Cooperative Adaptive Cruise Control Performance Analysis

Qi Sun

N° d'ordre : | 3 | 0 | 3 |

CENTRALE LILLE

THÈSE

présentée en vue d'obtenir le grade de

DOCTEUR

Spécialité : Automatique, Génie Informatique, Traitement du Signal et des Images

par

SUN Qi

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

Analyse de Performances de Régulateurs de Vitesse Adaptatifs Coopératifs

Soutenue le 15 décembre 2016 devant le jury :

M. Pierre BORNE	Ecole Centrale de Lille	Président
M. Noureddine ELLOUZE	Ecole Nationale d'Ingénieurs de Tunis	Rapporteur
Mme. Shaoping WANG	Université de Beihang, Chine	Rapporteur
M. Hamid AMIRI	Ecole Nationale d'Ingénieurs de Tunis	Examinateur
M. Abdelkader EL KAMEL	Ecole Centrale de Lille	Directeur de Thèse
Mme. Zhuoyue SONG	Université de technologie de Pékin, Chine	Examinateur
Mme. Liming ZHANG	Université de Macao, Chine	Examinateur

Thèse préparée dans le 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

Serial N° : 3 0 3

CENTRALE LILLE

THESIS

presented to obtain the degree of

DOCTOR

Topic : Automatic control, Computer Engineering, Signal and Image Processing

by

SUN Qi

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

Cooperative Adaptive Cruise Control Performances Analysis

Defended on December 15, 2016 in presence of the committee :

Mr. Pierre BORNE	Ecole Centrale de Lille	President
Mr. Noureddine ELLOUZE	Ecole Nationale d'Ingénieurs de Tunis	Reviewer
Mrs. Shaoping WANG	Université de Beihang, China	Reviewer
Mr. Hamid AMIRI	Ecole Nationale d'Ingénieurs de Tunis	Examiner
Mr. Abdelkader EL KAMEL	Ecole Centrale de Lille	PhD Supervisor
Mrs. Zhuoyue SONG	Université de technologie de Pékin, China	Examiner
Mrs. Liming ZHANG	Université de Macao, 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.

Acknowledgement

This research work has been realized at "Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL)" in École Centrale de Lille, with the research group "Optimisation : Modèles et Applications (OPTIMA)" from September 2013 to December 2016. This work is financially supported by China Scholarship Council (CSC). Thanks to the founding of CSC, it is my great honor having this valuable experience in France.

First and foremost I offer my sincerest gratitude to my PhD supervisor, Prof. Abdelkader EL KAMEL, for his supervision, valuable guidance, continuous encouragement as well as given me extraordinary experiences through out my Ph.D. experience. I could not have imagined having a better tutor and mentor for my Ph.D. study.

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. Noureddine ELLOUZE and Prof. Shaoping WANG, who have kindly accepted the invitation to be reviewers of my Ph.D. thesis, for their encouragement, insightful comments and interesting questions. My gratitude to Prof. Hamid AMIRI, Prof. Zhuoyue SONG and Prof. Liming ZHANG, 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 administration. 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 residence life.

My sincere thanks also goes to Dr. Tian ZHENG, Dr. Yue YU, Dr. Daji TIAN,

Dr. Chen XIA and Dr. Bing LIU, for offering me useful suggestion during my research in the laboratory as well as after their graduation.

I would like to take the opportunity to express my gratitude and to thank my fellow workmates in CRIStAL: Yihan LIU, Jian ZHANG for the stimulating discussions for the hard teamwork. Also I wish to thank my friends and colleagues: Qi GUO, Hongchang ZHANG, Lijie BAI, Jing BAI, Ben LI, Xiaokun DING, Jianxin FANG, Hengyang WEI, Lei ZHANG, Chang LIU etc., for their friendship in the past three years. All of them have given me support and encouragement in my thesis work. Special thanks to Meng MENG, for her accompany, patience, and encouragement.

All my gratitude goes to Ms. Hélène 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 and history, enriched my experience in France.

My acknowledgements to all the professors and teachers in École Centrale de Pékin, 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. Zongxia JIAO 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, Yibo SUN and Yumei LI, for supporting me to pursue this degree and to accept my absence for four years of living abroad.

Villeneuve d'Ascq, France *Sun Qi*

November, 2016

Contents

List of Figures

List of Algorithms

Abbreviations

ADAS - Advanced Driver Assistant Systems

AHS - Automated Highway Systems

CA - Collision Avoidance

CACC - Cooperative Adaptive Cruise Control

CC - Cruise Control

CCTV - Closed Circuit Tele-Vision

CTH - Constant Time Headway

DSRC - Dedicated Short-Range Communications

DTVACACC - Degraded Two-Vehicle-Ahead Cooperative Adaptive Cruise Control

GPS - Global Positioning System

IRL - Inverse Reinforcement Learning

ITS - Intelligent Transportation Systems

LCA - Lane Change Assistant

LfD - Learning from Demonstration

MDP - Markov Decision Process

NNQL - Neural Network Q-Learning

RL - Reinforcement Learning

TVACACC - Two-Vehicle-Ahead Cooperative Adaptive Cruise Control

VANETs - Vehicular Ad hoc Networks

V2V - Vehicle-to-Vehicle

V2I - Vehicle-to-Infrastructure

V2X - Vehicle-to-X

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 industrialized nations, the demand for transportation is much greater than any other period in history. More comfortable and more flexible, private vehicles are selected by many families. Besides, the development of automobile industry reduces the cost to own a car, thus vehicle ownership has been growing rapidly all over the world, especially in big cities. However, the increasing number of vehicles makes our society to suffer from traffic congestion, 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. Researches 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 solution 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 developed 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 pre-define some parameters in the system, and then he/she can decide to follow the advisory assistance or not. Recently, with

the improvement of artificial intelligence and sensing technology, companies and institutes have been committed to the research and development of autonomous driving. In some 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 inter-vehicle space will improve the usage of road capacity. However, autonomous cars still need driver's anticipation in these scenarios with complicated traffic situation or limited information. The inner layout of autonomous vehicles would not be much different from current ones, because steering wheel and pedals are still indispensable. 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 congestion and pollution problem will be relieved.

Another solution 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 intelligent 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. Therefore, if the interaction between vehicles and environment is available, the notifications can be transferred via Vehicle-to-X (V2X) communications, thus vehicles can be recognized 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, individual 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.

Main contributions

The main contributions of the thesis are summarized as follows:

- A novel decentralized Two-Vehicle-Ahead Cooperative Adaptive Cruise Control (TVACACC) longitudinal tracking control framework is proposed in this thesis. It is shown that the feed forward controller enables small inter-vehicle distances, using a velocity-dependent spacing policy. Moreover, a frequency-domain approach of string stability is theoretically analyzed. By using the TVA-wireless communication among the vehicles, a better string stability is proved compared to the conventional system, resulting in lower disturbance. Vehicle platoon in Stop-and-Go scenario is simulated with both normal and degraded V2V communication. It is shown that the proposed system yields a string-stable behavior, in accordance with the theoretical analysis, which also indicates a larger traffic flux and a better comfort.

- A graceful degradation technique for Cooperative Adaptive Cruise Control (CACC) is presented, serving as an alternative fallback scenario to Adaptive Cruise Control (ACC). The concept of the proposed approach is to obtain the minimum loss of functionality of CACC when the wireless link fails or when the preceding vehicle is not equipped with wireless communication units. The proposed strategy, which is referred to as Degraded TVACACC (DT-VACACC), uses the technique of estimation of the preceding vehicle's current acceleration to replace the desired acceleration, which would normally be communicated over a wireless V2V communication for the conventional CACC system.

