (Dedicated Road Infrastructure and Vehicle Environment) program was set up by the European authorities (Buckley *et al.* (1990)). In the United States, in the late 80s, the team *Mobility 2000* begins the formation of the IVHS (Intelligent Vehicle

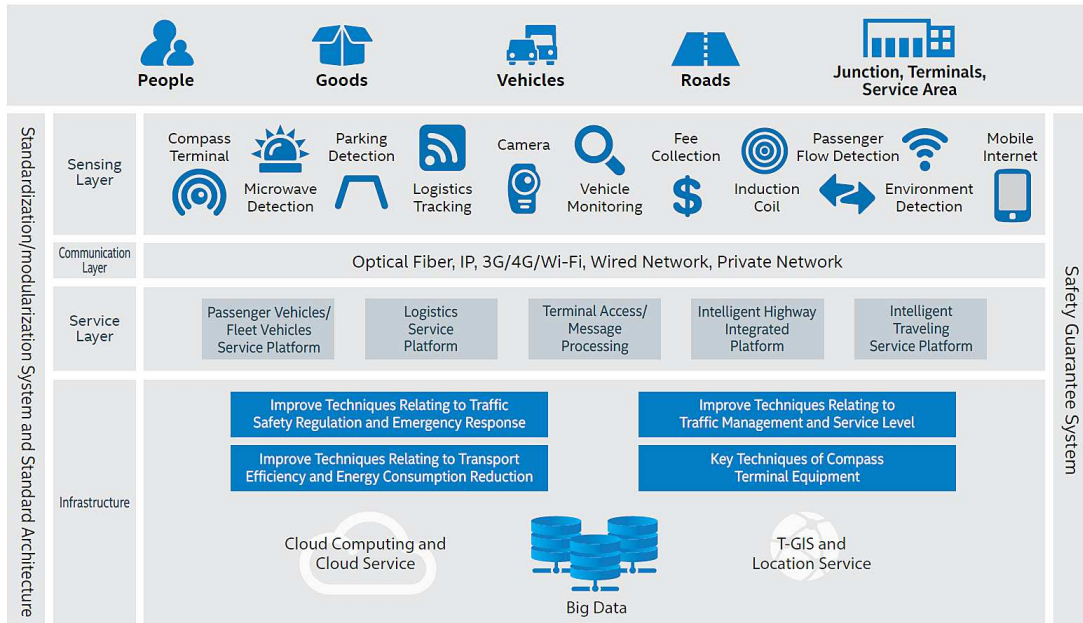


Figure 1.2: Instance for road ITS system layout

Highway Systems) (Betsold (1989)), which is a forum for consolidating ITS interests and promoting international cooperation. In 1994, USDOT (United States Department of Transportation) changed the name to ITS America (Intelligent Transportation Society of America). A key project, AHS (Automated Highway System) was conducted by NAHSC (National Automated Highway System Consortium) formed by the US Department of Transportation, General Motors, University of California and other institutions. Under this project various fully automated test vehicles were demonstrated on California highways (Janet (1992)).

In the second stage, the technologies for vehicle active safety, collision avoidance and intelligent vehicle were rapidly developed. The DEMO' 97 (Ozguner *et al.* (1997)) was the most inspiring project in America. Meanwhile in Europe, ERTICO (European Road Transport Telematics Implementation Coordination Organization) was installed to provide support for refining and implementing the Europe's Transport Telematics Project (Figueiredo *et al.* (2001)). And the organization takes advantage of information and communication to develop active safety and autonomous driving. The Technische Universität at Braunschweig is currently working on the project *Stadtpilot* with the objective to drive fully autonomously on multi-lane ring road around Braunschweig's city (Nothdurft *et al.*

1. INTRODUCTION TO ITS



Figure 1.3: Stanley at Grand Challenge 2005

(2011); Reschka *et al.* (2012); Wille *et al.* (2010)).

In our opinion, the development of ITS is coming to a new stage, where autonomous vehicles, V2X communication and artificial intelligence will be integrated to bring the data acquisition, data processing and decision making into a new level, in which the system is optimized by the cooperation of all the participants of transportation. More details can be referred to the following sections in this chapter.

1.1.3 Self-driving cars

Intelligent vehicles are important roles in ITS, which are motivated by three desires: improved road safety, relieved traffic congestion and comfort driver experience (Sukthankar *et al.* (1998)). The intelligent vehicle senses the environment around them using sensors (such as radar, lidar or machine vision techniques) and strive to achieve more efficient vehicle operation either by assisting the driver (via advisories or warnings) or by taking complete control of vehicle (Baskar *et al.* (2011)).

Since 2003, Defense Advanced Research Projects Agency (DARPA) of USA has found a prize competition “Grand Challenge” to encourage the development of technologies needed to create the first fully autonomous ground vehicles. The Challenge required robotic vehicles to travel a 142-mile long course through the desert within 10 hours. Unfortunately, in the first competition, none of the 15 participants have ever completed more than 5% of the entire course. while in the

second competition in 2005, five of 23 vehicles successfully finished the course, and “Stanley” of Stanford (see Figure 1.3) became the winner with a result of 6 h 53 min (Russell (2006); Thrun *et al.* (2006)). This robotic car was a milestone in the research for modern self-driving cars. Then it comes to the “DARPA Urban Challenge” in 2007. This time the autonomous vehicles should travel 97km through a mock urban environment in less than 6 hours, interacting with other moving vehicles and obstacles and obeying all traffic regulations (Montemerlo *et al.* (2008); Urmson *et al.* (2008)). These vehicles were regarded as the initial prototype of Google self-driving cars.

In 2010, European Research Council sponsored a project: VisLab Intercontinental Autonomous Challenge (VIAC) to build four driver-less vans to accomplish a journey of 13,000 km from Italy to China. The vans have experienced all kinds of road conditions from high-rise urban jungle to broad expanses of Siberia (Broggi *et al.* (2012)).

Google’s self-driving car project is well-known in world wide and is considered to be currently the most successful project in the domain (Guizzo (2011)) (see in Figure 1.4(a)). On the top of the car, a laser is installed to generate a detailed 3D map of the environment. The car then combines the laser measurements with high-resolution maps of the world, producing different types of data models that allow it to drive itself while avoiding obstacles and respecting traffic laws. Other sensors are installed on board, which include: four radars, mounted on the front and rear bumpers, that allow the car to “see” far enough to be able to deal with fast traffic on freeways; a camera, positioned near the rear-view mirror, that detects traffic lights; and a GPS, inertial measurement unit, and wheel encoder, that determine the vehicle’s location and keep track of its movements. When road test, an engineer sits behind the steering wheel to take over if necessary. More details in terms of sensors can be seen in Figure 1.5. A more radical solution is also proposed by Google: a self-driving car without steering wheel or pedals, just an emergency button (see in Figure 1.4(b)).

It should be indicated that Google’s approach relies on very detailed maps of the roads and terrain to determine accurately where the car is, because usually the GPS has errors of several meters. And before the road test, the car is driven by human one or more times to gather environment data, then a differential

1. INTRODUCTION TO ITS

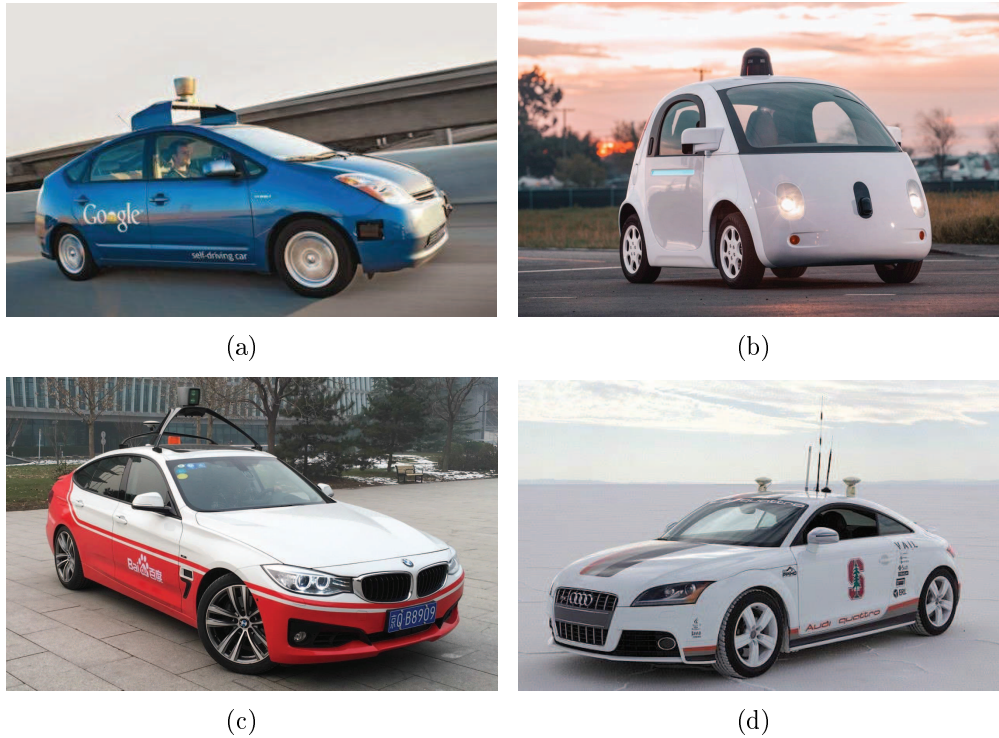


Figure 1.4: Self-driving cars

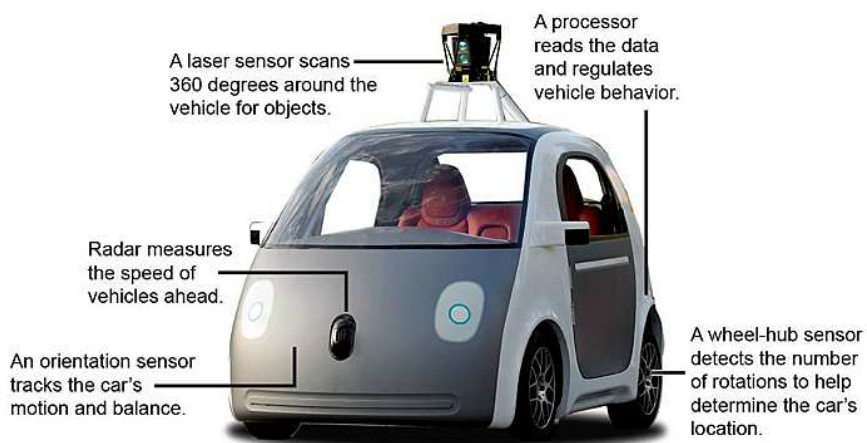


Figure 1.5: Sensors on Google's self-driving car (Source www.aventurine.com)

method is used when the car drives itself to compare the real-time signal with the recorded data in order to distinct pedestrians and stationary objects.

Similarly, in China, the company Baidu announced its autonomous car has successfully navigated a complicated route through Beijing (Davies (2015)). The car (see in Figure (1.4(c))) drove a 30 km route around the capital that included side streets as well as highways. The car successfully made turning, lane changing, overtaking, merging onto and off the highway.

The commercialization of self-driving cars can not be realized without automobile manufacturers. Some of them have launched their own self-driving projects targeting different scenarios (Chandler (2016)), such as “Drive Me” of Volvo (Ziegler (2015)), “Buddy” of Audi (Davies (2014)), Tesla (Korosec (2016)), etc. These prototypes are still at test stage, but it is a necessary step of self-driving car development.

Self-driving cars are considered to be capable to make better use of road space, therefore cars would drive closer to each other. They would react faster than humans to avoid accidents, potentially saving thousands of lives. Moreover, self-driving car could lower labor costs and bring the sharing economy to a higher level, thus people don’t need to own cars, only use them when needed. The number of vehicles would be reduced, then problems like congestion, pollution, public space loss, etc., could be subsequently solved.

However, the high price of sensors, especially the laser, may restrict the commercialization of self-driving car. Therefore, researchers and engineers are trying to use universal cameras combined with others cheap sensors to achieve the functions of the current system. Breakthroughs in computer vision are needed to make this come true (Templeton (2013)).

1.2 Vehicular Ad hoc Networks

1.2.1 Opportunities and challenges

Even if all the vehicles on road are replaced by intelligent self-driving vehicles, it does not mean that we have achieved the intelligence in system-level. One of the key features in ITS is the communication ability of all the participants which enables information exchange among them. For a traditional vehicle, the driver is

1. INTRODUCTION TO ITS

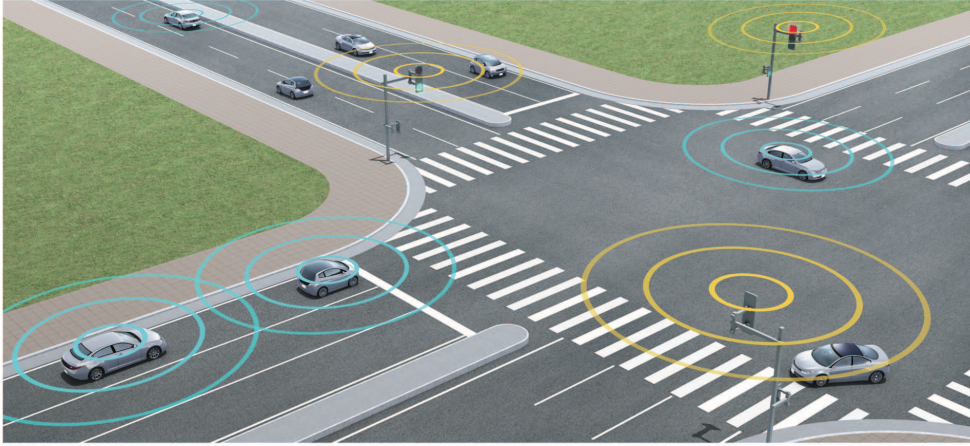


Figure 1.6: Vehicular Ad hoc Networks
(Source johndayautomotivelectronics.com)

aware of the situation within his sight, while for an autonomous vehicle, its “horizon of awareness” depends on the range and accuracy of its sensors. In both case, the information is analyzed and stored locally within a single vehicle. If communications among vehicles and with infrastructures along the road are enabled, the “horizon of awareness” can be considerably expanded, and a vehicle can decide its own move base on not only the current environment but also the prediction of the future environment, thus the individual intelligence can be turned into group intelligence and systematic intelligence. That is the motivation of the Vehicular Ad hoc Networks (VANET) conception. The communication between a vehicle and another agent is usually noted as V2X, including V2V (Vehicle-to-Vehicle), V2I (Vehicle-to-Infrastructure), V2P (Vehicle-to-Pedestrian) and V2C (Vehicle-to-Cloud) (Park & Min (2015)). With VANET, we can upgrade autonomous systems to cooperative systems.

While being promising in saving time and saving lives and being conceptually straightforward, design and deployment of VANET is technically and economically challenging (Hartenstein & Laberteaux (2010)). In technical terms, the challenges include the following issues:

- The most important challenge comes from the high mobility of the nodes. Due to high relative speed between cars, network’s topology changes very

fast, and many paths to disconnect before they can be utilized (Yousefi *et al.* (2006)). In (Wang (2004a,b)), the authors tried to find the approximation of link's lifetime. In (Artimy *et al.* (2004, 2005)), the authors attempted to capture some relationship between the model of vehicular mobility and connectivity of the networks. In (Blum *et al.* (2004)), the effective network diameter in a typical VANET is studied.

- The networks are self-organized, lacking of centralized management and coordination entity. It is difficult to synchronize and manage the transmission events of different nodes.
- The signal is sometimes blocked by obstacles, e.g. buildings in cities, the propagation models of the signal and the influence on the performance of VANET need to be studied.
- Standardization versus flexibility. Standardization is necessary to make the equipment from different manufactures compatible. while the manufactures will want to create some product differentiation.

In socio-economic terms, key challenges are as follows:

- Analyzing and quantifying the benefit of VANET for traffic safety and transport efficiency. As a new source of information, the impact of VANET on driving behavior needs to be studied.
- Analyzing and quantifying the cost-benefit relationship of VANET.
- Due to the “network effect”, users can benefit from VANET when the penetration rate is relatively high. There is the challenge of convincing manufactures to integrate VANET in their products and consumers to buy them, when it is rare to find a communication partner.
- As a part of the future ITS, the agents in VANET still need to interact with the other agents in the environment which are contacted in some other ways. The interface of VANET to the environment should also be developed.

1. INTRODUCTION TO ITS

1.2.2 Dedicated short range communications

In October 1999, the US Federal Communications Commission (FCC) allocated a 75MHz spectrum in the 5.9GHz band for the exclusive use of Intelligent Transportation Systems. The spectrum is known as Dedicated Short-Range Communications (DSRC). DSRC are one-way or two-way short-range to medium-range wireless communication channels specifically designed for automotive usage and a corresponding set of protocols and standards (Miller & Shaw (2001)). Usually, DSRC possess a range of 1km and a data rate from 6 to 27Mbps .

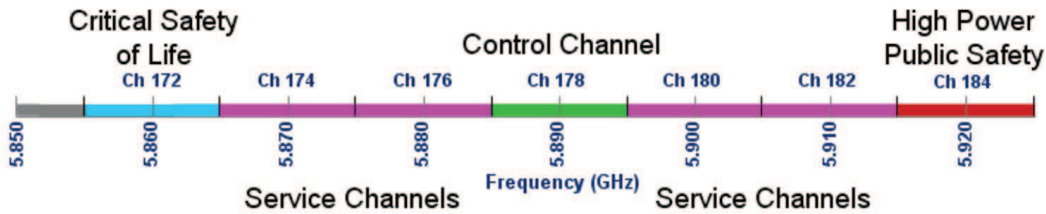


Figure 1.7: DSRC spectrum band and channels in the U.S. (Jiang & Delgrossi (2008))

As shown in Figure 1.7, the DSRC spectrum is structured into seven 10MHz wide channels (Jiang & Delgrossi (2008)). Channel 178 is the control channel (CCH), which is restricted to safety communications only. The two channels at the ends of the spectrum band are reserved for special uses. The rest are service channels (SCH) available for both safety and non-safety applications.

Standardization is a step that cannot be bypassed in the development of DSRC. Different standards have been formulated by several organizations, such as Institute of Electrical and Electronics Engineers (IEEE), European Committee for Standardization (CEN), European Telecommunications Standards Institute (ETSI) and Japanese Association of Radio Industries and Businesses (ARIB). Harmonization efforts government and standards organizations have begun to achieve global usage of DSRC. More details for the standards in different layers of DSRC protocol can be referred to (ISO 15628:2013 (2013); Jiang & Delgrossi (2008); Kenney (2011); Morgan (2010); Oyama *et al.* (2000)).

1.2.3 Applications in VANET

The interest in VANET is strongly motivated by the applications that could be enabled. Most of the applications can be classified into “Safety” and “Efficiency”. However, the aspects of safety and efficiency cannot be seen as completely disjoint sets of features (Hartenstein & Laberteaux (2010)). For example, vehicle crashes can lead to traffic jam. A message reporting an accident can be regarded as a safety message for the near-by vehicles on one hand, and regarded as an input to find an alternative route in a transport efficiency application on the other hand.

1.2.3.1 Vehicle-to-infrastructure applications

According to (Caveney (2010)), vehicle-to-infrastructure (V2I) applications are expected to be one of the first cooperative safety applications to be implemented in production vehicles because they demand low penetration rate to be effective in reducing vehicle-related fatalities. V2I communication can be applied but not limited in the following applications:

- *Intersection violation warning (IVW)*: The objective of IVW is to warn the driver when violating a red light seems imminent (Maile & Delgrossi (2009); Neale *et al.* (2007)). The application requires the location, light phase, light timing and intersection geometry broadcasted by the roadside unit and the vehicle’s predicted path. Lane-level accuracy is required in IVW applications, because different lanes are usually controlled by different signal phases and timing.
- *Speed advisory (SA)*: Approximately 7% of a vehicle’s energy is lost due to braking, therefore reducing braking is assumed a direct fuel saving strategy. If a driver is informed the upcoming signal phase and timing via V2I communication when approaching an intersection, the velocity of the vehicle can be regulated accordingly to avoid hard braking or accelerating, thereby reducing energy consumption and pollutant emissions (Rakha & Kamalanathsharma (2011); Ubiergo & Jin (2015)). Besides, it is possible to reduce energy consumption by preventing a vehicle from coming to a full stop at the intersections and by advising cruising velocities in order to catch as many green lights as possible (De Nunzio *et al.* (2015)). A velocity planning algorithm to achieve eco-driving is introduced in Chapter 3.

1. INTRODUCTION TO ITS

- *Traffic information systems (TIS)*: The core idea of these applications is to use vehicles as sensors which monitor parameters like traffic density, road and weather conditions, or parking space occupancy (Lochert *et al.* (2010)). The information is distributed to other vehicles as the input of efficiency applications. Relative research can be found in (Dornbush & Joshi (2007); Ibrahim & Weigle (2008); Lochert *et al.* (2005); Xu & Barth (2006)).
- *Autonomous intersection management (AIM)*: In the future, if all the vehicles become autonomous, the traffic can also be managed in an autonomous way. A manager would be installed at the intersection to solve the space-time conflict between vehicles coming in different streets, then the manager's instruction, which can be seen as a personal traffic signal to each vehicle, is distributed via V2I communication (Dresner & Stone (2008)). AIM is promising in improving the throughput of an intersection and reducing the delay of vehicles, besides, the fuel consumption and emission can also be decreased. Two different V2X-based intersection management protocols are proposed respectively in Chapter 4 and 5.

1.2.3.2 Vehicle-to-vehicle applications

Then V2V applications demands a relatively higher penetration rate of VANET. V2V communication can be used but not limited in the following applications:

- *Electronic brake warning (EBW)*: EBW is used to alert the driver when a preceding vehicle performs a violent braking maneuver, especially when the view of braking vehicle is blocked by other vehicles.
- *On-coming traffic warning (OTW)*: OTW alerts the driver the on-coming traffic on the opposite lane during the overtaking maneuvers. The relative positions including longitudinal and lateral need to be accurately predicted.
- *Lane change warning (LCW)*: LCW warns the driver who want to perform a lane change but it is unsafe to do so. Sometimes, the driver forgets to check the blind spot zone, or the view is blocked by other vehicles, in this case, the existence of the unnoticed vehicle can be informed to the driver.

- *Emergency vehicle warning (EVW)*: Instead of using sirens alone, this application helps making the right reaction when encountering emergency vehicles such as fire truck and ambulance (Buchenscheit *et al.* (2009)). EVW helps reduce accident risks during emergency response trips and save valuable time.
- *Cooperative Adaptive Cruise Control (CACC)*: With V2V communication, the Adaptive Cruise Control (ACC) which is currently available in the market can be turned into CACC (Van Arem *et al.* (2006)). Information like position, velocity, and acceleration can be transmitted to the following vehicle using inter-vehicle communication. The objective of CACC is to provide a secure longitudinal control to avoid collision and make the platoon compact. In CACC, a vehicle's behavior depends on not only the nearest preceding vehicle but also the vehicles further in front. Compared to ACC, the string stability of the platoon can be improved. More detailed introduction can be found in section 1.3. And a CACC algorithm is proposed in Chapter 3.

1.2.3.3 An instance of VANET application

As shown in Figure 1.8, an evolution road-map of the road transportation in urban and highway scenarios is given (SIEMENS (2015)).

- Currently, we are at “Assistance driving” stage, where the road speed limit can be given to driver by GPS and on-board navigation equipment. However, driver still needs to pay attention to temporary signs by himself, and make the right decision. Driver could get confused when meet information from different sources.
- In the near future, when communication devices are installed on vehicles and infrastructures, information can be given to drivers in a more precise way by VANET. Advisory applications implemented on board helps drivers to achieve better driving behavior. This stage is called “Automated driving”.
- After that, when auto-driving technology is popular, the “Autonomous driving” can be reached. At this stage, the vehicle would make decisions according to the VANET-based applications without any intervention of drivers.

1. INTRODUCTION TO ITS

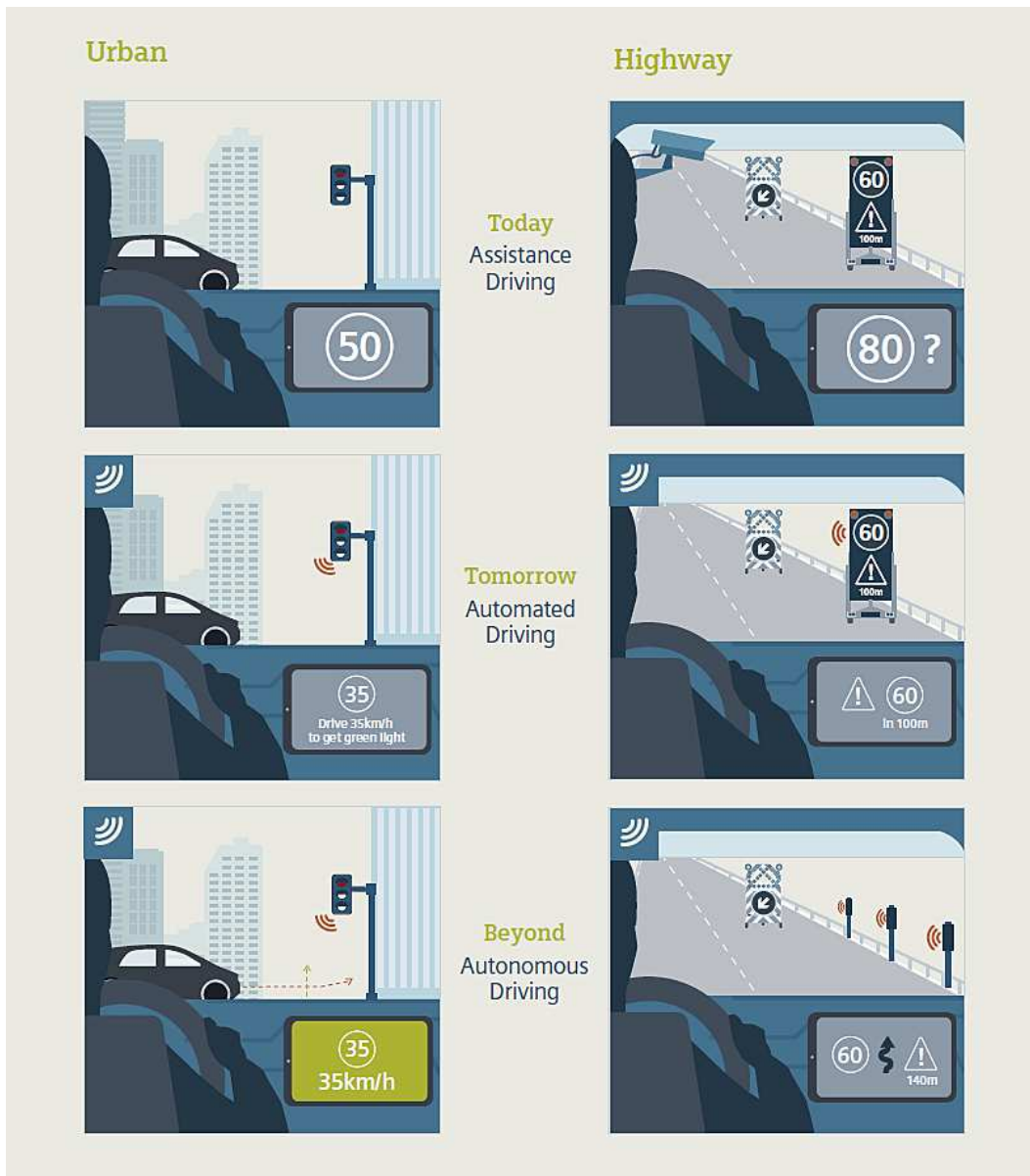


Figure 1.8: An instance of VANET-based application development
(Source: www.mobility.siemens.com)

1.3 Cooperative Adaptive Cruise Control

1.3.1 Evolution: from autonomous to cooperative

As introduced previously, for decades, people are trying to develop ITS in order to make transport safer and more efficient. In vehicle terms, advanced driver-assistant systems (ADAS) are developed aiming at enhancing driving comfort, reducing driving errors, improving safety, increasing traffic capacity and reducing fuel consumption. The main applications of ADAS includes Adaptive Cruise Control (ACC) (Vahidi & Eskandarian (2003)), Automatic Parking (Xu *et al.* (2000)), Lane Departure Warning (Dahmani *et al.* (2013)), Lane Change Assistance (Morris *et al.* (2011)), Blind Spot Monitor (Lin *et al.* (2012)), etc. Although the objective of ADAS is not to replace people in driving, it can help relief people from repetitive and boring labor, such as lane keeping, lane changing, space keeping, cruising, etc. Besides, the technologies developed in ADAS could also be used in autonomous driving.

One of the most important ADAS is adaptive cruise control (ACC), which is now commercially available in a wide range of passenger vehicles. ACC systems are an extension of cruise control (CC) systems. CC is used to maintain vehicle's velocity to a decided value, and the driver does not have to depress the pedals, therefore he/she can be more focused on steering wheel. CC can be turned off both explicitly and automatically when the driver depresses the brake. For ACC, if there is no preceding vehicle within a certain range, it works as a CC system; else, it utilizes the range sensor (such as lidar, radar and camera) to measure the distance and the relative velocity to the preceding vehicle. Then the ACC system calculates and estimates whether or not the vehicle can still travel at the user-set velocity. If the preceding vehicle is too close or is traveling slowly, ACC shifts from velocity control to time headway control by control both the throttle and brake (Xiao & Gao (2010)). However, ACC still has its own limits: in general, ACC system is limited to be operated within a velocity range from 40km/h to 160km/h and under a maximum braking deceleration of 0.5g (Rajamani (2011)). The operations outside these limits are still in the charge of driver, because it is very difficult to anticipate the preceding vehicle's motion only by using range sensors, so the vehicle cannot react instantly.

1. INTRODUCTION TO ITS

With the development of communication technologies and the international standard of DSRC (ISO 15628:2013 (2013); Miller & Shaw (2001)), researchers have gradually paid attention to cooperative longitudinal following control based on V2X communication in order to truly improve traffic safety, capacity, flow stability and driver comfort De Bruin *et al.* (2004); Lu *et al.* (2002); Xu & Sengupta (2003).

1.3.2 Development of ACC

The term “ACC” is firstly proposed by Broqua (1991) within the programme PROMETHEUS (Williams (1988)) initiated in 1986 in Europe. Actually, a large proportion of the work in this programme was conducted as propriety development work by automakers and their suppliers rather than publicly funded academic research. Therefore, most of the results and methods are not documented in open literature, but kept secret in order to enhance competitive advantage (Xiao & Gao (2010)). In USA, in 1986, the California Department of Transportation and the Institute of Transportation Studies at the University of California Berkeley initiated the state-wide programme called PATH (Shladover (1995)) to study the use of automation in vehicle-highway systems. Then the programme was extended in national scope named as Mobility 2000 (Figueiredo *et al.* (2001)), which grouped intelligent vehicle highway system technologies into four functional areas covering ACC systems. A large-scale ACC system field operations test was conducted by Fancher’s group (Fancher (1998)) from 1996 to 1997, in which 108 volunteers drove 10 ACC-equipped vehicles to determine the safety effects and user-acceptance of ACC systems.

Any design of an ACC system begins with the selection and design of a spacing policy. The spacing policy refers to the desired steady state distance between two successive vehicles. In 1950s, the “law of separation” (Pipes (1953)) is proposed, which is the sum of the distance that is proportional to the velocity of the following vehicle and a given minimum distance of separation when the vehicles are at rest. Then, three basic spacing policies (constant distance, constant time headway) and constant safety factor spacing have been proposed for the personal rapid transit (PRT) system (Mackinnon (1975)). Some nonlinear spacing policies (Wang & Rajamani (2004); Zhou & Peng (2005)) have been proposed to improve

1.3 Cooperative Adaptive Cruise Control

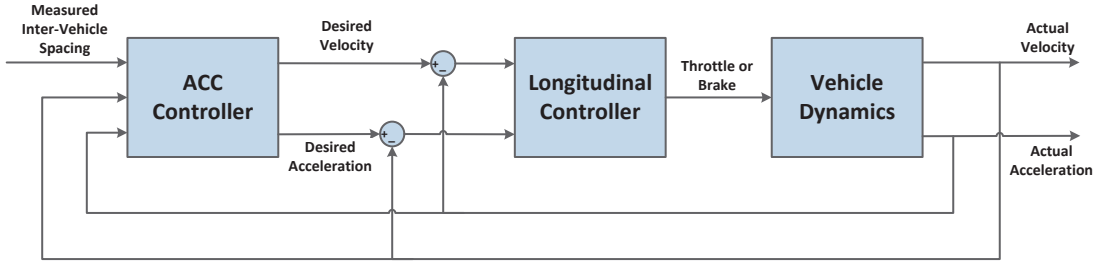


Figure 1.9: Structure of longitudinal control system

traffic flow stability, which are called constant stability spacing policies. In order to improve the user-acceptance rate, a drive-adaptive range policy (Han & Yi (2006)) is proposed, which is called the constant acceptance spacing policy. Considering feasibility, stability, safety, capacity and reliability (Swaroop *et al.* (1994)), the constant time headway (CTH) spacing policy is applied to ACC systems by manufacturers.

The longitudinal control system architecture of an ACC-equipped vehicle is typical hierarchical (see Figure 1.9), which is composed of an upper level controller and a lower level controller (Rajamani (2011)). Usually, the former is called the ACC controller and the latter is called the longitudinal controller. The ACC controller determines the desired acceleration or velocity to the longitudinal controller. The longitudinal controller determines the throttle and/or brake to track the desired accelerations and returns the fault messages to the ACC controller.

The ACC controller should be designed to meet three performance specifications:

- *Individual stability*: if the spacing error of the ACC vehicle converges to zero when the preceding vehicle is operating at constant speed. If the preceding vehicle is accelerating or decelerating, then the spacing error is expected to be non-zero. Spacing error is defined as the difference between the actual spacing from the preceding vehicle and the desired inter-vehicle spacing.
- *String stability*: this property is defined as the spacing errors are guaranteed not to amplify as they propagate towards the tail of the string.

1. INTRODUCTION TO ITS

- *Traffic flow stability*: this property is similar to string stability, but in a macroscopic way: any density or velocity disturbances must attenuate as it propagates upstream. It is indicated that using CTH spacing policy leads to unacceptable characteristics for automated traffic flows (Li & Shrivastava (2002); Yi & Horowitz (2006)).

1.3.3 Related work in CACC

CACC is a further development of ACC that adds V2V communications, providing the ACC system with more and better information about the preceding vehicles. With more accurate information, the ACC controller will be able to better anticipate problems, makes it to be safer and smoother in response (Van Arem *et al.* (2006)).

The concept of automated highway systems (AHS) is defined as vehicle-highway systems that support autonomous driving on dedicated highway lanes. In 1997, the National Automated Highway System Consortium (NAHSC) demonstrated several highway automation technologies. The highlight of the event was a fully automated highway system (Rajamani *et al.* (1998); Thorpe *et al.* (1997)). The goal of the AHS demonstration was a proof-of-concept of an AHS architecture that enhanced highway capacity and safety. Capacity increase was achieved by organizing the movement of vehicles in closely spaced platoons. Each vehicle had actuated-steering, braking and throttle that were controlled by the on-board computer. Safety was increased because the computer was connected to sensors that provided about itself, the vehicle's location within the lane, the relative speed and distance to the preceding vehicle. Most importantly, an inter-vehicle communication system formed a local area network to exchange information with other vehicles in the neighborhood, as well as to permit a protocol among neighboring vehicles to support cooperative maneuvers such as lane-changing, joining a platoon, and sudden braking. Computer-controlled driving eliminated driver misjudgment, which is a major cause of accidents today. At the same time, a suite of safety control laws ensured fail-safe driving despite sensor, communication and computer faults. The AHS experiment also showed that it could significantly reduce fuel consumption by greatly reducing driver-induced acceleration and deceleration surges during congestion.

1.3 Cooperative Adaptive Cruise Control



Figure 1.10: Vehicle platoon in GCDC 2011

The effects on capacity of increasing market penetration of ACC and CACC vehicles, relative to fully-manually driven vehicles, was examined by using microscopic-traffic simulation (Van Arem *et al.* (2006); Vander Werf *et al.* (2002)). The analyses were initially conducted for situations where manually driven vehicles, ACC-equipped vehicles and CACC-equipped vehicles separately have 100% penetration rate. The results shows that capacity in these situations are respectively 2050, 2200 and 4550 vehicles per hour, thus the route's capacity can be greatly improved using CACC. Then mixed vehicle populations were also analyzed, and it was concluded that CACC can potential double the capacity of a highway lane at high penetration rate.

The CHAUFFEUR 2 project is initiated in order to reduce a truck driver's workload by developing truck-platooning capacity (Bonnet (2003)). A truck can automatically follow any other vehicle with a safe following distance using ACC and a lane-keeping system. Also, three trucks can be coupled in a platooning mode. The leading vehicle is driven conventionally, and the other trucks follow. Due to the V2V systems installed on the trucks, the following distance can be reduced to $6 \sim 12m$. Simulation results show that the systems have better usage of road capacity, up to 20% reduction in fuel consumption and increased traffic safety.

1. INTRODUCTION TO ITS

In 2011, the Netherlands Organization for Applied Scientific Research (TNO), together with the Dutch High Tech Automotive Systems innovation programme (HTAS) organized the Grand Cooperative Driving Challenge (GCDC) (Geiger *et al.* (2012); Guvenc *et al.* (2012); Kianfar *et al.* (2012); Ploeg *et al.* (2012); van Nunen *et al.* (2012)). The 2011 GCDC mainly focused on CACC. Nine international teams participated in the challenge (see Figure 1.10), and they need to form a two-lane platoon with the help of V2X technologies and longitudinal control strategies. However, the algorithms running at each vehicle are different and not available to each other. The competition successfully showed cooperative driving of different vehicles ranging from a compact vehicle to a heavy-duty truck. Several issues should be addressed in the future like dealing with the flawed or missing data from other vehicles and lateral motions such as merging and splitting to be closer to realistic situations.

1.4 Autonomous Intersection Management

As autonomous vehicles become more and more prevalent, autonomous interactions among multiple vehicles become possible. And the current methods of vehicle coordination designed to work with human drivers will be outdated. Most modern intersections are controlled with traffic lights or stop signs, the former usually reserved for larger and busier intersections. Large extremely expensive cloverleaf junctions are built for even larger intersections. The first generation of autonomous vehicles are designed to mimic the way of human driver to recognize street signs, traffic lights, to keep large enough distance to other vehicles. However, when the penetration rate of autonomous vehicles is high enough, new coordination methods, which should be suitable and more efficient to autonomous vehicles, need to be designed. Then vehicular communications, which can make information more conveniently accessible, should be taken advantage of. However, intersection is a place where vehicles in different directions cross paths, and it is a more complicated scenario compared to highways, thus much more challenging in designing coordination protocols. A survey of traffic control strategy with vehicular communications is given in (Li *et al.* (2014)).

1.4.1 Isolated Intersection Control

One of the basic intersection control strategies is the fixed-time traffic signal. This strategy is effective to solve the space-time conflict inside the intersection. However, it is not efficient or fair: the vehicles should stop before the intersection even if there is no vehicle in other lanes, which leads to longer delay at the intersection; during this period, the engines are still idling, more fuel is consumed, which leads to more pollution. Thus the key to make intersection more efficient in terms of time and energy is to reduce the waiting time.