- A novel approach to obtain an autonomous longitudinal vehicle controller

is proposed. To achieve this objective, a vehicle architecture with its CACC subsystem has been presented. With this architecture, we have also described the specific requirements for an efficient autonomous vehicle control policy through Reinforcement Learning (RL) and the simulator in which the learning engine is embedded. A policy-gradient algorithm estimation has been introduced and has used a back propagation neural network for achieving the longitudinal control.

Outline of the thesis

This thesis is divided into 5 chapters:

In **Chapter 1**, the concept of intelligent road transportation systems is introduced in detail. 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 will be introduced in this chapter as well. CACC system, which is an extension of ACC systems by enabling the communication among the vehicles in a platoon is presented. CACC can not only relief the driver from repetitive jobs like adjusting speed and distance to the preceding vehicle like ACC, but also has safer and smoother response than ACC systems. Then Dedicated Short-Range Communications (DSRC) 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. Different kinds of applications using VANET are developed in order to make the road transportation safer, more efficient and user friendly. Finally, the technology of machine learning will be introduced, which can be applied on intelligent vehicles.

In **Chapter 2**, instead of the individual stability of each vehicle, another stability criterion known as the string stability is also described. For a cascaded system, e.g. a platoon of automated vehicles, stability of each component system itself is not sufficient to guarantee a good performance of all systems, such as the non-convergence of spacing error for two consecutive vehicles. Therefore, the string stability is con-

sidered as the most important criterion to evaluate the performance of intelligent vehicle platoon. In the second part, the Markov decision processes, which are the underlying structure of reinforcement learning, are described. Several classical algorithms for solving Markov decision process (MDP) are also briefly introduced. The fundamental concepts of the reinforcement learning is then brought.

In **Chapter 3**, we concentrate on the vehicle longitudinal control system design. The spacing policy and its associated control law are designed with the constrains of string stability. The CTH spacing policy is adopted to determine the desired spacing from the preceding vehicle. It will be shown that the proposed TVACACC system could ensure both the string stability. In addition, through the comparisons between the TVACACC and the conventional CACC and ACC systems, we could find the obvious advantages of the proposed system in improving traffic capacity especially in the high-density traffic conditions. The above proposed longitudinal control system will be validated to be effective through a series of simulations with normal and degraded V2V communication.

In **Chapter 4**, wireless communication faults must be taken into account to accelerate practical implementation of CACC in everyday traffic. To this end, a degradation technique for CACC is presented, used as an alternative fallback strategy to ACC. The concept of the proposed approach is to remain the minimum loss of functionality of CACC when the wireless link fails or when the preceding vehicle is not equipped with wireless communication units. The proposed strategy, which is referred to as DTVACACC, uses Filter Kalman to estimate the preceding vehicle's current acceleration to replace to the desired acceleration. In addition, a switch criterion from TVACACC to DTVACACC is presented. Both theoretical as well as experimental results of the DTVACACC system will be shown with respect to string stability characteristics by reducing the minimum string-stable headway time.

In **Chapter 5**, a novel approach to obtain an autonomous longitudinal vehicle CACC controller is proposed. To achieve this objective, a vehicle architecture with its CACC subsystem is presented. Using this architecture, specific requirements for an efficient autonomous vehicle control policy through RL and the simulator are de-

scribed, in which the learning engine is embedded. The policy-gradient algorithm estimation will be introduced and has used a back propagation neural network for achieving the longitudinal control. Then, experimental results, through simulation, show that this design approach can result in efficient behavior for CACC.

Chapter 1

Introduction to ITS

Sommaire

1.1. General traffic situation

The global vehicle production rises significantly thanks to the development of automobile industry during past years. [44] reported that there were 41 million cars being produced around the world only in the year 2000. Then, in 2005, 47 million cars were produced worldwide. Specially in 2015, almost 70 million passenger cars were produced, as seen in Fig. 1.1. Except in 2008 and 2009, car sales dried up on account of the economic crisis. Due to the increased demand, the volume of automobiles sold is back to pre-crisis levels today, especially from Asian markets. The passenger car sales are expected to continuous increase to about 100 million units in 2017 worldwide. China is ranked as the largest passenger car manufacturer in the world, having produced more than 18 million cars in 2013, and making up for more than 22 percent of the world's passenger vehicle production. Transport infrastructure investment is projected to grow at an average annual rate of about 5% worldwide over the period of 2014 to 2025. Roads will likely remain the biggest area of investment, especially for growth markets. This is partly due to the rise in prosperity and, hence, car ownership in developing countries 1.2.

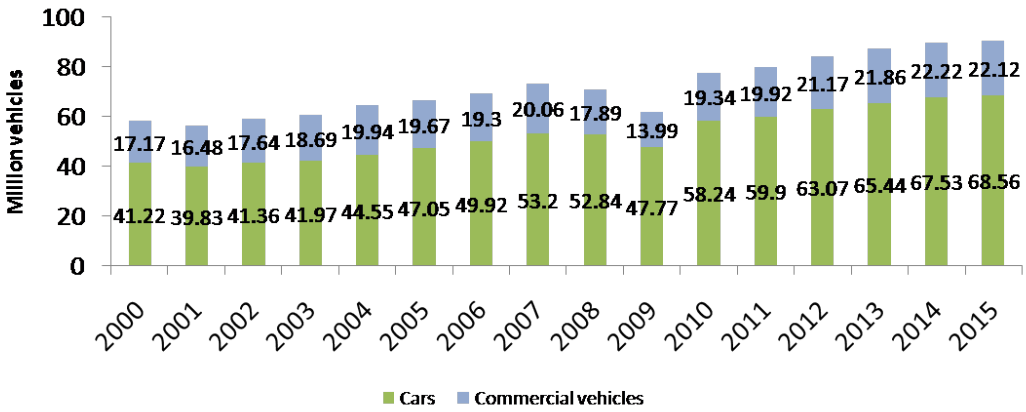


Figure 1.1 – *Worldwide automobile production from 2000 to 2015 (in million vehicles)*

Along within this augmentation, on one hand, we benefit the vehicles in different aspects. Like Europe, road transport is the largest share of intra-EU transport. The share of EU-28 [1] inland freight that was transported by road (74.9%) was more

[1] EU-28: The European Union (EU) was established on 1 November 1993 with 12 Member States. Their number has grown to the present 28 on 1 July 2013, through a series of enlargements.

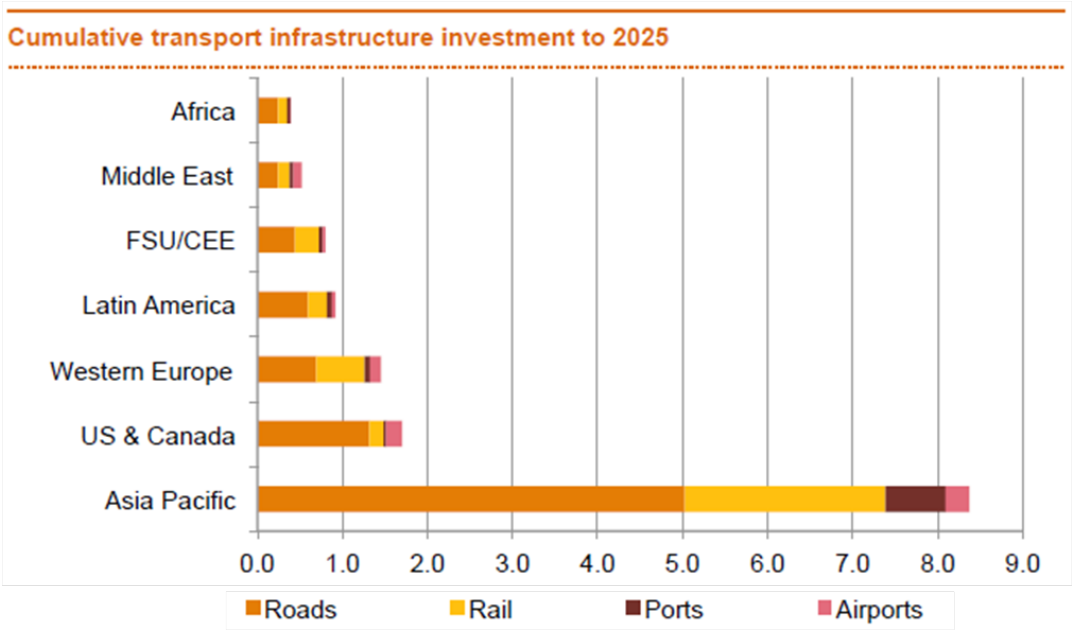


Figure 1.2 – *Cumulative transport infrastructure investment (in trillion dollars)*

than four times as high as the share transported by rail (18.2%), while the remainder (6.9%) of the freight transported in the EU-28 in 2013 was carried along inland waterways. The total inland freight transport in the EU-28 was over 2,200 billion tonne-kilometers in 2013[35]. Passenger cars accounted for 83.2% of inland passenger transport in the EU-28 in 2013, with motor coaches, buses and trolley buses (9.2%) and trains (7.6%) both accounting for less than a tenth of all traffic [36].

On the other hand, we have to face the spreading traffic problems:

- *Accidents and safety.* Ascending 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. A large proportion of accidents are caused by incorrect driving behaviors, such as violate regulations, speeding, fatigue driving and drunken driving.

- *Congestion.* Traffic jam is a very common transport problem in urban agglomerations. It is usually due to the lag between infrastructure construction and the increasing vehicle ownership. There are another reasons can be referred to improper traffic light signal, inappropriate road construction and accidents.

- *Environment impacts.* Noise pollution and air 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.

We can see from the White paper of 2004, the European Commission has set the ambitious aim of decreasing the number of road traffic fatalities by 2014. Much progress has been achieved. The total number of fatalities in road traffic accidents decreased by 45% between 2004 and 2014 (Figure 1.3) at the level of the EU-28. Road mobility comes at a high price in terms of lives lost: in 2014, slightly over 25 thousand persons lost their lives in road accidents within the EU-28. A general trend towards fewer road traffic fatalities has long been observed in all countries in Europe. However, at the level of the EU, this downward trend has come to a standstill as the total number of fatalities registered in 2014 remained at the same level as in 2013 [37].

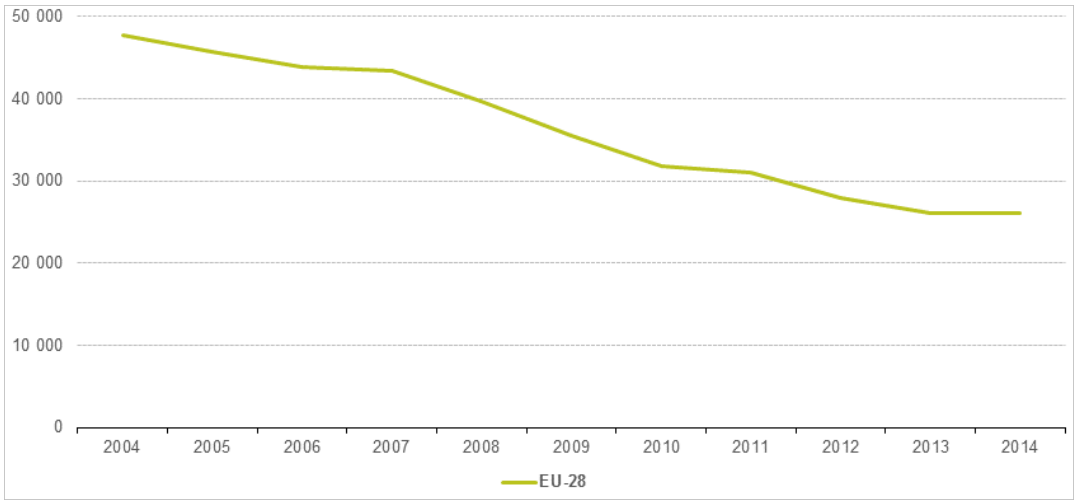


Figure 1.3 – *Total number of fatalities in road traffic accidents, EU-28*

A solution to the traffic problems is to build adequate highways and streets.

However, the fact that it is becoming increasingly difficult to build additional highway, for both financial and environmental reasons. Data shows that the traffic volume capacity added every year by construction lags the annual increase in traffic volume demanded, thus making traffic congestion increasingly worse. Therefore, the solution to the problem must lie in other approaches, one of which is to optimize the use of highway and fuel resources, provide safe and comfortable transportation, while have minimal impact on the environment. It is a great challenge to develop vehicles that can satisfy these diverse and often conflicting requirements. To meet this challenge, the new approach of "Intelligent Transportation System" (ITS) has shown its potential of increasing the safety, reducing the congestion, and improving the driving conditions. Early studies show that it is possible to cut accidents by 18%, gas emissions by 15%, and fuel consumption by 12% by employing ITS approach [161].

1.2. Intelligent Transportation Systems

1.2.1. Definition of ITS

A concept transportation system named "Futurama" was exhibited at the World's Fair 1940 in New York. At the same time, the origin of Intelligent Transportation System (ITS) appeared. After a long story via many researches and projects between 1980 to 1990 in Europe, North America and Japan, today's mainstream of ITS was formed. ITS is a transport system which is comprised of an advanced information and telecommunications network for users, roads and vehicles. By sharing vital information, ITS allows people to get more from transport networks, in greater safety, efficiency, and with less impact to the environment. The Conceptual principle of ITS is illustrated in Figure. 1.4.

For example, [64] designed an architecture of road ITS for commercial vehicles. This system is used to reduce fuel consumption through fuel-saving advice, maintain driver and vehicle safety with remote vehicle diagnostics and enable drivers to access information more conveniently. Generally speaking, there are three layers in ITS system, see Figure. 1.5:

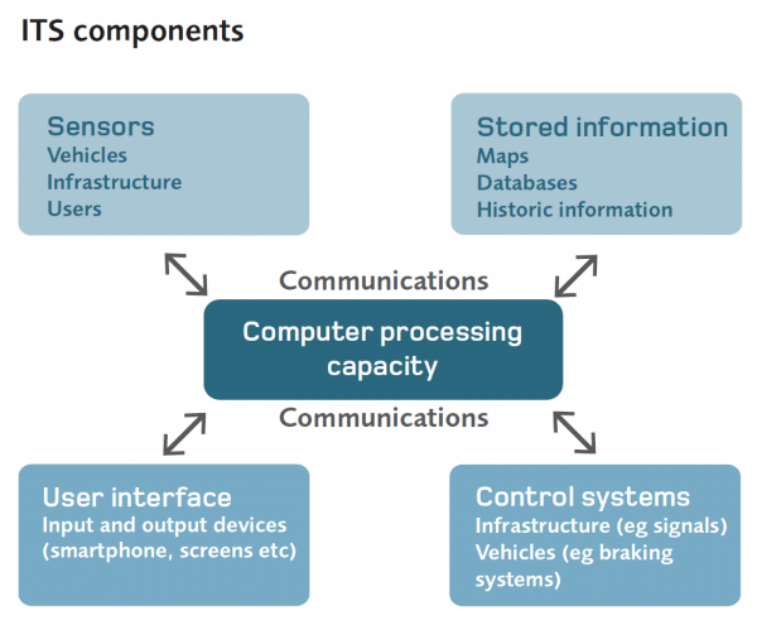


Figure 1.4 – *Conceptual principal of ITS*

- *Information collection*: This layer employs a vehicle terminal which is equipped with roadside surveillance including vehicle sensors, CCTV and camera, intelligent vehicle identification, etc. Meanwhile, it enables the information exchange with other units and infrastructures, such as parking information system, dynamic bus information center, police radio station traffic division dispatch center and center of freeway bureau.