The earliest attempt is actuated traffic signal control system. Limited measurements are provided by inductive loop installed tens of meters upstream to the stop lines. Each detected vehicle will generate a call to ask for an additional green time. Then it comes to adaptive traffic signal control systems, such as PRODYN (Henry *et al.* (1984)), CRONOS (Boillot (1992)), SCOOT (Hunt *et al.* (1982)) and RHODES (Mirchandani & Wang (2005)). Induction loops are still used, but not just to detect the existence of vehicles, also how long it occupies the detector and its speed. Prediction models are also employed to forecast the vehicle arrivals and estimate the queue lengths in each direction. In this case, the performance of time plan depends on the prediction accuracy.

Instead of considering the vehicles individually, we can also treat them in a collective way: a group of vehicles traveling together with short inter-vehicle headway can be defined as a platoon, meanwhile the distance between consecutive platoons can be quite large, therefore it is possible to design a timing plan that allows the platoons coming from different directions to pass the intersection without being interrupted. This kind of strategy is called the platoon-based control (He *et al.* (2012); Jiang *et al.* (2006)). One difficulty of deploying platoon-based control strategy is how to identify the platoons quickly, conveniently and robustly (Gaur & Mirchandani (2001)). If vehicles can send their movement information via V2X communications, there will be no platoon identification problem (Pandit *et al.* (2013); Priemer & Friedrich (2009)).

In addition to regulate the timing plan of lights to adapt the traffic demands, another idea is to regulate the movements of vehicles to further improve driving efficiency. The concept of cooperative driving was firstly introduced in the early 1990s (Hedrick *et al.* (1994); Tsugawa (2002)). Initially, it only refers to flexible platooning of automated vehicles. Actually, this conception has been extended

1. INTRODUCTION TO ITS

to implementing collision-free vehicle movement around unsignalized intersection using inter-vehicle communications (Li & Wang (2006)). In cooperative driving, a vehicle can choose to accelerate to join in an already formed platoon or decelerate to depart from it. Several vehicles can merge together to form a new platoon as well.

The objective of cooperative driving is to find a collision-free and feasible schedule that makes the vehicles get through the intersection in the shortest time. The feasibility of schedule is often difficult to test, since we need to consider whether the vehicle can produce the trajectory within its dynamical limits. Detailed discussions can be found in (Kowshik *et al.* (2011); Lee & Park (2012)).

1.4.2 Motion Planning at Intersection

Under light traffic conditions, the whole solution space can be enumerated to find the best schedule (Alonso *et al.* (2011); Milanés *et al.* (2010); Naumann *et al.* (1998); Omae *et al.* (2010); Zohdy & Rakha (2012)). However, when facing heavy traffic, the problem becomes much more complicated.

The first kind of solution is planning-based approaches, in which mathematical programming or heuristic algorithms can be used to find some near-optimal solutions. The reported algorithms include genetic algorithm (Teodorovic *et al.* (2001)), linear programming (Ghaffarian *et al.* (2012)), ant-colony algorithm (Wu *et al.* (2012)), Petri Nets (Ahmane *et al.* (2013)), etc.. Another kind of solution is negotiation-based approaches, in which the vehicles that almost arrive at the intersection are taken into consideration, and short-term driving plans are formulated by bilateral negotiations (Baber *et al.* (2005); Dresner & Stone (2008); Lee *et al.* (2013); Schepperle *et al.* (2008); Zohdy & Rakha (2012)). Thus the dimension of solution space is much smaller and the research cost can be greatly reduced.

The performance of negotiation-based control is sometimes lower than that of planning-based control, however, negotiation-based approach consumes less computational costs and generates agile reactions under any unexpected changes (Li *et al.* (2014)).

In order to reduce the computational cost of the planning-based approaches, decoupling methods can be used to decompose the original problem into multiple planning problems in lower-dimensional configuration spaces. This not only

decreases computational complexity, but also restricts the solution space because not all combinations of actions remain possible (Frese & Beyerer (2011a)). Instead of planning the motions for all vehicles at the same time, the motions can be planned separately for each vehicle in the order of priorities. For each vehicle, the plans of the higher-priority vehicles are regarded as moving obstacles, while the vehicles with lower-priorities are ignored (Bekris *et al.* (2007)). The priorities can be described as a permutation of vehicles. Strategies for choosing or optimizing the permutation were proposed in (Bennewitz *et al.* (2001); Van Den Berg & Overmars (2005)). Criteria based on traffic rules could also be used for selecting the permutation. For instance, the leading vehicle should have higher priority than the following vehicles on the same lane.

1.5 Conclusion

This chapter gives a detailed introduction to intelligent road transportation systems. Firstly, the current road transportation, which could cause economical and environmental problems, is not safe or efficient enough to response our demand, therefore it should be ameliorated and related technologies should be developed. Then several historical researches worldwide are presented. As a promising solution to reduce the accidents caused by human errors, autonomous vehicles are being developed by research organizations and companies all over the world. The state-of-art in autonomous vehicle development is introduced in this chapter as well.

Secondly, a key aspect in developing ITS: the communication is introduced. Specific to road transportation systems, it is V2X communications, including V2V communication and V2I communication. By enabling communications among these agents, the vehicular ad hoc networks (VANETs) are formed. With VANET, autonomous systems can be upgraded into cooperative systems, in which a vehicle's range of awareness can be extended, therefore it can anticipate in advance in an optimal way. Different kinds of applications using VANET are developed in order to make the road transportation safer, more efficient and user friendly.

Thirdly, the cooperative adaptive cruise control (CACC) is presented. CACC is an extension of ACC systems by enabling the communication among the vehicles in a platoon. CACC can not only relief the driver from repetitive jobs like

1. INTRODUCTION TO ITS

adjusting speed and distance to the preceding vehicle like ACC, but also has safer and smoother response than ACC systems.

Fourthly, the concept of autonomous intersection management (AIM) is described. AIM is designed to replace the traditional light or sign control strategies in the future. By using VANET and autonomous driving technologies, collision-free trajectories of vehicles in the vicinity of the intersection can be planned and followed, which can improve the intersection's throughput and reduce the vehicles' travel delays. Different design philosophies of AIM are also introduced.

Safety and efficiency are two most demanded features of ITS. Therefore, in this thesis, we focus on an isolated intersection scenario, different applications are designed for both signal controlled and unsignalized intersection in order to improve the intersection's throughput while guarantee safety in the crossing motions, by controlling the actions of vehicle platoons or individual vehicles. Different features such as heterogeneity of vehicles, constraints of dynamics, computational cost, passenger comfort, etc. are considered in our design philosophy, which is detailed in the following chapters.

Chapter 2

Decentralized Platoon Control

Contents

| | | |
|------------|----------------------------------|-----------|
| 2.1 | Introduction | 33 |
| 2.2 | Modeling | 36 |
| 2.2.1 | Vehicle kinematic model | 36 |
| 2.2.2 | Platoon model | 37 |
| 2.3 | PSO based Platoon control | 40 |
| 2.3.1 | Cost function design | 40 |
| 2.3.2 | Optimization algorithm | 42 |
| 2.4 | Simulation of algorithm | 45 |
| 2.5 | Conclusion | 48 |

2.1 Introduction

Arriving at one place safely in a certain period is the basic requirement of transportation. However, today's road transportation is far from perfect. Incorrect driving behaviors like drunken driving, fatigue driving and speeding are thought to be the main reasons for road accidents which on one hand cause injury, death, and property damage, on the other hand make vehicles keep larger distance from each other, thus the road capacity is not made full use of. Moreover, congestions caused by incorrect driving behaviors, accidents, improper signal timing

2. DECENTRALIZED PLATOON CONTROL

have become a global phenomenon which has economically and ecologically negative effects, so that people have to spend more time on road and more fuel is consumed, which leads to more pollution.

More efficient, better space utilization and elimination of human error, self-driving or semi self-driving car developed by Google and automobile manufacturers all over the world is a potentially revolutionizing technology to solve these problems ([Poczter & Jankovic \(2014\)](#)). However, the intelligence of individual vehicles does not represent the intelligence of the whole transportation system.

V2X technology, including Vehicle to Vehicle (V2V) communication via wireless ad hoc network and Vehicle to Infrastructure (V2I) via Dedicated Short-Range Communication (DSRC), is considered to be the next step to construct the intelligent transportation system (ITS). Instead of detecting the environment all by the vehicle itself, this technology enables the communication between vehicles themselves and with infrastructure along the road. Possible applications of V2X in terms of safety, information dissemination and efficiency are illustrated in ([Hartenstein & Laberteaux \(2010\)](#)).

V2X is the key technology to make some current systems more intelligent. For example, the adaptive cruise control (ACC) system, which is commercially available in the market, aims at automatically maintaining a constant speed or a safe distance to the preceding vehicle. Currently, the ACC system is mainly based on ranging sensors, e.g., radars and lasers, which are capable to measure the range or its changing rate. Generally, ACC system can relieve drivers from performing repetitive and boring jobs like adjusting speed and distance to the preceding vehicle in highway scenarios. However, current ACC systems still have disadvantages, like incapability to more complicated urban scenarios and limit anticipatory capabilities to sudden maneuver of the preceding vehicle ([Desjardins & Chaib-draa \(2011\)](#)). Moreover, ACC system cannot deal with shock-wave effect. Due to delays in sensor and actuator system of the ACC host vehicle, amplifications may appear, especially when constant spacing policy is used. Thus passenger comfort cannot be guaranteed and more fuel is consumed.

With V2X technology, ACC can be extended into cooperative ACC (CACC), which is considered to be highly potential to improve traffic flow capacity and smoothness and to reduce congestions ([Van Arem *et al.* \(2006\)](#)). V2X integrated vehicle can not only get information from the preceding vehicle as in ACC system,

but also the vehicles before the preceding one, even beyond line of sight (Milanés *et al.* (2014)). Research shows that with CACC, vehicles can have smaller spacing distance compared to ACC, and the shock-wave effect can be greatly mitigated (Calvert *et al.* (2011); Schakel *et al.* (2010)). A safety spacing policy (SSP) is introduced in (Zhao *et al.* (2009)) which utilizes the state of the preceding vehicle combined with the braking capacity to adjust the position and velocity of the controlled vehicle. The SSP can ensure vehicle string stability and it can also yield stable traffic flow and higher traffic capacity than the traditional Constant Time Gap (CTG) policy.

String stability is an important goal to be achieved in CACC system design (Milanés *et al.* (2014)). A platoon of vehicles is called string stable only if disturbances propagated from the leading vehicle to the rest of the platoon can be attenuated (Ploeg *et al.* (2011)). Researchers have tried different methodologies in designing platoon controllers. Most of the projects are relied on classic control theory like proportional-derivative feedback/feedforward controllers (Guvenc *et al.* (2012); Lidstrom *et al.* (2012); Milanés *et al.* (2014); Nieuwenhuijze *et al.* (2012)). De Bruin *et al.* (2004) summarized the block diagram to design a CACC system using classic control theory and suggest the four techniques to be used in the system. The concepts of "expected velocity" and "expected acceleration" are introduced in (Han *et al.* (2013)) and the CACC problem is turned into a tracking problem of spacing error, velocity and acceleration error in the platoon, where the control input of each vehicle is calculated by using an optimal control method. Model predictive control (MPC) technique can also be found in CACC research and implementation (Geiger *et al.* (2012); Kianfar *et al.* (2012)). A multi-objective CACC controller is designed in (Li *et al.* (2011)), which takes the tracking capacity, fuel economy and driver desired response into account.

In this chapter, the vehicle kinematic model and the platoon model are described; then a platoon control algorithm is proposed with following design objectives:

- Constraints like speed limit and power limit of the vehicles must be considered;
- The safety inside the platoon must be guaranteed, thus a safety space must be maintained to avoid rear-end collision;

2. DECENTRALIZED PLATOON CONTROL

- The velocity of vehicles in the platoon should be consistent;
- The passengers' comfort should be taken into account, that's to say, sharp change of the acceleration should be avoided.

2.2 Modeling

2.2.1 Vehicle kinematic model

As we assume that no lane shifting maneuver is taken during the process, a longitudinal vehicle model is sufficient. Hence, a simple three-state space linear model is used (Han *et al.* (2013)). For the i th vehicle V_i , we have:

$$\begin{aligned} \dot{x}_i(t) &= \Phi_i x_i(t) + \Pi_i u_i(t) \\ x_i(t) &= [p_i(t) \quad v_i(t) \quad a_i(t)]^T \\ \Phi_i &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1/\tau_i \end{bmatrix}, \quad \Pi_i = \begin{bmatrix} 0 \\ 0 \\ 1/\tau_i \end{bmatrix} \end{aligned} \quad (2.1)$$

where $x_i(t) \in R^3$ is the i th vehicle's system state at time t , and $p_i(t)$, $v_i(t)$ and $a_i(t)$ are separately the position, velocity and acceleration of the vehicle; $u_i(t) \in R$ represents the control input like thrust and brake; τ_i is the time constant coming from the vehicle's dynamical system which is heterogeneous for different vehicles. The continuous-time system above should be turned into discrete-time domain considering the implementation issue.

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k) \quad (2.2)$$

Moreover, constraints should be added on the vehicle system,

$$v_i(k) \leq v_{limit} \quad (2.3)$$

$$\begin{aligned} \frac{P_{tractive,i}(k)}{\eta_i} &\leq P_{engine,i} \\ P_{tractive,i}(k) &= \frac{M_i}{1000} \cdot v_i(k) \cdot a_i(k) + \left(M_i \cdot g \cdot C_r + \frac{\rho}{2} \cdot v_i^2(k) \cdot A_i \cdot C_a \right) \cdot \frac{v_i(k)}{1000} \end{aligned} \quad (2.4)$$

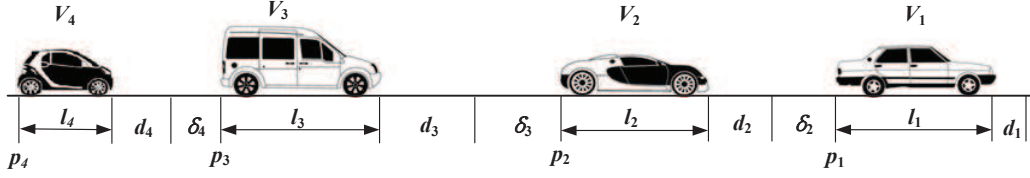


Figure 2.1: Platoon illustration

$$u_{min,i} \leq u_i(k) \leq u_{max,i} \quad (2.5)$$

$$|\Delta a_i(k)| \leq |\Delta a|_{max,i} \quad (2.6)$$

Equation (2.3) represents the road speed limit. Equation (2.4) is the power constraint of the i th vehicle, and it means that the tractive power demanded by the vehicle cannot exceed its engine power, where η_i is the transmission efficiency of the vehicle. The tractive power is composed of two parts, the first part comes from the acceleration of the vehicle, while the second part comes from the rolling resistance and the aerial resistance, where g is the gravitational constant, C_r is the rolling resistance coefficient, ρ is the mass density of air, A_i is the vehicle cross sectional area, and C_a is the aerodynamic drag coefficient. In consideration of reality, we add equation (2.5) as the constraint on the control input because the thrust and the brake ability are usually limited. As the passenger's comfort should be guaranteed, the acceleration cannot change sharply, so the changing rate of acceleration is also constrained by equation (2.6).

2.2.2 Platoon model

We define the vehicle's position p_i as the position of its rear bumper. l_i denotes the length of the i th vehicle V_i , while d_i denotes the safety spacing which is defined as:

$$d_i(k) = \gamma_i \cdot d_{min,i} + h_i \cdot v_i(k) \quad (2.7)$$

where γ_i is a safety coefficient which can be selected by the driver in accordance with the road condition and driver's preference. For instance, a bigger γ_i should

2. DECENTRALIZED PLATOON CONTROL

be set to get a larger safety spacing when the road is wet. $d_{min,i}$ is the minimal constant inter-distance, and h_i represents a time delay to recognize a hard brake of the preceding vehicle. l_i , $d_{min,i}$ and h_i are all the specific features of V_i . Hence, the safety spacing d_i is composed of two parts: $\gamma_i \cdot d_{min,i}$ is the constant part and $h_i \cdot v_i(k)$ is the velocity dependent part. An instance can be referred to the platoon G_3 in Fig. 2.1.

The spacing error of V_i to its preceding vehicle V_{i-1} can be defined as:

$$\delta_i(k) = p_{i-1}(k) - p_i(k) - d_i - l_i \quad (2.8)$$

with

$$\delta_i(k) > -d_i \quad (2.9)$$

(2.9) is a strict constraint to be respected in the control process, which aims at avoiding collision in the platoon. As introduced in the control objectives, the platoon should be compact to take full use of the space, while a safety spacing should be maintained to avoid rear-end collision. This objective can be explained as a zero spacing error that we should achieve during the control process.

For the velocity and acceleration coordination of the platoon, conceptions of “expected velocity” and “expected acceleration” are proposed in (Han *et al.* (2013)). In this platoon model, the host vehicle V_i can not only receive the information from its nearest preceding vehicle but also the leading vehicle V_l of the platoon, so that it can make the decision from a larger perspective and the control can be more intelligent. The advantage of this setting is that the host vehicle has the ability to predict the behavior of the platoon, thus a smaller delay of movement transmission can be achieved, and the platoon can converge faster to the steady state. The influence from the platoon leader V_l to the host vehicle V_i is related to the position of the host vehicle in the platoon. The greater the distance to the platoon leader, the host vehicle get less influence from the leader and more influence from the nearest preceding vehicle. The expected velocity and acceleration of the host vehicle V_i are defined as follows:

$$\begin{aligned} v_{r,i}(k) &= (1 - w_i) \cdot v_{i-1}(k) + w_i \cdot v_l(k) \\ a_{r,i}(k) &= (1 - w_i) \cdot a_{i-1}(k) + w_i \cdot a_l(k) \end{aligned} \quad (2.10)$$

where w_i is the influence weight from the platoon leader, which corresponds to the vehicle's position in the platoon, and it decreases with the distance to the leader. We can notice that the vehicle right behind the leader is completely influenced by the latter.

Therefore, we can define the velocity error and the acceleration error as:

$$\begin{aligned}\Delta v_{e,i}(k) &= v_i(k) - v_{r,i}(k) \\ \Delta a_{e,i}(k) &= a_i(k) - a_{r,i}(k).\end{aligned}\tag{2.11}$$

The objectives of platoon control can be partially expressed as to make the spacing, velocity, and acceleration error as small as possible. However, these objectives are contradictory in some cases. For example, when the vehicles in the platoon are at the same constant velocity while the spacing error between two vehicles is not zero, the posterior vehicle have to accelerate to eliminate the spacing error, whereas the velocity and acceleration error have to be increased. Thus, a priority must be defined for these three different control objectives.

In order to make decisions individually, each vehicle in the platoon should be capable to measure the three errors based on its own situation and the traffic information coming from related vehicles via V2V communication. Therefore, equations (2.8) and (2.10) should be integrated into the discrete vehicle model (2.2). The measurement output can be divided into two parts according to the information source: the measurement based on its own situation, and the measurement based on the information of the platoon leader and the nearest preceding vehicle. The former can be defined as y_i :

$$\begin{aligned}x_i(k+1) &= A_i x_i(k) + B_i u_i(k) \\ y_i(k) &= C_i x_i(k)\end{aligned}\tag{2.12}$$

where

$$C_i = \begin{bmatrix} -1 & -h_i & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

And the latter can be defined as z_i :

$$z_i(k) = H_i \xi_i(k)\tag{2.13}$$

2. DECENTRALIZED PLATOON CONTROL

where

$$H_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & w_i - 1 & 0 & -w_i & 0 \\ 0 & 0 & w_i - 1 & 0 & -w_i \end{bmatrix}$$

$$\xi_i = \begin{bmatrix} x_{i-1}(k) - l_i - \gamma_i \cdot d_{min,i}(k) \\ v_{i-1}(k) \\ a_{i-1}(k) \\ v_l(k) \\ a_l(k) \end{bmatrix}.$$

Therefore, the tracking error of vehicle V_i can be expressed as follows:

$$e_i(k) = y_i(k) + z_i(k) \quad (2.14)$$

where

$$e_i(k) = [\delta_i(k) \quad \Delta v_{e,i}(k) \quad \Delta a_{e,i}(k)]^T.$$

2.3 PSO based Platoon control

2.3.1 Cost function design

In order to find an optimal control input for every vehicle at each time step with an integrated consideration of the contradictory between different objectives, a cost function needs to be defined. In this paper, we use a quadratic function of the tracking errors and the control input to find a trade-off between the tracking objectives and the control cost as well.

$$J(V_i) = e_i^T(k)Q_i e_i(k) + u_i^T(k)R_i u_i(k) \quad (2.15)$$

where Q_i and R_i are respectively the weights of the tracking errors and the control input,

$$Q_i = \begin{bmatrix} q_{i,1} & 0 & 0 \\ 0 & q_{i,2} & 0 \\ 0 & 0 & q_{i,3} \end{bmatrix}, \quad R_i = cont_i.$$

In fact, the cost function of vehicle V_i is defined as the weighted sum of its tracking errors and the control input at step k , where the weights represent the priorities of the different objectives. In the function, the greater the weight is, the more important the objective will be, so the controller will tend to achieve

this objective in priority. For example, if the weight of spacing error is greater than that of velocity error, the controller will adjust the spacing at first, when the spacing error is small enough, the vehicle will begin to adjust its velocity.

In this paper, constraints like velocity limit, power limit, collision avoidance, and passenger comfort should also be considered. The feasible domain of control input is constrained and varying at each time step. Therefore, the studied problem is a constrained optimization problem. We utilize a penalty function technique (Parsopoulos *et al.* (2002)) to solve this constrained optimization problem through a sequence of unconstrained optimization problems.

The penalty function is defined using a trail-and-error method, and the idea is quite straightforward: a higher penalty value is given when a solution goes outside the feasible domain. In fact, the penalty value must be given properly: if it is too high, the optimization algorithm usually get trapped in local optima, while if it is too low, the feasible optimal solutions may not be found easily. The penalty value can be either stationary or non-stationary: for the former, the penalty value is fixed during the optimization process; and for the later, the penalty value is variant with the iteration number. In this paper, we use a non-stationary penalty value.

Another term for the penalty is added on the original cost function (2.15):

$$J_{constraint}(V_i) = J(V_i) + h(n) \cdot H(x_i) \quad (2.16)$$

where $J(V_i)$ is the original cost function that we defined in equation (2.15), $h(n)$ is the penalty value at the current iteration number n , and $H(x_i)$ is the penalty factor which is defined as bellow:

$$H(x_i) = \sum_{j=1}^m \theta(q_j(x_i)) \cdot q_j(x_i)^{r(q_j(x_i))} \quad (2.17)$$

where m is the number of constraints, and in this paper, we have speed limit, power limit, collision avoidance, and change rate of acceleration as the four con-

2. DECENTRALIZED PLATOON CONTROL

straints, hence $m = 4$. $q_j(x_i)$ are defined as follows:

$$\begin{aligned} q_1(x_i) &= \max(0, v_i - v_{limit}), \\ q_2(x_i) &= \max(0, \frac{P_{tractive,i}}{\eta_i} - P_{engine,i}), \\ q_3(x_i) &= \max(0, -\delta_i - d_i) \\ q_4(x_i) &= \max(0, |\Delta a_i| - |\Delta a|_{max}). \end{aligned} \quad (2.18)$$

$\theta(q_j(x_i))$ is a multi-stage assignment function, $r(q_j(x_i))$ is the power of the penalty function, and their forms are given as bellow:

$$\theta(q_j(x_i)) = \begin{cases} \nu_1, & \text{if } 0 < q_j(x_i) < \mu_1 \\ \nu_2, & \text{if } \mu_1 \leq q_j(x_i) < \mu_2 \\ \vdots & \\ \nu_n, & \text{if } q_j(x_i) \geq \mu_{n-1} \end{cases} \quad (2.19)$$

$$r(q_j(x_i)) = \begin{cases} 1, & \text{if } 0 < q_j(x_i) \leq 1 \\ 2, & \text{if } q_j(x_i) > 1. \end{cases} \quad (2.20)$$

The constraints on control input (2.5) is not included in the cost function, they can be added into the initialization procedure of the optimization algorithm.

Therefore, at each time step k , the contradictory control objectives can be summarized as to find the control input $u_i^*(k)$ which minimizes the cost function value. $U_i(k)$ is the feasible domain of control input with respect to all the constraints above.

$$u_i^*(k) = \arg \min_{u \in U_i(k)} [J_{constraint}(V_i)]. \quad (2.21)$$

2.3.2 Optimization algorithm

An optimization algorithm needs to be applied to solve the constrained optimization problem (2.16) that we have established. Due to the nonlinearity of the objectives and constraints, the optimal input could be difficult to find using deterministic algorithms. Thus, we tend to utilize heuristic algorithms which can handle both nonlinear objective and constraint functions without requiring gradient information. Actually, there are four popular heuristic algorithms derived from natural phenomena: Simulated Annealing Algorithm (SA), Genetic Algorithm (GA), Ant Colony Optimization Algorithm (ACO) and Particle Swarm Op-

timization Algorithm (PSO). Comparisons among these algorithms can be found in (Hassan *et al.* (2005); Meihong & Wenhua (2010); Selvi & Umarani (2010)).

The particular swarm optimization (PSO) is a population-based stochastic algorithm developed by Eberhart and Kennedy (Eberhart *et al.* (1995); Kennedy (2010); Stefanoiu *et al.* (2014)), and it is inspired by the social behavior of animal swarm like group of birds. In this algorithm, a particle is like an individual bird with flies through a multi-dimensional search space where each position in the space is evaluated by a fitness function; each particle adjusts its own trajectory towards its previous best position and the best position of all the particles in a local neighborhood (the neighborhood can refer to the whole swarm). Thus, a particle can benefit from the discovery of its neighbors and the previous discoveries. PSO has the best integrated performance among these four algorithms (Meihong & Wenhua (2010)), due to its relatively simple formulation to perform exploration and exploitation, thus lower computational cost can be achieved. Unlike genetic algorithm, there is no selection and mutation procedure, and all the particles survive till the end.

The pseudo code of PSO is summarized in Algorithm 1. At the beginning, the swarm is initialized in the feasible area of the moment with randomized positions and velocities. Basically, each particle in the swarm updates its velocity and position according to the following two equations (2.22) and (2.23) at the current iteration n .

$$V_{pi}^{n+1} = \phi \cdot V_{pi}^n + c_1 \cdot r_1 \cdot (P_{bi} - X_{pi}^n) + c_2 \cdot r_2 \cdot (P_{gb} - X_{pi}^n) \quad (2.22)$$

$$X_{pi}^{n+1} = X_{pi}^n + V_{pi}^n \quad (2.23)$$

where pi refers to the i th particle and $i = 1, 2, \dots, N_{swarm}$, N_{swarm} is the population of the swarm.

The equation (2.22) is used to update the velocity of the particle based on the current velocity V_{pi}^k , the best position P_{bi} that the particle has reached so far, and the current best position P_{gb} among the neighborhood. ϕ is the inertia weight, r_1 and r_2 are two independent random number, and c_1 and c_2 are called the learning factors. The three parts on the right side of the equation (2.22) are separately called the inertia part, the cognition part and the social part. The first part represents the exploration process while the second and the third part represent the exploitation process of the algorithm. The inertia weight make the particle

2. DECENTRALIZED PLATOON CONTROL

Algorithm 1 Pseudo Code of PSO

Input:

Swarm size N_{swarm} , searching area's dimension D_{search} , area's boundaries (X_{min}, X_{max}) , velocity limit V_{max} ;

Output:

Global best solution P_{gbest} ;

- 1: Initialization of particles' positions and velocities within the boundaries;
 - 2: **repeat**
 - 3: **for** particle pi in N_{swarm} **do**
 - 4: **if** $f(X_{pi}) < f(P_{bi})$ **then**
 - 5: $P_{bi} = X_{pi}$;
 - 6: **end if**
 - 7: **if** $f(P_{bi}) < f(P_{gb})$ **then**
 - 8: $P_{gb} = P_{bi}$;
 - 9: **end if**
 - 10: **end for**
 - 11: **for** particle pi in N_{swarm} **do**
 - 12: Update the velocity of the particle in every dimension by (2.22), and adjust the velocity if it exceeds the boundaries;
 - 13: Update the position of the particle in every dimension by (2.23), and adjust the position if it exceeds the boundaries;
 - 14: **end for**
 - 15: **until** $iteration > Max_iteration$
-

Table 2.1: Vehicles' parameters setting.

| Vehicle No. | p_i (m) | v_i (m/s) | l_i (m) | γ_i | h_i (s) | $d_{max,i}$ (m) | δ_i (m) |
|----------------|--------------|----------------|--------------|------------|--------------|--------------------|-------------------|
| V_1 | 42.95 | 10 | 5 | 1.0 | 0.40 | 4.5 | - |
| V_2 | 35.65 | 10 | 4 | 1.1 | 0.30 | 3.0 | -3 |
| V_3 | 23.25 | 10 | 4.5 | 1.0 | 0.40 | 4.5 | -0.6 |
| V_4 | 10.65 | 10 | 3.5 | 1.1 | 0.30 | 3.0 | 2.8 |
| V_5 | 0.00 | 10 | 4.5 | 1.1 | 0.35 | 3.5 | -1.2 |

tend to explore new areas of the searching space since it cannot easily change its velocity towards the best solutions. The three weights should be tuned carefully, so that PSO can establish trade-off between exploration and exploitation. If not, the algorithm may suffer from premature convergence, trapping in a local optima and stagnation. After the velocity update, the equation (2.23) updates the position of the particle.

2.4 Simulation of algorithm

In this section, our PSO-based platoon control algorithm is tested. In the designed scenario, a platoon is formed by five vehicles in which V_1 is the platoon leader. And their initial positions, velocities and other parameter settings can be found in Table 2.1. All the vehicles are traveling at $v = 10m/s$ in the beginning, and according to equation 2.8, the spacing errors of the following vehicles can be calculated, which is also shown in 2.1. The control input u_i is limited as $u_i \in [-1.5, 1.5]$. And the changing rate of acceleration is limited as $\Delta a_i \in [-0.5m/s^3, 0.5m/s^3]$. The total simulation time is $T = 50s$, and the sampling time is set to $T_s = 0.02s$. For the PSO algorithm, the parameters are set as followings: $N_{swarm} = 10$, $Max_iteration = 30$, $\phi = 0.729$, $c_1 = 2.988$, $c_2 = 2.988$. The simulation is executed in *Matlab* 2013a on a computer with *IntelCoreTMi5 - 2410M 2.3GHz*.

The leading vehicle V_1 decelerates with $u = -1$ from $t = 6s$ to $t = 12s$, then it accelerates with $u = 1$ from $t = 21s$ to $t = 27s$. In the platoon, V_2 only receives the information from V_1 , because its nearest preceding vehicle is the platoon leader. The other following vehicles receive information from both V_1

2. DECENTRALIZED PLATOON CONTROL

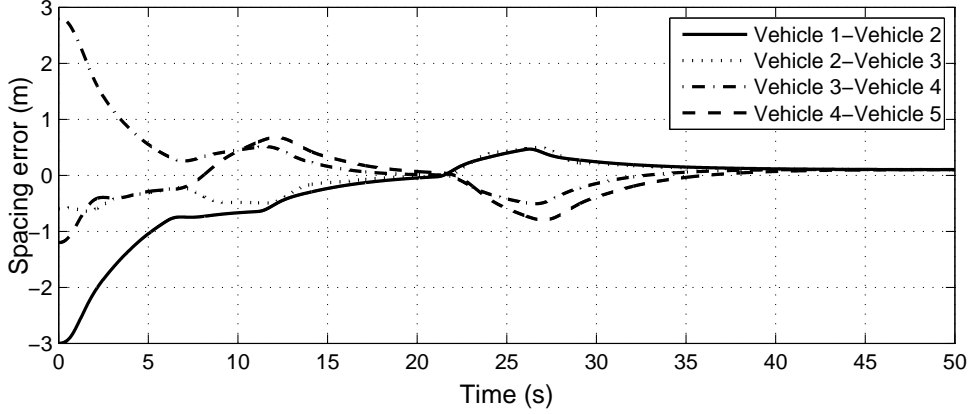


Figure 2.2: Spacing error in the platoon

and its nearest preceding vehicle. The following vehicles try to find the optimal control input within the defined constraints and eliminate all the tracking errors currently exist and those caused by the leader's motions. The simulation results are shown in the figures 2.2, 2.3, 2.4 and 2.5.

As shown in the figures, in the first five seconds, the following vehicles try to reduce the existing spacing error, because spacing error is dominant for now in the current cost. Thus the vehicles must take actions, then the spacing error decrease quickly while the velocity error and acceleration error increase. At $t = 5s$, all the tracking errors are reduced. From $5s$ to $10s$, the spacing error become lower than $1m$ while velocity error augment because V_1 begins to accelerate. At $t = 12s$, the acceleration error is reduced compared to the error at $t = 6s$. From $12s$ to $21s$, when the leading vehicle is cruising at constant speed, all the tracking errors tend to attenuate. Between $21s$ and $27s$, all the following vehicles are capable to track the leader's motion. From $t = 45s$, all the errors are eliminated. The platoon is stabilized when facing the influence of the leader's variate behavior, and rear-end collision is absolutely avoided during the whole process.

In this scenario, the computational cost of the PSO algorithm is quite low: by applying the setup mentioned previously, to find a solution for a vehicle in one time step, it costs at most $0.032s$, at least $0.015s$, on average $0.017s$ which is smaller than the sampling time $T_s = 0.020s$. However, it still cannot be guaranteed that the vehicle can find a solution during the sampling time. In the

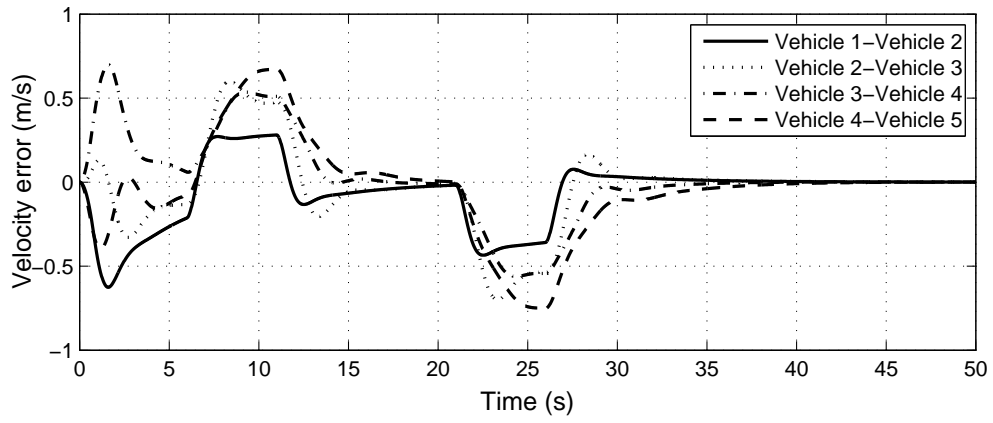


Figure 2.3: Velocity error in the platoon

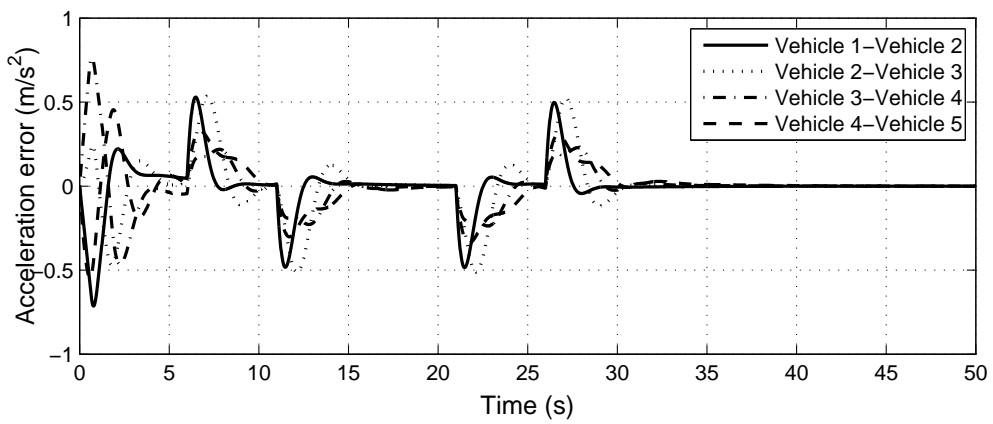


Figure 2.4: Acceleration error in the platoon

2. DECENTRALIZED PLATOON CONTROL

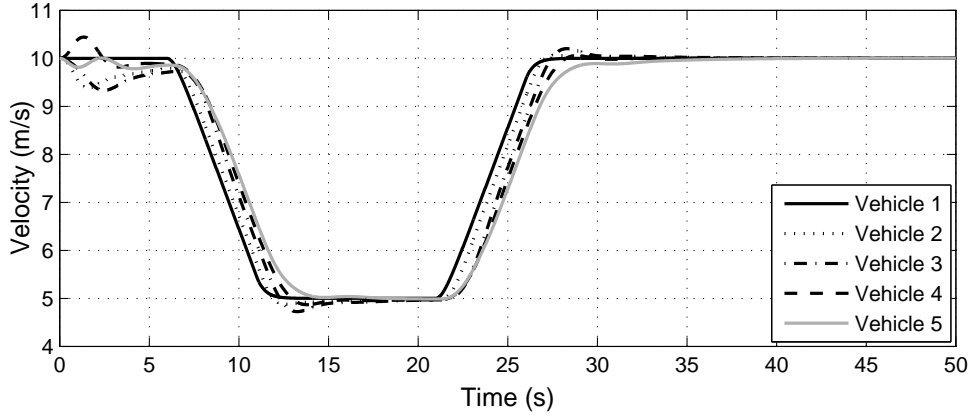


Figure 2.5: Velocities of vehicles

near future, when higher computational ability can be obtained more cheaply, this PSO based platoon control algorithm can be implemented for real-time usage.

2.5 Conclusion

In this chapter, a PSO based platoon control algorithm is described. At each time step, a vehicle tries to find the optimal control input to reduce the three tracking errors in the platoon, according to the information coming from the platoon leader and the nearest preceding vehicle via V2V communications. The vehicles' dynamic capabilities and passenger comfort are also considered in this algorithm.

Simulation results prove that our algorithm is effective to control vehicles in the platoon. The platoon can be stabilized with reduced tracking errors, and influence from the acceleration/deceleration of leading vehicle to the rest of the platoon can be eliminated.

In future research, additional properties such as data lose, delay and disturbance in communication will be considered to make the algorithm more robust when facing realistic scenarios.

Chapter 3

Improving Traffic Efficiency via V2X and CACC

Contents

| | | |
|------------|-------------------------------|-----------|
| 3.1 | Introduction | 49 |
| 3.2 | Problem statement | 51 |
| 3.2.1 | Scenario description | 51 |
| 3.2.2 | Design objectives | 53 |
| 3.3 | Methodology | 54 |
| 3.3.1 | Overview of CACC-VI | 54 |
| 3.3.2 | Platoon Reorganization | 56 |
| 3.3.3 | Mixed Control Strategy | 65 |
| 3.4 | Simulation Experiments | 66 |
| 3.4.1 | Setup | 66 |
| 3.4.2 | Simulation results | 68 |
| 3.5 | Conclusion | 75 |

3.1 Introduction

V2X technology, including Vehicle to Vehicle (V2V) communication via wireless ad hoc network and Vehicle to Infrastructure (V2I) via Dedicated Short-Range

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

Communication (DSRC), is considered to be the next step to construct the intelligent transportation system (ITS). Instead of detecting the environment all by the vehicle itself, this technology enables the communication between vehicles themselves and with infrastructure along the road, thus autonomous systems can be upgraded into cooperative systems. Possible applications of V2X in terms of safety, information dissemination and efficiency are illustrated in (Hartenstein & Laberteaux (2010)).