- *Communication*: This layer ensures real-time, secure and reliable transmission between each layer via different networks, such as 3G/4G, Wi-Fi, Bluetooth, wired networks and optical fiber.

- *Information processing*: 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.

- *Information publishing and strategy execution*: In this layer, each individual ve-

hicle transfers information of their state and control strategy to the different centers. Therefore, these centers are able to publish traffic condition, manage all connected vehicles and execute complete strategy based on collected information in different situations, e.g. lane change, traffic light and intersection, freeway, etc.

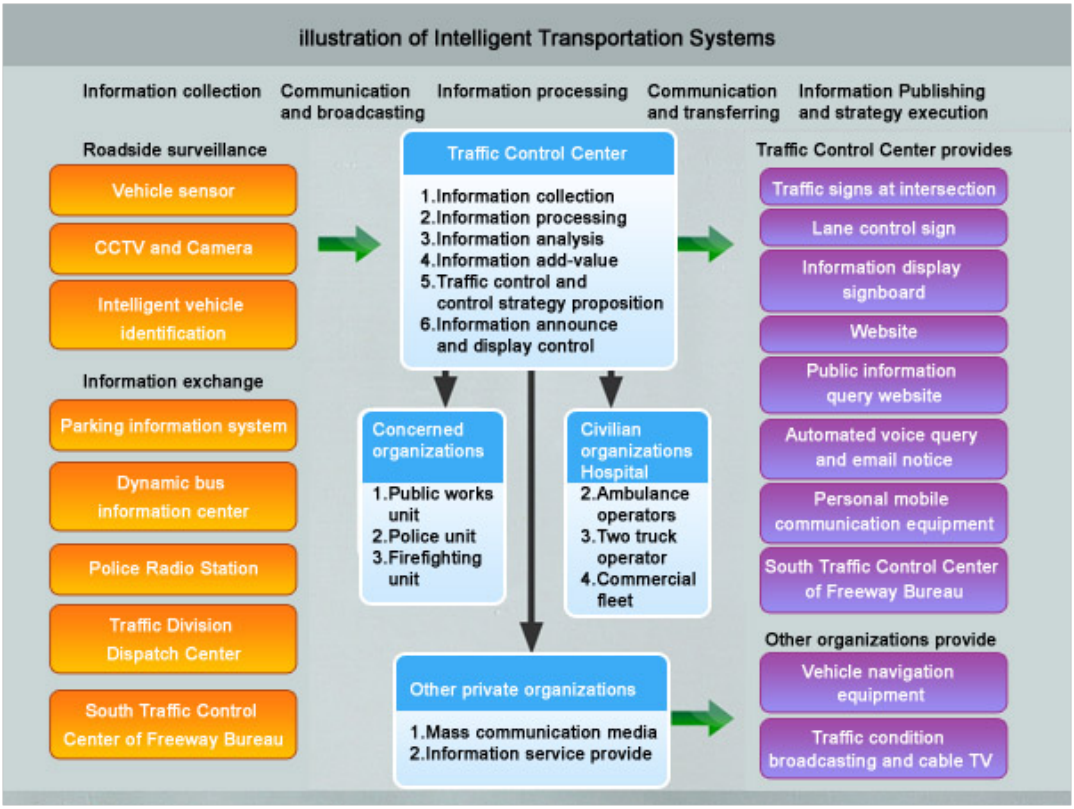


Figure 1.5 – *Instance for road ITS system layout*

1.2.2. ITS applications

Although ITS may refer to all types of transport, EU Directive 2010/40/EU (7 July 2010) defines ITS as systems in which information and communication technologies are applied in the field of road transport, including infrastructure, vehicles and users, and in traffic management and mobility management, as well as for interfaces with other modes of transport, see Figure. 1.6.

ITS is actually a big system which concerns a broad range of technologies and diverse activities.

Figure 1.6 – *ITS applications*

- *Adaptive Cruise Control (ACC)*: ACC systems perform longitudinal control by controlling the throttle and brakes so as to maintain a desired spacing from the preceding vehicle. A significant benefit of using ACC is to avoid rear-end collisions. The SeiSS study reported that it could save up to 4 000 accidents in Europe in 2010 if only 3% of the vehicles were equipped [3].

- *Lane Change Assistant (LCA) system*. The LCA will check for obstacles in a vehicle's course when the driver intends to change lanes. The same study estimated that 1 500 accidents could be avoided in 2010 given a penetration rate of only 0.6%, while a penetration rate of 7% in 2020 would lead to 14 000 fewer accidents.

- *Collision Avoidance (CA)*: CA system operates like a cruise control system to maintain a constant desired speed in the absence of preceding vehicles. If a preceding vehicle appears, the CA system will judge the operation speed is

safe of not, if not, the CA will reduce the throttle and/or apply brake so as to slow the vehicle down, at the same time a warning is provided to the driver.

- *Drive-by-wire*: This technology replaces the traditional mechanical and hydraulic control systems with electronic control systems using electromechanical actuators and human-machine interfaces such as pedal and steering feel emulators. The benefits of applying electronic technology are improved performance, safety and reliability with reduced manufacturing and operating costs. Some sub-systems using "by-wire" technology have already appeared in the new car models.

- *Vehicle navigation system*: It typically uses a GPS navigation device to acquire position data to acquire position data to locate the user on a road in the unit's map database. Using the road database, the unit can give directions to other locations along roads also in its database.

- *Emergency vehicle notification systems*: The in-vehicle eCall is generated either manually by the vehicle occupants or automatically via activation of in-vehicle sensors after an accident. When activated, the in-vehicle eCall device will establish an emergency call carrying both voice and data directly to the nearest emergency point. The voice call enables the vehicle occupant to communicate with the trained eCall operator. At the same time, data about the incident will be sent to the eCall operator receiving the voice call, including time, precise location, the direction the vehicle was traveling, and vehicle identification.

- *Automatic road enforcement*: A traffic enforcement camera system, consisting of a camera and a vehicle-monitoring device, is used to detect and identify vehicles disobeying a speed limit or some other road legal requirement and automatically ticket offenders based on the license plate number. Traffic tickets are sent by mail.

- *Variable speed limits*: Recently some jurisdictions have begun experimenting with variable speed limits that change with road congestion and other factors.

Typically such speed limits only change to decline during poor conditions, rather than being improved in good ones. Initial results indicated savings in journey times, smoother-flowing traffic, and a fall in the number of accidents, so the implementation was made permanent in 1997.

- *Dynamic traffic light sequence*: Dynamic traffic light circumvents or avoids problems that usually arise with systems that use image processing and beam interruption techniques. With appropriate algorithm and database, a dynamic time schedule was worked out for the passage of each column. The simulation showed the dynamic sequence algorithm could adjust itself even with the presence of some extreme cases.

1.2.3. ITS benefits

For automated driving, the development of products and systems is one of the central issues of the long-term technology strategy that aims, stage by stage, to introduce fully automated driving by 2025. With this kind of system on board, drivers will in future be able to decide whether they want to drive themselves or let themselves be driven by automated means. By pre-defining a time-effective, low-consumption or schedule-oriented drive strategy, drivers can choose between traveling according to their own, customized schedule or according to inclination (e.g. fuel-saving), on the basis of comprehensive "real-time floating car data". While awaiting the launch of highly automated vehicles in around 2020, drivers can for the time being devote themselves to other activities than driving for selected driving tasks or sections of journeys (e.g. stop-and-go driving). For example, they can surf the Internet or visual media, or use the infotainment system. This opens up a whole new scope to drivers, transforming driving times from "wasted time" to useful time. At the same time, the automated car, and consequently traffic as a whole, will be substantially safer, as responsibility for driving the vehicle, which currently accounts for the majority of accidents (more than 90%), will be taken out of the driver's hands.

The potential benefits that might acquire from the implementation of ITS could

be summarized as follows. Note that some of the benefits are fairly speculative, the system they would depend upon are not yet in practical application.

- *Road capacity*: Vehicles travel in closely packed platoons can provide a highway capacity that is three times the capacity of a typical highway [168].
- *Safety*: Human error is involved in almost 93% of accidents, and in almost three-quarters of the cases, the human mistake is solely to blame [25]. Only a very small percentage of accidents are caused by vehicle equipment failure or even due to environmental conditions (for example, slippery roads). Since automated systems reduce driver burden and provide driver assistance, it is expected that the employment of well-designed automated systems will certainly lead to improve traffic safety.
- *Weather*: Weather and environmental conditions will impact little on high performance driving. Fog, haze, blowing dirt, low sun angle, rain, snow, darkness, and other conditions affecting driver visibility and thus, safety and traffic flow will no longer impede progress.
- *Mobility*: It offers enhanced mobility for the elderly, and less experienced drivers, etc.
- *Energy consumption and air quality*: Fuel consumption and emissions can be reduced. In the short term, these reductions will be accomplished because vehicles travel in a more efficient manner, lesser traffic congestion occurs.
- *Land use*: ITS help us to use the road efficiently, thus using the land in a efficient way.
- *Travel time saving*: Travel time is saved by reducing congestion in urban highway travel, and permitting higher cruise speed than today's driving.
- *Commercial and transit efficiency*: More efficient commercial operations and transit operations. Commercial trucking can realize better trip reliability and transit operations can be automated, extending the flexibility and convenience of the transit option to increase ridership and service.

1.2.4. Previous researches

The development of ITS in different countries can be divided into two steps [184]. The first step is mainly concerned about transportation information acquisition and processing intellectualization. In the 70s the CACS (Comprehensive Automobile Traffic Control System) was developed in Japan, in which different technological programs were conducted to tackle the large number of traffic deaths and injuries as well as the structural ineffective traffic process [80]. While in Europe, the first formalized transportation telematics program named PROMETHEUS (Programme for European Traffic with Highest Efficiency and Unprecedented Safety) was initiated by governments, companies and universities in 1986 [174]. In 1988, DRIVE (Dedicated Road Infrastructure and Vehicle Environment) program was set up by the European authorities [17]. In the United States, during the late 80s, the team *Mobility 2000* begins the formation of the IVHS (Intelligent Vehicle Highway Systems), which is a forum for consolidating ITS interests and promoting international cooperation [11]. 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 [68].

In the second step, the technologies for vehicle active safety, collision avoidance and intelligent vehicle were rapidly developed. The DEMO' 97 [113] 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 [41]. And the organization takes advantage of information and communication to develop active safety and autonomous driving. The Technische Universit at 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 [173, 108, 132].

In our opinion, the development of ITS is coming to a new stage, where autonomous vehicles, inter-vehicle communication and artificial intelligence will be integrated to bring the data acquisition, data transmission 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.

1.3. Intelligent vehicle

The Automated Highway System (AHS) is one of the most important items among the different topics in the research of ITS. The AHS concept defines a new relationship between vehicles and the highway infrastructure. The fully automated highway systems assume the existence of dedicated highway lanes, where all the vehicles are fully automated, with the steering, brakes and throttle being controlled by a computer [160]. AHS uses communication, sensor and obstacle-detection technologies to recognize and react to external infrastructure conditions. The vehicles and highway cooperate to coordinate vehicle movement, avoid obstacles and improve traffic flow, improving safety and reducing congestion. In brief, the AHS concept combines on-board vehicle intelligence with a range of intelligent technologies installed onto existing highway infrastructure and communication technologies that connect vehicles to highway infrastructure [21].

Implementation of AHS requires autonomous controlled vehicles. Nowadays, vehicles are becoming more and more "intelligent", with increasingly equipping with electromechanical sub-systems that employ sensors, actuators, communication systems and feedback control. Thanks to the advances in solid state electronics, sensors, computer technology and control systems during the last two decades, the required technologies to create an intelligent transportation system is already available, although still expensive for full implementation. According to Ralph [130], today's cars normally have 25 to 70 ECUs (Electronic Control Unit), which perform the monitoring and controlling tasks. Few people realize, in fact, that today's car has four times the computing power of the first Apollo moon rocket [5].

Intelligent vehicles are important roles in ITS, which are motivated by three de-

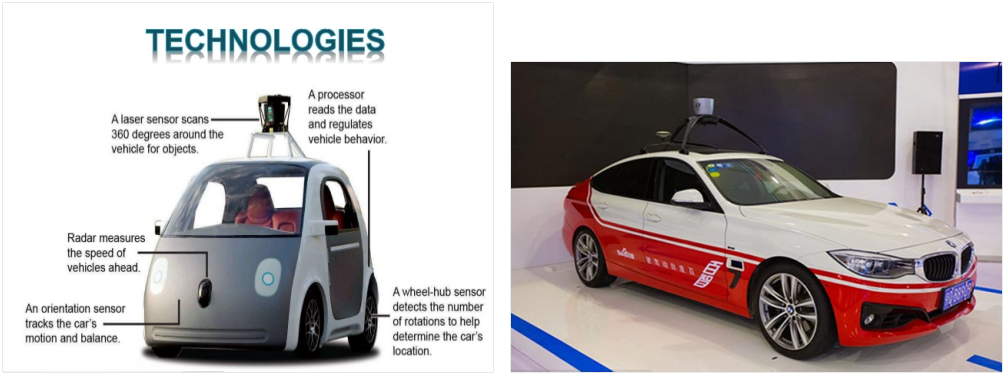
sires: improved road safety, relieved traffic congestion and comfort driver experience [150]. The intelligent vehicles strive to achieve more efficient vehicle operation either by assisting the driver (via advisories or warnings) or by taking complete control of vehicle [9].