For instance, the cooperative collision warning systems proposed in (Sengupta *et al.* (2007)) provides warnings or situation awareness displays to drivers based on information about the motions of surrounding vehicles obtained via V2V communications from those vehicles, without use of any ranging sensors. In (Wedel *et al.* (2009)), a traffic congestion recognition and avoidance algorithm is proposed which utilizes V2X technology to share information about the current local traffic situation and to use this information to optimize the routes. A velocity planning algorithm is designed in (Mandava *et al.* (2009)) based on traffic signal information obtained by V2I communication, the idea is to minimize the acceleration/deceleration rates while ensuring the vehicle get through the intersection without coming to a full stop. Researches shows that vehicle fuel consumption and emissions can be reduced by avoiding sharp acceleration/deceleration patterns and decreasing idling period (Asadi & Vahidi (2011); Madireddy *et al.* (2011); Mandava *et al.* (2009)). Similarly, different optimal speed advising algorithms based on V2I have been proposed by other researchers (Kamalanathsharma & Rakha (2012); Kishore Kamalanathsharma & Rakha (2014); Malakorn & Park (2010)). In these algorithms, the maneuver responsibility is left to the driver, and no direct inter-vehicle communication is considered.

In this chapter, a decentralized cooperative adaptive cruise control algorithm using V2X for vehicles in the vicinity of intersections (CACC-VI) is proposed. The main idea is to reorganize the already existed platoons according to the traffic signal timing and the heterogeneous characteristics and capabilities of vehicles to take advantage of the unused road capacity in order to improve the throughput of intersection. Meanwhile, many other features and limits are also taken into account, such as safety, fuel consumption, speed limit, passenger comfort, etc. Each vehicle in the platoon can obtain the position, velocity, and acceleration information from the platoon leader and the preceding vehicle. A particular

swarm optimization (PSO) algorithm with constraints is used to find the optimal inputs at every step to eliminate the tracking errors. When approaching the intersection, an “opportunity space” is calculated according to the traffic signal timing and platoons’ future positions. An optimal trajectory is calculated for each platoon leader in consideration of fuel economy within the constrained domain. A space arrangement approach is designed in order to maximize the number of vehicles which could get through the intersection during the current green phase and minimize the distance to be covered for each vehicle. If the opportunity space is larger than a platoon’s length and the all the platoon members are capable to produce the desired trajectory, this platoon can get through the intersection within the limited period by accelerating. If not, the platoon splits into two parts, the first one gets through the intersection by accelerating, while the second one decelerates to a lower speed to get through when the lights turns green. If the opportunity space has not been fully distributed, a vehicle can choose to join in the preceding platoon to take advantage of this space after verifying its own capability.

The rest of the chapter is organized as follows. In section 3.2, the scenario of our research is described, the control objectives are established. In section 3.3, a platoon control algorithm using constrained PSO is given, and a platoon reorganization method, including our space arrangement approach and trajectory planning algorithm, is introduced in details. In section 3.4, simulation results are presented to prove the validation of our algorithm. Finally, some concluding remarks and perspectives are given in Section 3.5.

3.2 Problem statement

3.2.1 Scenario description

The object of our study is the group of vehicles approaching an intersection controlled by a signal timing plan which is produced by the intersection manager using some kind of superior signal control algorithm to coordinate the vehicles coming from different directions. The traffic signal control systems have been widely investigated (He *et al.* (2012); Jiang *et al.* (2006); Mirchandani & Wang (2005)) and will not be covered in the scoop of this chapter. In our research, the intersection manager does not need to plan the trajectory of each vehicle

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

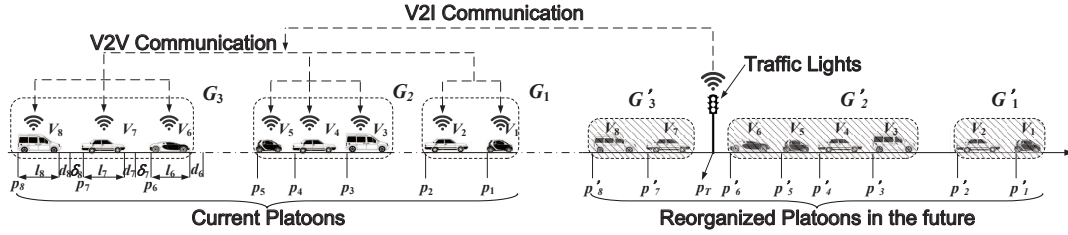


Figure 3.1: Scenario illustration

because its computational capability is usually limited, it produces only near optimal schedule including the “safe time window” and the “danger time window” to permit or prohibit the vehicles in a certain lane to pass. In order to facilitate the expression, we consider the “safe time window” as a “green light”, while the “danger time window” as a “red light”. The plan can be expressed directly by physical traffic lights or be sent to vehicles via V2I communication.

Each vehicle is supposed to be equipped with V2V device which is capable to send/receive the information such as position, velocity and acceleration to/from the corresponding vehicle. The intersection manager can broadcast the signal timing to the vehicles in the corresponding lane. We assume that lane-shifting maneuver has been completed before arriving at the effect area of our algorithm, which is defined as the area can be covered by the V2I communication. So all the lanes with green light can be considered as independent. Therefore, we can extract a single lane from the intersection as our study object in which the vehicles’ trajectories should be planned according to the signal time plan already designed by the intersection manager.

To briefly explain the idea of our method, we take the scenario in Figure 3.1 as an instance. All the vehicles circulate from the left to the right. The vehicles which have passed the intersection are not displayed in the figure and they are not considered in our algorithm. Platoons have been formed already when entering the effect area of our algorithm. Instead of a homogeneous vehicle group in which all the vehicles are of the same kind, a heterogeneous group is discussed in this chapter where vehicles can be different in type, size, engine power, and so on. Besides, the difference in driver settings can also lead to the heterogeneity of vehicles.

In Figure 3.1, there are three platoons formed by eight vehicles approaching

the intersection in the same lane. The i th vehicle is denoted as V_i , while the i th platoon is denoted as G_i , and the traffic lights are installed at p_T which indicates the position of intersection entrance. The traffic signal timing is designed by the intersection manager and is broadcasted to the vehicles. The first platoon G_1 can get through the intersection during the current green light by maintaining its current speed, while the rest platoons G_2 and G_3 cannot. A ameliorated solution can be found by our algorithm: the vehicle V_6 can choose to join in the platoon G_2 , and the new platoon G'_2 accelerates to pass during the limited period. At the same time, the new platoon G'_3 decelerates to a smaller velocity so that it can get through the intersection when the light turns green again while full stop is avoided.

3.2.2 Design objectives

According to the vehicles' behaviors in this scenario, the design objectives of our algorithm are established as followings:

- Instead of a centralized algorithm, a distributed one should be proposed in order to reduce the computational cost at the intersection manager by distributing the cost to vehicles;
- In consideration of the heterogeneity of vehicles, it is necessary to design an algorithm, in which each vehicle must make decisions individually based on its own features, the driver's desired response and real-time traffic information from other vehicles and the intersection manager;
- The number of vehicles which can get through the intersection during the current green phase should be maximized;
- The platoons should be flexible, a vehicle can choose to join in a platoon or depart from it;
- Trajectories need to be planned for vehicles in different cases, meanwhile their fuel consumption should be minimized;
- Constraints like speed limit and power limit of the vehicle must be respected;

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

- The velocity of a platoon should be the same to its preceding platoon to prevent collision after getting through the intersection;
- The safety of the whole process must be guaranteed, thus a safety space must be maintained to avoid rear-end collision;
- The passengers' comfort should be taken into account, in other words, sharp change of the acceleration should be averted.

It can be noticed that the last two objectives are the same as the objectives in the platoon control algorithm.

3.3 Methodology

In this section, based on the control objectives that we proposed and the models of vehicle and platoon, our CACC-VI algorithm is introduced in details.

3.3.1 Overview of CACC-VI

First of all, an overview of CACC-VI is given as shown in Figure 3.2. The conception of “Opportunity Space” is proposed to evaluate the redundant road capacity, and three sub-algorithms are designed to establish the integrated CACC-VI algorithm. The first sub-algorithm is PSO based platoon control algorithm (2.3), which is applied by the vehicles in a platoon except the leader to eliminate the tracking errors that we defined in the previous section; the second one is the trajectory planning algorithm (3.3.2.1) which aims at finding an optimal input profile for a vehicle V_i to arrive at the decided future position p'_i at time t with a specific velocity v ; the third one is space arrangement (3.3.2.2), which is utilized in coordination with the trajectory planning to distribute the opportunity space to as many vehicles as possible, while minimize the distance to be covered by each vehicle.

At first, the intersection manager broadcasts the signal timing to the vehicles in the lane, while the platoons send their current positions and velocities to the manager; then the opportunity space is calculated and distributed to vehicles using space arrangement approach and trajectory planning algorithm; if there is a vehicle in the accelerating platoon which cannot find an input profile, the last

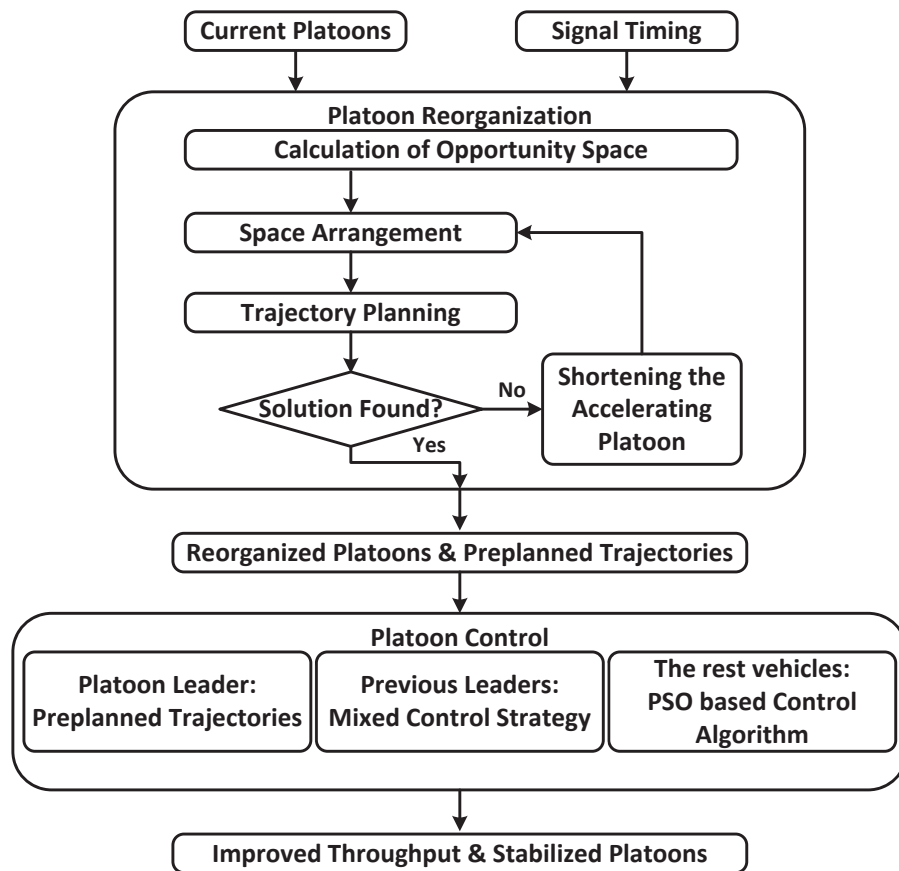


Figure 3.2: Overview of CACC-VI

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

vehicle in the platoon is reorganized into the decelerating platoon; then the process is launched again until all the vehicles in the accelerating platoon can find a solution. After that, the platoon reorganization process is finished. The newly reorganized platoons are controlled by different strategies: the platoon leaders are controlled by following the preplanned trajectories; the previous platoon leaders, which are merged into a new platoon, are controlled by a mixed control strategy that we define in the section 3.3.3; the other vehicles are controlled by the PSO based platoon control algorithm. By using the CACC-VI algorithm, the intersection throughput can be improved by making use of the redundant road capacity; and platoons' motion is controlled with stabilized behaviors in consideration of fuel consumption, heterogeneous vehicular features and passenger comfort.

3.3.2 Platoon Reorganization

In this subsection, we focus on how to reorganize the platoons in consideration of the road capacity and the power limit of vehicles to maximize the throughput of the intersection. Several problems need to be solved:

- How to quantify the redundant road space that can be taken advantage of?
- How many vehicles to which can be distributed this space?
- How to verify if a vehicle is capable to arrive at the decided position with a certain velocity at the definitive moment?
- If so, how to find an optimal trajectory for this vehicle to save fuel?

These questions can be responded in our trajectory planning and space arrangement algorithms.

3.3.2.1 Trajectory planning algorithm

This algorithm is designed to find the solution of the last two questions. First of all, vehicles should be divided into three categories: vehicles which will get through the intersection with a constant velocity; vehicles which will get through the intersection by accelerating; the vehicles which will decelerate to a smaller velocity till the lights turns green again. These three categories are separately given the labels C_1 , C_2 and C_3 . Our objective is to plan the trajectory for the

vehicles with labels C_2 and C_3 . Basically, a vehicle needs to arrive at the decided position with a certain velocity at the definitive moment; at the same time, the constraints, like speed limit, power limit and stop-free, need to be followed; besides, the fuel consumption should be considered as well. Hence a feasible trajectory with least fuel consumption should be found.

In our scenario, the platoon of vehicles with label C_2 needs to accelerate at first, then to travel at a constant velocity; as the distance to the nearest preceding platoon of the whole platoon should be reduced, the platoon must have a greater cruise speed; at last, the platoon should decelerate to make its velocity in accordance with the preceding platoon. Thus we can divide the whole process into three sections: acceleration, cruise and deceleration. The platoon of vehicles with label C_3 has a similar three-section trajectory: the platoon should decelerate to a lower cruise speed, and then accelerate to the original speed to pass the intersection. The duration of the three sections are separately noted as t_1 , t_2 and t_3 .

In this chapter, we consider an uniform input for the acceleration and the deceleration section to simplify the trajectory planning process. And the acceleration and deceleration process have the opposite control input, while the input during the cruising process is zero. The research (Mandava *et al.* (2009)) shows that fuel consumption can be reduced by avoiding sharp acceleration/deceleration maneuver, hence the input value should be minimized in order to have a environment friendly driving process. Based on their algorithm, several modifications are proposed according to our scenario. Once the optimal input is found, we can determine the acceleration/deceleration at any time, and the velocity profile can be planed. We note the current position and velocity of the vehicle V_i as p_i and v_i , its future position as p'_i , the time to the moment when the light turns red/green as T_r/T_g . Then the trajectory planning problem can be translated into solving the following optimization program:

$$\text{minimize } |u_i| \tag{3.1}$$

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

with constraints for vehicles with label C_2 :

$$t_1 + t_2 + t_3 \leq T_r, \quad t_1, t_2, t_3 \in (0, T_r) \quad (3.2)$$

$$v_i(t_1 + t_2 + t_3) = v_{target} \quad (3.3)$$

$$\frac{P_{tractive,i}(v_i(t_1))}{\eta_i} \leq P_{engine,i} \quad (3.4)$$

where v_{target} is the velocity of the last platoon of vehicles with label C_1 .

Constraints for vehicles with label C_3

$$t_1 + t_2 + t_3 \geq T_g, \quad t_1, t_2, t_3 \in (0, T_g) \quad (3.5)$$

$$v_i(t_1 + t_2 + t_3) = v_{original} \quad (3.6)$$

where $v_{original}$ is the original velocity of the platoon.

Constraints for all the vehicles

$$d(v_i(0), u_i, t_1) + d(v_i(t_1), 0, t_2) + d(v_i(t_1 + t_2), -u_i, t_3) = p'_i - p_i \quad (3.7)$$

$$0 < v_i(t_1 + t_2) \leq v_{limit} \quad (3.8)$$

where $d(v_i(t), u_i, t_j)$ is the distance covered by the vehicle with initial speed $v_i(t)$ and control input u_i , during time t_j .

The constraints (3.2, 3.5), (3.3, 3.6) and (3.7) are separately the constraints of time, velocity and position. (3.4) and (3.8) are the power limit and the speed limit. The distance and velocity of each moment are calculated by using the vehicle's kinematic model (2.1). For the vehicles with label C_2 , the greatest demanded tractive power is at time t_1 . And for vehicles with label C_2/C_3 , the greatest/smallest velocity is achieved at time $t_1 + t_2$, thus this velocity should be constrained below the speed limit and above zero to respect the traffic rules and to avoid total stop.

This trajectory planning algorithm can also be used to verify if a vehicle with label C_2 is capable to follow an accelerating trajectory within constraints. If an input value can be found for vehicle V_i , it means V_i is capable to get through the intersection within the current green phase by accelerating, thus V_i can be added into the accelerating platoon. If not, the vehicle is abandoned by the platoon, and it should change its label to C_3 to get through the intersection till the light become green again.

3.3.2.2 Space arrangement approach

This part is used to solve the first two problems that we propose in the beginning of this section. In order to apply the trajectory planning algorithm that we introduce before, the vehicle's future position p'_i must be decided. Since the traffic signal timing is broad-casted to all the vehicles within the V2I range, each platoon can find out if it can totally get through the intersection in time by maintaining the current speed. If the last vehicle of the last platoon which can pass is V_j , then we can define the space between the future position p'_j of V_j and the position of traffic lights p_T as the "opportunity space" S .

$$S = p'_j - p_T = p_j + v_j * T_r - p_T \quad (3.9)$$

where p_T is the position of the traffic lights; for unsignalized intersection, it is the position of the intersection entrance.

As we introduced in the platoon model, a vehicle needs a safety spacing d_i to the preceding one, which is a function of its own features and velocity. We can define the "demanding space" of V_i as the sum of the vehicle length and the safety spacing according to the targeted velocity v_{target} .

$$S_{D,i} = l_i + d'_i = l_i + \gamma_i \cdot d_{min,i} + h_i \cdot v_{target} \quad (3.10)$$

The idea of our space arrangement approach is to distribute the "opportunity space" to as many vehicles as possible according to their demanding spaces, while to reduce their distances to the future positions.

In the platoon reorganization procedure of our algorithm, after receiving the broad-casted traffic signal timing via V2I communication, by evaluating the future position of the last vehicle in the platoon, each platoon can predict if it can pass by maintaining the current speed. If so, each vehicle in this platoon is classified into the first category, and its label is set to C_1 ; if not, it is temporarily classified into second category with label C_2 . Then the last vehicle of the platoon with label C_1 sends immediately a message to the intersection manager indicating its future position, thus the opportunity space can be derived. The vehicle's response after it receives a broad-casted message is summarized in Algorithm 2.

After the intersection manager has decided the opportunity space, it sends immediately this information combined with the targeted velocity to the first

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

Algorithm 2 Vehicle's response to a broad-casted message

Input:

The broad-casted message M_{broad} coming from the intersection manager;

Output:

The vehicle V_i 's label and message to intersection manager;

- 1: **if** The message is well received **then**
 - 2: Based on the signal timing T_r and the current velocity v_i , predict the future position p'_i of the last vehicle of the platoon;
 - 3: **if** The Platoon can pass in time, $p'_i > p_T$ **then**
 - 4: Set labels of vehicles in this platoon as C_1 ;
 - 5: Send a message to the intersection manager indicating the future position, so that the opportunity space can be calculated;
 - 6: **else**
 - 7: Set labels of vehicles in this platoon as C_2 ;
 - 8: Send no message;
 - 9: **end if**
 - 10: **end if**
-

Table 3.1: Upstream message

| |
|-------------------|
| Message type |
| Sender vehicle ID |
| Remaining space |

platoon with label C_2 . Then each vehicle in this platoon can find its own demanding space according to the targeted velocity. A message propagates from the leader to the tail of the platoon in order to distribute this opportunity space. We call this message the "upstream message", because it goes from downstream to upstream of the travel direction. The message is formed by three fields separately indicating the message type, the sender's ID, and the remaining space which is the opportunity space in the upstream message it receives minus its demanding space. The structure of an upstream message is shown in Table 3.1.

If a vehicle in the platoon receives an upstream message in which the remaining

Table 3.2: Downstream message

| |
|-------------------|
| Message type |
| Vehicle ID |
| Demanding spaces |
| Corresponding IDs |

opportunity space is less than its demanding space, the vehicle and its following ones must depart from the current platoon and these vehicles should change their labels from C_2 to C_3 . Several operations need to be done in this case: firstly, this vehicle sends an abandoning message to its following vehicles in the platoon so that they can classify themselves to the third category; secondly, this vehicle sends a V2I message to the intersection manager indicating that the opportunity space has been totally distributed, and this message will be rebroadcasted to the following platoons with label C_2 , so that they can be informed to change their labels to C_3 ; thirdly, this vehicle sends a message to the nearest preceding vehicle which is composed of four fields as shown in Table 3.2. This message is called the "downstream message" because of its opposite propagation direction to the upstream message. For a vehicle with label C_3 , the "Demanding spaces" field is set to 0; while for a vehicle with label C_2 , this field is the set of the demanding spaces of all its following vehicles with label C_2 and itself. The sum of this field is the future position of the nearest preceding vehicle. This setting can ensure that the vehicles' future positions are the nearest to the current ones, thus the distances to be covered are minimal, and fuel consumption is minimized. The future position of vehicle V_i can be expressed as:

$$p'_i = \sum_{j>i, j \in C_2} S_{D,j} + p_T \quad (3.11)$$

If the upstream message propagates to the tail of the platoon while the remaining opportunity space is still larger than the demanding space, it means the opportunity space has not been totally distributed, the platoon sends a V2I message to the intersection manager indicating the remaining opportunity space. Then the manager sends this remaining space to the next platoon to restart the distribution process till the space is totally distributed. The vehicle's response to an upstream message is summarized in Algorithm 3.

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

Algorithm 3 Vehicle's response to an upstream message

Input:

The upstream message M_{up} coming from the nearest preceding vehicle;

Output:

The vehicle V_i 's label, upstream message, downstream message and V2I message;

- 1: **if** The message is well received **then**
 - 2: **if** The remaining opportunity space is greater than the demanding space **then**
 - 3: Calculate the new remaining opportunity space;
 - 4: **if** V_i is the tail of platoon **then**
 - 5: Send a V2I message to the intersection manager indicating the remaining opportunity space;
 - 6: **else**
 - 7: Send an upstream message to the nearest following vehicle V_{i+1} ;
 - 8: **end if**
 - 9: **else**
 - 10: Abandon joining in the vehicle platoon, set its label as C_3 and send the abandoning messages to the following vehicles;
 - 11: Set the demanding space as 0 and send the downstream message to the nearest preceding vehicle V_{i-1} ;
 - 12: Send a V2I message to the intersection manager indicating the opportunity space is totally distributed;
 - 13: **end if**
 - 14: **end if**
-

Algorithm 4 Vehicle's response to a downstream message

Input:

The downstream message M_{down} coming from the nearest following vehicle;

Output:

The vehicle V_i 's label and a downstream message or a confirmation message or an abandoning message;

- 1: **if** The message is well received **then**
 - 2: Calculate its future position using the information in the downstream message;
 - 3: Lunch the trajectory planning procedure;
 - 4: **while** Solution does not exist && V_i is not the last vehicle in the current platoon **do**
 - 5: Abandon the last vehicle, recalculate the future position p'_i and lunch the velocity planning again;
 - 6: **end while**
 - 7: **if** Solution exists **then**
 - 8: **if** Vehicle V_i is the platoon leader **then**
 - 9: Send the confirmation message to all the vehicles with label C_2 ;
 - 10: **else**
 - 11: Send the downstream message to the nearest preceding vehicle V_{i-1} ;
 - 12: **end if**
 - 13: **else**
 - 14: Abandon joining in the platoon;
 - 15: **end if**
 - 16: **end if**
-

When a vehicle with label C_2 receives a downstream message, it sums up all the items in the "Demanding spaces" to obtain its own future position, and then lunches the trajectory planning algorithm to find an optimal input profile. If the solution exists and the vehicle is the platoon leader, it sends a confirmation message to all the vehicles with label C_2 , and a new platoon is formed; if the solution exists and vehicle is not the leader, it adds information to the last two fields of the downstream message and sends it to the nearest preceding vehicle; if the solution does not exist, the vehicle sends an abandoning message to the last vehicle with label C_2 , recalculates the future position and lunch the trajectory

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

planning again; this procedure is repeated until the vehicle finds a solution. When the vehicle becomes the last one in the platoon but the solution still does not exist, it must depart from the accelerating platoon and send a downstream message with zero demanding space. This process is repeated until all the vehicles in the platoon are capable to arrive at the decided future positions with respect to the speed and power limits. The vehicle's response to a downstream message is summarized in Algorithm 4.

An example of the space arrangement approach is given in Figure 3.3. We assume that V_1 is the last vehicle with label C_1 , thus we can define the opportunity space as shown in the figure; with the propagation of upstream messages, the opportunity space is distributed to five vehicles; the vehicle V_7 has to abandon joining in the accelerating platoon because its demanding space is greater than the remaining opportunity space, therefore V_7 sends abandon messages to its following vehicles, a downstream message to V_6 , and a V2I message to the intersection manager; then each vehicle temporally with label C_2 begins to plan its input profile after receiving a downstream message in which the relative information can be used to calculate its future position; the vehicle V_6 can find an optimal input profile while the vehicle V_5 can't, therefore V_5 sends an abandoning message to V_6 , and the trajectory planning is lunched again; the solution still does not exist, therefore V_5 sends a downstream message with a zero demanding space to V_4 ; V_2 , V_3 and V_4 are all capable to find the solution according to their future positions, therefore a platoon of three vehicles is formed.

The future positions of vehicles with label C_3 are quite easy to be decided: the first vehicle in C_3 is denoted as V_k , and its future position at T_g is $p_T - l_k$, because it can only enter the intersection after T_g ; the future positions of the rest can be decided by:

$$p'_i = p_T - l_k - \sum_{j>k, j \in C_3} S_{D,j} \quad (3.12)$$

where $S_{D,j}$ is the demanding space of a vehicle V_j with label C_3 :

$$S_{D,j} = l_j + d'_j = l_j + \gamma_j \cdot d_{min,j} + h_j \cdot v_{original} \quad (3.13)$$

Then the trajectory of vehicles in C_3 can be decided by the trajectory planning algorithm. The platoon should slow down to a smaller velocity then accelerates

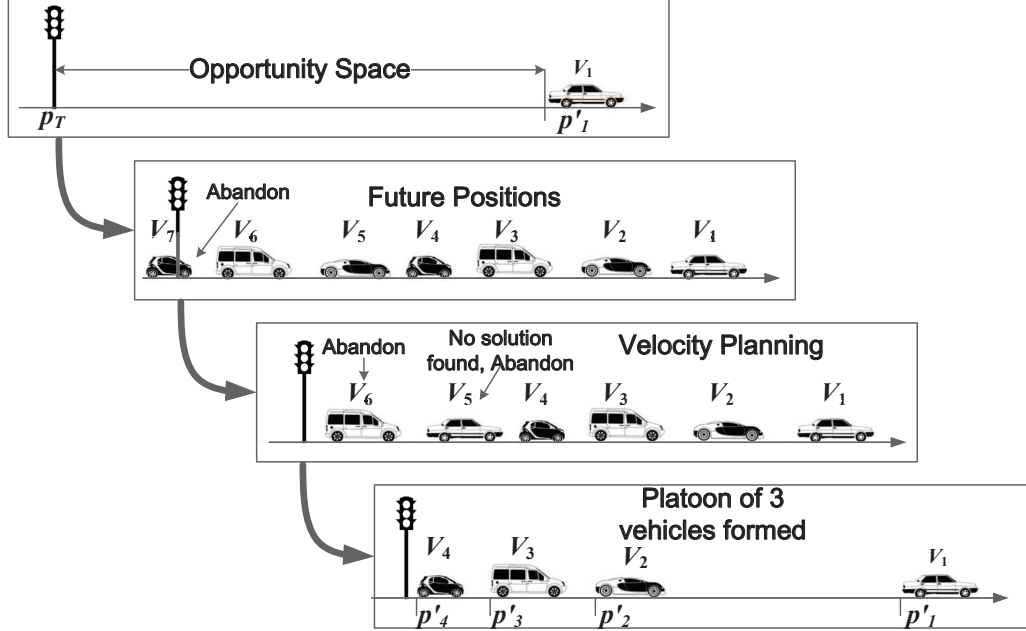


Figure 3.3: An example for space arrangement approach

to its original velocity when the light turns green again. If an optimal control input can be found, the vehicle can get through the intersection without full stop.

By using the trajectory planning algorithm and the space arrangement approach, all the four questions can be responded. The platoons are reorganized and their trajectory are planned separately: the platoons of vehicles with label C_1 get through the intersection with constant velocity; all the vehicles with label C_2 reform a new platoon, and this platoon gets through the intersection by accelerating; all vehicles with label C_3 reform another new platoon which gets through the intersection by the next green phase.

3.3.3 Mixed Control Strategy

In fact, the trajectories of vehicles in the platoons with label C_2 and C_3 are planned in advance, and the vehicles in these platoons are capable to arrive at the decided position with a certain velocity at the definitive moment; however, the trajectories are planned individually for each vehicle according to the heterogeneous characteristics, without any consideration of the existence of other

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

vehicles; therefore, a collision-free procedure cannot be guaranteed by following these preplanned trajectories. Thus, in our algorithm, only a platoon leader's preplanned trajectory is kept, while the rest vehicles in the platoon are still piloted by the PSO based platoon control algorithm with V2V communication.

For the vehicles newly reorganized in a platoon, a mixed strategy is designed: they are treated as a "sub-platoon" in the beginning, which merges into its preceding platoon when the spacing error becomes small enough. In other words, the previous platoon leader utilizes the its preplanned trajectory at first, then it switches to the PSO based control strategy when the distance to the preceding vehicle is small enough; and the other vehicles in the "sub-platoon", if there is any, are always controlled by the PSO based control algorithm. The switch from preplanned trajectory to PSO based algorithm is not reversible. The mixed control strategy can be expressed as following:

$$u_i(k) = \begin{cases} u_{preplanned,i}(k), & \text{if } \delta_i(k) \geq E \\ u_{PSO,i}(k), & \text{if } \delta_i(k) < E \end{cases} \quad (3.14)$$

where E is the threshold of switch.

3.4 Simulation Experiments

3.4.1 Setup

In this section, the CACC-VI algorithm is tested and verified by a designed scenario, in which, nine vehicles circulate in the same lane, the current phase of the traffic lights is green, and it turns to red in $T_r = 18s$, then the next green phase is at $T_g = 36s$. The vehicles are divided into three platoons: $V_1 \sim V_3$ are in platoon G_1 ; $V_4 \sim V_6$ are in platoon G_2 ; and $V_7 \sim V_9$ are in platoon G_3 . Vehicles V_1 , V_4 and V_7 are separately the leaders of these three platoons. It is assumed that all the vehicles are cruising at the same constant velocity $v = 10m/s$ in the beginning, thus the targeted velocity v_{target} and the original velocity $v_{original}$ are equal to $10m/s$, therefore the platoons with labels C_2 and C_3 should enter the intersection with velocities of this value. The platoons are controlled by the proposed control algorithm, and the tracking errors in each platoon at $t = 0s$ including spacing error, velocity error and acceleration error are all zeros. The road speed limit is set to $v_{limit} = 50km/h$ ($13.89m/s$) which

3.4 Simulation Experiments

| Vehicle No. | p_i (m) | l_i (m) | $P_{engine,i}$ (kw) | τ_i (s) | h_i (s) | $d_{max,i}$ (m) | γ_i | A_i (m ²) | Platoon |
|-------------|-----------|-----------|---------------------|--------------|-----------|-----------------|------------|-------------------------|---------|
| V_1 | -80.00 | 5.0 | 150 | 0.45 | 0.40 | 4.5 | 1.0 | 1.5 | G_1 |
| V_2 | -90.80 | 4.5 | 140 | 0.30 | 0.30 | 3.0 | 1.1 | 2.0 | G_1 |
| V_3 | -103.30 | 4.0 | 130 | 0.45 | 0.40 | 4.5 | 1.0 | 1.8 | G_1 |
| V_4 | -165.00 | 4.5 | 100 | 0.30 | 0.30 | 3.0 | 1.1 | 1.7 | G_2 |
| V_5 | -175.85 | 3.5 | 140 | 0.40 | 0.35 | 3.5 | 1.1 | 1.6 | G_2 |
| V_6 | -190.85 | 5.0 | 150 | 0.30 | 0.40 | 5.0 | 1.2 | 2.0 | G_2 |
| V_7 | -223.20 | 5.0 | 120 | 0.30 | 0.35 | 3.5 | 1.1 | 2.0 | G_3 |
| V_8 | -233.20 | 3.5 | 100 | 0.40 | 0.35 | 3.0 | 1.0 | 2.2 | G_3 |
| V_9 | -243.05 | 3.0 | 125 | 0.45 | 0.30 | 3.5 | 1.1 | 1.8 | G_3 |

Table 3.3: Vehicles' parameters setting.

is a common value of urban road. The control input u_i of each vehicle is limited as $u_i \in [-1.5, 1.5]$. To guarantee the passenger comfort, the changing rate of acceleration should be limited as $\Delta a_i \in [-0.5m/s^3, 0.5m/s^3]$. The traffic lights which indicate the entrance of intersection are installed at $p_T = 0m$, while the initial positions and features of the nine vehicles are shown in Table 3.3. This table shows that the vehicles are heterogeneous in terms of engine power, length, user preference and many other features, and this heterogeneity accords with the design objective of our CACC-VI algorithm. The threshold in the mixed control strategy is set as $E = 4m$. The total simulation time is $T = 40s$, and the sample time is set to $T_s = 0.02s$. As for the PSO based platoon control algorithm, the parameters are set as followings: $N_{swarm} = 10$, $Max_iteration = 30$, $\phi = 0.729$, $c_1 = 2.988$, $c_2 = 2.988$. The simulation is executed in *Matlab* 2010b on a computer with *IntelCore™i5 – 2410M 2.3GHz*.

Based on the signal timing of traffic lights and the current velocities of vehicles, it can be predicted that the first platoon formed by $V_1 \sim V_3$ can fully get through the intersection by maintaining the current velocity, so they are classified into the first category and given the label C_1 . Therefore the opportunity space, that we defined in (3.9) as the space left behind the last vehicle with label C_1 at moment T_r , can be decided as $76.7m$.

3.4.2 Simulation results

3.4.2.1 Platoon reorganization

The opportunity space is distributed, using the trajectory planning algorithm and the space arrangement approach, to as many vehicles as possible. According to the targeted velocity and the heterogeneous features, the demanding spaces can be calculated, therefore the opportunity space could be distributed to the rest six vehicles.

However, after launching the trajectory planning algorithm for the first time, none of the vehicles can find a feasible solution within constraints for the designed future positions. Thus vehicle V_9 is abandoned, then the future positions are recalculated for the second trajectory planning. This time, V_4 , V_5 and V_6 can find a solution while V_7 and V_8 cannot. So vehicle V_8 should abandon joining in the accelerating platoon. In the third trajectory planning with updated future positions, all the four vehicles V_4 , V_5 , V_6 and V_7 can find feasible solutions. Therefore, their labels should stay unchanged as C_2 , and the new platoon G'_2 is formed. This platoon is capable to get through the intersection during the current green phase by accelerating while respecting the constraints. As for the rest two vehicles V_8 and V_9 , they should change their labels to C_3 , and a new platoon G'_3 which will get through the intersection in the next green phase, is also formed. In other words, in the platoon reorganization procedure, the vehicle V_7 departs from the current platoon G_3 and joins in the preceding platoon G_2 to form a new platoon G'_2 ; at the same time, the rest of G_3 becomes a new platoon G'_3 .

This is the optimal result regarding to constraints such as space, velocity, engine power and time. The results of the platoon reorganization are summarized in Table 3.4.

3.4.2.2 Accelerating platoon

For the platoon G'_2 , which has the opportunity to get through the intersection by accelerating during the current green phase of traffic lights, the leader V_4 utilizes the preplanned trajectory while the previous leader V_7 of platoon G_3 utilizes the mixed control strategy, and the rest is controlled by the platoon control algorithm based on PSO. For illustration, the position, the velocity and the acceleration of the vehicles are shown in Figure 3.4, the tracking errors are shown in Figure

| Vehicle No. | $S_{D,i}$ (m) | 1^{st} Arrangement | | 2^{nd} Arrangement | | 3^{rd} Arrangement | | Labels | platoons |
|-------------|----------------------|----------------------|--------------|----------------------|--------------|----------------------|-----------|--------|----------|
| | | Positions | Solutions | Positions | Solutions | Positions | Solutions | | |
| V_4 | 10.80 | 61.05 | Not Exist | 51.20 | Exist | 41.20 | Exist | C_2 | G'_2 |
| V_5 | 10.85 | 50.20 | Not Exist | 40.35 | Exist | 30.35 | Exist | C_2 | G'_2 |
| V_6 | 15.00 | 35.20 | Not Exist | 25.35 | Exist | 15.35 | Exist | C_2 | G'_2 |
| V_7 | 12.35 | 22.85 | Not Exist | 13.00 | Not Exist | 3.00 | Exist | C_2 | G'_2 |
| V_8 | 10.00 | 12.85 | Not Exist | 3.00 | Not Exist | - | - | C_3 | G'_3 |
| V_9 | 9.85 | 3.00 | Not Exist | - | - | - | - | C_3 | G'_3 |

Table 3.4: Platoon reorganization of CACC-VI algorithm.

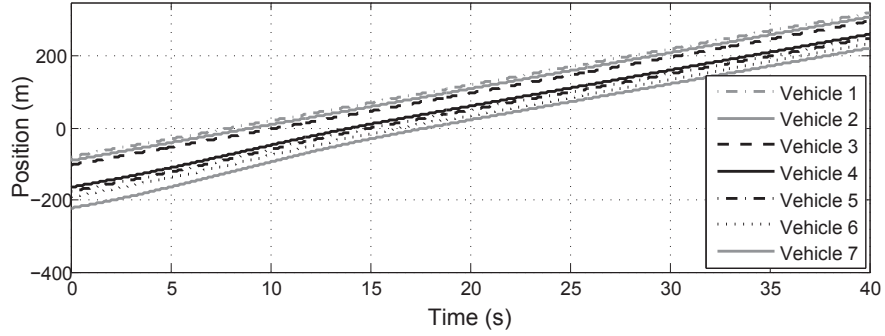
3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

3.5, and the control inputs are shown in Figure 3.6. By the result of trajectory planning, V_4 needs to accelerate for $t(V_4)_1 = 7s$ with control input $u(V_4)_1 = 0.4$; then cruising with $u(V_4)_2 = 0$ for $t(V_4)_2 = 3s$; finally, decelerate with negative input $u(V_4)_3 = -0.4$ in $t(V_4)_3 = 7s$. Due to the equality of the original velocity and the targeted one, the acceleration and the deceleration process have the same duration. The previous platoon leader V_7 tends to utilize the preplanned trajectory at first, because it is still far away from its nearest preceding vehicle V_6 . The preplanned trajectory for V_7 is to accelerate with the input $u(V_7)_1 = 0.8$ during $t(V_7)_1 = 5s$; then to cruise with $u(V_7)_2 = 0$ for $t(V_7)_2 = 7s$; at last to decelerate with $u(V_7)_3 = -0.8$ during $t_3 = 5s$.

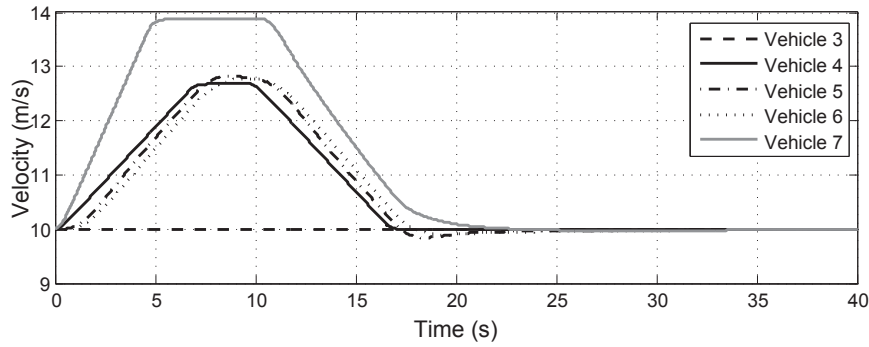
The members of G'_2 except V_4 need to find the optimal control input within the constraints based on its current state and the information coming from related vehicles: the platoon leader and the nearest preceding vehicle. In this scenario, the vehicle V_5 receives the information only from V_4 , because the platoon leader and the nearest preceding vehicle are the same one; while V_6 receives the information from V_4 and V_5 , and V_7 from V_5 and V_6 .

The positions of the vehicles in G'_2 at $T_r = 18s$ are separately $41.10m$, $30.61m$, 15.51 and $2.15m$, so the whole platoon passes the intersection. The intersection throughput during the current green phase is improved from three vehicles to seven vehicles. Although there are some errors to the future positions decided in the platoon reorganization process, the main objective of the algorithm is achieved. All the vehicles can arrive at the targeted velocity $10m/s$ after $t = 20s$, so the design objective of velocity is also achieved. The PSO based platoon control algorithm works well for V_5 and V_6 , because they are capable to track the trajectory of leader V_4 , and all the tracking errors are well limited and eliminated. From Figure 3.4(b), the three-section velocity trajectory of V_4 and V_7 can be clearly noticed, and their top velocities are well limited below $v_{limit} = 13.89m/s$. The initial spacing error of V_7 is $\delta_7(0) = 20m$, therefore it must catch up with the majority of G'_2 : at first, it follows the preplanned trajectory with helps it to arrive at the speed limit in five seconds, so that the spacing error can be quickly reduced, whereas the velocity error and acceleration error increase in this period; when V_7 is close enough to the preceding vehicle V_6 ($\delta_7 < 4m$) at around $t = 10s$, it switches to PSO based platoon control algorithm in order to eliminate all the tracking errors. Due to the limit of acceleration changing rate, the passenger will

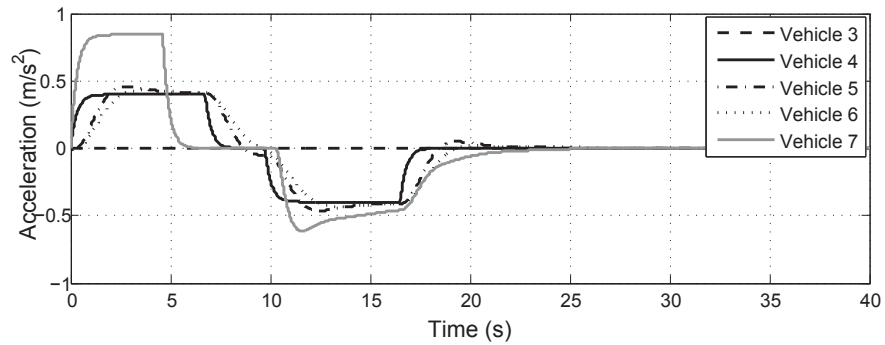
3.4 Simulation Experiments



(a) Positions: All the vehicles in the platoon get through the intersection at $T_r = 18s$ with positions $p_4 = 41.10m$, $p_5 = 30.61m$, $p_6 = 15.51m$ and $p_7 = 2.15m$; the intersection throughput is improved from 3 to 7.



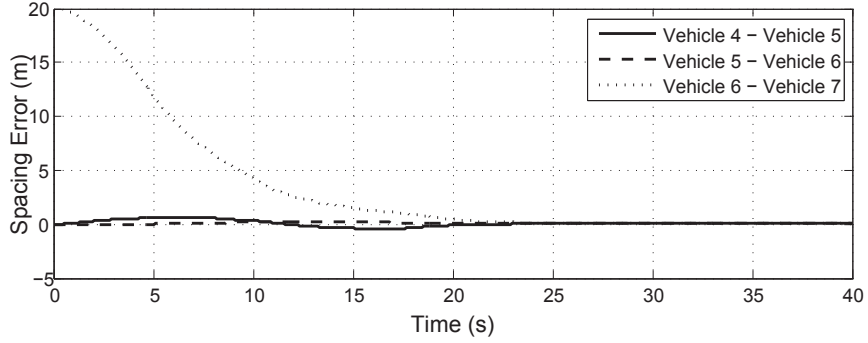
(b) Velocities: The velocities are well limited below the speed limit $v_{limit} = 13.89m/s$; the velocities tend to converge to the targeted velocity $v_{target} = 10m/s$.



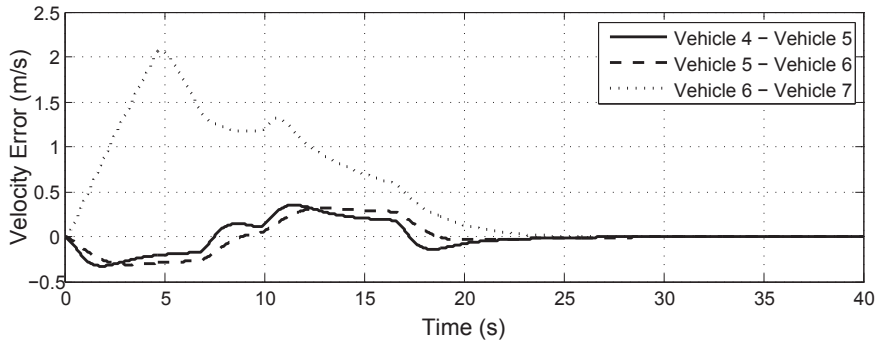
(c) Accelerations: V_4 utilizes the preplanned trajectory, V_5 and V_6 utilizes the PSO based control algorithm, and V_7 utilizes the mixed control strategy; the delay between V_6 and V_5 is smaller than that between V_5 and V_4 .

Figure 3.4: Motion of accelerating platoon G'_2

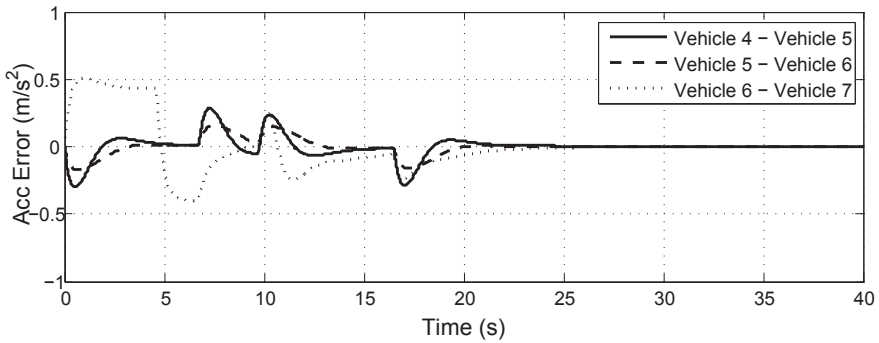
3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC



(a) Spacing errors: The spacing error δ_7 reduces with time; all the spacing errors become zero after $t = 25s$; the rear-end collision is absolutely avoided.



(b) Velocity errors: The velocity error $\Delta v_{e,7}$ increases during the first 5s; then all the velocity errors in the platoon G'_2 are limited and converge to 0 after the platoon passes the intersection.



(c) Acceleration errors: Acceleration errors in the platoon G'_2 are also limited and converge to 0 after the platoon passes the intersection.

Figure 3.5: Tracking errors in the platoon G'_2

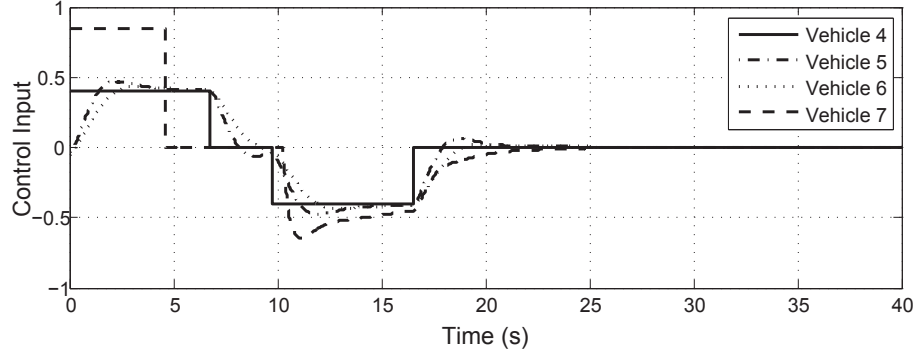


Figure 3.6: Control inputs of vehicles in the platoon G'_2

not feel the switch process. The relative information can be referred to Figure 3.4(b), 3.4(c), 3.5 and 3.6.

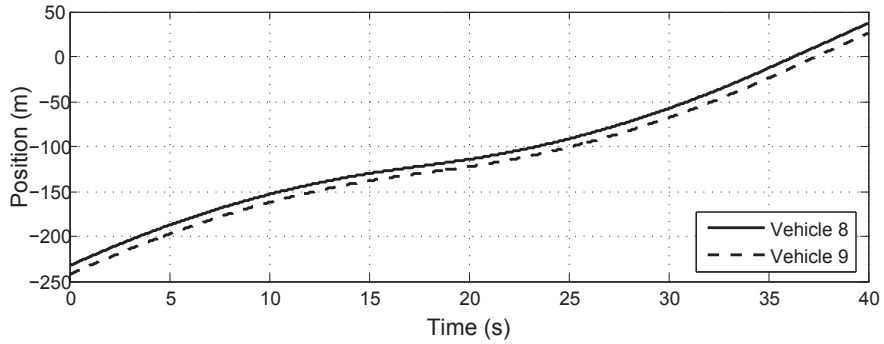
All the tracking errors of platoon G'_2 are eliminated after $t = 25s$; rear-end collision is absolutely avoided; the platoon is stabilized when facing the influence of the leader's changing behavior. Moreover, the difference between the two communication topologies can also be noticed: in Figure 3.4(b) and 3.4(c), we can find a smaller delay between V_5 and V_6 than that between V_4 and V_5 . Due to the communication with the platoon leader and the nearest preceding vehicle, V_6 can predict the behavior of V_5 , so that it can execute the maneuver in advance.

3.4.2.3 Decelerating platoon

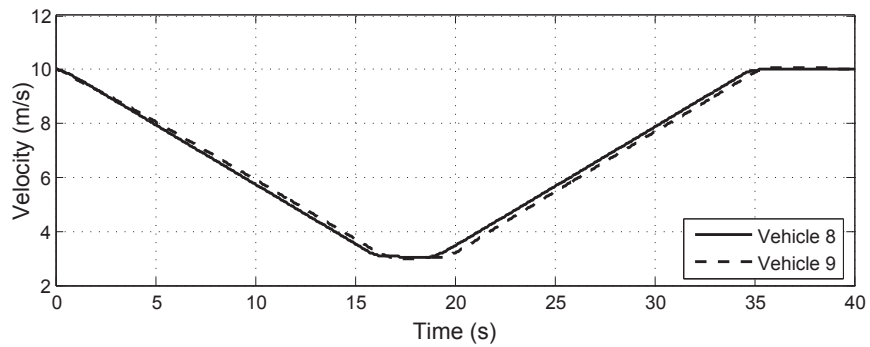
For the platoon G'_3 , which is formed by V_8 and V_9 , it cannot get through the intersection within the current green phase, therefore it chooses to decelerate to a lower speed without full stop and to pass the intersection during next green phase.

Similar to the platoon G'_2 , the leader V_8 follows the preplanned trajectory: V_8 should decelerate with $u(V_8)_1 = -0.44$ during $t(V_8)_1 = 16s$, then cruise with $u(V_8)_2 = 0$ in $t(V_8)_2 = 4s$, at last, accelerate with $u(V_8)_3 = 0.44$ within $t(V_8)_3 = 16s$. V_9 calculates the near optimal solution using the PSO based control algorithm according to the designed constraints and the information from V_8 . The results can be found in Fig. 3.7. At $T_g = 36s$ when the next green phase begins, the positions of V_8 and V_9 are separately $p_8 = -3.55m$ and $p_9 = -14.10m$ which are very close to their distributed future position $-3.50m$ and $-13.35m$.

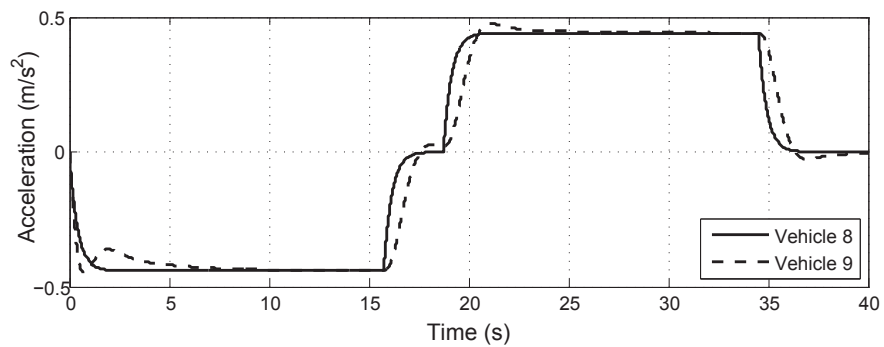
3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC



(a) Positions: At $T_g = 36s$, the positions of V_8 and V_9 are separately $p_8 = -3.55m$ and $p_9 = -14.10m$; the rear-end collision is absolutely avoided with limited spacing error.



(b) Velocities: The lowest speed is $3m/s$, the platoon G'_3 does not need to fully stop; the velocities at $T_g = 36s$ are all equal to $10m/s$.



(c) Accelerations: V_8 utilizes the preplanned trajectory while V_9 applies the PSO algorithm to track its behavior.

Figure 3.7: Motion of decelerating platoon G'_3

The lowest speed is $3m/s$, in other words, the platoon G'_3 does not need to fully stop to wait for the next green phase. The velocities at $T_g = 36s$ are all equal to $10m/s$, it means the platoon recovers to its original velocity. Thus the control objectives of G'_3 are all achieved.

The simulation results can prove that the CACC-VI algorithm that we designed achieves all the control objectives that we established in the beginning.

3.5 Conclusion

The most important contribution of this chapter is the presentation of a potential application of V2X communication to ameliorate the road transportation systems in terms of safety, time-saving, and environment friendly. In this chapter, we have proposed a decentralized cooperative adaptive cruise control algorithm for vehicles in the vicinity of intersections (CACC-VI), which aims at maximizing the throughput of intersection within a limited green phase of traffic lights, by making use of the remnant road capacity, while considering safety, fuel saving, velocity limitation, heterogeneous dynamics of vehicles and passenger comfort.

The CACC-VI algorithm composes of two main parts: platoon reorganization and platoon control. In the first part, a trajectory planning algorithm is introduced which is used to find a feasible input profile to accomplish designed motion within constraints; and a space arrangement approach is designed, in which the vehicles use different types of messages to share the information of the related spaces, the targeted velocity and the pass capability in order to take full use of the opportunity space while reduce the distances should be covered by the vehicles in the platoon; with these two methods, the platoons become flexible: vehicles can choose to accelerate to join in the preceding platoon or decelerate to depart from the current platoon. In the second part, different control strategy are used: the current platoon leaders follow the preplanned trajectories designed in the first part; the previous leaders follow a mixed control strategy; and the rest vehicle use a constrained PSO algorithm to find the optimal control input to minimize the value of the cost function, which is defined considering the tracking errors, the control input and the different types of constraints.

Simulation experiments show that our algorithm is capable to control vehicles of different situations; the number of vehicles which can get through the inter-

3. IMPROVING TRAFFIC EFFICIENCY VIA V2X AND CACC

section during the green phase can be greatly improved; and the optimal control input can be found within the defined constraints so that the disturbances like accelerating and decelerating from the platoon leader to the rest of the platoon can be attenuated.

However, there are still some aspects that we have not taken into account. For example, in reality, the vehicles enter the effect area of the algorithm progressively, it is necessary to decide the activation moment of the algorithm; a larger scale simulation need to be conducted for fully functional intersections to further test the effectiveness of this algorithm. These aspects will be studied in our future research.

Chapter 4

Decentralized Autonomous Intersection Control for Light Traffic

Contents

| | | |
|------------|---|-----------|
| 4.1 | Introduction | 78 |
| 4.2 | Intersection model and Design objectives | 80 |
| 4.2.1 | Intersection | 80 |
| 4.2.2 | Assumption and Objectives | 81 |
| 4.3 | Algorithm Design | 82 |
| 4.3.1 | Collision Definition | 82 |
| 4.3.2 | Set Partition | 84 |
| 4.3.3 | Priority Setting | 85 |
| 4.3.4 | Cost Function | 86 |
| 4.4 | UML-based Testbed Design | 88 |
| 4.4.1 | Static Design | 88 |
| 4.4.2 | Dynamic Design | 91 |
| 4.5 | Simulation | 93 |
| 4.5.1 | Parameter Settings | 93 |
| 4.5.2 | Simulation Results | 95 |
| 4.6 | Conclusion | 97 |

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

4.1 Introduction

In urban traffic scenarios, intersections are considered to be one of the most high-risk places where accidents may occur. Besides, congestions caused by improper traffic signal setting make people spend more time on road and more fuel is consumed, which causes more pollution. Therefore, improving safety and efficiency at intersections needs to be researched.

Actuated traffic signal control systems are the one of the earliest attempts to improve the intersection's efficiency: inductive loops are used to detect vehicles in order to ask for additional green time (Papageorgiou *et al.* (2003)). Predictive models are added into adaptive signal control systems (Hunt *et al.* (1982); Mirchandani & Wang (2005)) to forecast the vehicle arrivals and estimate queue lengths at intersections.

More efficient, better space utilization and elimination of human distraction and misjudgment, autonomous vehicles are considered to be highly potential to improve safety. Integration of wireless communication technology in autonomous vehicles can bring many possibilities to design intelligent road transportation systems in the future (Hartenstein & Laberteaux (2010)). In this case, communications between vehicle and vehicle (V2V) and between vehicle and infrastructure (V2I) are enabled. Thus vehicle information like velocity, acceleration and vehicle type can be shared, and traffic information like vehicle density, accidents and light signal timing can be acquired and disseminated. A vehicle does not have to detect the environment all by itself, and it can be aware of the situation even beyond the line of sight. Therefore, the individual intelligence of a single autonomous vehicle can be extended into collective intelligence of vehicle groups and intersection agents.

By using V2X communication, revolutionary improvements can be achieved in traffic control systems for intersections. A survey of different design philosophy for traffic control using V2X communication is given in (Li *et al.* (2014)). Generally, there are two ways to improve the current systems. One way is platoon-based control strategy (He *et al.* (2012); Jiang *et al.* (2006); Pandit *et al.* (2013); Priemer & Friedrich (2009); Xie *et al.* (2011, 2012)), in which time plan of traffic lights is set referring to the information of platoons coming from different directions so that they can get through the intersection without being interrupted. If V2X device is integrated in the platoon, the information can be sent directly to the

intersection manager instead of identifying the platoon by inductive loops or other sensors (Priemer & Friedrich (2009)). In (He *et al.* (2012)), an arterial traffic control algorithm for intersections is proposed, which is capable to recognize the vehicle platoons and to determine future optimal signal plans based on the current traffic controller status, online platoon data and priority requests from special vehicles, such as transit buses.

Another way is cooperative driving, in which the intersection controller receives the vehicles' requests to get through the intersection and gives each vehicle the advice of an optimal course in consideration of their dynamic capacity of action ensuring no crash occurs, at the same time minimizing the intersection delay and fuel consumption (Dresner & Stone (2008); Li & Wang (2006); Liu & El Kamel (2016e); Lu & Kim (2016); Wu *et al.* (2012); Wuthishuwong *et al.* (2015); Zohdy & Rakha (2014)). In cooperative driving, a platoon becomes flexible, a vehicle can join in or depart from an already formed platoon; and several vehicles can also choose to form a new one (Li & Wang (2006); Liu & El Kamel (2016e)). There are two ideas to decide a feasible schedule: negotiation-based control (Dresner & Stone (2008); Lu & Kim (2016); Wu *et al.* (2012)) and planning-based control (Li & Wang (2006); Liu & El Kamel (2016e); Wuthishuwong *et al.* (2015); Zohdy & Rakha (2014)). The former consumes less computation costs because only the vehicles nearby the intersection are taken into account, whereas the latter sometimes can achieve higher performance. In (Dresner & Stone (2008)), a first-come-first-serve policy is used, and every vehicle must send a reservation message to the intersection manager, then the latter checks the availability of the space. If the reservation is not in conflict with the reservations of others, a car is allowed to pass through the intersection. Otherwise, the car has to generate and send new request messages while decelerating until it gets the permission from the manager. At the worst, the vehicle must stop before the intersection and start again. In (Wuthishuwong *et al.* (2015)), an incoming vehicle requests the state of intersection from the manager, then discrete mathematics is used to plan a safe trajectory for vehicle to pass the intersection; the related message protocol is also designed. A CACC-VI algorithm is designed in (Liu & El Kamel (2016e)) in order to improve the throughput of the intersection within the current green time by taking advantage of the redundant road capacity and reorganizing the platoons.

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

In this chapter, a decentralized autonomous intersection control (DAIC) algorithm is designed to improve the efficiency while guarantee the safety of intersection. Vehicles around the intersection are divided into three sets: cruising set, tuning set and free set. A vehicle in the cruising set must get through the intersection by maintaining the current speed, and a vehicle in the tuning set should change its velocity until find a collision-free value. Priorities are given to the vehicles in tuning set according to their types, routes and distances to the intersection, so that they can make decision one by one. Therefore computational cost at the intersection manager can be greatly reduced. The intersection is granulated into conflict zones and the related occupancy time window of vehicles is calculated and stored at the manager. An objective function for each vehicle is defined referring to the conflict time to the cruising vehicles coming from crossing lanes, the difference to safe velocity and the resulting delay. At each time step, the vehicle should utilize the control input with the lowest cost in its action set.

The rest of the paper is organized as follows. In section 4.2, the scenario of our research is described, and the control objectives are established. In section 4.3, our algorithm, including the set partition, the priority setting and the objective function, is introduced in details. In section 4.4, a simulation testbed to evaluate the algorithm is designed using UML. In section 4.5, simulation results are presented to prove the validation of our algorithm. Finally, some concluding remarks and perspectives are given in Section 4.6.

4.2 Intersection model and Design objectives

4.2.1 Intersection

In this chapter, a symmetrical four-leg intersection with three lanes for each incoming and outgoing streets is studied. As shown in Figure 4.1, in the incoming direction, lane on the left is for left turning, the middle one is for straight direction and lane on the right is for both straight direction and right turning. The area inside the intersection where collision of vehicles coming from different lanes may happen is defined as the "conflict zones". There are 36 conflict zones in total for this intersection model. The width of a conflict zone equals the lane width, which is denoted as w . The vehicles approaching the intersection are different not only in routes, but also in types, lengths, acceleration and deceleration abilities.

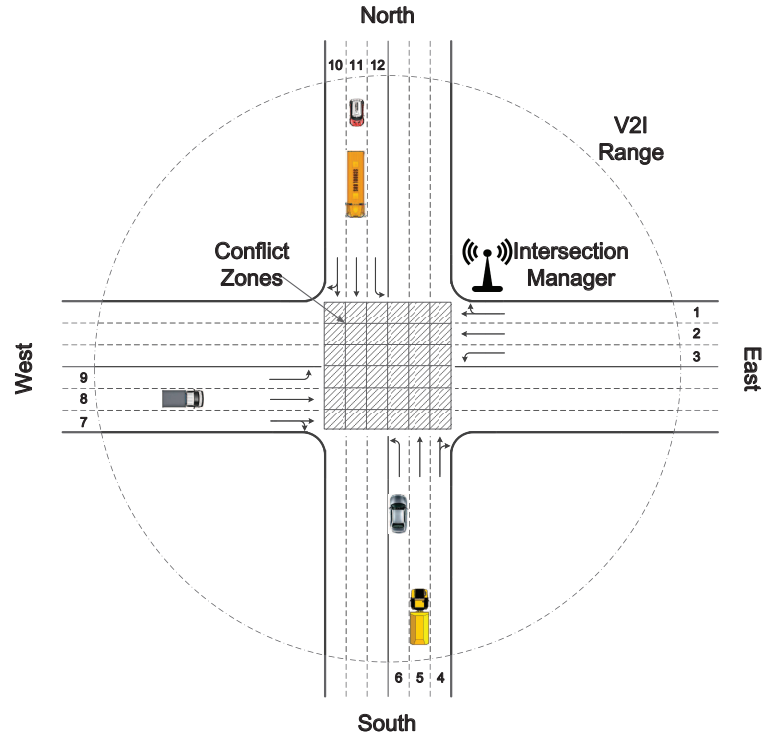


Figure 4.1: 6-lane 4-leg intersection

4.2.2 Assumption and Objectives

For traditional traffic light, only lanes without conflict can be given green lights simultaneously: the vehicles on a lane A must totally stop before the red light to give way to another lane B to avoid collisions. This is an effective protocol to solve the time and space conflict at the intersection. However, it is not efficient and not fair enough: a vehicle on lane A may have to wait even if it arrives earlier than some vehicles on lane B. Besides, compared to cruising, more fuel is consumed during this stop-idling-start process.

In connected vehicles and autonomous intersection management context, it is assumed that typical vehicles are all replaced by autonomous vehicles and traffic light is replaced by intersection manager. The manager is capable to communicate with the vehicles within its range R . And the vehicles are all equipped with V2X devices so that they can exchange necessary information with each other and with the manager. The network connection is activated all the time and with negligible transmission delay, packet loss, interference, etc. It is assumed that there is no lane shifting within the V2I range.

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

In this chapter, we try to develop a new protocol without traffic light in connected autonomous vehicle environment with the following design objectives: rear-end collision in the same lane and cross collision from two different lanes must be avoided; the average delay of vehicles should be reduced; vehicle should not stop except necessary; the decision making process should be local at each vehicle in order to reduce the computational cost at the intersection manager; the heterogeneity of vehicles must be considered.

4.3 Algorithm Design

Essentially, collision at intersection is due to the conflict in space and time. By using V2X technology, vehicle can get the occupation information of the intersection from larger distance, so that it can tune its velocity to obtain a collision-free trajectory at the intersection.

4.3.1 Collision Definition

4.3.1.1 Cross Collision

The cross collision is defined as collision inside the intersection coming from vehicles on different lanes. For vehicle V_i , the set of all the conflict zones which it will pass through is defined as SCZ_i ; its distance to the conflict zone CZ_k in SCZ_i on its lane is denoted as d_{V_i, CZ_k} ; its velocity is denoted as v_i and its length is denoted as l_i . If we note the current moment as t_0 , then the moment when V_i enters and leaves CZ_k , which are separately denoted as t_{1, V_i} and t_{2, V_i} , can be defined as follows

$$\begin{aligned} t_{1, V_i} &= t_0 + \frac{d_{V_i, CZ_k}}{v_i} \\ t_{2, V_i} &= t_0 + \frac{d_{V_i, CZ_k} + l_i + w}{v_i}. \end{aligned} \tag{4.1}$$

In reality, noise of sensor and error of actuators cause bias in position and velocity measurement, and the road traffic does not demand extremely accuracy; while in simulation, the discretization of time causes errors in position and velocity as well. These inaccuracies cannot be ignored for vehicles move in high speed.

Therefore a time buffer t_b is added on the original entering and leaving moment:

$$\begin{aligned} t'_{1,V_i} &= t_{1,V_i} - t_b \\ t'_{2,V_i} &= t_{2,V_i} + t_b. \end{aligned} \quad (4.2)$$

Therefore the conflict zone occupation time for V_i at CZ_k can be defined as

$$OT_{i,CZ_k} = [t'_{1,V_i}, t'_{2,V_i}]. \quad (4.3)$$

In the same way, the conflict zone occupation time for another vehicle V_j at CZ_k can be defined. The overlapped occupation time for V_i and V_j at CZ_k can be calculated as following:

$$OOT_{CZ_k,V_i,V_j} = \max(0, \min(t'_{2,V_i}, t'_{2,V_j}) - \max(t'_{1,V_i}, t'_{1,V_j})). \quad (4.4)$$

If OOT is bigger than 0, V_i and V_j will collide at CZ_k , which is noted as $V_i \overset{CZ_k}{\nabla} V_j$. The set of vehicles which have conflict with V_i at CZ_k in cross collision type is denoted as S_{V_i,CZ_k} :

$$S_{V_i,CZ_k} = \{V_j | V_j \overset{CZ_k}{\nabla} V_i\}. \quad (4.5)$$

4.3.1.2 Rear-end Collision

The rear-end collision is defined as the collision between two vehicles on the same lane. For vehicle V_i , its safe velocity to maintain a safe inter-distance and avoid creating an accident with the preceding vehicle V_{i-1} is calculated by using the Krauss model:

$$v_i^{safe}(t_0 + T_s) = v_{i-1}(t_0) + \frac{d_{i,i-1}(t_0) - v_{i-1}(t_0) \cdot T}{(v_i(t_0) + v_{i-1}(t_0))/2b_{max} + T}, \quad (4.6)$$

where T_s is the sampling time of the intersection control system; $d_{i,i-1}$ is the distance between the two vehicles; b_{max} is the maximum deceleration of V_i ; and T is the reaction time of human driver, while in case of autonomous system it should be equal to T_s .

The predicted velocity of V_i at $t_0 + T_s$ is defined as

$$v_i(t_0 + T_s) = \min(v_{limit}, v_i(t_0) + a_i(t_0) \cdot T_s) \quad (4.7)$$

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

If $v_i(t_0 + T_s) > v_i^{safe}(t_0 + T_s)$ and if the two vehicles maintain their velocities from $t_0 + T_s$, then the rear-end collision will occur.

4.3.2 Set Partition

The vehicles on the incoming street within V2I range can be divided into three different sets:

- Cruising Set S_C : The vehicles which don't have any conflict with other vehicles. They should maintain their current velocities and the conflict zone occupation time (CZOT) information is calculated and stored at the manager.
- Tuning Set S_T : The vehicles which currently have conflict with some other vehicles. Thus a vehicle V_i in S_T must require the occupation time of all the conflict zones in SCZ_i from the intersection manager via V2I, and require the position and velocity information from its nearest preceding vehicle via V2V. Then V_i have to tune its own velocity according to the acquired information to avoid both rear-end collision and cross collisions.
- Free Set S_F : All the other vehicles within V2I range. These vehicles are not controlled by our autonomous intersection control algorithm but by its local autonomous driving algorithms.

If we sort the vehicles on an incoming street by the distance to the intersection, S_C has the smallest distance, then it comes to S_T and finally S_F . The number of vehicles in S_C and S_F is not limited. Meanwhile, in order to reduce the computational cost, there is at most one vehicle for each lane in S_T . Therefore there are up to 12 vehicles in S_T for our intersection model.

A vehicle V_i in S_T can become a member of S_C if and only if with its tuned velocity, it will not have any rear-end collision and cross collision. Then V_i must maintain its tuned velocity until it fully get through the intersection. Then its nearest following vehicle in S_F , if exist, will become the tuning vehicle of this lane.

4.3.3 Priority Setting

The maximum acceleration and deceleration of vehicle V_i are separately denoted as a_{max} and b_{max} . It is assumed that the V_i can choose an acceleration to execute from several values between the two limits. The action set A_i of V_i is defined as:

$$A_i = \{a_{i,1}, a_{i,2}, \dots, a_{i,n}\}, \text{ where } a_{i,j} \in [-b_{max}, a_{max}]. \quad (4.8)$$

For different vehicles, the dimension $\|A_i\|$ of action set can also be different.

There are two different ways to find the optimal acceleration for each vehicle. The first one is centralized: all the vehicles in S_T are taken into account simultaneously; thus all the information like vehicles' positions and velocities should be sent to the manager, the latter is responsible to find the optimal acceleration for each vehicle; the searching space's dimension is in order of $\|A_i\|^{\|S_T\|}$, so the computational cost at the manager could be enormous. Another way is decentralized: the computation is local at each vehicle, and it demands the manager and its preceding vehicle for necessary information to find its own optimal acceleration; the dimension of searching space of each vehicle is only $\|A_i\|$.

In fact, the role of intersection manager is different in these two ways. For centralized intersection control, the manager is both the information hub and the computation center: it requires all the necessary information for optimization and gives orders to all the vehicles in S_T . For decentralized intersection control, the manager is principally information hub: it requires the information, then process it with light computation and transmits it to target vehicles. Clearly, the decentralized intersection control is less time consuming. Besides, it can take advantage of the local computational ability of autonomous vehicles and relief the manager from heavy duty.

When a vehicle computes its own acceleration, it needs the information of vehicles in S_C and S_T . The velocities of vehicles in S_C are constants and their CZOT information is stored at the manager. While for vehicles in S_T , they are still tuning their velocities and the occupation time is variable. However, the decision-making process can not rely on variable information. In order to solve this problem, a priority policy is designed: the vehicles in S_T make their decisions in turn by following sorted priorities; once a vehicle has arrived at a collision-free velocity, it sends its own information immediately to the manager, so that the

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

manager can calculate its CZOT and then store it in the memory; then it comes to a vehicle with lower priority, and the vehicle should require the manager for the CZOT of all the vehicles in S_C and the vehicles in S_T which has higher priorities.

According to their distances to the intersection, headings, and types, the vehicles in S_T are given scores in three parts n_d , n_h and n_t by following the principles as below:

1. Vehicles which have smaller distances to the intersection get a higher score in n_d ;
2. Vehicles turning left have higher score in n_h than vehicles going straight and turning right, because turning left needs to occupy more conflict zones;
3. Emergency vehicles get the highest score in n_t , then heavy duty vehicles like trucks and bus get higher score than others like sedans and hatchbacks.

The vehicle V_i 's total score is noted as n_{total,V_i} , which is the sum of n_d , n_h and n_t .

Thus the vehicles in S_T with higher priorities than V_i can be defined as S'_{T,V_i} :

$$S'_{T,V_i} = \{V_j | V_j \in S_T, n_{total,V_j} > n_{total,V_i}\}. \quad (4.9)$$

4.3.4 Cost Function

In order to find the best action at each time step in A_i , an objective function is needed. As presented in design objective, there are three parts to be considered in this function: cross collision prevention, rear-end collision prevention and minimized delay. Therefore, the objective function is composed of three parts.

Firstly, the cost of cross collision for V_i is defined as the sum of the overlapped occupation time defined in (4.4) coming from the vehicles which have conflict with V_i at all the conflict zones in SCZ_i :

$$J_{cross,i} = \sum_{CZ_k \in SCZ_i} \sum_{V_j \in S_{V_i,cf}} OOT_{CZ_k,V_i,V_j} \quad (4.10)$$

where $S_{V_i,cf}$ is the set of vehicles to be counted by V_i in its cost of cross collision, which is defined as

$$S_{V_i,cf} = S_{V_i,CZ_i} \cap (S_C \cup S'_{T,V_i}). \quad (4.11)$$

If $J_{cross,i} = 0$, it means that there will be no cross collision when V_i gets through the intersection.

Secondly, the cost of rear-end collision is measured by the difference between the predicted velocity and the safe velocity of V_i :

$$J_{rear-end,i} = \max(0, v_i(t_0 + T_s) - v_i^{safe}(t_0 + T_s)). \quad (4.12)$$

No rear-end collision with the preceding vehicle will happen if $J_{rear-end,i} = 0$.

Thirdly, the cost of delay is defined as the difference between the travel time if V_i utilizes its current velocity v_i and the travel time if it utilizes the tuned velocity:

$$J_{delay,i} = \max(0, t(v_{tuned}) - t(v_i)). \quad (4.13)$$

The total cost is the weighted sum of these three parts:

$$\begin{aligned} J_{total,i}(t_0 + T_s) = & k_1 \cdot J_{cross,i} \\ & + k_2 \cdot J_{rear-end,i} + k_3 \cdot J_{delay,i}. \end{aligned} \quad (4.14)$$

Intuitively, the first two weights should be much bigger than the third one, because collision avoidance is the first priority.

At each time step, a vehicle V_i in S_T must traverse its action set A_i to find all the acceleration values with which it can have a collision-free trajectory. In other words, $J_{rear-end,i}$ and $J_{cross,i}$ are all zeros. If this subset A'_i is not empty, then V_i should utilize the acceleration value which can achieve the minimal delay. Then V_i becomes a member of S_C and begins to maintain its velocity from $t_0 + T_s$ till it gets through the intersection. Otherwise, V_i should apply the acceleration value leading to the minimal total cost and keeps itself in S_T .

$$A'_i = \{a_{i,j} | J_{rear-end,i} = 0 \ \& \ J_{cross,i} = 0\} \quad (4.15)$$

$$a_i^*(t_0) = \begin{cases} \arg \min_{a_{i,j} \in A'_i} J_{delay,i}, & \text{if } A'_i \neq \emptyset; \\ \arg \min_{a_{i,j} \in A_i} J_{total,i}, & \text{otherwise.} \end{cases} \quad (4.16)$$

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

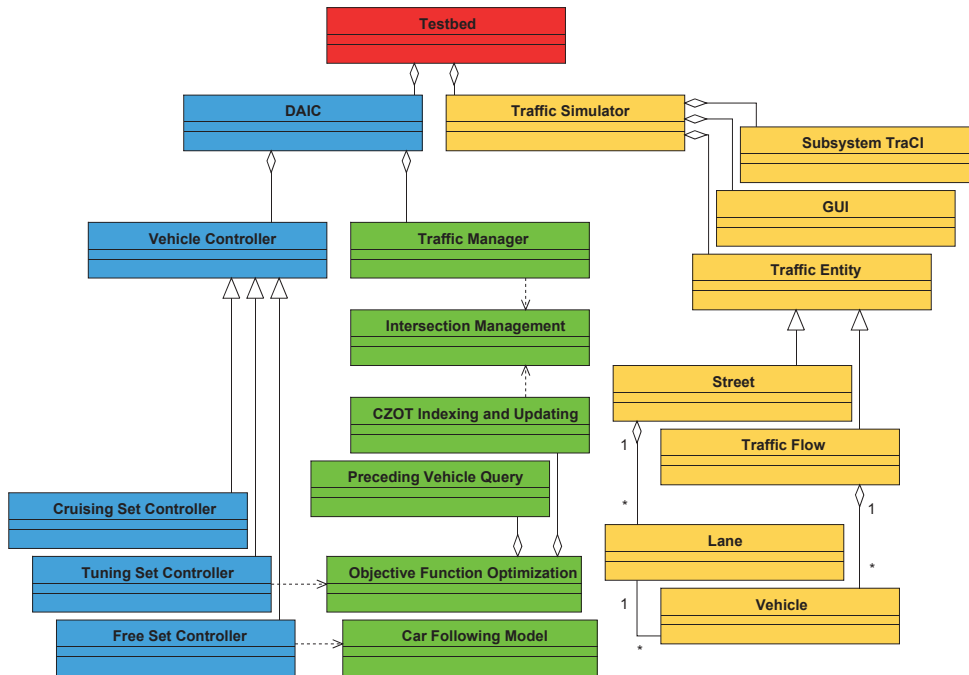


Figure 4.2: Class diagram of the testbed

4.4 UML-based Testbed Design

In order to evaluate the effectiveness of our decentralized autonomous intersection control (DAIC) algorithm, a simulation testbed is established. Unified Modeling Language (UML) is used at the conceptual design stage (Luo *et al.* (2011)). UML is a specification defining a graphical language for visualizing, specifying, constructing and artifacts of distributed object systems. It has become the standard for object-oriented modeling and design.