Figure 1.7 – *Stanley at Grand Challenge 2005*

Since 2003, Defense Advanced Research Projects Agency (DARPA) of USA founded a prize competition "Grand Challenge" to encourage the development of technologies needed to create the first fully autonomous ground vehicles. The Challenge required autonomous 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. 1.7) became the winner with a result of 6 h 53 min [159, 138]. 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 [162, 99]. These vehicles were regarded as the initial prototype of Google self-driving cars.

In 2010, a project is sponsored by the European Research Council: 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 [15].



(a) *Google's self-driving car* (b) *Baidu's self-driving car*

Figure 1.8 – *Self-driving vehicles*

For vehicle manufacturers, 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 of intelligent vehicles [50] (see in Figure. 1.8a). 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.

Note 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 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.

In China, the company Baidu announced its autonomous vehicle has success-

fully navigated a complicated route through Beijing [31]. The car (see in Figure. 1.8b) 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 vehicles can not be realized without automobile manufacturers. Some of them have launched their own self-driving projects targeting different scenarios [20], such as "Drive Me" of Volvo [197], "Buddy" of Audi [30], Tesla [79] etc. These prototypes are still at test stage, but it is a necessary step of self-driving car development.

Autonomous vehicles are considered to be capable to make better use of road capacity, therefore cars would drive closer to each other. They would react faster than humans to avoid accidents, potentially saving thousands of lives. Moreover, autonomous vehicles 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, such as 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 [157].

1.4. Adaptive Cruise Control

1.4.1. Evolution: from autonomous to cooperative

As mentioned previously, for decades, researchers are trying to develop ITS in order to obtain a safer and more efficient transport system. In vehicle terms, Advanced Driver-Assistant Systems (ADAS) has been 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) [163], Automatic Parking [182], Lane Departure

Warning [28], Lane Change Assistance [100], Blind Spot Monitor [84], etc. Although the objective of ADAS is not to completely replace people in driving, it is able to 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.

Among all ADAS, one of the most important is adaptive cruise control (ACC), which is actually available in a wide range of commercial passenger vehicles. ACC systems are an extension of cruise control (CC) systems. CC is able to maintain vehicle's velocity to a decided value, and the driver does not have to use the pedals, therefore the driver 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 distance, it works as the same as a conventional CC system; else, it utilities 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 [181]. 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 $0.5g$ [128]. 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.

With the development of inter-vehicle communication technologies and the international standard of DSRC [96, 66], 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 [183, 86, 32].

1.4.2. Development of ACC

The notion "ACC" is firstly proposed by [16] within the program PROMETHEUS [174] initiated in 1986 in Europe. Currently, a large proportion of the work in this program 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 [181]. In 1986, the California Department of Transportation and the Institute of Transportation Studies at the University of California Berkeley initiated the state-wide program called PATH [145] to study the use of automation in vehicle-highway systems. Then the program was extended in national scope named as Mobility 2000 [41], 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 [39] from 1996 to 1997, in which 108 volunteers drove 10 ACC-equipped vehicles to determine the safety effects and user-acceptance of ACC systems.

The 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" [116] 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 [89]. Some nonlinear spacing policies [170, 196] have been proposed to improve traffic flow stability, which are called constant stability spacing policies. In order to improve the user-acceptance rate, a drive-adaptive range policy ([54] is proposed, which is called the constant acceptance spacing policy. Considering feasibility, stability, safety, capacity and reliability [154], 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, which is composed of an upper level controller and a lower level controller [128]. The upper level controller determines the desired accelera-

tion or velocity. The lower level controller determines the throttle and/or brake to track the desired accelerations and returns the fault messages to the upper level controller.

The ACC controller should be designed to meet two 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.

1.4.3. Related work in CACC

By adding V2V communications, CACC is a extent version, 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 [164].

The notion of 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 [158, 126]. The objective of the AHS demonstration was a proof-of-concept of an AHS architecture that enhanced highway capacity and safety. In creased capacity was achieved by organizing the movement of vehicles in closely spaced platoons. Autonomous vehicles had actuated-steering, braking and throttle that were controlled by the on-board computer. Safety was improved 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. The 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[191, 192]. 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.

The influence on capacity of increasing market penetration of ACC and CACC vehicles, relative to fully-manually driven vehicles, was examined by using microscopic-traffic simulation [167, 164]. 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 launched in order to reduce a truck driver's workload by developing truck-platooning capacity [13]. A truck can automatically follow any other vehicle with a safe following distance using ACC and a lane-keeping system. Besides, 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.

Traffic simulation in virtual reality system plays an important part in the research of microscopic traffic behavior[97, 88, 187]. In 2014, Yu focuses on the modeling and simulation of microscopic traffic behavior in virtual reality system using multi-agent technology, a hierarchical modular modeling methodology and distributed simulation. Besides, the dynamic features of the real world have been considered in the simulation system in order to improve the microscopic traffic analysis [188]. [189] focuses on the modeling and simulation of the overtaking behavior in virtual reality traffic simulation system involving environment information. A de-

centralized CACC algorithm using V2X for vehicles in the vicinity of intersections is proposed in [85]. This algorithm is designed to improve the throughput of intersection by reorganizing the vehicle platoons around it, in consideration of safety, fuel consumption, speed limit, heterogeneous features of vehicles, and passenger comfort.

Figure 1.9 – *Vehicle platoon in GCDC 2011*

In 2011, the Netherlands Organization for Applied Scientific Researche (TNO), together with the Dutch High Tech Automotive Systems innovation programme (HTAS) organized the Grand Cooperative Driving Challenge (GCDC) [118, 45, 73, 53, 165]. The 2011 GCDC mainly focused on CACC. Nine international teams participated in the challenge (see Figure 1.9), 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.

1.5. Vehicle Ad hoc networks

Individual autonomous vehicles can not represent the whole intelligent vehicle system. The ITS emphasis on the interaction with other vehicles and also the environments such as pedestrian, obstacles, traffic lights in order to exchange these information in ITS all over the world. Dedicated Short-Range Communications (DSRC) provide communications between a vehicle and the roadside in specific locations, for example toll plazas. They may then be used to support specific Intelligent Transport System applications such as Electronic Fee Collection. The standards of Dedicated Short Range Communications (DSRC) technology have been formulated for use in the V2V and V2I communication. DSRC is a kind of one-way or two-way short-range multi-media wireless communication. Based on common communication protocols like IEEE802.11/3G/LTE, DSRC tends be a modified version specifically designed for high speed automotive use. The mainstream of DSRC standards systems are TC278 formulated by CEN (European Committee for Standardization) and TC204 formulated by ISO(International Organization for Standards). Other standardization organizations such as European Telecommunications Standards Institute (ETSI) and Japanese Association of Radio Industries and Businesses (ARIB) have also been involved in the process of formulating DSRC standards. DSRC systems are used in the majority of European Union countries, but these systems are currently not totally compatible. Therefore, standardization is essential in order to ensure pan-European interoperability, particularly for applications such as electronic fee collection, for which the European imposes a need for interoperability of systems. Standardization will also assist with the provision and promotion of additional services using DSRC, and help ensure compatibility and interoperability within a multi-vendor environment. Cooperation and harmonization efforts among government and standards organizations have been made for global utilization.[71] As intelligence vehicle is becoming an important method to decrease the rate of traffic accidents and relive the urban traffic rush, this work becomes important for the interpretability of systems and globalization of ITS. We can easily foresee the development track of the ITS.