4.4.1 Static Design

Class diagram is a capable tool to show the static structure of the design and the various relation between different modules and components. The structure of the testbed is illustrated in the class diagram shown Figure 4.2. Our testbed mainly contains two parts: the traffic simulator (marked in yellow) and the core algorithm DAIC (marked in blue and green).

In this design, the algorithm part is the aggregation of the vehicle controller and the traffic manager. The vehicle controller can be generalized into several

controller subclasses: according to the set to which the vehicle belongs, different controllers are applied. The cruising set controller is used to make vehicles in this set travel at constant collision-free velocities; the free set controller is applied to calculate the control inputs for vehicles according to the car-following model; the tuning set controller is conducted by the vehicles which need to further tune their velocities to avoid any collisions, and the objective function is implemented in this controller to find the optimal action for each vehicle. Since the objective function is defined based on information from the preceding vehicle and the intersection manager, two modules separately represent the V2V communication and the V2I communication should be integrated in the construction of testbed.

Another part, the traffic simulator can be seen as the backbone of the testbed. Firstly, the geometry of intersection and streets, and the links between two lanes are described; secondly, the traffic flows are generated according to the definition including the arriving rate of vehicles, different vehicle types and their appearance probabilities; thirdly, an interface subsystem must be established to build the communication link between traffic simulator and the algorithm part built outside; a GUI illustrating the vehicles' movement and states update during the simulation is necessary to give an intuitive view of the algorithm's outcome.

In this paper, the DAIC algorithm is implemented in Matlab, and SUMO (Krajzewicz *et al.* (2012)) is applied as the traffic simulator. "Simulation of Urban MObility" (SUMO) (Krajzewicz *et al.* (2012)) is an open-source, microscopic, inter- and multi-modal, space-continuous and time-discrete traffic flow simulation platform. Its development is started by German Aerospace Center (DLR) in 2001. SUMO is not only a traffic simulator after development of more than a decade. It has evolved into a full featured suite of traffic modeling applications including a road network importer capable of reading different source formats, demand generation and routing utilities, which use a high variety of input source, a high performance simulation usable for single junctions as well as whole cities including a "remote control" interface to adapt the simulation on-line and a large number of additional tools and scripts.

Since 2006, the simulation in SUMO is extended by the possibility to interact with an external application via a socket connection. This API is called "Traffic Control Interface (TraCI)", which has currently become a part of SUMO's official

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

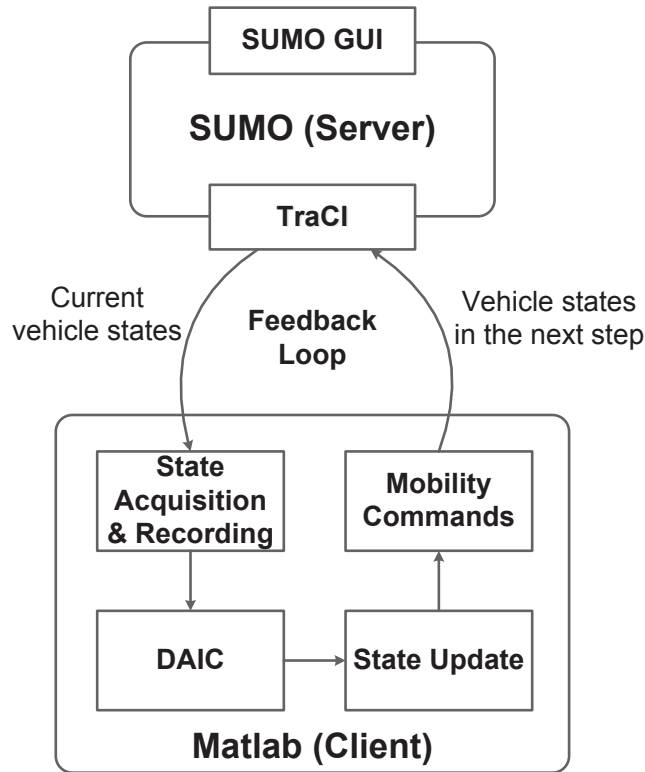


Figure 4.3: Simulation using Matlab and SUMO

release. To enable on-line interaction, SUMO has to be started with an additional option, which obtains the port number to listen to. After the simulation has been loaded, SUMO acts as a server to listen on this port for an incoming connection. After being connected, the client (external application) is responsible for triggering simulation steps in SUMO and for closing down the connection which also forces the simulation to quit. The client has access to values of almost all simulation artifacts, such as intersections, lanes, traffic lights, and individual vehicles. The client can also change values, for example, set the timing plan of traffic lights, change a vehicle's velocity or force it to change a lane. TraCI allows complex interaction such as adaptive transition of traffic lights or modeling special behavior of individual vehicles.

After the connection between Matlab and SUMO is established, SUMO acts as a server and Matlab acts as a client. The DAIC algorithm running in Matlab is regarded as external application. As shown in Figure 4.3, at each time step, Matlab requires vehicles' data, including position and velocity, from SUMO via

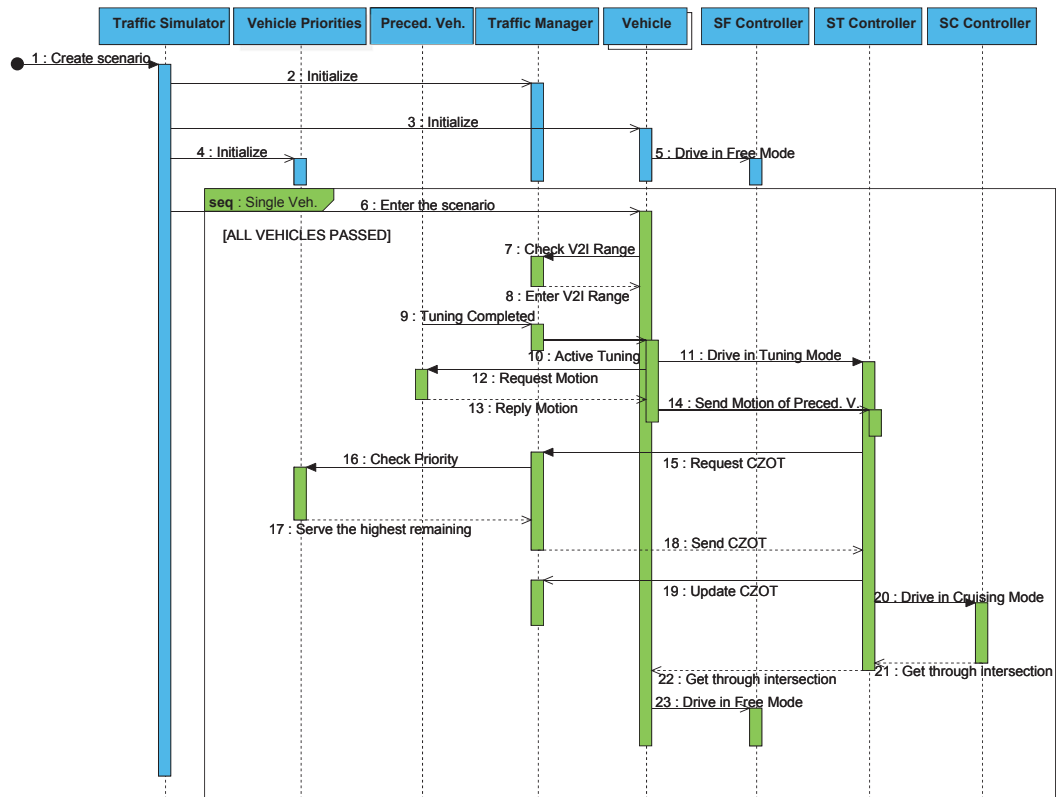


Figure 4.4: Sequence diagram of the testbed

TraCI, then the data is stored for later usage; after that, related vehicles need to find the control input for the current step using DAIC; based on the control input and their current states, the vehicles' velocities in the next step can be decided and set; finally, Matlab triggers the simulation step in SUMO to complete a feedback loop.

4.4.2 Dynamic Design

The interaction between the objects we defined previously can be summarized in the sequence diagram shown in Figure 4.4. The state transition of a vehicle object is shown in the statechart diagrams in Figure 4.5 and 4.6.

After launching the simulation, the scenario including the geometry of the intersection and streets is created according to the configuration file, and a route file indicating the appearance time in the scenario and the characteristics of each vehicle is generated. The vehicles are loaded in the scenario by following the route

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

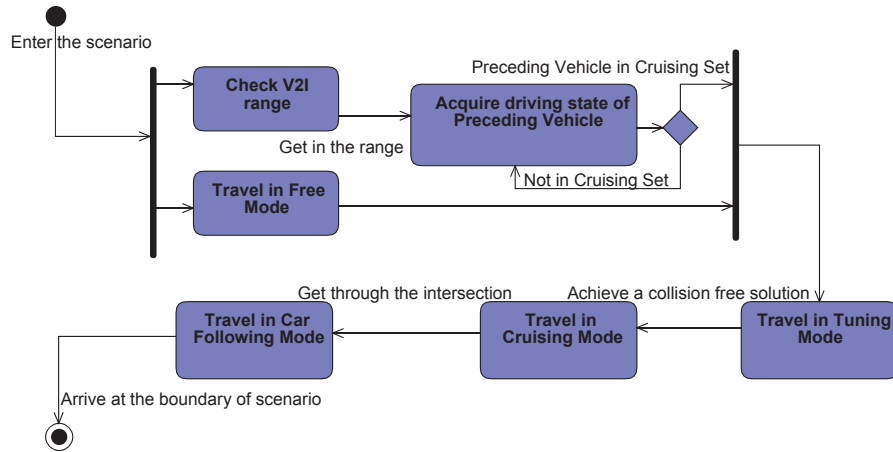


Figure 4.5: Statechart diagram of a vehicle object

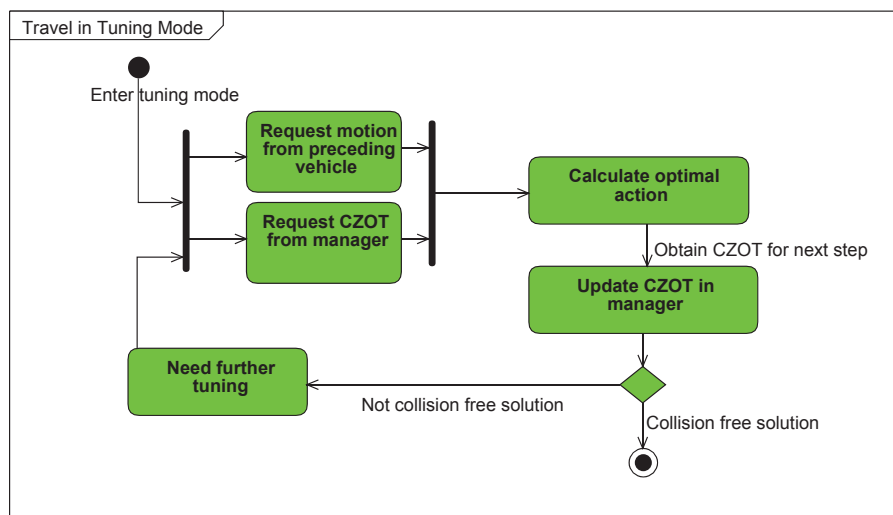


Figure 4.6: Statechart diagram of a vehicle in tuning mode

file and the sequence loop of individual vehicle is repeated until all the vehicles get through the intersection.

In term of a vehicle, it travels by following its local autonomous driving algorithms in the beginning. It becomes a member of the free set S_F immediately when entering the V2I range. When its preceding vehicle has found a collision-free velocity thus becomes a member of cruising set S_C , then it will receive an active message from the manager, so that the vehicle must join in the tuning set S_T . After that it should send the information like position, route and vehicle type to the manager, at the same time, it demands the related CZOT information. Furthermore, it should require the future motion of the preceding vehicle. After getting the required information, the vehicle can find an optimal control input for the next step. The new CZOT is calculated and given back to the manager. If the velocity in the next step will not lead to any collision, the vehicle should join in S_C and maintain that velocity until it gets through the intersection. After that, the vehicle should switch to car-following mode again. The strategy of a vehicle within V2I range is summarized in Algorithm 5.

In term of the manager, when receiving the requests from vehicles, it divides the set S_T into service groups; then it gives vehicles scores according to the principles established previously, thus the service priorities and information dependence between vehicles can be decided. Then the manager begins to response the vehicles in the decided order. The manager should wait for the reply of the served vehicles, then it can reply the vehicles in lower priorities. After all the vehicles in S_T get replied in the current step, the manager can start the next service cycle.

4.5 Simulation

4.5.1 Parameter Settings

The intersection in simulation is same as the one shown in Figure 5.3. The length of each incoming street is $500m$, and the V2I communication range R of the manager is set to $200m$. $V_{limit} = 70km/h(19.44m/s)$ is set to be the speed limit of all the incoming streets. The appearance of vehicles on each street is assumed to be a Poisson process with arriving rate λ . Three different scenarios is defined to verify our algorithm whose arriving rates on different lanes are shown in Table 4.1. The traffic flows on each lane is randomly generated by following the

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

Algorithm 5 Vehicle V_i 's strategy in DAIC at each time step

Input:

V2V message: information message M_{Info} , V2I messages: activation message M_{Active} , CZOT message M_{CZOT} , update message M_{Update} ;

Output:

V_i 's motion in the next step;

```
1: repeat
2:   if  $V_i \in S_C$  then
3:     Maintain the current velocity in the next step;
4:   end if
5:   if  $V_i \in S_F$  then
6:     if Activation message  $M_{Active}$  from the preceding vehicle received then
7:        $V_i$  joins in  $S_T$ ;
8:     else
9:       Find control input  $a_i$  for the next step using Krauss car-following
        model;
10:    end if
11:  end if
12:  if  $V_i \in S_T$  then
13:    Demand related CZOT from the manager;
14:    Demand the preceding vehicle's information;
15:    if  $M_{info}$  and  $M_{CZOT}$  are received then
16:      Calculate the optimal input  $a_i^*$  for the next step using 4.16;
17:      Predict its own CZOT and send it to the manager;
18:    end if
19:    if A collision-free velocity will be achieved in the next step then
20:       $V_i$  joins in  $S_C$ ;
21:      Send an update message  $M_{Update}$  to the manager indicating an update
        demand of  $S_T$ ;
22:    end if
23:  end if
24: until  $V_i$  gets through the intersection.
```

| Scenarios | Arriving Rate λ | | |
|------------|-------------------------|-------------|-----------|
| | Right Lane | Middle Lane | Left Lane |
| Scenario 1 | 0.10 | 0.10 | 0.10 |
| Scenario 2 | 0.15 | 0.15 | 0.13 |
| Scenario 3 | 0.20 | 0.20 | 0.15 |

Table 4.1: Different scenarios in the simulation.

| Types | Lengths | a_{max} | b_{max} | Appearance Probabilities | Action Set Dimension |
|-----------|---------|-----------|-----------|--------------------------|----------------------|
| Sedan | 3.5 | 2.5 | 5 | 0.35 | 9 |
| Hatchback | 3 | 3 | 4.5 | 0.35 | 9 |
| Wagon | 4 | 2.7 | 4.5 | 0.1 | 7 |
| Truck | 8 | 1.5 | 3.5 | 0.1 | 5 |
| Bus | 10 | 1.5 | 3.5 | 0.1 | 5 |

Table 4.2: Traffic flow composition.

arriving rate λ and the appearance probabilities of five vehicle types which are different in lengths and acceleration/deceleration abilities. The related parameter settings can be referred to Table 4.2. The flows begins at $t = 0s$ and ends at $t = 100s$. And the sample time of simulation is set as $Ts = 0.1s$.

4.5.2 Simulation Results

Comparisons are made between the same intersection controlled separately by fixed-period traffic light and our DAIC algorithm in different traffic scenarios. The length of green light for straight direction and right turning is $25s$, for left turning is $20s$, and the yellow light between two phases is $3s$. Therefore, the phase period of the traffic lights is $102s$. The generated traffic flows for these two methods are exactly the same. The performance of different methods are evaluated by the following criteria:

1. Evacuation time: the least time during which all the generated vehicles in the first $100s$ get through the intersection.
2. Throughput: the ratio between the number of vehicles getting through the intersection and the number of vehicles entering the V2I range on the

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

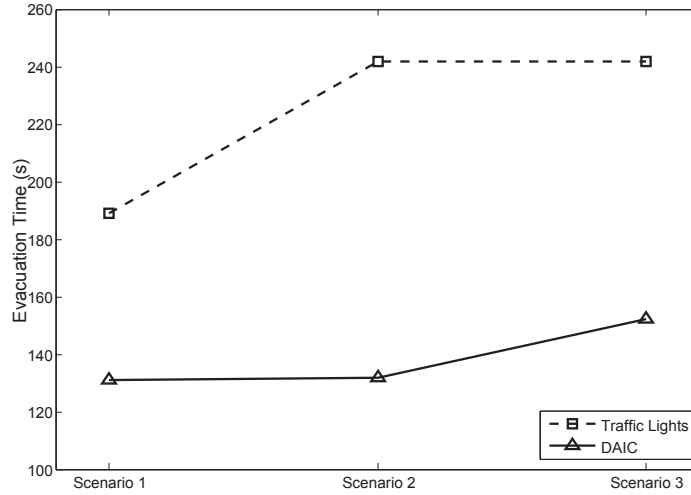


Figure 4.7: Evacuation time comparison

incoming streets during a whole signal period of Fixed Lights. It should be clarified that there is no V2I communication in Fixed Lights, so its "V2I range" is virtual and it is just used for statistic.

3. Delay: the increase in travel time due to the presence of intersection. Travel time is defined as the period between the moment when vehicle gets through the intersection and the moment when vehicle enters the V2I range. Then delay can be calculated as the difference between the real travel time under intersection control and the estimated travel time if the vehicle travels at the speed limit for the entire journey.

As shown in Figure 4.7, for traditional traffic lights, the evacuation time for the three scenarios are separately 189s, 240s and 242s; while by using our DAIC algorithm, the evacuation time are separately 130s, 132s and 152s. It can be concluded that DAIC takes much less time to evacuate all the vehicles in each scenario. From lower arriving rate in scenario 1 to higher value in scenario 3, the evacuation time increases for 53s under traffic light control, and for 22s under DAIC which is only the half of the former.

Then all the vehicles' delays are counted in Figure 4.8. The average delays under traffic control are 33s, 50s and 47s, while the maximum value increases from 83s to 169s. And the average delays under DAIC are 0.7s, 1.4s and 2.3s, and the maximum value goes from 6.9s to 20.5s. Much smaller gap in maximum

delay can be found in DAIC. It can be seen that DAIC overtakes traffic lights largely on this criteria.

The throughput during a signal period, which is 102s, is shown in Figure 4.9. The throughputs controlled by DAIC in the 3 scenarios are all above 80%. While the values for traffic lights are around 30% ~ 40%. Therefore the throughputs using DAIC is more than twice of the ones using traffic lights.

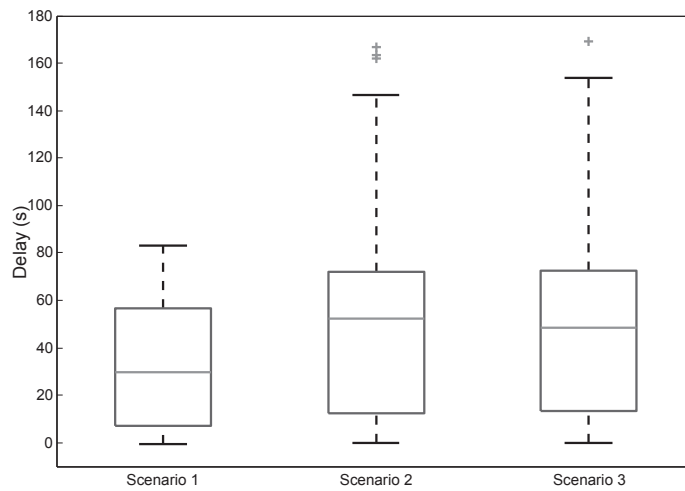
A more intuitional view can be seen in Figure 4.10. Screen shots at the same moment $t = 70s$ are taken for comparison. It can be noted that, there are much more vehicles in the view for intersection controlled by traffic lights than the one controlled by DAIC. For traffic lights, flows having intersection point in their trajectories can not get into the intersection simultaneously, so that vehicles have to be held before the intersection. Especially in light traffic case, if the signal period is fixed, a vehicle still have to wait even if there is no vehicle on the cross road. Meanwhile this limit is broken through in DAIC, the intersection is divided into smaller conflict zones, and vehicles can be inside the intersection at the same time if they don't take any conflict zone simultaneously. Thus the intersection space is used much more efficiently.

Based on the simulation results, it can be concluded that DAIC has huge advantage in evacuation time, delay and throughput compared to traditional traffic lights in the designed scenarios.

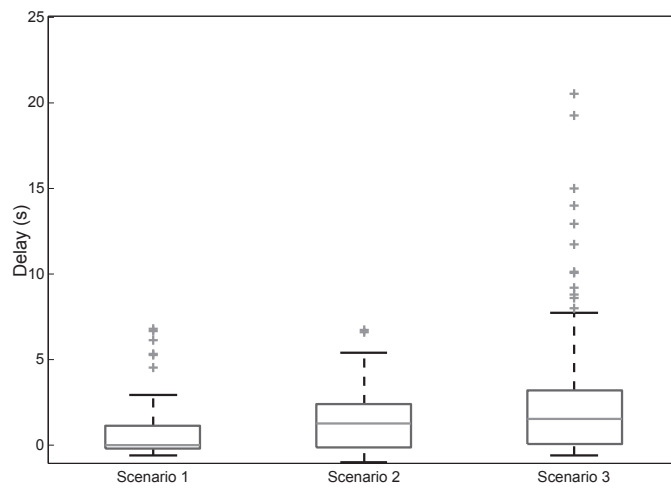
4.6 Conclusion

In this chapter, a decentralized autonomous intersection control (DAIC) algorithm is developed. The vehicles in the V2I range are divided into three switchable sets. At each time step, the vehicles in the tuning set should make their own decision in turn following decided priorities deduced from the vehicles' distances to the intersection, types and routes. In order to find the optimal acceleration value in its action set, each vehicle in the tuning set must require the conflict zone occupation time of the vehicles in cruising set and the ones with higher priority in tuning set from the intersection manager via V2I and the related information from the preceding vehicle via V2V. When the vehicle has found a collision-free velocity, it should maintain this velocity until it gets through the intersection.

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC



(a) Traffic lights



(b) DAIC

Figure 4.8: Delay in different scenarios

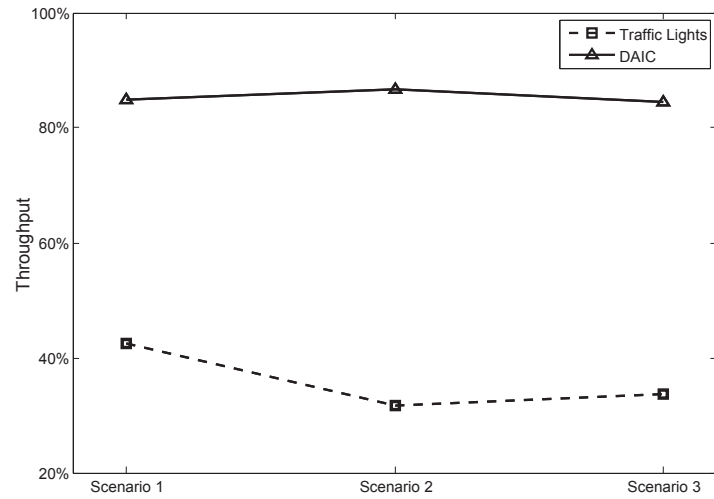


Figure 4.9: Throughput comparison

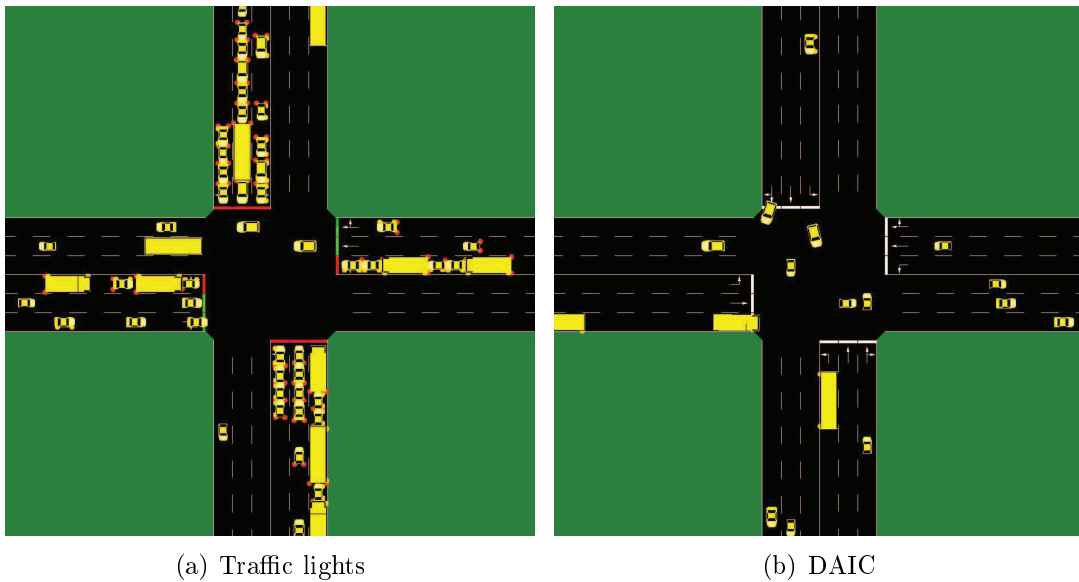


Figure 4.10: Screen shots at $t = 70s$ using traffic lights and DAIC

4. DECENTRALIZED AUTONOMOUS INTERSECTION CONTROL FOR LIGHT TRAFFIC

Simulation results shows that DAIC algorithm overtakes greatly the traditional traffic lights in evacuation time, delay and throughput under light traffic conditions.

It should be indicated that our DAIC algorithm still has some disadvantages. For example, the current algorithm may not work under heavy traffic conditions. When the vehicles' arriving rate is high, the average speed on the incoming roads, especially on left turning lanes, decreases with time till stop. Thus a restart policy needs to be designed to guide the vehicles after the stop.

Chapter 5

Trajectory Planning for Autonomous Intersection Management

Contents

| | | |
|------------|--|------------|
| 5.1 | Introduction | 101 |
| 5.2 | Assumptions and Design objectives | 103 |
| 5.3 | Protocol Design | 104 |
| 5.3.1 | Overview | 104 |
| 5.3.2 | Definitions | 105 |
| 5.3.3 | Window Searching | 110 |
| 5.3.4 | Trajectory Planning | 113 |
| 5.4 | Simulation | 119 |
| 5.4.1 | Settings | 119 |
| 5.4.2 | Results | 120 |
| 5.5 | Conclusion | 124 |

5.1 Introduction

In Chapter 4, a Decentralized Autonomous Intersection Control (DAIC) is designed to regulate the each vehicle' velocity to find a value which permits the

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

vehicle get through the intersection with no collision and less delay. In light traffic conditions, DAIC is capable to find a collision-free velocity. However, when facing heavier traffic, this algorithm may be struggling. Because vehicles may not have other solution than stopping themselves before entering the intersection, but DAIC has no policy to restart a stopping vehicle, which is a main drawback of DAIC. Besides, the regulation is often in one-direction: accelerating or decelerating, and after finding a collision-free value, the vehicle will maintain this velocity until fully getting through the intersection. Thus the vehicle's trajectory is not flexible, and there could have better trajectories. Basically, DAIC is a negotiation-based (with decision priority) algorithm, at each time step, the related vehicles make decision based on the current situation, however, the trajectories afterwards are not planned. In order to overcome the drawbacks of DAIC, we intend to solve the problems in a new way: another philosophy for autonomous intersection management is attempted, which is planning-based approach.

Several methods can be used to plan the motion for vehicles approaching the intersection (Frese & Beyerer (2011b)), including tree search (Frese & Beyerer (2010)), mixed-integer linear programming (Schouwenaars *et al.* (2001)) and elastic bands (Hilgert *et al.* (2003)). If the vehicles' motion is planned simultaneously, the computational cost increases exponentially with the number of vehicles. Thus prioritized planning is developed (Van Den Berg & Overmars (2005)) to decompose the problem into lower-dimension problems. And the problem can also be decomposed in time scale using partial motion planning (PMP) if the complete motion to the goal cannot be found within the available time (Petti & Fraichard (2005)).

In this chapter, a trajectory planning based autonomous intersection management protocol (TP-AIM) is developed to guarantee safety while improving efficiency of an intersection in connected autonomous vehicles context. A manager is installed at the intersection which acts as an information hub where conflict zone occupancy time (CZOT) is stored. When entering the V2I range, vehicle demands the related CZOT from the intersection manager; then the latter responses these demands in turn following a priority defined by the vehicles' types, headings and distances to the intersection. After having got the response, a window searching algorithm is used to find an entering time and velocity to enter the intersection with no collision and minimal delay. Then the vehicle begins to plan its trajectory

using a segmented dynamic programming algorithm based on the decided entering window and the future motions of the preceding vehicle. Backup entering windows should be used to restart trajectory planning if the vehicle can not find an feasible trajectory within its dynamical limits. TP-AIM is tested and verified using a traffic simulator SUMO under different traffic densities, and results prove that TP-AIM greatly outperforms traditional traffic lights and adaptive traffic lights.

The rest of the chapter is organized as follows. In section 5.2, the scenario of our research is described, and the control objectives are established. In section 5.3, our protocol, including priority setting, window searching algorithm and trajectory planning algorithm, is introduced in details. In section 5.4, simulation results are presented to prove the validation of our algorithm, and comparisons are made with other intersection management methods. Finally, some concluding remarks and perspectives are given in Section 5.5.

5.2 Assumptions and Design objectives

The study object is the same intersection described in Chapter 4. The conflict zones are also defined in the same way. Besides, the research is based on the same assumptions: vehicles are completely autonomous in the vicinity of intersection; vehicles are all equipped with V2X devices; the communications are capable and effective; lane-changing maneuvers are accomplished before entering the V2I communication range.

In this chapter, a new protocol for autonomous intersection management is developed with following design objectives:

- heterogeneity of vehicles must be considered;
- the decision should be made locally at each vehicle in order to reduce the computational cost at the intersection manager;
- entering windows for each vehicle indicating the possible moments and velocities to enter the intersection needs to be found in order to avoid cross collisions;
- a vehicle's trajectory before entering the intersection must be planned to avoid rear-end collision and to match with the entering window;

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

- fail-safe mode should be designed to deal with situations where the trajectory planning cannot find a solution for the given entering window;
- the average delay of vehicles should be reduced.

5.3 Protocol Design

5.3.1 Overview

An overview of TP-AIM protocol is given in Figure 5.1. When approaching an intersection, before entering the V2I range, a vehicle circulates according to its implemented self-driving algorithm; after getting inside the range, the vehicle demands the intersection manager for the related conflict zone occupancy time (CZOT), which is defined in the following subsection. The manager processes these requests in a decided order: the vehicles which have proposed the demand are divided into service groups of which the priority is given by the vehicles' types, headings, and distances to the intersection. A vehicle can only get the CZOT information after its related vehicles in the previous group, which shares conflict zones with it, have finished the trajectory planning process.

Given the related CZOT, the vehicle should utilize the window searching algorithm (5.3.3) to find an optimal entering moment and velocity which leads to a collision-free trajectory inside the intersection with minimal delay, as well as backup solutions which are also collision free but have larger delays. It is supposed that vehicles circulate at constant speed in the intersection.

After that, the vehicle needs to plan its own trajectory based on its current position, velocity and its acceleration/deceleration ability to enter the intersection at the decided entering moment with the correct entering velocity. On one hand, if the vehicle currently has a preceding vehicle on the same lane, the former should request the latter for future motions which have been planned; then a Segmented Dynamic Programming algorithm (5.3.4.1) is used to plan actions in order to achieve intermediate goals and to avoid running into the preceding vehicle. On the other hand, if the vehicle is the first one on its lane, it should use a three-section acceleration template (5.3.4.2) to plan its trajectory. If the trajectory planning succeeds, the vehicle must send a message to the manager and the CZOT it has requested is formally determined. If the trajectory planning could

not find feasible inputs within the vehicle's dynamical limits, in other words, the vehicle can not profit the current entering window, therefore it should enter latter. Thus the requested CZOT should be canceled and a backup solution of the window searching is used to restart the trajectory planning process.

The design philosophy of TP-AIM is quite different from DAIC: in DAIC, at each time step, the vehicle tries to find the optimal control input which leads to minimal cost value, and after arriving a collision-free speed, the vehicle will maintain this speed until getting through the intersection; in TP-AIM, the possible entering windows are decided at first, then the vehicle plans its trajectory based on these windows and other information, thus the vehicle's speed is non-constant before entering the intersection.

5.3.2 Definitions

5.3.2.1 Conflict Zone Occupation Time

Because of the difference in design philosophy, the definition of conflict zone occupancy time (CZOT) needs to be changed. The distance from the intersection's entrance to the conflict zone CZ_k in SCZ_i is denoted as d_{CZ_k} . For vehicle V_i , the set of all the conflict zones which it will pass through is defined as SCZ_i ; its velocity inside the intersection is denoted as v_i and its length is denoted as l_i . If we note the moment at which V_i enters the intersection as t_{enter} , then the moment when V_i enters and leaves CZ_k , which are separately denoted as t_{1,V_i} and t_{2,V_i} , can be defined as follows

$$\begin{aligned} t_{1,V_i} &= t_{enter} + \frac{d_{CZ_k}}{v_i} \\ t_{2,V_i} &= t_{enter} + \frac{d_{CZ_k} + \delta_{V_i,CZ_k}(l_i, w)}{v_i}, \end{aligned} \tag{5.1}$$

where $\delta_{V_i,CZ_k}(l_i, w)$ is the distance to be covered by V_i to get through CZ_k , which is related to V_i 's heading and the position of CZ_k in the intersection.

Then the usage of time buffer and the definition of overlapped occupancy time are conducted in the same way as in 4.2 and 4.4.

It should be indicated that the CZOT stored at the manager is anonymous and only related to conflict zone index. The occupancy time of conflict zone CZ_k

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

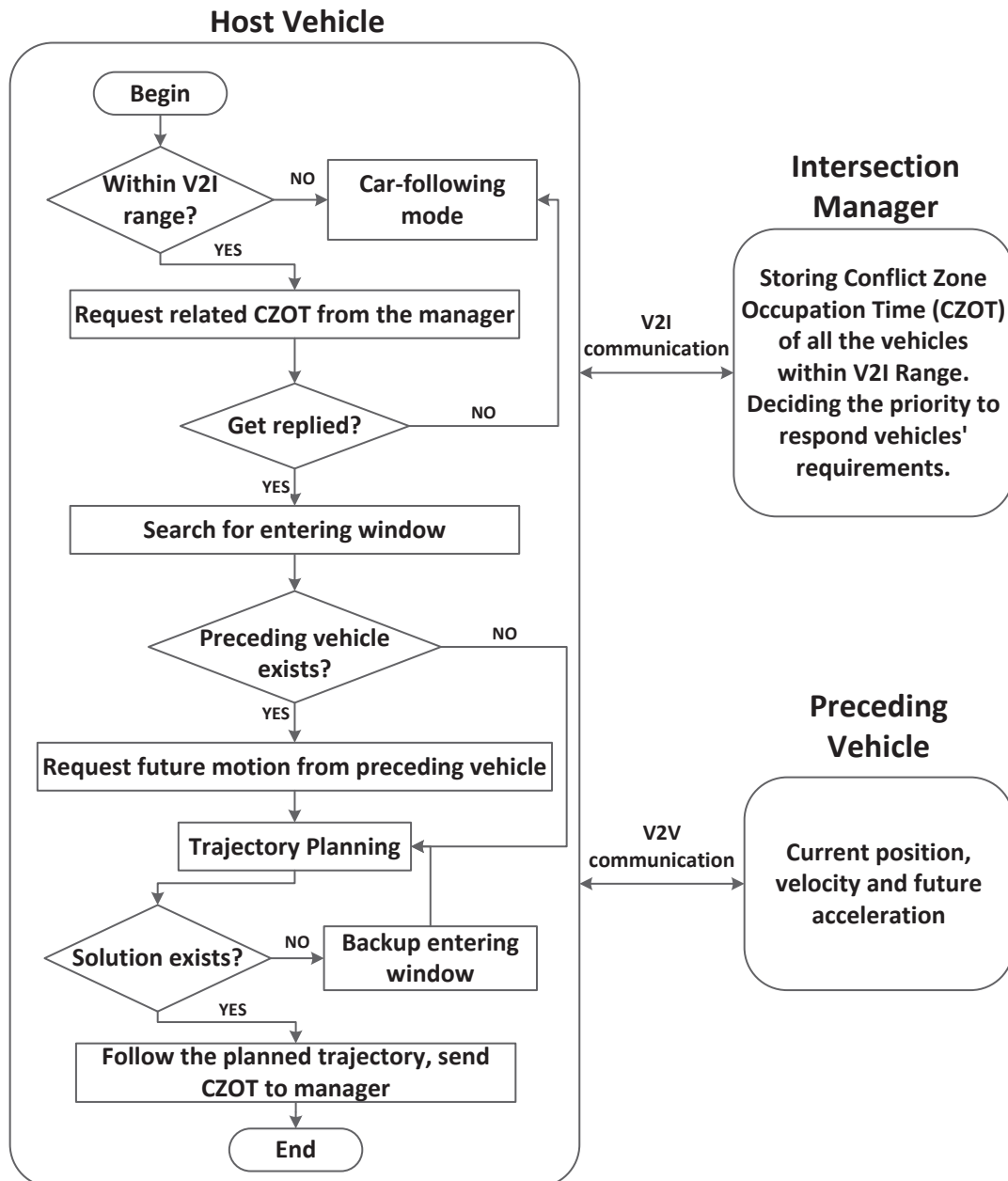


Figure 5.1: Overview of TP-AIM

is defined as

$$CZOT(CZ_k) = \{OT_{V_i, CZ_k} | \forall V_i\}. \quad (5.2)$$

5.3.2.2 Vehicle set notations

The vehicles on the incoming street within V2I range can be divided into three different sets:

- Free Set S_F : The vehicles which have demanded the CZOT, but have not got the response from the manager.
- Planning Set S_P : The vehicles which have obtained the response from the manager, and they are trying to find a feasible and collision-free trajectory to get through the intersection.
- Cruising Set S_C : The vehicles which have found their trajectories to get through the intersection, and their CZOT is stored at the intersection.

On the same lane, if we sort the vehicles on an incoming street by the distance to the intersection, S_C has the smallest distance, then it comes to S_P and finally S_F . The number of vehicles in S_C and S_F is not limited. Meanwhile, there is at most one vehicle for each lane in S_P . Because the CZOT of a vehicle can be decided if and only if it has find a trajectory, then it becomes a member of S_C , and its nearest following vehicle can get the response from the manager and become a member of S_P . Therefore there are up to 12 vehicles in S_P for our intersection model.

5.3.2.3 Service Priority

After demanding the CZOT information from the intersection manager, the vehicle could not get the response immediately. Because the CZOT information stored at the manager may not be complete, and this incomplete information would cause incorrect decision of the current vehicle. For example, vehicle V_i and V_j send their demands simultaneously, and the two vehicles have conflict zone in common; the manager cannot answer them at the same time using the current incomplete CZOT information, because the two vehicles are related. Therefore, the two vehicles must be responded one by one, the first one only need to consider the current CZOT while the second vehicle have to take the first one into account

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

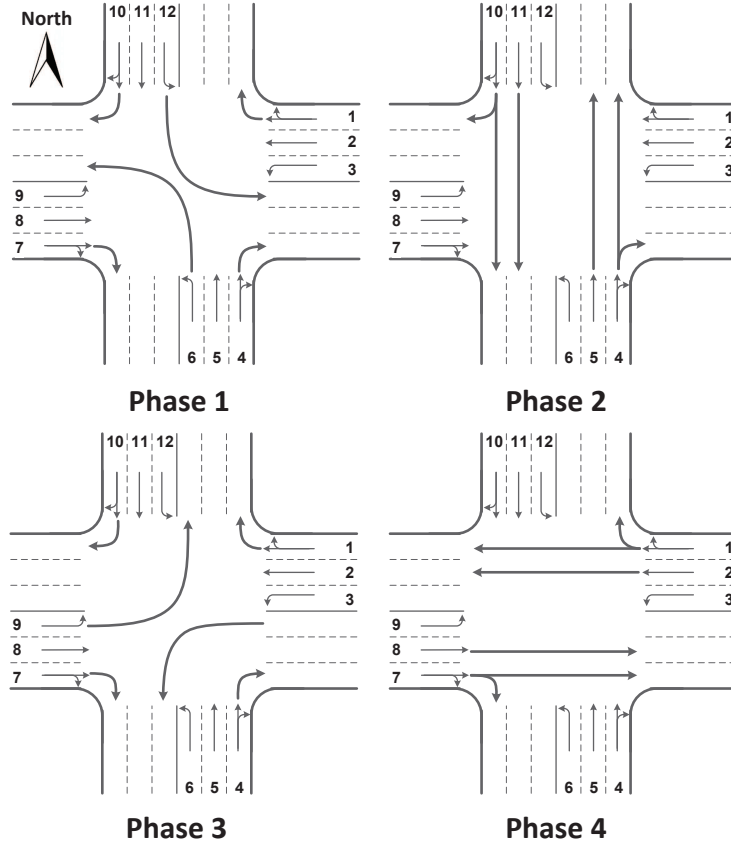


Figure 5.2: Phases for 4 serving groups

as well. Thus a fair and efficient service priority needs to be defined. A more efficient service rule compared to the one in the previous chapter is designed.

As it is assumed that there is no lane-shifting and overtaking operation within the V2I range, therefore on the same lane, a vehicle must get through the intersection before its following vehicles. Thus for the vehicles on the same lane, a first-come first-served principle is used. It is sufficient to only take the first vehicle in the free set S_F on each lane into consideration at a time, and this subset of S_F is denoted as S'_F . There are at most 12 vehicles in S'_F .

Firstly, the vehicles in S'_F is divided into four groups according to their headings. This division is inspired by the four phases of traditional traffic lights shown in Fig.5.2. Each phase is a biggest set of collision-free headings in which vehicles can circulate simultaneously, so the vehicles in the same group are independent and they can get their requested CZOT at the same time. Secondly, the vehicles in S'_F are given scores according to their distances to the intersection, headings,

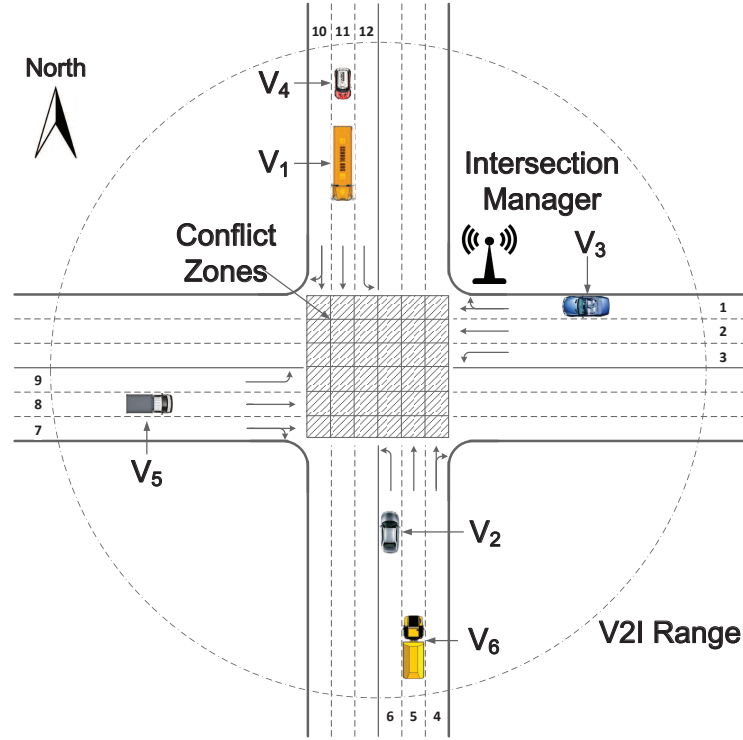


Figure 5.3: Intersection illustration

and types, by using the principles defined in 4.3.3. Thirdly, the groups' serving priority is sorted according to the maximal score in the group. The group with the highest priority is denoted as G_1 , then it comes to G_2 , G_3 and G_4 . The manager should send the requested CZOT information to vehicles in G_1 at first. Then these vehicles begin to search for the entering windows and plan the trajectories. After having found a feasible collision-free trajectory, the vehicle sends a confirmation message to the manager. The manager does not have to wait for all the definite answer from G_1 to begin to serve the members of G_2 . If all the related CZOT of a vehicle in G_2 is decided, then it can be served before all the vehicles in G_1 have found a solution. After all the vehicles in the current S'_F have found their trajectories, S'_F is updated to their nearest following vehicles in S_F , then the manager begins to repeat the three steps to find the priority to serve the new S'_F .

For instance, in Fig.5.3 six vehicles are approaching the intersection, and they have all proposed their requirement to the manager. Among them there are five vehicles belongs to S'_F : V_1 , V_2 , V_3 , V_5 , V_6 . According to their headings, V_2 is in

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

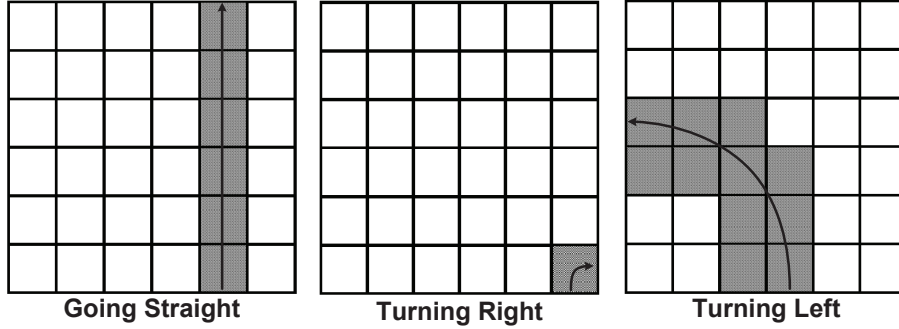


Figure 5.4: Conflict zone occupancy

phase 1, V_1 and V_6 are in phase 2, V_3 and V_5 are in phase 4. Therefore the vehicles in S'_F can be divided into three service groups. V_1 gets the highest score because it is a school bus and it is close to the intersection. And V_2 gets higher score than V_3 and V_5 because it will turn left. Thus V_1 and V_6 is in the first service group G_1 , so they should be served at first; V_2 is in the second service group G_2 , therefore once V_1 has sent its confirmation message to the manager, it can get all the necessary information; finally V_3 and V_5 can get their demanded CZOT after V_2 has found a feasible trajectory.

It should be clarified that the service priority does not represent the priority to get through the intersection. A vehicle with lower service priority needs the CZOT of all related vehicles with higher priority to plan its own trajectory, in other words, the lower the service priority is, the more the vehicles to be considered.

5.3.3 Window Searching

The nature of intersection management is to solve the space-time conflict so that the vehicles can share the limited space resource inside the intersection. When a vehicle is replied by the manager, it has the occupancy time of all the conflict zones that it will pass through. The conflict zones that the vehicles with different headings will occupy is shown in Fig. 5.4. The set of conflict zones to be occupied when vehicle V_i goes through the intersection is noted as SCZ_i .

If we unroll the CZOT in time dimension, the CZOT information in the response from the manager can be seen as obstacles, and finding an entering window can be interpreted as finding a trajectory to bypass these obstacles. For example, for a vehicle going straight, its unrolled CZOT is shown in Fig. 5.5. $CZOT(CZ_k)$

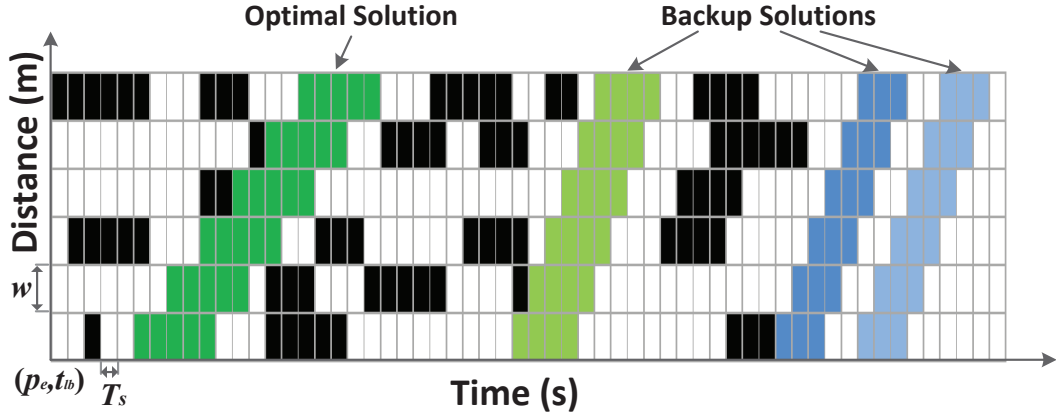


Figure 5.5: Window Searching for a vehicle going straight

is presented as a line of grids in the figure. The distance axis begins from the entrance of the intersection p_e , and each segment represents the width w of a conflict zone. The time axis begins from the lower boundary t_{lb} of the searching time interval which is defined later and each segment on this axis represents the protocol's sampling time T_s . The black grids in the figure means that the related conflict zone is occupied by other vehicles during the indicated time, and the occupancy is already stored at the manager, so it cannot be changed. Conversely, the blank grids indicates that the conflict zone is available so that it can be occupied by the host vehicle.

Obviously, the vehicles in S_C are countable, so their CZOT stored at the manager is not infinite, therefore at least the host vehicle can get inside the intersection after all the vehicles in S_C having done occupying the zones in SCZ_i . In this case, if the vehicle enters the intersection in the speed limit, it would be minimally delayed.

However, the vehicle could insert earlier by taking advantage of the available grids. That's why the window searching algorithm is designed. As assumed before, the vehicles circulates in a constant speed in the intersection, so the vehicle should maintain the entering velocity until it totally gets through. Thus according to the entering time and velocity, the vehicles' occupancy time at different conflict zones can be predicted. The objective is to search for an earlier entering window, including entering time and velocity, which produces an collision-free trajectory in the intersection, in other words, the grids that the vehicle plans to occupy should not overlap with the black grids.

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

For a pair of entering time and velocity noted as (t_{enter}, v_{enter}) , the occupancy time OT_{V_i, CZ_k} of $CZ_k \in SCZ_i$ can be deduced using (5.1) and (4.2). Then the overlapped occupancy time with $CZOT(CZ_k)$ can be calculated in the same way as (4.4):

$$OOT_{CZ_k, V_i} = \sum_{OT_j \in CZOT(CZ_k)} f_{OOT}(OT_{V_i, CZ_k}, OT_j), \quad (5.3)$$

where OT_j is a successive occupancy time in $CZOT(CZ_k)$. Then the total OOT for all $CZ_k \in SCZ_i$ is defined as:

$$OOT_{total, V_i} = \sum_{CZ_k \in SCZ_i} OOT_{CZ_k, V_i}. \quad (5.4)$$

If OOT_{total, V_i} equals to 0, then V_i will not collide other vehicles when it enters at t_{enter} with the velocity v_{enter} . Otherwise, the pair (t_{enter}, v_{enter}) is not a safe entering window for V_i .

Another objective is to minimize the delay of vehicles, and it can be expressed as a minimized exiting time t_{exit} for getting out of the intersection, which is also the function for t_{enter} and v_{enter} . Combining these two objectives, a cost function can be defined:

$$J_{cost, V_i}(t_{enter}, v_{enter}) = k_1 \cdot OOT_{total, V_i} + k_2 \cdot (t_{exit} - t_0), \quad (5.5)$$

where t_0 is the current moment, and k_1, k_2 are the weights of the two parts. As safety is the first priority, k_1 should be much bigger than k_2 .

The searching area needs to be bounded in order to get close to reality and to produce solutions which have higher success rate in trajectory planning. The lower boundary t_{lb} of t_{enter} is defined as

$$t_{lb} = \max(t_{fullspeed}, t_{enter, preced}), \quad (5.6)$$

where $t_{fullspeed}$ is the moment when the vehicle enters the intersection if it accelerates to the road's speed limit, and $t_{enter, preced}$ is the entering time of the nearest preceding vehicle. t_{lb} is set like so because the vehicle cannot enter the intersection earlier than using the speed limit or before its preceding vehicle. The upper boundary t_{ub} of t_{enter} is defined as the moment when all the conflict zones

in $CZ(V_i)$ are not occupied afterwards:

$$t_{ub} = \max_{CZ_k \in SCZ_i} \max(CZOT(CZ_k)). \quad (5.7)$$

Then a lower boundary v_{lb} should be given to v_{enter} in order to guarantee the traffic's fluency inside the intersection. And the upper boundary v_{ub} of v_{enter} is defined as the road's speed limit v_{limit} .

Thus the window searching problem is transformed to a two-dimension optimization problem, of which the objective is to find the value of an entering window (t_{enter}, v_{enter}) within constraints that produces the minimal J_{cost, V_i} . Several optimization algorithms can be used to solve this problem. In this chapter, we utilize the Particle Swarm Optimization (PSO) Algorithm (Kennedy (2010)) because of its simplicity in exploration and exploitation.

During the searching process, all the pairs of (t_{enter}, v_{enter}) having zero overlapped occupancy time are recorded in the vehicle's cache as backup solutions, and the optimal solution would be attempted at first for trajectory planning. If the planning fails to find a feasible trajectory, then a backup solution is used to restart the planning. If window searching algorithm cannot find any collision-free solution, it means that there is no chance to enter the intersection earlier and the vehicle should insert using the speed limit after all vehicles in S_C having got out of its way. An example of optimal solution and backup solutions can be referred to Fig. 5.5.

5.3.4 Trajectory Planning

After finding entering windows, the vehicle needs to plan its trajectory before entering the intersection according to these windows. Two different algorithms are designed for two situations the vehicles can encounter: with/without a preceding vehicle in the same lane. For the first situation, vehicle's trajectory is planned by a segmented dynamic programming algorithm; while for the second situation, a three-section acceleration template is used. Normally, these two situations happen successively: the preceding vehicle enters the intersection during the planning scope, so the two algorithms need to be deployed one after the other. It is assumed that the whole planning process can be finished in a computation

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

period δ_c . During δ_c , the vehicle circulates according to the self-driving algorithm and the position and velocity at $t_0 + \delta_c$ are assumed to be predictable.

5.3.4.1 With preceding vehicle

If there is a preceding vehicle V_j before V_i , then the first goal of V_i is to avoid rear-end collision with V_j , and catching up with the decided entering window becomes the second. In this situation, the vehicle should plan its trajectory from the moment $t_0 + \delta_c$ to $t_{enter,preced}$, when the preceding vehicle enters the intersection.

The maximum acceleration and deceleration of vehicle V_i are separately denoted as a_{max} and b_{max} . It is assumed that the V_i can choose an acceleration to execute, called an action, from several values between the two limits. The action set A_i of V_i is defined as:

$$A_i = \{a_{i,1}, a_{i,2}, \dots, a_{i,n}\}, \text{ where } a_{i,j} \in [-b_{max}, a_{max}]. \quad (5.8)$$

For different vehicles, the dimension $\|A_i\|$ of action set can also be different. Since the preceding vehicle V_j 's trajectory has been successfully planned, the host vehicle V_i can request necessary information via V2V to predict V_j 's position and velocity in the future.

The vehicle's trajectory can be planned using dynamic programming (DP) ([Wuthishuwong et al. \(2015\)](#)), because this problem can be simplified by breaking it down into a sequence of decision steps over time, and a sub-problem can be solved recursively by using the results of its previous steps. And the results of overlapping subproblems are memorized and looked up subsequently. As shown in Fig. 5.6, each node represents a state (p, v) including a position and a velocity. The planning begins with an initial state (p_0, v_0) , and terminates with a target state (p_{target}, v_{target}) . Each node has $\|A_i\|$ actions to produce new states at every step, but not all the child-nods are qualified: the state that a child-node represents may be out of the velocity's limit or may cause collision with the preceding vehicle, so it should be deleted and cannot produce new child-nodes. The nodes produced at the final step are called the final states. The objective is to find the final state which has the smallest error to the target state. The nodes belongs to different layers, if a node is the j th in the same layer and behind it there are N steps to be planned afterwards, then this node is noted as $n_{N,j}$, and the set of all its

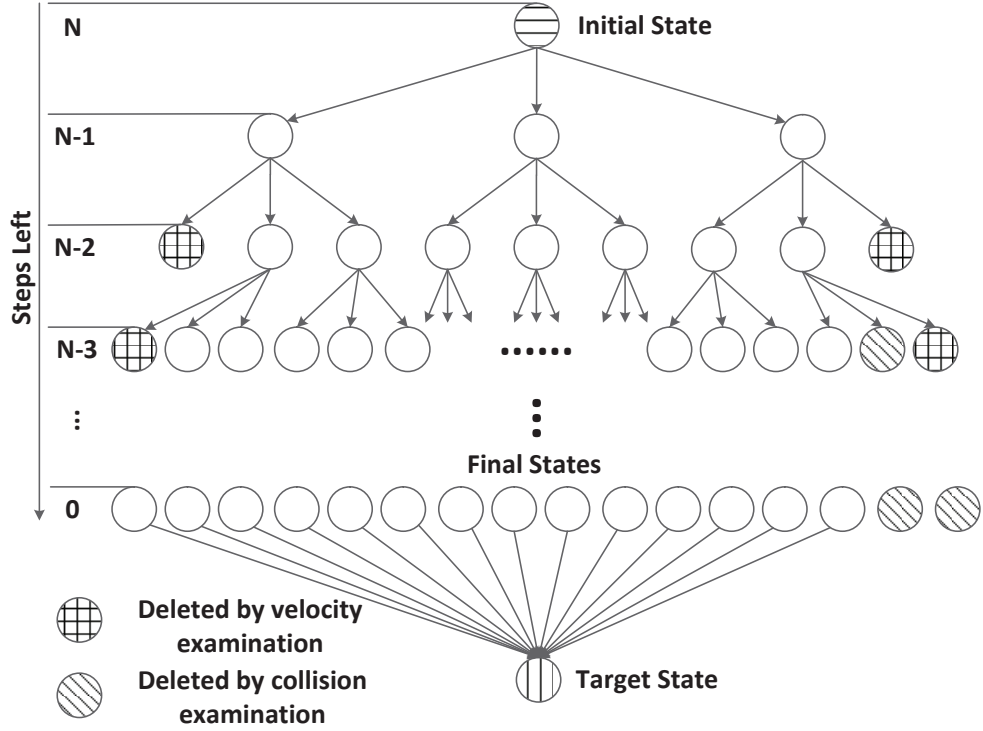


Figure 5.6: Planning trajectory using Dynamic programming

qualified child-nodes is noted as $CN(n_{N,j})$. If two nodes at the same layer have the same state, their child-nodes will be exactly the same as well. Thus they can be regarded as overlapping subproblems.

If DP is deployed directly using T_s as the step length, it could be considerably expensive in computation. For example, if we need to plan the trajectory for the future $20s$, $T_s = 0.2s$ and $\|A_i\| = 5$, by using DP, there are 100 steps to be planned, and the number of nodes at the last step are $O(5^{100})$. Therefore, it is supposed that an action is maintained during the planning step ΔT , which is an integer multiple of T_s . The total number of steps to be planned is

$$N_{step} = \text{ceil} \left(\frac{t_{enter,preced} - t_0 - \delta_c}{\Delta T} \right), \quad (5.9)$$

where $\text{ceil}(x)$ is the function which returns the smallest integer larger than or equal to x . The target state of DP is defined as

$$\begin{aligned} p_{target} &= p_{enter} - v_{enter} \cdot (t_{enter} - \Delta T \cdot N_{step} - t_0 - \delta_c) \\ v_{target} &= v_{enter} \end{aligned} \quad (5.10)$$

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

And it represents an ideal case: when the preceding vehicle enters the intersection, the host vehicle can reach v_{enter} and it does not have to make adjustment before t_{enter} .

If ΔT is too big, the error between the final state and the target could be large; whereas if ΔT is too small, N_{step} is still big which leads to high computational cost. Therefore, a segmented dynamic programming is designed to solve this dilemma: the original DP process is divided into several successive and smaller DP process with limited steps to compute and intermediate goals to achieve. For the same example defined previously, if $\Delta T = 2s$, then $N_{step} = 10$; if the original DP is divided into 2 successive DP of 5 steps, then the number of nodes in final states are $O(5^5)$. Thus the computational time can be greatly reduced compared to the one in the previous example.

The number of steps in the i th segmented DP is defined as N_i , and the maximum number of steps in a segmentation is denoted as N_{max} , so

$$\sum N_i = N_{step}, N_i < N_{max}. \quad (5.11)$$

The moments in i th segmented DP are noted as $t_{i,N_i}, t_{i,N_i-1} \dots t_{i,1}, t_{i,0}$. Then the velocity and position of the child-nodes are examined by

$$\begin{aligned} 0 \leq v(t_{i,j+1}) \leq v_{limit}, t_{i,j} - t_{i,j+1} = \Delta T \\ p(t) < p_{preced}(t) - l_{preced}, \forall t \in [t_{i,j}, t_{i,j+1}]. \end{aligned} \quad (5.12)$$

where p_{preced} is the position of the preceding vehicle, and l_{preced} is its length. The child-nodes which are outside these limits should be deleted and they cannot generate new child-nodes in the next step.

The intermediate target of the whole DP is the target of i th segmented DP, which is defined as followings

$$p_{temp} = p_0 + \frac{\sum_{j \leq i} N_j}{N_{step}} \cdot (p_{target} - p_0) \quad (5.13)$$

$$p_{target,i} = \min(p_{preced}(t_{i,0}) - l_{preced}, p_{temp})$$

$$v_{temp} = v_0 + \frac{\sum_{j \leq i} N_j}{N_{step}} \cdot (v_{target} - v_0) \quad (5.14)$$

$$v_{target,i} = \min(v_{preced}(t_{i,0}), v_{temp})$$

This definition is an integrated consideration of the final target and the motion of the preceding vehicle.

If $n_{N_i,j}$ is a node which represents a final state of the i th segmented DP, thus $N_i = 0$, then its cost $C_{0,j}$ is defined as

$$C_{0,j} = \omega_1 \cdot \text{abs}(p_{0,j} - p_{target,i}) + \omega_2 \cdot \text{abs}(v_{0,j} - v_{target,i}), \quad (5.15)$$

where $p_{target,i}$ and $v_{target,i}$ are the target position and velocity of the i th segmented DP, and ω_1 and ω_2 are the weights of position and velocity error. This cost is called the terminal cost.

Otherwise, for a node with $N_i > 0$, its cost is defined as

$$C_{N_i,j} = \min(\{C_{N_i-1,k} + \text{abs}(a_{N_i,k}) | n_{N_i-1,k} \in CN(n_{N_i,j})\}) \quad (5.16)$$

$$a_{N_i,m} = \arg \min(\{C_{N_i-1,k} + \text{abs}(a_{N_i,k})\}), \quad (5.17)$$

where $n_{N_i-1,k}$ is a child-node of $n_{N_i,j}$, and $a_{N_i,k}$ is the action used to produce it. The term $\text{abs}(a_{N_i,k})$ is defined as the control cost.

By calculating the cost of the node at the initial state, an action profile with minimal cumulated control cost can be found which can lead to a node having small error to the target state. Then the action profiles of segmented DP are connected successively to generate vehicle's trajectory from $t_0 + \delta_c$ to $t_{enter,preced}$.

5.3.4.2 Without preceding vehicle

Normally, this situation happens when the preceding vehicle has entered the intersection. Because of the existence of the preceding vehicle and the limited action set, a vehicle may not reach the target state (p_{target}, v_{target}) , so adjustment should be done to catch up with the entering window (t_{enter}, v_{enter}) . In this case, the vehicle only needs to consider the entering window because the preceding vehicle's occupancy time is already considered in the window searching process, so there will not be any rear-end collision.

A three-section acceleration template is designed to plan the vehicle's trajectory from $t_{enter,preced}$ to t_{enter} . It is assumed that the trajectory for this duration has three sections: acceleration, cruise and deceleration, or deceleration, cruise and acceleration. An uniform acceleration is considered to simplify the planning

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

Algorithm 6 Trajectory planning in a segmentation using DP

Input:

Steps left N_{left} , Nodes in this step $\mathbb{N}(P, V)$, Action set \mathbb{A} , Target state (p_{target}, v_{target}) , Speed limit v_{limit} , Position of the preceding vehicle p_{preced} ;

Output:

Nodes' action profiles $AP(\mathbb{N}, N_{left})$ and costs $C(\mathbb{N}, N_{left})$;

$[AP, C] = dp_planning(N_{left}, \mathbb{N}, \mathbb{A}, p_{target}, v_{target}, v_{limit}, p_{preced})$

```

1: if  $N_{left} = 0$  then
2:   Calculate the cost  $C_i$  for  $\forall n_i \in \mathbb{N}$  using 5.15;
   /*The nodes represent the final states */
3:    $AP \leftarrow \emptyset$ 
4: else
5:   Generate child-nodes  $\mathbb{CN}$  of  $\mathbb{N}$  using actions in  $\mathbb{A}$ ;
   /* Velocity Examination and Collision Examination */
6:   for  $cn_i(p_i, v_i) \in \mathbb{CN}$  do
7:     if  $p_i$  has collision with  $p_{preced}$  or  $v_i > v_{limit}$  then
8:        $\mathbb{CN} \leftarrow \mathbb{CN} \setminus cn_i$ ;
9:     end if
10:  end for
11:  if  $\mathbb{CN} = \emptyset$  then
12:    No qualified child-node, trajectory planning failed;
13:  else
14:     $\mathbb{S} \leftarrow \emptyset$ ;
15:    for  $cn_i(p_i, v_i) \in \mathbb{CN}$  do
16:      if  $\exists cn_j \in \mathbb{S}, p_i = p_j$  and  $v_i = v_j$  then
17:        Establish mapping between  $cn_i$  and  $cn_j$  /* Overlapped
        subproblems */
18:      else
19:        Add  $cn_i$  in  $\mathbb{S}$ ;
20:      end if
21:    end for
22:    Calculate  $[AP(\mathbb{S}, N_{left} - 1), C(\mathbb{S}, N_{left} - 1)] = dp\_planning(N_{left} - 1, \mathbb{S}, \mathbb{A}, p_{target}, v_{target}, v_{limit}, p_{preced})$ 
23:    Build  $AP(\mathbb{CN}, N_{left} - 1)$  and  $C(\mathbb{CN}, N_{left} - 1)$  using overlapped mapping;

24:    Calculate the cost  $C_i$  for  $\forall n_i \in \mathbb{N}$  using 5.16;
25:     $AP(\mathbb{N}, N_{left}) \leftarrow [a^*, AP(\mathbb{CN}, N_{left} - 1)]$ 
26:  end if
27: end if
28: return  $AP, C$ 

```

process. The first and the third section have the same value of acceleration, and the duration of the three sections are separately T_1 , T_2 and T_3 . The acceleration value should be minimized in order to reduce the fuel consumption. Then the trajectory planning problem can be translated into solving the following optimization program:

$$\text{minimize } |a| \quad (5.18)$$

with constraints

$$t_{enter,preced} + T_1 + T_2 + T_3 = t_{enter} \quad (5.19)$$

$$v_{final} + a \cdot (T_1 - T_3) = v_{enter} \quad (5.20)$$

$$\begin{aligned} d(v_{final}, a, T_1) + d(v_{final} + a \cdot T_1, 0, T_2) \\ + d(v_{final} + a \cdot T_1, -a, T_3) = p_{enter} - p_{final} \end{aligned} \quad (5.21)$$

$$0 \leq v_{final} + a \cdot T_1 \leq v_{limit} \quad (5.22)$$

$$-b_{max} \leq a \leq a_{max} \quad (5.23)$$

where (p_{final}, v_{final}) is the optimal final state given by the segmented DP, and $d(v, a, t)$ is the distance covered by the vehicle with initial speed v and acceleration a , during time t . The constraints (5.19), (5.20) and (5.21) are separately the constraints of time, velocity and position. (5.22) and (5.23) are the speed limit and acceleration limit.

If a value of a can be found within the constraints, then the trajectory from $t_0 + \delta_c$ to t_{enter} is successfully planned, and the vehicle can catch up with the current entering window. The vehicle should send a confirmation message to the manager, and its related CZOT is stored. Otherwise, the vehicle is not capable to take advantage of the entering window, therefore it should enter the intersection later. A backup solution in its cache is used to restart the trajectory planning process.

5.4 Simulation

5.4.1 Settings

In order to prove the effectiveness of our TP-AIM protocol, simulations are performed by using the same testbed setting in Chapter 4 composed of Matlab,

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

| Scenarios | Arriving Rate λ | | |
|-----------|-------------------------|-------------|-----------|
| | Right Lane | Middle Lane | Left Lane |
| S1 | 0.10 | 0.10 | 0.10 |
| S2 | 0.20 | 0.20 | 0.15 |
| S3 | 0.30 | 0.30 | 0.20 |
| S4 | 0.40 | 0.40 | 0.25 |
| S5 | 0.50 | 0.50 | 0.30 |

Table 5.1: Arriving rate in different scenarios.

SUMO and TraCI.

The intersection in simulation is same as the one shown in Fig. 5.3. The length of each incoming street is $500m$, and the V2I communication range R of the manager is set to $150m$. $V_{limit} = 70km/h(19.44m/s)$ is set to be the speed limit of all the incoming streets. The appearance of vehicles on each street is assumed to be a Poisson process with arriving rate λ . Five scenarios are designed for the simulation of which the arriving rates of vehicles for different lanes are given in TABLE 5.1. From scenario S1 to S5, the arriving rates increase so that the traffic becomes increasingly heavy. The traffic flows on each lane is randomly generated by following the arriving rate λ and the appearance probabilities of five vehicle types including sedan, hatchback, wagon, truck and bus, which are different in lengths and acceleration/deceleration abilities. The flows begins at $t = 0s$ and ends at $t = 100s$. The sample time of simulation is set as $Ts = 0.2s$. The time buffer $t_b = 0.2s$.

For the window searching algorithm, the lower limit of velocity inside the intersection v_{lb} is set to $10m/s$, and 10 particles search for the entering window in 50 iterations using PSO. In trajectory planning algorithm, each segmented DP can have at most $N_{max} = 4$ steps, and the dimension of a vehicle's action set $\|A_i\| = 7$. The simulation is executed in *Matlab* 2013a on a computer with *IntelCoreTMi5 - 2410M 2.3GHz*.

5.4.2 Results

Comparisons are made between the same intersection controlled separately by fixed-period traffic lights, adaptive traffic lights (Priemer & Friedrich (2009)) and our TP-AIM protocol in different scenarios. In the following, we use "Fixed

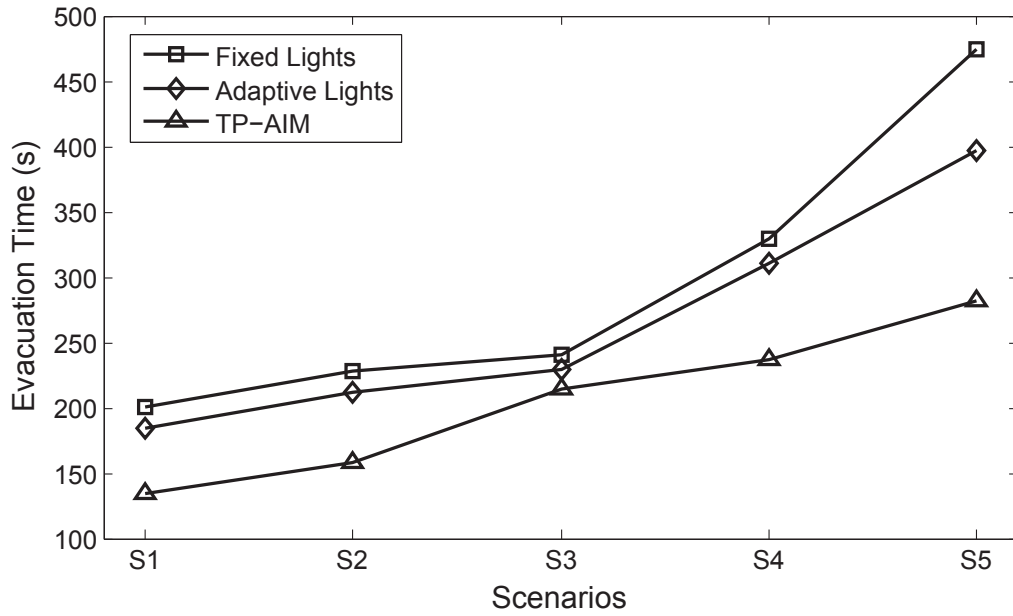
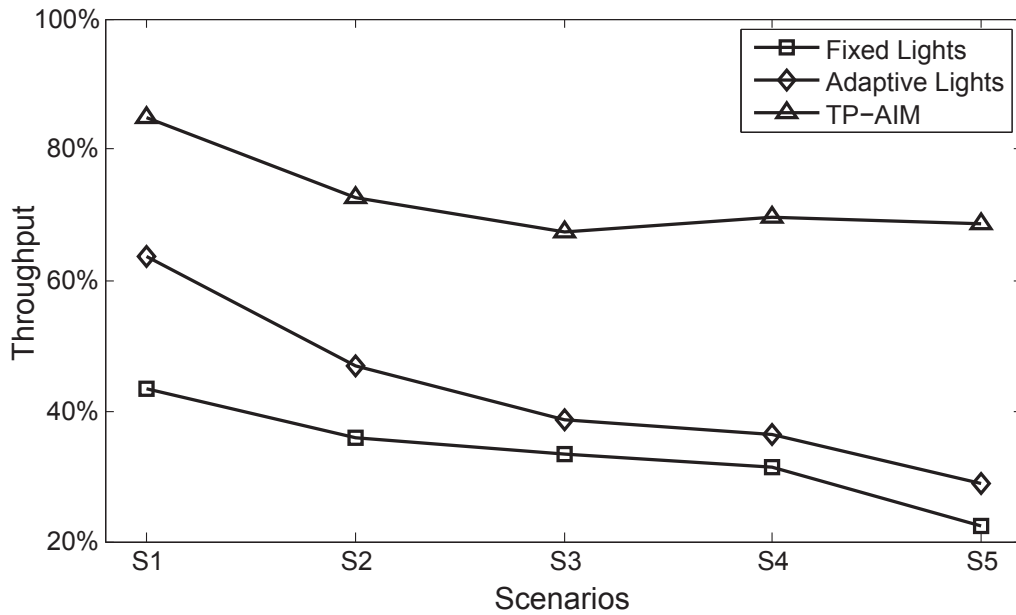
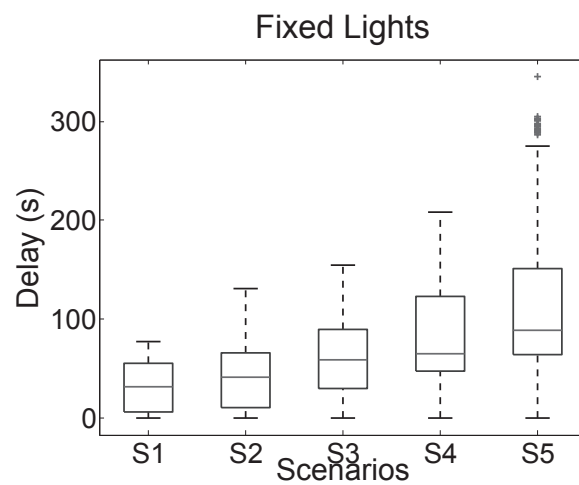


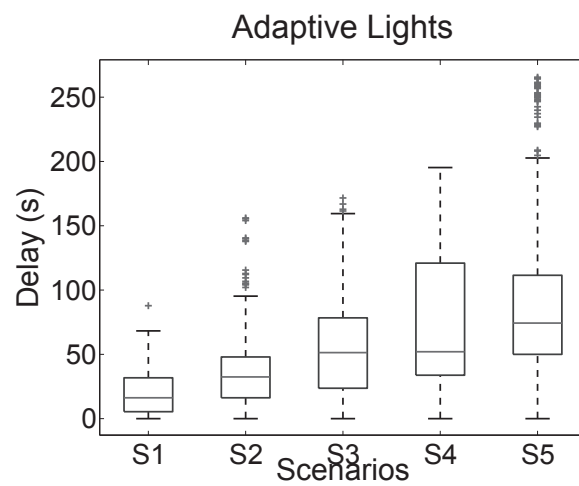
Figure 5.7: Evacuation time comparison

Figure 5.8: Throughput comparison at $t = 98s$

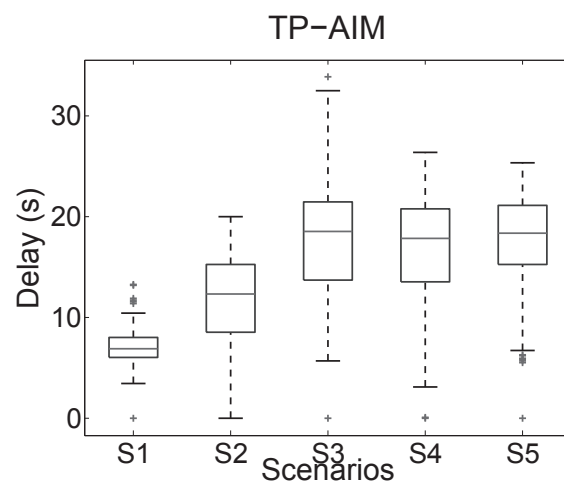
5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT



(a)



(b)



(c)

Figure 5.9: Delay Comparison

Lights", "Adaptive Lights" and "TP-AIM" to represent these three methods. For Fixed Lights, the length of green light for straight direction and right turning is 25s, for left turning is 20s, and the yellow light between two phases is 2s. Therefore, the phase period of Fixed Lights is 98s. The generated traffic flows for these three methods are exactly the same. The performance of different methods are evaluated by the same criteria defined in section 4.4.