DSRC tackles two main tasks: V2V communication and V2I communication. V2V communications carry out through a MANET (mobile ad hoc network), in which the word "ad hoc" comes from Latin and it means "for this purpose" and MANET is a self-configuring infrastructureless network of mobile devices connected by wireless. But the V2V network is still a little different from ad hoc and cellular systems in resource availability and mobility characteristics. Therefore, adopting existing wireless networking solutions to this environment may result in low performance in delay, throughput, and fairness. The vehicle-to-infrastructure communication transfers information between vehicles and the immobile infrastructures. The protocols may be also different from V2V networks because a rush traffic may cause a concentration of the information. The V2I network will support high throughput, low delay, and fair access to available resources.

Originally designed for ETC (Electronic toll collection) system, DSRC technology has been developed and applied in many other typical fields, such as Cooperative Adaptive Cruise Control, Cooperative Forward Collision Warning, Emergency warning, Advanced Driver Assistance Systems, Vehicle safety inspection, Electronic parking payments.

Although the DSRC standardization is in process, a number of institutes and companies did some early researches on the DSRC applications with well developed short range communication systems such as Bluetooth[61, 58, 59, 48], Zigbee[33, 34] and WiFi[98, 40, 57, 74], because they are off-the-shelf commercially ready solutions.

- Bluetooth. Bluetooth network forms a Piconet with one master and a collection of slaves is called. There can only be one master and up to seven active slaves in a single Piconet. The slaves only have a direct link to the master, and not with each other. Multiple Piconets can be joined together to form a Scatternet. A frequency-hopping channel based on the address of the master defines each piconet. The master's transmissions may be either point-to-point or point-to-multipoint. Also, besides in an active mode, a slave device can be in the parked or standby modes so as to reduce power consumptions. The effective range of the original version of Bluetooth is less than 10 meters. But

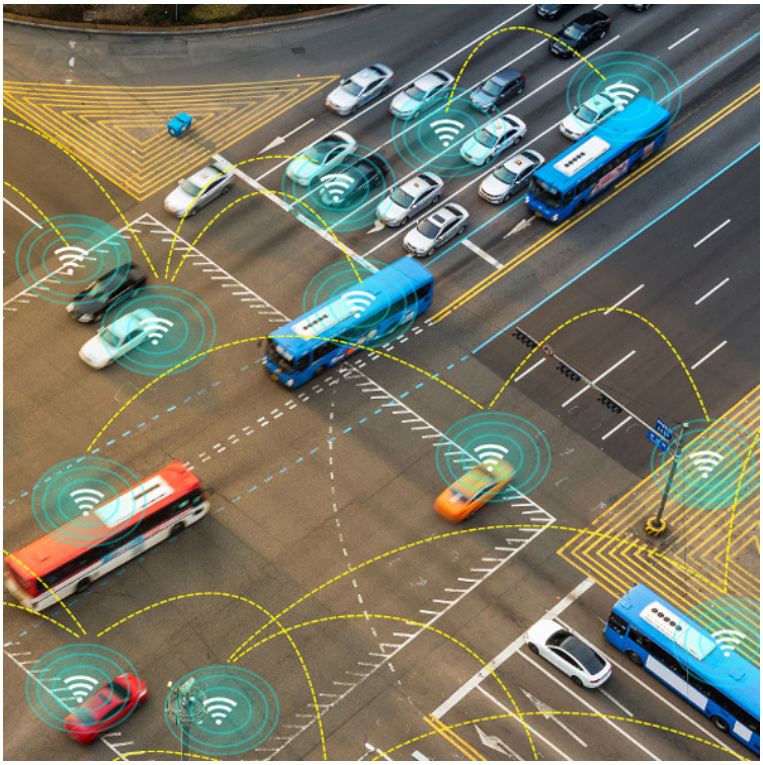


Figure 1.10 – *DSRC demonstration*

the later version promoted the function of Bluetooth and allows the device to transmit data at the distance up to 100 meters, the data rate is also been promoted.

- ZigBee. ZigBee is another short range wireless communication protocol designed specifically for individual remote controls. ZigBee was designed costless and "sleeping" strategy leads to low power consumption so that a Zigbee device would work for over years without changing the battery. But the transmission has a lower data rate comparing to Bluetooth. Zigbee system is widely used in the industrial environments which have lower requirement on the data rate.

- Wireless fidelity (Wi-Fi). A series of standards for wireless local area networks (WLAN). Wi-Fi is a wireless version of a common wired Ethernet network, and requires configuration to set up shared resources, transmit data. Wi-Fi uses the same radio frequencies as Bluetooth, but with higher power, resulting

in higher data rates. As Wi-Fi is connected to the World Wide Web, it is easy to exchange information with the database long distance away. It is more complicate in installing the infrastructures and configuration, so using in V2V communication maybe less advantaged compare to the former two systems. But in the V2I communication which has the requirement of data rate and node tolerance, the Wi-Fi would be more suitable.

In[34] a simulation of the performance of V2V communication with the uses of AODV routing protocol is presented. Two different wireless protocols of IEEE802.11 (WLAN) and IEEE802.15.1(Zigbee) were compared under the same condition. The result showed that when the number of vehicle nodes increases, the transmission in WLAN yields the higher successful rate and shorter delay than that in Zigbee. In addition, when the number of vehicle nodes increases, WLAN yields less number of hops and tends to be constant while the average number of hops in Zigbee network keeps increasing as the network density increases. From the comparison of these short range communication systems, we can see they all have advantages and disadvantages in applying in ITS. [48] focused on issues relating to ad-hoc network formation in a mobile environment using Bluetooth technology, the author found Bluetooth is a good choice for inter-vehicle communication because the nodes (vehicles) are constantly moving in and out of range of the master node and local piconets. Though Bluetooth provides a strong foundation in forming ad-hoc networks for mobile vehicles, problems like large connection time and topological changing for the mobile nodes have been showed too. While for IEEE802.11 connection, the result of [74] showed the Wi-Fi protocol have also the problems in routing overheads in the environment of long distance and high velocities. Besides, IEEE802.11 (Wi-Fi) standard were designed to provide a replacement for wired infrastructure networks, so it is highly infrastructure-depended. However, the short-range communication system dedicated for ITS should have high flexibility with respect to asymmetric data flows, allows the communication over large distances and supports high velocities.

Table 1.1 – *Comparison of the short range communication systems*

Standard	Bluetooth	Zigbee	Wi-Fi
IEEE specification	802.15.1	802.15.4	802.11a/b/g
Maximum data rate	24Mb/s	250kb/s	54Mb/s
Transmission range	100m(class1)	100m	300m
Maximum number of nodes	7(single piconet)	65536+	2007

1.6. Machine Learning

Autonomous vehicles cannot always be programmed to execute predefined actions because one does not always know in advance the unpredicted situations that the vehicle might encounter. Today, however, most vehicles used in the researches are pre-programmed and require a well-defined and controlled environment. Reprogramming is often a costly process requiring an expert. By enabling vehicles to learn tasks either through autonomous self-exploration or through guidance from a human teacher, task reprogramming can be simplified. Vehicles can be regarded like intelligent robots that are able to learn.

Recent researches has shown a drift toward artificial intelligence approaches to improve the robot autonomous ability based on accumulated experiences, and artificial intelligence methods can be computationally less expensive than classical ones. Machine learning approaches are often applied, to each the burden on system engineers. Learning therefore has become a central topic in modern robotics research.