In Fig. 5.7, it can be found that for the same scenario, TP-AIM protocol can always evacuate all the vehicles during the least time, and much bigger improvement in evacuation time can be seen in TP-AIM than in Adaptive Lights, especially under light and heavy traffic conditions; as the traffic gets heavier, evacuation time using the three methods all increases; however, the increase for TP-AIM (147.6s) is smaller than that for Fixed Lights (211.8s) and Adaptive Lights (273.8s).

Fig. 5.8 shows the throughput of the intersection at $t = 98s$, the end of the first signal period of Fixed Lights. The results show that TP-AIM always outperforms Fixed Lights and Adaptive Lights with great advantage. A throughput around 70% can still be maintained under heavy traffic conditions. For Adaptive Lights, it can outperform Fixed Lights greatly under light traffic conditions; however, the advantage reduces when the traffic gets heavier.

As for the delay comparison shown in Fig. 5.9, first of all, it can be noticed that vehicles controlled by TP-AIM have much smaller delay in every scenario, and the median values of delay are only 20% ~ 30% of Fixed Lights and 21% ~ 42% of Adaptive Lights. Then for Fixed Lights and Adaptive Lights, when more vehicles intend to get through the intersection, the maximum value and median value of delay increase; while for TP-AIM, the median value of delay saturates to around 18.3s. Under heavy traffic conditions in scenario $S5$, the delay of a vehicle can be as high as 345s if the intersection is controlled by Fixed Lights, and the maximum delay can be reduced to 265s when Adaptive Lights is applied, whereas if our TP-AIM protocol is implemented, the maximum delay can greatly decrease to less than 10% of these two values. When comparing the first two sub-figures in Fig. 5.9, it can be found that Adaptive Lights cannot guarantee to have lower maximum delays than Fixed Lights in light and medium traffic, however, the median values are always lower.

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

| Scenarios | Attempt Times | | | Max CT (s) | Average CT (s) | Average CT(WS) (s) | Average CT(TP) (s) |
|-----------|---------------|-------|-------|------------|----------------|--------------------|--------------------|
| | 1 | 2 | 3 | | | | |
| S1 | 96.64% | 2.52% | 0.84% | 2.21 | 1.26 | 0.98 | 0.28 |
| S2 | 95.81% | 3.38% | 0.81% | 2.56 | 1.70 | 1.38 | 0.32 |
| S3 | 91.70% | 7.30% | 1.00% | 2.59 | 1.68 | 1.33 | 0.34 |
| S4 | 94.20% | 5.50% | 0.30% | 2.45 | 1.92 | 1.45 | 0.47 |
| S5 | 94.50% | 4.72% | 0.78% | 2.60 | 1.88 | 1.43 | 0.45 |

Table 5.2: Attempt times and computation time.

Simulation results show that Adaptive Lights have absolute advantage in evacuation time and throughput compared to Fixed Lights, while our TP-AIM protocol considerably outperforms the other protocols in all the three criteria. For traffic lights, flows having intersection point in their trajectories can not get into the intersection simultaneously, so that vehicles have to be held before the intersection. Meanwhile this limit is broken through in TP-AIM, the intersection is divided into smaller conflict zones, and vehicles can be inside the intersection at the same time if they don't take any conflict zone simultaneously. Thus the intersection space is used much more efficiently. Besides, vehicles don't have to stop only if necessary, therefore travel delay can be reduced.

In terms of computation, as shown in TABLE 5.2, for all the simulation scenarios, more than 90% of vehicles can find a feasible trajectory for the first given entering window, and the others need to use backup solutions in their cache to plan the trajectory again. The results show that a vehicle tries at most three times to find a feasible trajectory. By using the setup mentioned previously, it takes at most 2.6s for a vehicle to find entering windows and to plan a feasible trajectory. On average, the computation time is less than 2s, of which 78% is spent on window searching (WS), and the rest on trajectory planning (TP). In near future, when vehicles have efficient on-board computation ability, TP-AIM can be implemented for real-time usage.

5.5 Conclusion

In this chapter, a trajectory planning based autonomous intersection management protocol (TP-AIM) is developed to control an intersection under connected

autonomous vehicles context in a safer and more efficient way.

The intersection is controlled by a manager which is used to collect and distribute conflict zone occupancy information. When entering the V2I range, vehicle demands the occupancy time of its related conflict zone from the manager; then the manager responds these demands in priority by dividing the vehicles into service groups. After having got the response, a window searching algorithm is used in order to find an entering window which can produce a collision-free trajectory with minimal delay, as well as some other entering windows as backups. Then the vehicle begins to plan its trajectory using a segmented dynamic programming algorithm based on the decided entering window and the future motions of the preceding vehicle. If the current trajectory planning fails, a backup entering window is applied to restart the planning.

Simulation results prove that our V2X-based TP-AIM protocol is capable to deal with different traffic scenarios, and all the design objectives are fulfilled. Furthermore, TP-AIM can bring qualitative change to the current intersection management systems. For future research, multiple intersections can be taken into account to learn the coordination of several managers.

5. TRAJECTORY PLANNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

Conclusions and Perspectives

This chapter is aiming at concluding the thesis. The results of each chapter is summarized here, then some limits and drawbacks in this work are analyzed, and several ideas to improve and extend this research are also proposed.

Conclusions

This thesis is devoted to study the potential applications of autonomous vehicles and V2X communications to improve the safety and efficiency of road transportation systems. This research is a glance of the future in which the transportation could be completely different from nowadays: the vehicles can drive themselves, interact with each other to avoid collisions and to optimize their trajectories to arrive at destinations earlier; passengers can safely enjoy their journey on board.

In this thesis, we have been focused on designing algorithms to realize some possible interactions between vehicles and infrastructures in different scenarios. We hope that our work could be helpful and inspiring for future research and development of intelligent transportation systems.

The work has been presented as follows:

Chapter 1 mainly presents the motivation, history and state-of-art of the development of intelligent transportation systems. Due to the drawbacks like accidents, congestions and pollution, people are trying to make the transportation safer, more efficient and environmentally friendly. Currently, autonomous vehicle is a very popular R&D domain, in which people hope to develop artificial intelligence to replace human in driving, so that accidents due to human error can be reduced. In order to facilitate information exchange between transportation participants, V2X communications are developed. By forming vehicular ad

CONCLUSIONS AND PERSPECTIVES

hoc networks, vehicles can communicate with each other and with infrastructures. As two main interests in our research, cooperative adaptive cruise control (CACC) and autonomous intersection management (AIM) are reviewed in terms of development path and mainstream techniques.

Chapter 2 presents a platoon control algorithm. The vehicular kinetic model and platoon model we use are described; the cost function indicating safe spacing and accordance of velocity and acceleration in the platoon is established; a trail-and-error method is used to integrate different kinds of constraints in the cost function; the topology of communication between vehicles is described which is a fusion of information from the platoon leader and the preceding vehicle; a bionic particle swarm optimization (PSO) algorithm is used to find near-optimal solution for following vehicles; simulation results show that the platoon can be stabilized with reduced tracking errors, the influence from acceleration/deceleration of platoon leader can be attenuated. The corresponding results can be found in (Liu & El Kamel (2016e); Liu *et al.* (2016)).

Chapter 3 has established the CACC-VI algorithm. The algorithm is designed to improve the throughput of a signal controlled intersection with the help of V2X communications and CACC technology. CACC-VI is composed of two main parts: platoon reorganization and platoon control. In the first part, a trajectory planning algorithm is designed to plan a fuel-saving trajectory to accomplish desired motion within constraints; a space arrangement approach is proposed to distribute the opportunity space to vehicles which are capable to find feasible solutions in trajectory planning; the related communication message and vehicles' behaviors are also described. In the second part, the strategies of vehicles in different cases are designed: the platoons leaders follow the preplanned trajectories; the following vehicles are controlled by the PSO-based platoon control algorithm; the former platoons leaders applies both the two strategies with a switch condition between them. Simulation experiments show that our algorithm is capable to reorganize the platoons safely according to the opportunity space and vehicles' dynamical abilities; the throughput of intersection can be improved and the reorganized platoons can be stabilized. The corresponding results have been published in (Liu & El Kamel (2016e)).

Chapter 4 has focused on autonomous intersection control in light traffic conditions. The DAIC algorithm is developed to let vehicles get through the

CONCLUSIONS AND PERSPECTIVES

intersection without full stop, so that the average delay of vehicles can be reduced. The conception of conflict zone occupancy time (CZOT) is defined to quantify the space-time conflict inside the intersection. In order to reduce the dimension of solution space, the vehicles make their decisions in a prioritized way. A cost function representing the crossing collision and the rear-end collision is established. Vehicles must choose the input from its action set to minimize the cost value, and they should keep adjusting their velocities until arriving at a collision-free value. A simulation testbed is designed using several functional diagrams in UML and the testbed is established using Matlab and SUMO, an open source traffic simulator. Simulation results prove that DAIC algorithm overtakes greatly traditional traffic lights in evacuation time, delay and throughput under light traffic conditions. The corresponding results can be found in (Liu & El Kamel (2016b,c)).

Chapter 5 has expounded the TP-AIM algorithm. This algorithm is developed to guarantee safety while improving efficiency of an intersection under different traffic densities. The CZOT information is stored at the intersection manager. When entering the V2I range, the vehicle demands the related CZOT. If multiple demands received at the same time, the manager responses by following an improved service priority arrangement method. After getting the response, the vehicle uses the window searching algorithm to find possible entering moments and velocities. In order to reduce the computation complexity, the original dynamic programming process is divided into several smaller processes linked by intermediate goals. The vehicle's trajectory is planned by this segmented dynamic programming method based on the decided entering window and the future motion of the preceding vehicle. A backup mechanism is developed to improve the robustness of the algorithm. Simulation results show that TP-AIM can deal with even heavy traffic scenarios. Besides, it can greatly reduce vehicles' delays and improve the efficiency of intersection, compared to time-fixed traffic lights and adaptive traffic lights. The corresponding contributions can be found in (Liu & El Kamel (2016a,d)).

Perspectives

When developing the algorithms in this thesis, several assumptions have been made. Therefore, when applying our method in realistic environment, there are still some aspects to work on. Based on the results of this thesis, some perspective remarks must be considered:

- In the TP-AIM, the trajectories are planned for individual vehicles. Maybe the computation cost can be reduced if we plan the trajectories for vehicular platoons. The idea of platoon reorganization in CACC-VI could be combined with window searching algorithm in TP-AIM. Instead of simply searching for the existence of entering window, the windows' widths, indicating the number of vehicles which can profit the window, could also be quantified. Then the platoons can be reorganized according to the widths and vehicles' dynamical characteristics. Thus the following vehicle does not have to utilize trajectory planning algorithms but the platoon control algorithm, which is less computationally expensive.
- When developing the algorithms, we have assumed that all the lane-changing maneuvers have been accomplished before entering the traffic manager's range. So we have only considered the longitudinal interaction between vehicles. However, it is not the case in reality: the headings of vehicles in the platoon may not be the same, so a vehicle may need to depart from the current platoon and joins a platoon on the lane nearby. Therefore, the platoon reorganization should not only be longitudinal but also be lateral. For example, in the platoon reorganization we designed in Chapter 3, vehicle V_i have to abandon joining in the accelerating platoon because its preceding vehicle V_j is not capable to find a feasible trajectory; then V_j could change to another lane nearby if possible to give way to V_i , so that the throughput can be further improved.
- Currently, our algorithms are all developed for an isolated intersection. Their performance when applied on multiple adjacent intersections has not been investigated. A coordination mechanism among the intersection managers might need to be developed to better use the road space between two intersections.

CONCLUSIONS AND PERSPECTIVES

- The communication in our research is assumed to be perfect with negligible transmission delay, packet loss, interference, etc. In future work, the impact of communication quality should be studied, and ways to improve the robustness when facing delay and data loss should be found. The achievements of Dr. Tian ZHENG may help in the research in this direction ([Zheng et al. \(2013\)](#)).
- Moreover, the penetration rate of autonomous vehicles and V2X communication devices are assumed to be 100%; but we can not achieve this value in one day, thus the impact of penetration rate on vehicular networks should be investigated.
- In our research, the vehicles are modeled in a simple way. However, in reality, a vehicle is a quite complicated system, so more complete vehicle model can be used to better represent the vehicle's behaviors. Besides, the location system is assumed to be perfectly accurate. In future work, the errors in data acquisition and transmission could be taken into account.
- The impacts of natural environment and road conditions on vehicles and how this influences the system performance should be studied. As there are already some results on this aspect in our research group ([Luo et al. \(2012\)](#); [Yu et al. \(2013, 2014\)](#)), it will be better to integrate all the results on one single platform.
- The surrounding vehicles in the same street and the vehicles coming from other streets can be seen as moving obstacles. Navigation algorithm can be designed to plan the vehicle's trajectory in real time to avoid collisions with these obstacles and to achieve social benefit like minimal delay and minimal energy consumption. Machine learning method can be used for trajectory planning and historical actions can be recorded to serve as training examples to ameliorate future motions. When facing unexpected scenarios, the system could also be capable to make right decisions. Dr. Chen XIA's work ([Chen & EL KAMEL \(2015\)](#); [Xia & El Kamel \(2016\)](#)) could be inspiring in this research direction.

CONCLUSIONS AND PERSPECTIVES

- It could also be interesting to build physical testbed to evaluate the algorithms in small scales, so that the work flow of our algorithms can be tested and potential problems in real time implementation could be discovered.

Résumé Étendu en Français

Introduction

Cette thèse est consacrée à étudier des applications potentielles des véhicules autonomes et des communications V2X pour améliorer la sécurité et l'efficacité des systèmes de transport routier. Cette recherche est un coup d'œil de l'avenir dans lequel le transport pourrait être complètement différent de nos jours: les véhicules peuvent conduire eux-mêmes, d'interagir les uns avec les autres pour éviter les collisions et d'optimiser leurs trajectoires pour arriver aux destinations plus tôt; les passagers peuvent profiter en toute sécurité leur voyage à bord.

Dans cette thèse, nous sommes attachés sur la conception des algorithmes pour réaliser les interactions entre les véhicules et les infrastructures dans les différents situations. Nous espérons que notre travail pourrait être utile et inspirant pour la recherche et le développement futur des systèmes de transport intelligents.

Motivation de la recherche

L'économie mondiale est fortement dépendante de la sécurité et l'efficacité des systèmes de transport pour déplacer des passagers et des marchandises entre les origines et les destinations multiples. En arrivant à un endroit en toute sécurité dans une certaine période est l'exigence fondamentale de transport. Cependant, le transport routier d'aujourd'hui est loin d'être parfait.

Comportements de conduite incorrects comme l'alcool au volant, la fatigue au volant et la vitesse sont considérés comme les causes principales des accidents de la route qui, d'une part causent des blessures, la mort et des dommages matériels, d'autre part rendent les véhicules garder plus grande distance entre eux, donc la capacité de la route n'est pas utilisé complètement. En outre, les congestions

RÉSUMÉ ÉTENDU EN FRANÇAIS

causés par des comportements de conduite incorrectes, les accidents, la notifications de trafic inconvenantes sont devenues un phénomène mondial qui cause des effets négatifs au niveau de l'économique et de l'écologique, de sorte que les gens doivent passer plus de temps sur la route et plus de carburant est consommé, ce qui conduit à plus de pollution. Face à tous ces inconvénients des systèmes de transport routier actuels, améliorations au niveau du système et de l'individuel doivent être étudiées.

Dans ce contexte, la concept "Systèmes de transport intelligents (STI)" est proposée. Les STI peuvent être largement définis comme l'application des technologies de pointe, telles que les communications, les capteurs, et l'informatique, aux systèmes de transport, afin de fournir des informations en temps réel pour améliorer la sécurité, l'efficacité et le confort, en même temps de réduire l'impact sur l'environnement.

Systemes Autonomes dans STI

L'idée principale des systèmes autonomes est d'utiliser les applications autonomes pour aider/remplacer les manœuvres et les décisions humaines. Les systèmes avancés d'assistance au conducteur (ADAS) sont conçus pour aider les conducteurs en les alertant lorsque le danger (par exemple, la tenue de voie, alerte de collision avant), en acquérant plus d'information pour la prise de décision (par exemple, planification de l'itinéraire et évitement d'encombrement) et en les libérant de manoeuvres répétitives et astuce (par exemple, régulateur de vitesse adaptatif, stationnement automatique). Dans les systèmes semi-automatiques, le processus de conduite a encore besoin de l'intervention du conducteur humain : le conducteur doit régler certains paramètres dans le système, et il/elle peut décider de suivre l'assistance consultative ou non.

Récemment, avec l'amélioration de la technologie de détection et de l'intelligence artificielle, les entreprises et les instituts ont été commis à la recherche et le développement de la conduite autonome (Figure 1). Dans certains scénarios (par exemple les autoroutes et les routes principales), avec l'aide de capteurs précis et la carte de haute précision, la conduite sans intervention humaine seraient atteinte. L'élimination de l'erreur humaine rendra le transport routier beaucoup

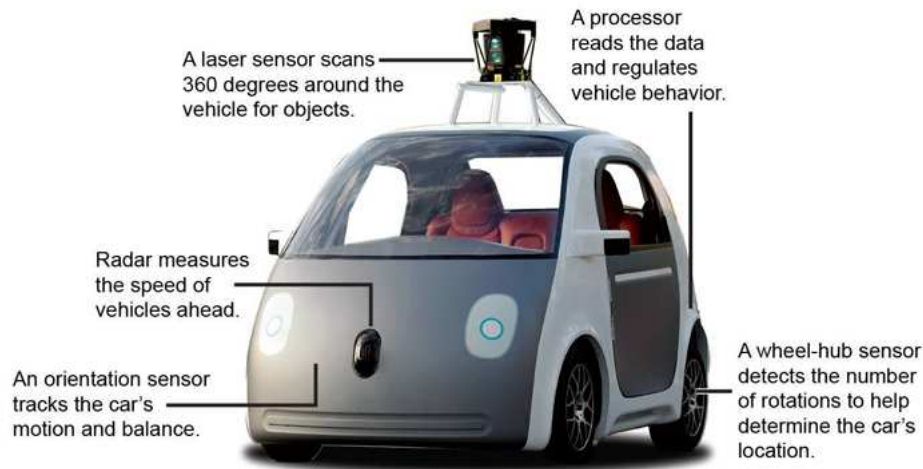


Figure 1: La voiture sans conducteur "Bubble" de Google

plus sûr, et une meilleure utilisation de l'espace permettra de mieux profiter la capacité routière. Cependant, les voitures autonomes doivent encore l'anticipation du conducteur dans certains scénarios avec la situation du trafic compliqué ou avec les informations limitées. La disposition intérieure des voitures autonomes ne serait pas très différent des voitures actuelles, parce que le volant et les pédales sont encore nécessaires. La prochaine étape de la conduite autonome est la conduite sans conducteur, dans laquelle la voiture est totalement dirigée par elle-même. Le siège dédié pour le conducteur disparaîtrait et les personnes à bord se concentreraient sur leur propre travail. L'économie auto-partage basée sur les voitures sans conducteur serait énorme: à l'avenir, les gens préfèrent appeler une voiture sans conducteur lorsqu'il est nécessaire à posséder une voiture privée. Ainsi les congestions et les pollutions pourraient être soulagées.

Systemes Coopératifs dans STI

Il est évident que les notifications de transport routier actuels sont conçus pour les conducteurs humains, tels que les feux de circulation, les clignotants et les panneaux le long de la route. Les véhicules autonomes actuels sont équipés de caméras dédiées à détecter ces signes. Cependant, les notifications conçus pour les humains ne sont pas suffisamment efficaces pour les véhicules autonomes,

RÉSUMÉ ÉTENDU EN FRANÇAIS

parce que l'utilisation des caméras est limitée par la distance et de la visibilité, et les algorithmes doivent être mis en œuvre pour reconnaître ces signes. Si l'interaction entre les véhicules et l'environnement est activée, les notifications peuvent être livrées via les communications V2X (Figure 2), donc les véhicules peuvent être remarqué dans une plus grande distance même en dehors de la vue, et l'information originale est plus précise que les informations détectées par des capteurs. Lorsque le taux de pénétration des voitures sans conducteur est suffisamment élevé, il ne serait pas nécessaire d'avoir des feux et des panneaux de signalisation physiques. Le signe personnel virtuel de la circulation peut être communiqué aux véhicules individuels par le gestionnaire de trafic. Dans les systèmes coopératifs, un individu n'a pas besoin d'acquérir l'information tout par ses propres capteurs, mais avec l'aide des autres via la communication. Par conséquent, l'intelligence autonome peut être étendue en intelligence coopérative.

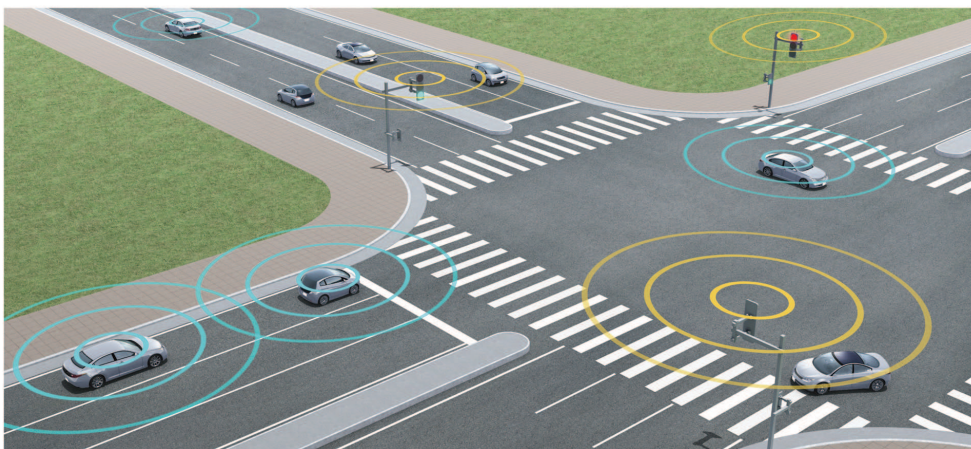


Figure 2: Réseaux ad hoc des véhicules au voisinage d'une intersection

La recherche présentée dans cette thèse se concentre sur le développement d'applications pour améliorer la sécurité et l'efficacité des systèmes de transport intelligents dans le contexte des véhicules autonomes et des communications V2X. Ainsi, cette recherche est dans le domaine des systèmes coopératifs. Stratégie de contrôle sont conçus pour définir la façon dont les véhicules interagissent les uns avec les autres et avec les infrastructures.

Contrôle Décentralisé de Caravane

Le système du régulateur de vitesse adaptatif (ACC), qui est disponible dans le marché, vise à maintenir automatiquement une vitesse constante ou une distance de sécurité au véhicule précédent. Actuellement, le système ACC est principalement basé sur des capteurs de télémètre, par exemple, les radars et les lasers, qui sont capables de mesurer la distance et son taux de variation. En règle générale, le système ACC peut dégager les conducteurs de réaliser des tâches répétitives et ennuyeuses comme réglage de la vitesse et la distance au véhicule précédent sur l'autoroute. Cependant, les systèmes d'ACC actuels ont encore des inconvénients, comme l'incapacité à des scénarios plus complexes et les capacités limitées d'anticipation à manœuvre brusque du véhicule précédent.

Grâce à la technologie V2X, ACC peut être étendu à ACC coopérative (CACC). Véhicule équipé de V2X peut non seulement obtenir des informations du véhicule précédent comme dans le système ACC, mais aussi les véhicules avant la précédente, même dehors la vue. CACC est considéré hautement potentiel d'améliorer la capacité d'écoulement et la douceur du trafic et de réduire les congestions.

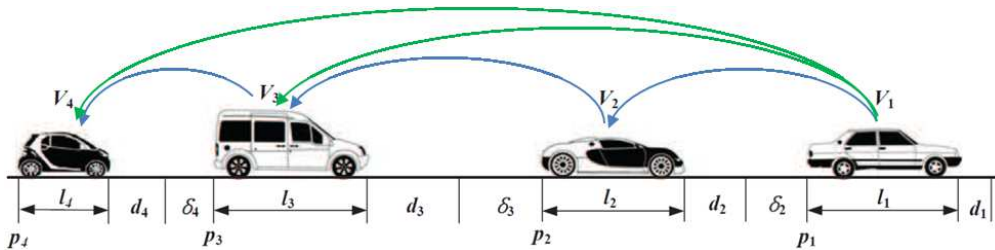


Figure 3: La topologie de communication dans la caravane

Dans cette thèse, un algorithme de contrôle de caravane basé sur l'optimisation par essaim de particules (PSO) est décrit. A chaque pas de temps, un véhicule cherche les l'entrée de commande optimale pour réduire les trois erreurs de suivi dans la caravane, selon les informations provenant du leader de caravane et le véhicule précédent le plus proche via les communications V2V. Les capacités dynamiques des véhicules et le confort des passagers sont également pris en compte dans cet algorithme.

Les résultats de simulation montrent que notre algorithme est efficace pour contrôler les véhicules dans la caravane. La caravane peut être stabilisée avec des

erreurs de suivi réduites et l'influence de l'accélération/décélération du premier véhicule au reste de la caravane peut être éliminée.

Amélioration de l'efficacité d'intersection via V2X et CACC

Ensuite, nous avons designé un algorithme : régulateur de vitesse adaptatif coopératif au voisinage d'intersection (CACC-VI) , qui vise à augmenter le débit d'une intersection sous contrôle des feux de circulation (Figure 4). Il est supposé que les caravanes de véhicules sont déjà formées avant d'arriver à l'intersection. Un gestionnaire est installé à l'intersection qui peut communiquer avec les caravanes via V2I.

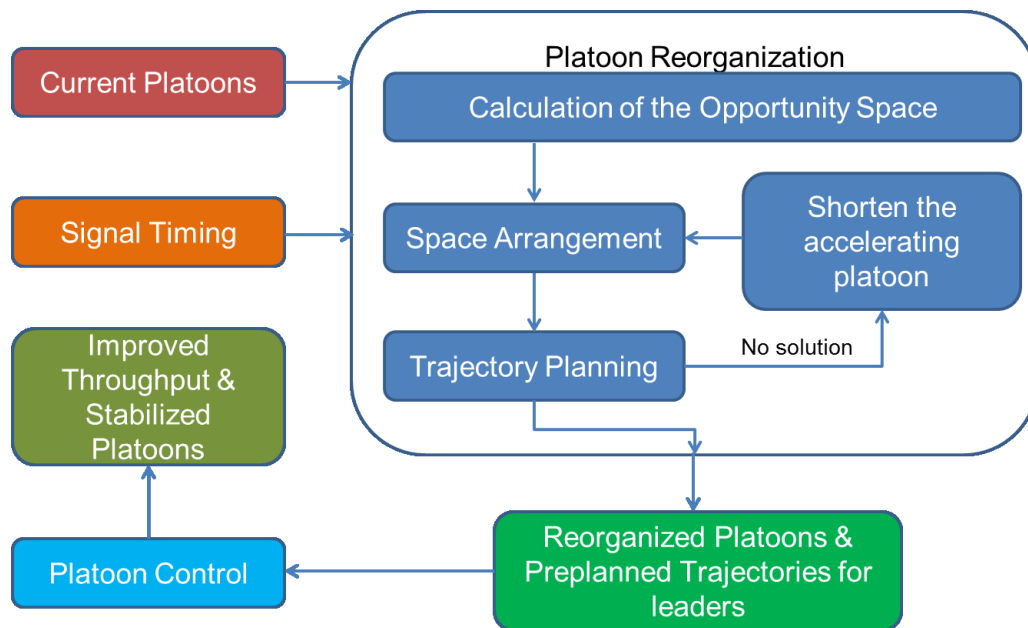


Figure 4: L'aperçu de l'algorithme CACC-VI

Dans un premier temps, le gestionnaire d'intersection diffuse le réglage du temps du feu aux véhicules dans la voie, tandis que les caravanes envoient ses positions et ses vitesses actuelles au gestionnaire; alors l'espace d'opportunité est calculé et distribué aux véhicules en utilisant l'approche d'aménagement de l'espace et de l'algorithme de planification de trajectoire; s'il y a un véhicule

dans la caravane accélérante qui ne peut pas trouver un profil d'entrée, le dernier véhicule est réorganisé dans la caravane décélérante; alors le processus est lancé à nouveau jusqu'à tous les véhicules dans la caravane accélérante peut trouver une solution. En ce moment-là, le processus de réorganisation du peloton est terminé. Les caravanes nouvellement réorganisées sont contrôlées par des stratégies différentes: les leaders de caravane sont contrôlés en suivant les trajectoires préétablies; les leaders précédents, qui sont fusionnés dans une nouvelle caravane, sont contrôlés par une stratégie de contrôle mixte. Les autres véhicules sont contrôlés par l'algorithme de contrôle de caravane basé sur PSO.

En utilisant l'algorithme CACC-VI, le débit d'intersection peut être amélioré en profitant la capacité redondantes de la route; et le mouvement des caravanes est contrôlé avec des comportements stabilisés en considération de la consommation de carburant, les caractéristiques hétérogènes des véhicules et le confort des passagers.

Planification de Trajectoire pour la Gestion Autonome d'Intersection

Dans le contexte des véhicules autonomes connectés, comme les notifications des feux peuvent être communiquées via V2I communications, s'il n'y a pas de passages piétons qui traversent la rue, les feux physiques pourraient disparaître. Donc les intersections sans feu ni panneau pourraient être réalisées, et les véhicules au voisinage d'intersection seraient conduits dans une façon coopérative.

Dans la conduite coopérative, le gestionnaire d'intersection reçoit les demandes des véhicules pour traverser l'intersection et donne à chaque véhicule les conseils d'un itinéraire optimal en tenant compte de leur capacité d'action dynamique pour assurer qu'aucun accident ne se produit, en même temps, minimiser le retard d'intersection et de la consommation de carburant. Une caravane devient flexible, un véhicule peut rejoindre ou quitter une caravane déjà formée; et plusieurs véhicules peuvent également choisir d'en former une nouvelle.

Un protocole de gestion autonome d'intersection fondé sur la planification de trajectoire (TP-AIM) est développé afin de garantir la sécurité tout et d'améliorer l'efficacité d'une intersection dans le contexte des véhicules autonomes connectés. Une vue d'ensemble du protocole TP-AIM est donnée dans la Figure 5.

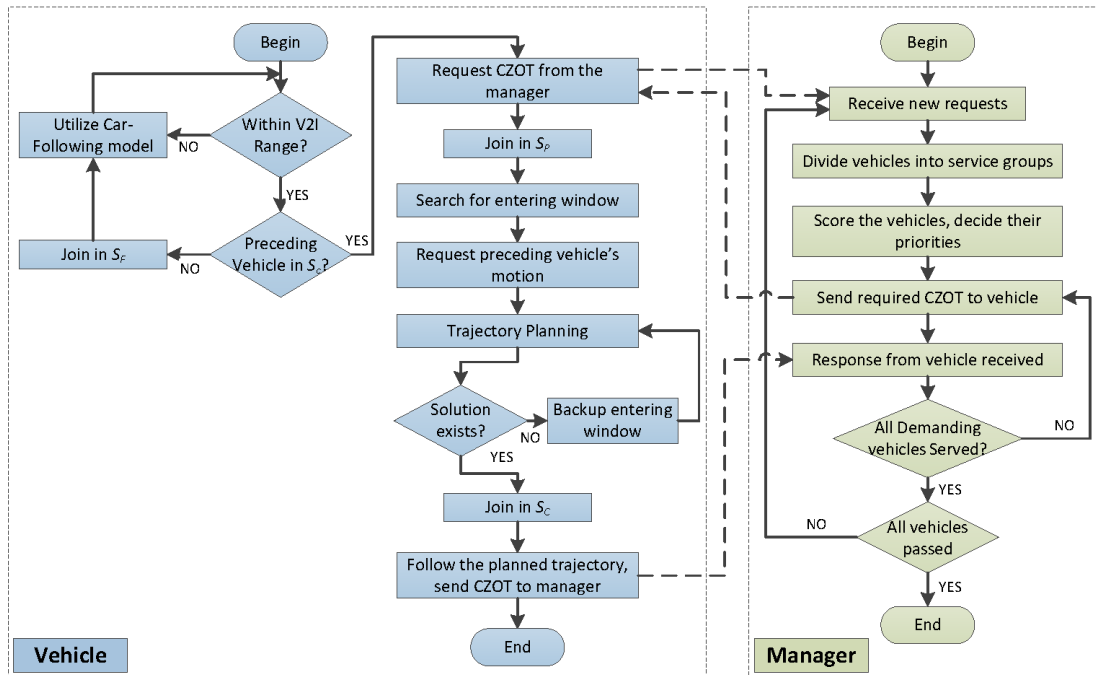


Figure 5: Le diagramme de TP-AIM

En s'approchant une intersection, avant d'entrer dans le rayon de V2I, un véhicule circule selon son algorithme auto-conduite mis en œuvre; après être entré à l'intérieur du rayon, le véhicule demande au gestionnaire d'intersection le temps d'occupation de la zone de conflit (CZOT). Le gestionnaire traite ces demandes dans un ordre décidé: les véhicules qui ont proposé la demande sont divisés en groupes de services, dont la priorité est donnée par les types de véhicules, les directions et les distances à l'intersection. Un véhicule ne peut obtenir l'information CZOT après ses véhicules associés dans le groupe précédent, qui partage des zones de conflit avec lui, ont terminé le processus de planification de la trajectoire.

Etant donné le CZOT associé, le véhicule doit utiliser l'algorithme de fenêtre recherche pour trouver un moment d'entrée et une vitesse optimale qui conduit à une trajectoire sans collision à l'intérieur de l'intersection avec le retard minimal, ainsi que des solutions de sauvegarde qui sont également sans collision, mais possèdent de plus grands retards. Il est supposé que les véhicules circulent à des vitesses constantes dans l'intersection.

Ensuite, le véhicule doit planifier sa propre trajectoire en fonction de sa position actuelle, de la vitesse et de sa capacité d'accélération / décélération pour

entrer dans l'intersection, au moment d'entrée décidé à la vitesse d'entrée correcte. D'une part, si le véhicule dispose actuellement d'un véhicule précédent sur la même voie, le premier devrait demander à cette dernière pour les motions futures qui ont été prévues; puis un algorithme de programmation dynamique segmenté est utilisé pour planifier des actions afin d'atteindre les objectifs intermédiaires et pour éviter d'heurter le véhicule précédent. D'un autre part, si le véhicule est le premier sur sa voie, il faut utiliser un modèle d'accélération à trois sections pour planifier sa trajectoire. Si la trajectoire planification réussite, le véhicule doit envoyer un message au gestionnaire et le CZOT qu'il veut réserver est formellement déterminé. Si la planification de trajectoire n'a pas pu trouver les entrées possibles dans les limites dynamiques du véhicule, en d'autres termes, le véhicule ne peut pas profiter de la fenêtre d'entrer actuelle, il devrait donc entrer plus tard. Ainsi, le CZOT qu'il prévoit doit être annulé et une solution de sauvegarde de fenêtre recherche est utilisée pour relancer le processus de planification de trajectoire.

Conclusions et Perspectives

Cette thèse est consacrée à étudier les applications potentielles de véhicules autonomes et communications V2X pour construire les systèmes de transport intelligents. Tous les algorithmes conçus sont testés et vérifiés avec succès par des simulations dans scénarios différents.

Contributions

- Premièrement, le comportement de caravane dans un environnement de véhicule connecté est étudié. Un algorithme de commande de caravane est conçu pour obtenir l'espacement sécuritaire ainsi que la conformité de la vitesse et de l'accélération.
- Deuxièmement, à plus grande échelle, les caravanes autour d'une intersection sont considérées. Le débit pendant une période de signal de trafic peut être améliorée en profitant la capacité redondante de la route. Dans diverses contraintes, les véhicules peuvent choisir d'accélérer à se joindre à la caravane précédente ou à décélérer de déroger à l'actuel.

RÉSUMÉ ÉTENDU EN FRANÇAIS

- Troisièmement, une intersection sans signalisation en VANET est considérée. Dans des conditions de faible trafic, les véhicules peuvent réguler leur vitesse avant d'arriver à l'intersection en fonction du temps d'occupation de la zone de conflit (TOZC) stocké au niveau du gestionnaire, afin qu'ils puissent traverser l'intersection sans collision ni arrêt. Le délai peut être réduit en conséquence.
- Enfin, un algorithme de gestion d'intersection autonome universelle, qui peut fonctionner même avec le trafic lourd, est développé. Le véhicule cherche des fenêtres sécuritaires d'entrée dans le TOZC. Ensuite, sur la base des fenêtres trouvées et le mouvement du véhicule qui précède, les trajectoires des véhicules peuvent être planifiées en utilisant une méthode de programmation dynamique segmentée.

Perspectives

- Lors de l'élaboration des algorithmes, nous avons supposé que toutes les manoeuvres de changement de voie ont été accomplies avant d'entrer dans le rayon du gestionnaire de trafic. Donc, nous avons seulement considéré comme l'interaction longitudinale entre les véhicules. Meilleurs résultats pourraient être obtenus, si nous activions les manoeuvres latérales.
- Actuellement, nos algorithmes sont tous développés pour une intersection isolée. Leur performance lorsqu'ils sont appliqués sur plusieurs intersections adjacentes n'a pas été étudiée. Un mécanisme de coordination entre les gestionnaires d'intersection pourrait avoir besoin d'être développé pour mieux utiliser l'espace routier entre deux intersections.
- En outre, le taux de pénétration des véhicules autonomes et des dispositifs de communication V2X sont supposés d'être 100%; mais nous ne pouvons pas atteindre cette valeur en un jour, donc il est nécessaire d'étudier l'impact du taux de pénétration sur les réseaux de véhicules.

Bibliography

- AHMANE, M., ABBAS-TURKI, A., PERRONNET, F., WU, J., EL MOUDNI, A., BUISSON, J. & ZEO, R. (2013). Modeling and controlling an isolated urban intersection based on cooperative vehicles. *Transportation Research Part C: Emerging Technologies*, **28**, 44–62. [30](#)
- ALONSO, J., MILANÉS, V., PÉREZ, J., ONIEVA, E., GONZÁLEZ, C. & DE PEDRO, T. (2011). Autonomous vehicle control systems for safe crossroads. *Transportation research part C: emerging technologies*, **19**, 1095–1110. [30](#)
- ARTIMY, M.M., ROBERTSON, W. & PHILLIPS, W.J. (2004). Connectivity in inter-vehicle ad hoc networks. In *Electrical and Computer Engineering, 2004. Canadian Conference on*, vol. 1, 293–298, IEEE. [17](#)
- ARTIMY, M.M., PHILLIPS, W.J. & ROBERTSON, W. (2005). Connectivity with static transmission range in vehicular ad hoc networks. In *Communication Networks and Services Research Conference, 2005. Proceedings of the 3rd Annual*, 237–242, IEEE. [17](#)
- ASADI, B. & VAHIDI, A. (2011). Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time. *IEEE Transactions on Control Systems Technology*, **19**, 707–714. [50](#)
- BABER, J., KOLODKO, J., NOËL, T., PARENT, M. & VLACIC, L. (2005). Co-operative autonomous driving: intelligent vehicles sharing city roads. *Robotics & Automation Magazine, IEEE*, **12**, 44–49. [30](#)
- BASKAR, L.D., DE SCHUTTER, B., HELLENDORRN, J. & PAPP, Z. (2011). Traffic control and intelligent vehicle highway systems: a survey. *Intelligent Transport Systems, IET*, **5**, 38–52. [12](#)

BIBLIOGRAPHY

- BEKRIS, K.E., TSIANOS, K.I. & KAVRAKI, L.E. (2007). A decentralized planner that guarantees the safety of communicating vehicles with complex dynamics that replan online. In *IROS 2007. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007.*, 3784–3790, IEEE. [31](#)
- BENNEWITZ, M., BURGARD, W. & THRUN, S. (2001). Optimizing schedules for prioritized path planning of multi-robot systems. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation, 2001.*, vol. 1, 271–276, IEEE. [31](#)
- BETSOLD, R. (1989). Intelligent vehicle/highway systems for the united states-an emerging national program. In *Proceedings of JSK International Symposium- Technological Innovations for Tomorrow's Automobile Traffic and Driving Information Systems*, 53–59. [11](#)
- BLUM, J.J., ESKANDARIAN, A. & HOFFMAN, L.J. (2004). Challenges of intervehicle ad hoc networks. *Intelligent Transportation Systems, IEEE Transactions on*, **5**, 347–351. [17](#)
- BOILLOT, F. (1992). Optimal signal control of urban traffic networks. In *International Conference on Road Traffic Monitoring and Control (6th: 1992: London, England). 6th International Conference on Road Traffic Monitoring and Control*. [29](#)
- BONNET, C. (2003). Chauffeur 2 final report. *Deliverable D24, Version*, **1**. [27](#)
- BROGGI, A., MEDICI, P., ZANI, P., COATI, A. & PANCIOLO, M. (2012). Autonomous vehicles control in the vislab intercontinental autonomous challenge. *Annual Reviews in Control*, **36**, 161–171. [13](#)
- BROQUA, F. (1991). Cooperative driving: basic concepts and a first assessment of "intelligent cruise control" strategies. In *DRIVE Conference (1991: Brussels, Belgium). Advanced telematics in road transport. Vol. II*. [24](#)
- BUCHENSCHNEIT, A., SCHAUB, F., KARGL, F. & WEBER, M. (2009). A vanet-based emergency vehicle warning system. In *Vehicular Networking Conference (VNC), 2009 IEEE*, 1–8, IEEE. [21](#)

- BUCKLEY, T.F., JESTY, P., HOBLEY, K. & WEST, M. (1990). Drive-ing standards: a safety critical matter. In *Proceedings of the Fifth Annual Conference on Computer Assurance, Systems Integrity, Software Safety and Process Security*, 164–172, Gaithersburg, USA. 10
- CALVERT, S., VAN DEN BROEK, T. & VAN NOORT, M. (2011). Modelling cooperative driving in congestion shockwaves on a freeway network. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 614–619, IEEE. 35
- CAVENEY, D. (2010). Cooperative vehicular safety applications. *Control Systems, IEEE*, 30, 38–53. 19
- CHANDLER, M. (2016). Google, baidu, tesla gunning self-driving car development. www.investors.com/news/technology. 15
- CHEN, X. & EL KAMEL, A. (2015). A reinforcement learning method of obstacle avoidance for industrial mobile vehicles in unknown environments using neural network. In *Proceedings of the 21st International Conference on Industrial Engineering and Engineering Management 2014*, 671–675, Springer. 131
- DAHMANI, H., CHADLI, M., RABHI, A. & EL HAJJAJI, A. (2013). Road curvature estimation for vehicle lane departure detection using a robust takagi-sugeno fuzzy observer. *Vehicle System Dynamics*, 51, 581–599. 23
- DAVIES, A. (2014). Audi’s self-driving car hits 150 MPH on an F1 track. www.wired.com/2014/10/audis-self-driving-car-hits-150-mph-f1-track/. 15
- DAVIES, A. (2015). Baidu’s self-driving car has hit the road. www.wired.com. 15
- DE BRUIN, D., KROON, J., VAN KLAVEREN, R. & NELISSE, M. (2004). Design and test of a cooperative adaptive cruise control system. In *Intelligent Vehicles Symposium, 2004 IEEE*, 392–396, IEEE. 24, 35
- DE NUNZIO, G., WIT, C.C., MOULIN, P. & DI DOMENICO, D. (2015). Eco-driving in urban traffic networks using traffic signals information. *International Journal of Robust and Nonlinear Control*. 19

BIBLIOGRAPHY

- DESJARDINS, C. & CHAIB-DRAA, B. (2011). Cooperative adaptive cruise control: A reinforcement learning approach. *IEEE Transactions on Intelligent Transportation Systems*, **12**, 1248–1260. [34](#)
- DORNBUSH, S. & JOSHI, A. (2007). Streetsmart traffic: Discovering and disseminating automobile congestion using vanets. In *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, 11–15, IEEE. [20](#)
- DRESNER, K. & STONE, P. (2008). A multiagent approach to autonomous intersection management. *Journal of artificial intelligence research*, 591–656. [20](#), [30](#), [79](#)
- EBERHART, R.C., KENNEDY, J. *et al.* (1995). A new optimizer using particle swarm theory. In *Proceedings of the sixth international symposium on micro machine and human science*, vol. 1, 39–43, New York, NY. [43](#)
- FANCHER, P. (1998). Intelligent cruise control field operational test. Tech. rep., University of Michigan Transportation Research Institute. [24](#)
- FIGUEIREDO, L., JESUS, I., MACHADO, J.T., FERREIRA, J. & DE CARVALHO, J.M. (2001). Towards the development of intelligent transportation systems. In *Intelligent Transportation Systems*, vol. 88, 1206–1211. [9](#), [11](#), [24](#)
- FRESE, C. & BEYERER, J. (2010). Planning cooperative motions of cognitive automobiles using tree search algorithms. In *KI 2010: Advances in Artificial Intelligence*, 91–98, Springer. [102](#)
- FRESE, C. & BEYERER, J. (2011a). A comparison of motion planning algorithms for cooperative collision avoidance of multiple cognitive automobiles. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, 1156–1162, IEEE. [31](#)
- FRESE, C. & BEYERER, J. (2011b). A comparison of motion planning algorithms for cooperative collision avoidance of multiple cognitive automobiles. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, 1156–1162, IEEE. [102](#)
- GAUR, A. & MIRCHANDANI, P. (2001). Method for real-time recognition of vehicle platoons. *Transportation Research Record: Journal of the Transportation Research Board*, 8–17. [29](#)

- GEIGER, A., LAUER, M., MOOSMANN, F., RANFT, B., RAPP, H., STILLER, C. & ZIEGLER, J. (2012). Team annieway's entry to the 2011 grand cooperative driving challenge. *IEEE Transactions on Intelligent Transportation Systems*, **13**, 1008–1017. [28](#), [35](#)
- GHAFFARIAN, H., FATHY, M. & SORYANI, M. (2012). Vehicular ad hoc networks enabled traffic controller for removing traffic lights in isolated intersections based on integer linear programming. *Intelligent Transport Systems, IET*, **6**, 115–123. [30](#)
- GUIZZO, E. (2011). How google's self-driving car works. *IEEE Spectrum Online, October*, **18**. [13](#)
- GUVENC, L., UYGAN, I.M.C., KAHRAMAN, K., KARAAHMETOGLU, R., ALTAY, I., SENTURK, M., EMIRLER, M.T., KARCI, A.E.H., GUVENC, B.A., ALTUG, E. *et al.* (2012). Cooperative adaptive cruise control implementation of team mekar at the grand cooperative driving challenge. *IEEE Transactions on Intelligent Transportation Systems*, **13**, 1062–1074. [28](#), [35](#)
- HAN, D. & YI, K. (2006). A driver-adaptive range policy for adaptive cruise control. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, **220**, 321–334. [25](#)
- HAN, S.Y., CHEN, Y.H., WANG, L. & ABRAHAM, A. (2013). Decentralized longitudinal tracking control for cooperative adaptive cruise control systems in a platoon. In *2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE. [35](#), [36](#), [38](#)
- HARTENSTEIN, H. & LABERTEAUX, K. (2010). *VANET: vehicular applications and inter-networking technologies*, vol. 1. Wiley Online Library. [16](#), [19](#), [34](#), [50](#), [78](#)
- HASSAN, R., COHANIM, B., DE WECK, O. & VENTER, G. (2005). A comparison of particle swarm optimization and the genetic algorithm. In *Proceedings of the 1st AIAA multidisciplinary design optimization specialist conference*, 18–21. [43](#)

BIBLIOGRAPHY

- HE, Q., HEAD, K.L. & DING, J. (2012). Pamscod: Platoon-based arterial multi-modal signal control with online data. *Transportation Research Part C: Emerging Technologies*, **20**, 164–184. [29](#), [51](#), [78](#), [79](#)
- HEDRICK, J., TOMIZUKA, M. & VARAIYA, P. (1994). Control issues in automated highway systems. *Control Systems, IEEE*, **14**, 21–32. [29](#)
- HENRY, J.J., FARGES, J.L. & TUFFAL, J. (1984). The PRODYN real time traffic algorithm. In *IFACIFIPIFORS Conference on Control*. [29](#)
- HILGERT, J., HIRSCH, K., BERTRAM, T. & HILLER, M. (2003). Emergency path planning for autonomous vehicles using elastic band theory. In *2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2003. AIM 2003. Proceedings.*, vol. 2, 1390–1395, IEEE. [102](#)
- HUNT, P., ROBERTSON, D., BRETHERTON, R. & ROYLE, M.C. (1982). The scoot on-line traffic signal optimisation technique. *Traffic Engineering & Control*, **23**. [29](#), [78](#)
- IBRAHIM, K. & WEIGLE, M.C. (2008). Optimizing cascade data aggregation for vanets. In *Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008. 5th IEEE International Conference on*, 724–729, IEEE. [20](#)
- INTEL (2015). Building an intelligent transportation system with the the internet of things (iot). <http://www.intel.cn/content/www/cn/zh/internet-of-things>. [9](#)
- ISO 15628:2013 (2013). Intelligent transport systems – dedicated short range communication (DSRC) – dsrc application layer. http://www.iso.org/iso/home/store/catalogue_ics/. [18](#), [24](#)
- JANET (1992). *Strategic Plan for IVHS in the United States*. IVHS, AMERICA. [11](#)
- JIANG, D. & DELGROSSI, L. (2008). Ieee 802.11 p: Towards an international standard for wireless access in vehicular environments. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, 2036–2040, IEEE. [vii](#), [18](#)

- JIANG, Y., LI, S. & SHAMO, D.E. (2006). A platoon-based traffic signal timing algorithm for major–minor intersection types. *Transportation Research Part B: Methodological*, **40**, 543–562. [29](#), [51](#), [78](#)
- KAMALANATHSHARMA, R.K. & RAKHA, H. (2012). Agent-based modeling of eco-cooperative adaptive cruise control systems in the vicinity of intersections. In *2012 15th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 840–845, IEEE. [50](#)
- KENNEDY, J. (2010). Particle swarm optimization. In *Encyclopedia of Machine Learning*, 760–766, New York, NY, USA: Springer-Verlag. [43](#), [113](#)
- KENNEY, J.B. (2011). Dedicated short-range communications (dsrc) standards in the united states. *Proceedings of the IEEE*, **99**, 1162–1182. [18](#)
- KIANFAR, R., AUGUSTO, B., EBADIGHAJARI, A., HAKEEM, U., NILSSON, J., RAZA, A., TABAR, R.S., IRUKULAPATI, N.V., ENGLUND, C., FALCONE, P. *et al.* (2012). Design and experimental validation of a cooperative driving system in the grand cooperative driving challenge. *IEEE Transactions on Intelligent Transportation Systems*, **13**, 994–1007. [28](#), [35](#)
- KISHORE KAMALANATHSHARMA, R. & RAKHA, H.A. (2014). Leveraging connected vehicle technology and telematics to enhance vehicle fuel efficiency in the vicinity of signalized intersections. *Journal of Intelligent Transportation Systems*, 1–12. [50](#)
- KOROSEC, K. (2016). Tesla: This is our most significant step towards safe self-driving cars. <http://fortune.com/2016/02/09/tesla-self-parking/>. [15](#)
- KOSHI, M. (1989). Development of the advanced vehicle road information systems in japan—the cacs project and after. In *Proceedings of JSK International Symposium-Technological Innovations for Tomorrow’s Automobile Traffic and Driving Information Systems*, 9–19. [10](#)
- KOWSHIK, H., CAVENEY, D. & KUMAR, P. (2011). Provable systemwide safety in intelligent intersections. *Vehicular Technology, IEEE Transactions on*, **60**, 804–818. [30](#)

BIBLIOGRAPHY

- KRAJZEWICZ, D., ERDMANN, J., BEHRISCH, M. & BIEKER, L. (2012). Recent development and applications of sumo–simulation of urban mobility. *International Journal on Advances in Systems and Measurements*, **5**, 128–138. [89](#)
- LEE, J. & PARK, B. (2012). Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment. *Intelligent Transportation Systems, IEEE Transactions on*, **13**, 81–90. [30](#)
- LEE, J., PARK, B.B., MALAKORN, K. & SO, J.J. (2013). Sustainability assessments of cooperative vehicle intersection control at an urban corridor. *Transportation Research Part C: Emerging Technologies*, **32**, 193–206. [30](#)
- LI, L. & WANG, F.Y. (2006). Cooperative driving at blind crossings using intervehicle communication. *Vehicular Technology, IEEE Transactions on*, **55**, 1712–1724. [30](#), [79](#)
- LI, L., WEN, D. & YAO, D. (2014). A survey of traffic control with vehicular communications. *Intelligent Transportation Systems, IEEE Transactions on*, **15**, 425–432. [28](#), [30](#), [78](#)
- LI, P.Y. & SHRIVASTAVA, A. (2002). Traffic flow stability induced by constant time headway policy for adaptive cruise control vehicles. *Transportation Research Part C: Emerging Technologies*, **10**, 275–301. [26](#)
- LI, S., LI, K., RAJAMANI, R. & WANG, J. (2011). Model predictive multi-objective vehicular adaptive cruise control. *IEEE Transactions on Control Systems Technology*, **19**, 556–566. [35](#)
- LIDSTROM, K., SJOBERG, K., HOLMBERG, U., ANDERSSON, J., BERGH, F., BJADE, M. & MAK, S. (2012). A modular CACC system integration and design. *IEEE Transactions on Intelligent Transportation Systems*, **13**, 1050–1061. [35](#)
- LIN, B.F., CHAN, Y.M., FU, L.C., HSIAO, P.Y., CHUANG, L.A., HUANG, S.S. & LO, M.F. (2012). Integrating appearance and edge features for sedan vehicle detection in the blind-spot area. *IEEE Transactions on Intelligent Transportation Systems*, **13**, 737–747. [23](#)

- LIU, B. & EL KAMEL, A. (2016a). Cooperative driving for connected autonomous vehicles at intersection. *8th International Workshop on Planning, Perception and Navigation for Intelligent Vehicles* – A bridge between Robotics and ITS technologies, Accepted. [129](#)
- LIU, B. & EL KAMEL, A. (2016b). Decentralized autonomous intersection control in connected vehicle environment. *6th International Workshop on Artificial Transportation Systems and Simulation*, Accepted. [129](#)
- LIU, B. & EL KAMEL, A. (2016c). Decentralized autonomous intersection control using V2X communications. *Journal of Intelligent Transportation Systems*, Under Review. [129](#)
- LIU, B. & EL KAMEL, A. (2016d). Trajectory planning for autonomous intersection management using V2X communications. *IEEE Transactions on Intelligent Transportation Systems*, 1st Revision. [129](#)
- LIU, B. & EL KAMEL, A. (2016e). V2X-based decentralized cooperative adaptive cruise control in the vicinity of intersections. *IEEE Transactions on Intelligent Transportation Systems*, **17**, 644–658. [79](#), [128](#)
- LIU, B., SUN, Q. & EL KAMEL, A. (2016). Improving the intersection’s throughput using V2X communication and cooperative adaptive cruise control. In *4th IFAC International Conference on Intelligent Control and Automation Sciences (ICONS 2016)*, 365–370, IFAC. [128](#)
- LOCHERT, C., BARTHEL, A., CERVANTES, A., MAUVE, M. & CALISKAN, M. (2005). Multiple simulator interlinking environment for ivc. In *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, 87–88, ACM. [20](#)
- LOCHERT, C., SCHEUERMANN, B. & MAUVE, M. (2010). *Information dissemination in VANETs*. John Wiley & Sons. [20](#)
- LU, Q. & KIM, K.D. (2016). Intelligent intersection management of autonomous traffic using discrete-time occupancies trajectory. *Journal of Traffic and Logistics Engineering*, **4**. [79](#)

BIBLIOGRAPHY

- LU, X.Y., HEDRICK, J.K. & DREW, M. (2002). Acc/cacc-control design, stability and robust performance. In *American Control Conference, 2002. Proceedings of the 2002*, vol. 6, 4327–4332, IEEE. [24](#)
- LUO, M., EL KAMEL, A. & GONG, G. (2011). Uml-based design of intelligent vehicles virtual reality platform. In *2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 115–120, IEEE. [88](#)
- LUO, M., EL KAMEL, A. & GONG, G. (2012). Simulation of natural environment impacts on intelligent vehicle based on a virtual reality platform. In *Control in Transportation Systems*, vol. 13, 116–121. [131](#)
- MACKINNON, D. (1975). High capacity personal rapid transit system developments. *IEEE Transactions on Vehicular Technology*, **24**, 8–14. [24](#)
- MADIREDDY, M., DE COENSEL, B., CAN, A., DEGRAEUWE, B., BEUSEN, B., DE VLIET, I. & BOTTELDOOREN, D. (2011). Assessment of the impact of speed limit reduction and traffic signal coordination on vehicle emissions using an integrated approach. *Transportation research part D: transport and environment*, **16**, 504–508. [50](#)
- MAILE, M. & DELGROSSI, L. (2009). Cooperative intersection collision avoidance system for violations (cicas-v) for avoidance of violation-based intersection crashes. *Enhanced Safety of Vehicles*. [19](#)
- MALAKORN, K.J. & PARK, B. (2010). Assessment of mobility, energy, and environment impacts of intellidrive-based cooperative adaptive cruise control and intelligent traffic signal control. In *IEEE International Symposium on Sustainable Systems and Technology (ISSST) 2010*, 1–6, IEEE. [50](#)
- MANDAVA, S., BORIBOONSOMSIN, K. & BARTH, M. (2009). Arterial velocity planning based on traffic signal information under light traffic conditions. In *ITSC'09. 12th International IEEE Conference on Intelligent Transportation Systems, 2009.*, 1–6, IEEE. [50](#), [57](#)
- MEIHONG, W. & WENHUA, Z. (2010). A comparison of four popular heuristics for task scheduling problem in computational grid. In *2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, 1–4, IEEE. [43](#)

- MILANÉS, V., RASTELLI, J.P., ONIEVA, E. & GONZÁLEZ, C. (2010). Controller for urban intersections based on wireless communications and fuzzy logic. *IEEE Transactions on Intelligent Transportation Systems*. 30
- MILANÉS, V., SHLADOVER, S.E., SPRING, J., NOWAKOWSKI, C., KAWAZOE, H. & NAKAMURA, M. (2014). Cooperative adaptive cruise control in real traffic situations. *IEEE Transactions on Intelligent Transportation Systems*, 15, 296–305. 35
- MILLER, H.J. & SHAW, S.L. (2001). *Geographic information systems for transportation: principles and applications*. Oxford University Press on Demand. 18, 24
- MIRCHANDANI, P. & WANG, F.Y. (2005). Rhodes to intelligent transportation systems. *IEEE Intelligent Systems*, 10–15. 29, 51, 78
- MONTEMERLO, M., BECKER, J., BHAT, S., DAHLKAMP, H., DOLGOV, D., ETTINGER, S., HAEHNEL, D., HILDEN, T., HOFFMANN, G., HUHNKE, B. *et al.* (2008). Junior: The stanford entry in the urban challenge. *Journal of field Robotics*, 25, 569–597. 13
- MORGAN, Y.L. (2010). Notes on dsrc & wave standards suite: Its architecture, design, and characteristics. *Communications Surveys & Tutorials, IEEE*, 12, 504–518. 18
- MORRIS, B., DOSHI, A. & TRIVEDI, M. (2011). Lane change intent prediction for driver assistance: On-road design and evaluation. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, 895–901, IEEE. 23
- NAUMANN, R., RASCHE, R. & TACKEN, J. (1998). Managing autonomous vehicles at intersections. *Intelligent Systems and their Applications, IEEE*, 13, 82–86. 30
- NEALE, V.L., PEREZ, M.A., LEE, S.E. & DOERZAPH, Z.R. (2007). Investigation of driver-infrastructure and driver-vehicle interfaces for an intersection violation warning system. *Journal of Intelligent Transportation Systems*, 11, 133–142. 19

BIBLIOGRAPHY

- NIEUWENHUIJZE, M.R., VAN KEULEN, T., ONCU, S., BONSEN, B. & NIJMEIJER, H. (2012). Cooperative driving with a heavy-duty truck in mixed traffic: Experimental results. *IEEE Transactions on Intelligent Transportation Systems*, **13**, 1026–1032. [35](#)
- NOTHDURFT, T., HECKER, P., OHL, S., SAUST, F., MAURER, M., RESCHKA, A. & BÖHMER, J.R. (2011). Stadtpilot: first fully autonomous test drives in urban traffic. In *2011 14th International IEEE Conference on Intelligent Transportation Systems*, 919–924, Washington, USA. [11](#)
- OMAE, M., OGITSU, T., HONMA, N. & USAMI, K. (2010). Automatic driving control for passing through intersection without stopping. *International Journal of Intelligent Transportation Systems Research*, **8**, 201–210. [30](#)
- OYAMA, S., TACHIKAWA, K. & SATO, M. (2000). Dsrc standards and etc systems development in japan. In *Proceedings of the 7th world congress on intelligent systems*. [18](#)
- OZGUNER, U., BAERTLEIN, B., CAVELLO, C., FARKAS, D., HATIPOGLU, C., LYTLE, S., MARTIN, J., PAYNTER, F., REDMILL, K., SCHNEIDER, S., WALTON, E. & YOUNG, J. (1997). The osu demo '97 vehicle. In *1997 IEEE Conference on Intelligent Transportation System*, 502–507, Boston, MA. [11](#)
- PANDIT, K., GHOSAL, D., ZHANG, H.M. & CHUAH, C.N. (2013). Adaptive traffic signal control with vehicular ad hoc networks. *Vehicular Technology, IEEE Transactions on*, **62**, 1459–1471. [29](#), [78](#)
- PAPAGEORGIU, M., DIAKAKI, C., DINOPOULOU, V., KOTSIALOS, A. & WANG, Y. (2003). Review of road traffic control strategies. *Proceedings of the IEEE*, **91**, 2043–2067. [78](#)
- PARK, Y. & MIN, D. (2015). Distributed traffic simulation using dds-communication based HLA for V2X. In *2015 Seventh International Conference on Ubiquitous and Future Networks (ICUFN)*, 450–455, IEEE. [16](#)
- PARSOPOULOS, K.E., VRAHATIS, M.N. *et al.* (2002). Particle swarm optimization method for constrained optimization problems. *Intelligent Technologies—Theory and Application: New Trends in Intelligent Technologies*, **76**, 214–220. [41](#)

- PERNG, J.W., WEN, Y.H., CHANG, W.J. & CIOU, M.Y. (2011). Design and implementation of an intelligent vehicle driving controller. In *2011 8th Asian Control Conference*, 1111–1116, Kaohsiung. [9](#)
- PETTI, S. & FRAICHARD, T. (2005). Safe motion planning in dynamic environments. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, 2210–2215, IEEE. [102](#)
- PIPES, L.A. (1953). An operational analysis of traffic dynamics. *Journal of applied physics*, **24**, 274–281. [24](#)
- PLOEG, J., SCHEEPERS, B., VAN NUNEN, E., VAN DE WOUW, N. & NIJMEIJER, H. (2011). Design and experimental evaluation of cooperative adaptive cruise control. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 260–265, IEEE. [35](#)
- PLOEG, J., SHLADOVER, S., NIJMEIJER, H. & VAN DE WOUW, N. (2012). Introduction to the special issue on the 2011 grand cooperative driving challenge. *Intelligent Transportation Systems, IEEE Transactions on*, **13**, 989–993. [28](#)
- POCZTER, S.L. & JANKOVIC, L.M. (2014). The google car: Driving toward a better future? *Journal of Business Case Studies (Online)*, **10**, 7. [34](#)
- PRIEMER, C. & FRIEDRICH, B. (2009). A decentralized adaptive traffic signal control using v2i communication data. In *Intelligent Transportation Systems, 2009. ITSC'09. 12th International IEEE Conference on*, 1–6, IEEE. [29](#), [78](#), [79](#), [120](#)
- RAJAMANI, R. (2011). *Vehicle dynamics and control*. Springer Science & Business Media. [23](#), [25](#)
- RAJAMANI, R., CHOI, S., HEDRICK, J. & LAW, B. (1998). Design and experimental implementation of control for a platoon of automated vehicles. In *Proceedings of the ASME Dynamic Systems and Control Division (1998)*. [26](#)
- RAKHA, H. & KAMALANATHSHARMA, R.K. (2011). Eco-driving at signalized intersections using v2i communication. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, 341–346, IEEE. [19](#)

BIBLIOGRAPHY

- RESCHKA, A., BOHMER, J., SAUST, F., LICHTER, B. & MAURER, M. (2012). Safe, dynamic and comfortable longitudinal control for an autonomous vehicle. In *2012 IEEE Intelligent Vehicles Symposium*, 346–351, Reschka, Andreas. [12](#)
- RODRIGUE, J.P. (2013). Urban transport challenges. <https://people.hofstra.edu/geotrans/eng/ch6en/conc6en/ch6c4en.html>. [8](#)
- RUSSELL, S. (2006). DARPA grand challenge winner: Stanley the robot! *Popular Science*. [13](#)
- SCHAKEL, W.J., VAN AREM, B. & NETTEN, B.D. (2010). Effects of cooperative adaptive cruise control on traffic flow stability. In *2010 13th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 759–764, IEEE. [35](#)
- SCHEPPERLE, H., BÖHM, K. & FORSTER, S. (2008). Traffic management based on negotiations between vehicles—a feasibility demonstration using agents. In *Agent-Mediated Electronic Commerce and Trading Agent Design and Analysis*, 90–104, Springer. [30](#)
- SCHOUWENAARS, T., DE MOOR, B., FERON, E. & HOW, J. (2001). Mixed integer programming for multi-vehicle path planning. In *Control Conference (ECC), 2001 European*, 2603–2608, IEEE. [102](#)
- SELVI, V. & UMARANI, D.R. (2010). Comparative analysis of ant colony and particle swarm optimization techniques. *International Journal of Computer Applications (0975-8887)*, **5**. [43](#)
- SENGUPTA, R. *et al.* (2007). Cooperative collision warning systems: Concept definition and experimental implementation. *Journal of Intelligent Transportation Systems*, **11**, 143–155. [50](#)
- SHLADOVER, S.E. (1995). Review of the state of development of advanced vehicle control systems (avcs). *Vehicle System Dynamics*, **24**, 551–595. [24](#)
- SIEMENS (2015). Vehicle-to-x communication technology infographic. www.mobility.siemens.com. [21](#)

BIBLIOGRAPHY

- STEFANOIU, D., BORNE, P., POPESCU, D., FILIP, F.G. & EL KAMEL, A. (2014). *Optimization in Engineering Sciences: Approximate and Metaheuristic Methods*. John Wiley & Sons. 43
- SUKTHANKAR, R., HANCOCK, J. & THORPE, C. (1998). Tactical-level simulation for intelligent transportation. *Mathematical and computer modelling*, **27**, 229–242. 12
- SWAROOP, D., HEDRICK, J., CHIEN, C. & IOANNOU, P. (1994). A comparison of spacing and headway control laws for automatically controlled vehicles1. *Vehicle System Dynamics*, **23**, 597–625. 25
- TEMPLETON, B. (2013). Cameras or lasers? www.templetons.com/brad/robocars/cameras-lasers.html. 15
- TEODOROVIC, D., LUCIC, P., POPOVIC, J. & KIKUCHI, S. (2001). Intelligent isolated intersection. In *The 10th IEEE International Conference on Fuzzy Systems.*, vol. 1, 276–279, IEEE. 30
- THORPE, C., JOCHEM, T. & POMERLEAU, D. (1997). The 1997 automated highway free agent demonstration. In *Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on*, 496–501, IEEE. 26
- THRUN, S., MONTEMERLO, M., DAHLKAMP, H., STAVENS, D., ARON, A., DIEBEL, J., FONG, P., GALE, J., HALPENNY, M., HOFFMANN, G. *et al.* (2006). Stanley: The robot that won the darpa grand challenge. *Journal of field Robotics*, **23**, 661–692. 13
- TSUGAWA, S. (2002). Inter-vehicle communications and their applications to intelligent vehicles: an overview. In *Intelligent Vehicle Symposium, 2002. IEEE*, vol. 2, 564–569, IEEE. 29
- UBIERGO, G.A. & JIN, W.L. (2015). Mobility and environment improvement of signalized networks through vehicle-to-infrastructure (v2i) communications. In *Transportation Research Board 94th Annual Meeting*, 15-4309. 19
- URMSON, C., ANHALT, J., BAGNELL, D., BAKER, C., BITTNER, R., CLARK, M., DOLAN, J., DUGGINS, D., GALATALI, T., GEYER, C. *et al.* (2008).

BIBLIOGRAPHY

- Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics*, **25**, 425–466. [13](#)
- VAHIDI, A. & ESKANDARIAN, A. (2003). Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE transactions on intelligent transportation systems*, **4**, 143–153. [23](#)
- VAN AREM, B., VAN DRIEL, C.J. & VISSER, R. (2006). The impact of cooperative adaptive cruise control on traffic-flow characteristics. *Intelligent Transportation Systems, IEEE Transactions on*, **7**, 429–436. [21](#), [26](#), [27](#), [34](#)
- VAN DEN BERG, J.P. & OVERMARS, M.H. (2005). Prioritized motion planning for multiple robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005.*, 430–435, IEEE. [31](#), [102](#)
- VAN NUNEN, E., KWAKKERNAAT, R., PLOEG, J. & NETTEN, B.D. (2012). Cooperative competition for future mobility. *IEEE Transactions on Intelligent Transportation Systems*, **13**, 1018–1025. [28](#)
- VANDER WERF, J., SHLADOVER, S., MILLER, M. & KOURJANSKAIA, N. (2002). Effects of adaptive cruise control systems on highway traffic flow capacity. *Transportation Research Record: Journal of the Transportation Research Board*, 78–84. [27](#)
- WANG, J. & RAJAMANI, R. (2004). Should adaptive cruise-control systems be designed to maintain a constant time gap between vehicles? *IEEE Transactions on Vehicular Technology*, **53**, 1480–1490. [24](#)
- WANG, S. (2004a). On the intermittence of routing paths in vehicle-formed mobile ad hoc networks on highways. In *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*, 803–809, IEEE. [17](#)
- WANG, S.Y. (2004b). Predicting the lifetime of repairable unicast routing paths in vehicle-formed mobile ad hoc networks on highways. In *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on*, vol. 4, 2815–2819, IEEE. [17](#)

- WEDEL, J.W., SCHÜNEMANN, B. & RADUSCH, I. (2009). V2X-based traffic congestion recognition and avoidance. In *2009 10th International Symposium on Pervasive Systems, Algorithms, and Networks*, 637–641, IEEE. [50](#)
- WILLE, J.M., S., F. & MAURER, M. (2010). Stadtpilot: driving autonomously on braunschweig’s inner ring road. In *2010 IEEE Intelligent Vehicles Symposium*, 506–511, San Diego, USA. [12](#)
- WILLIAMS, M. (1988). Prometheus-the european research programme for optimising the road transport system in europe. In *IEEE Colloquium on Driver Information*, 1/1–1/9, London, UK. [10](#), [24](#)
- WU, J., ABBAS-TURKI, A. & EL MOUDNI, A. (2012). Cooperative driving: an ant colony system for autonomous intersection management. *Applied Intelligence*, **37**, 207–222. [30](#), [79](#)
- WUTHISHUWONG, C., TRAECHTLER, A. & BRUNS, T. (2015). Safe trajectory planning for autonomous intersection management by using vehicle to infrastructure communication. *EURASIP Journal on Wireless Communications and Networking*, **2015**, 1–12. [79](#), [114](#)
- XIA, C. & EL KAMEL, A. (2016). Neural inverse reinforcement learning in autonomous navigation. *Robotics and Autonomous Systems*. [131](#)
- XIAO, L. & GAO, F. (2010). A comprehensive review of the development of adaptive cruise control systems. *Vehicle System Dynamics*, **48**, 1167–1192. [23](#), [24](#)
- XIE, X.F., BARLOW, G.J., SMITH, S.F. & RUBINSTEIN, Z.B. (2011). Platoon-based self-scheduling for real-time traffic signal control. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 879–884, IEEE. [78](#)
- XIE, X.F., SMITH, S.F., LU, L. & BARLOW, G.J. (2012). Schedule-driven intersection control. *Transportation Research Part C: Emerging Technologies*, **24**, 168–189. [78](#)

BIBLIOGRAPHY

- XU, H. & BARTH, M. (2006). An adaptive dissemination mechanism for inter-vehicle communication-based decentralized traffic information systems. In *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, 1207–1213, IEEE. [20](#)
- XU, J., CHEN, G. & XIE, M. (2000). Vision-guided automatic parking for smart car. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, 725–730. [23](#)
- XU, Q. & SENGUPTA, R. (2003). Simulation, analysis, and comparison of acc and cacc in highway merging control. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, 237–242, IEEE. [24](#)
- YAN, X., ZHANG, H. & WU, C. (2012). Research and development of intelligent transportation systems. In *2012 11th International Symposium on Distributed Computing and Applications to Business, Engineering Science*, 321–327, Wuhan, China. [10](#)
- YI, J. & HOROWITZ, R. (2006). Macroscopic traffic flow propagation stability for adaptive cruise controlled vehicles. *Transportation Research Part C: Emerging Technologies*, **14**, 81–95. [26](#)
- YOUSEFI, S., MOUSAVI, M.S. & FATHY, M. (2006). Vehicular ad hoc networks (vanets): challenges and perspectives. In *ITS Telecommunications Proceedings, 2006 6th International Conference on*, 761–766, IEEE. [17](#)
- YU, Y., EL KAMEL, A. & GONG, G. (2013). Modeling intelligent vehicle agent in virtual reality traffic simulation system. In *2013 2nd International Conference on Systems and Computer Science (ICSCS)*, 274–279, IEEE. [131](#)
- YU, Y., EL KAMEL, A., GONG, G. & LI, F. (2014). Multi-agent based modeling and simulation of microscopic traffic in virtual reality system. *Simulation Modelling Practice and Theory*, **45**, 62–79. [131](#)
- ZHAO, J., OYA, M. & EL KAMEL, A. (2009). A safety spacing policy and its impact on highway traffic flow. In *Intelligent Vehicles Symposium, 2009 IEEE*, 960–965, IEEE. [35](#)

BIBLIOGRAPHY

- ZHENG, T., EL KAMEL, A. & WANG, S. (2013). Performance analysis of fuzzy controller with delay. In *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, 794–799, IEEE. 131
- ZHOU, J. & PENG, H. (2005). Range policy of adaptive cruise control vehicles for improved flow stability and string stability. *IEEE Transactions on Intelligent Transportation Systems*, **6**, 229–237. 24
- ZIEGLER, C. (2015). Volvo will run a public test of self-driving cars with 100 real people in 2017. www.theverge.com. 15
- ZOHDY, I.H. & RAKHA, H. (2012). Game theory algorithm for intersection-based cooperative adaptive cruise control (cacc) systems. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, 1097–1102, IEEE. 30
- ZOHDY, I.H. & RAKHA, H.A. (2014). Intersection management via vehicle connectivity: The intersection cooperative adaptive cruise control system concept. *Journal of Intelligent Transportation Systems*, 1–16. 79

BIBLIOGRAPHY

Contrôle et Optimisation des Systèmes de Transport Intelligents dans le Voisinage des Intersections

Résumé: Cette thèse est consacrée à étudier les applications potentielles de véhicules autonomes et communications V2X pour construire les systèmes de transport intelligents. Premièrement, le comportement de caravane dans un environnement de véhicule connecté est étudié. Un algorithme de commande de caravane est conçu pour obtenir l'espacement sécuritaire ainsi que la conformité de la vitesse et de l'accélération. Deuxièmement, à plus grande échelle, les caravanes autour d'une intersection sont considérées. Le débit pendant une période de signal de trafic peut être améliorée en tirant profit de la capacité redondante de la route. Dans diverses contraintes, les véhicules peuvent choisir d'accélérer à se joindre à la caravane précédente ou à décélérer de déroger à l'actuel. Troisièmement, une intersection sans signalisation en VANET est considérée. Dans des conditions de faible trafic, les véhicules peuvent réguler leur vitesse avant d'arriver à l'intersection en fonction du temps d'occupation de la zone de conflit (TOZC) stocké au niveau du gestionnaire, afin qu'ils puissent traverser l'intersection sans collision ni arrêt. Le délai peut être réduit en conséquence. Enfin, un algorithme de gestion d'intersection autonome universelle, qui peut fonctionner même avec le trafic lourd, est développé. Le véhicule cherche sécurité fenêtres entrant dans le TOZC. Ensuite, sur la base des fenêtres trouvées et le mouvement du véhicule qui précède, les trajectoires des véhicules peuvent être planifiées en utilisant une méthode de programmation dynamique segmentée. Tous les algorithmes conçus sont testés et vérifiés avec succès par des simulations dans scénarios différents.

Mots-clés: Système de Transport Intelligents, Véhicules Autonomes, Réseau Ad-hoc de Véhicules, Régulateur de Vitesse Adaptatif Coopératif, Gestion Autonome d'Intersection, Optimisation sous Contrainte.

Control and Optimization for Intelligent Transportation Systems in Vicinity of Intersections

Abstract: This thesis is devoted to study the potential applications of autonomous vehicles and V2X communications to contribute to the construction of intelligent transportation systems. Firstly, the behavior of platoon in connected vehicle environment is studied. A platoon control algorithm is designed to obtain safe spacing as well as accordance of velocity and acceleration for vehicles in the same lane. Secondly, in larger scale, the platoons around the intersection are considered. The intersection's throughput in a traffic signal period can be improved by taking advantage of the redundant road capacity while respecting diverse constraints. Vehicles can choose to accelerate to join in the preceding platoon or to decelerate to depart from the current one. Thirdly, an unsignalized intersection in VANET is considered. In light traffic conditions, vehicles can regulate their velocities before arriving at the intersection according to the conflict zone occupancy time (CZOT) stored at the manager, so that they could get through the intersection without collision or stop. The delay can be reduced accordingly. Finally, an universal autonomous intersection management algorithm, which can work even with heavy traffic, is developed. The vehicle searches for safe entering windows in the CZOT. Then based on the found windows and the motion of preceding vehicle, the trajectories of vehicles can be planned within short time using a segmented dynamic programming method. All the designed algorithms are successfully tested and verified by simulations in various scenarios.

Keywords: Intelligent Transportation Systems, Autonomous Vehicles, Vehicular Ad hoc Network (VANET), Cooperative Adaptive Cruise Control, Autonomous Intersection Management, Constrained Optimization.