Learning consists of a multitude of machine learning approaches, particularly reinforcement learning, imitation learning, inverse reinforcement learning, and regression methods, that have been adapted sufficiently to domain so that they allow learning in complex robot systems such as helicopters, flapping-wing flight, legged robots, anthropomorphic arms and humanoid robots. While classical artificial intelligence-based robotics approaches have often attempted to manually generate a set of rules and models that allows the robot systems to sense and act in the real-world, robot learning centers around the idea that it is unlikely that we can foresee all interesting real-world situations sufficiently accurate.

While robot learning covers a wide range of fields, from learning to perceive, to plan, to make decisions, etc., we focus our work on applying learning approaches to intelligent vehicles. In general, learning control refers to the process of acquiring a particular control system and a particular task by trial and error [141]. Reinforcement Learning (RL) and learning from Demonstration (LfD) are mentioned as two popular families of algorithms for learning policies for sequential decision problems [24].

- Reinforcement learning algorithms solve sequential decision problems posed as Markov Decision Processes (MDPs), learning a policy by letting the agent explore the effects of different actions in different situations while trying to maximize a sparse reward signal. RL has been successfully applied to a variety of scenarios.

- Learning from demonstration is an approach to agent learning that takes as input demonstrations from a human in order to build action or task models. There are a broad range of approaches that fall under the umbrella of LfD research[6]. These demonstrations are typically represented as state-action tuples, and the LfD algorithm learns a policy mapping from states (input) to actions (output) based on the examples seen in the demonstrations. Inverse reinforcement learning (IRL), as one important branch of LfD methods, addresses the problem of estimating the reward function of an agent acting in a dynamic environment.

Another approach is to provide a mapping from sensory inputs to actions that statistically capture the key behavioral objectives without needing a model or detailed domain knowledge [26]. Such methods are well-suited to domains where the tools available to learn from past experience and adapt to emergent conditions are limited.

With the advent of increasingly efficient learning methods, one can observe a growing number of successful applications in this area, such as autonomous helicopter control [106, 2, 1], self-driving car [159, 99, 162], autonomous underwater

vehicles (AUVs) control [18], mobile robot navigation [69], robot soccer control [135].

Recently, several interesting applications have appeared. [81] worked with a Willow Garage Personal Robot 2 (PR2), named Berkeley Robot for the Elimination of Tedious Tasks (BRETT), and empowered BRETT has acquired the ability to learn to perform various tasks on its own via trial and error, without pre-programmed details about its surroundings. Those tasks include assembling a wheel part onto a toy airplane, stacking a Lego block, and screwing a cap on a water bottle. [102] used imitation and reinforcement learning techniques to enable a Barrett WAM arm to learn successful hitting movements in table tennis. [78] taught a robot to flip a pancake. Other successful robot learning applications also include [131, 77, 179, 180, 176, 175, 178].

1.7. Conclusion

This chapter gives a detailed introduction to intelligent road transportation systems. Firstly, the background of the current traffic situation and problems were introduced. 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, we briefly introduced ITS, AHS and intelligent vehicle, which were considered as the most promising solutions to the traffic problems.

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

Fourthly, a key aspect in developing ITS: the communication is introduced. Specific to road transportation systems, it is V2X communications, including V2V com-

munication and V2I communication. By enabling communications among these agents, the 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.

Finally, the technology of machine learning is introduced, which can be applied on intelligent vehicles.

Safety and efficiency are two most demanded features of ITS. Therefore, in this thesis, we focus on an Stop-and-Go scenario with different applications designed in order to improve the throughput while guarantee safety and stability by controlling the actions of vehicle platoons or individual vehicles.

Chapter 2

String stability and Markov decision process

Sommaire

2.1. String stability

2.1.1. 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, congestion caused by incorrect driving behaviors, accidents, improper signal timing 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 [120]. However, the intelligence of individual vehicles does not represent the intelligence of the whole transportation system.

A completely different concept proposed by the California PATH Program, is vehicle "platoon", where the vehicles travel together with a close separation [127]. String stability is an important goal to be achieved in (C)ACC system design [95]. 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 [117]. As opposed to conventional stability notions for dynamical systems, which are basically concerned with the evolution of system states over time, string stability focuses on the propagation of system responses along a cascade of systems. Several approaches exist regarding string stability, as reviewed below.

2.1.2. Previous research

Probably the most formal approach is based on Lyapunov stability, of which [143] provides an early description, comprehensively formalized in [152]. In this ap-

proach, the notion of Lyapunov stability is employed, focusing on initial condition perturbations. Consequently, string stability is interpreted as asymptotic stability of interconnected systems [29]. Recently, new results appeared in [75], regarding a one-vehicle lookahead topology in a homogeneous vehicle platoon. In this brief, the response to an initial condition perturbation of a single vehicle in the platoon is considered, thereby conserving the disturbance-propagation idea behind string stability. The drawback of this approach, however, is that only this special case is regarded, ignoring the effect of initial condition perturbations of other vehicles in the platoon, as well as the effect of external disturbances to the interconnected system. Consequently, the practical relevance of this approach is limited, since external disturbances, such as velocity variations of the first vehicle in a platoon, are of utmost importance in practice. The perspective of infinite-length strings of interconnected systems [27] also gave rise to a notion of string stability, described in [93] in the context of a centralized control scheme and in [23] for a decentralized controller. Various applications regarding interconnected systems are reported in [8] and [83], whereas [7] and [27] provide extensive analyzes of the system properties. In this approach, the system model is formulated in the state space and subsequently transformed using the bilateral Z-transform. The Z-transform is executed over the vehicle index instead of over (discrete) time, resulting in a model formulated in the "discrete spatial frequency" domain [7], related to the subsystem index, as well as in the continuous-time domain. String stability can then be assessed by inspecting the eigenvalues of the resulting state matrix as a function of the spatial frequency. Unfortunately, the stability properties of finite-length strings, being practically relevant, might not converge to those of infinite-length strings as length increases. This can be understood intuitively by recognizing that in a finite-length platoon, there will always be a first and a last vehicle, whose dynamics may significantly differ from those of the other vehicles in the platoon, depending on the controller topology. Consequently, the infinite-length platoon model does not always serve as a useful paradigm for a finite length platoon as it becomes increasingly long [27].

The most important macroscopic behaviors of ACC vehicles is string stability. Such stability has been first recognized by D. Swaroop [155]. The string stability

of a string of vehicles refers to a property in which spacing errors are guaranteed not to amplify as they propagate towards the tail of the string [154, 140]. This property ensures that any spacing error present at the head of the string does not amplify into a large error at the tail of the string. A general method to evaluate string stability is to examine the transfer function from the spacing error of the proceeding vehicle to that of the following vehicle. If the infinite norm of this transfer function is less than 1, string stability is ensured [153, 169].

For an interconnected system, such as a platoon of automated vehicles, stability of each component system itself is not sufficient to guarantee a certain level of performance, such as the boundedness of the spacing errors for all the vehicles. This is reasonable because our research object is a string of vehicles instead of only one vehicle. Therefore, besides the individual stability of each vehicle, another stability criterion known as the string stability is also required [62, 125].

Finally, a performance-oriented approach for string stability is frequently adopted, since this appears to directly offer tools for controller design for linear cascaded systems. This approach is employed for the control of a vehicle platoon with and without lead vehicle information in [144], whereas [129] and [104] apply inter-vehicle communication to obtain information of the preceding vehicle. In [149], a decentralized optimal controller is designed by decoupling the interconnected systems using the so-called inclusion principle, and in [72], optimal decentralized control is pursued by means of nonidentical controllers. Furthermore, [94] extensively investigated the limitations on performance, whereas in [47], a controller design methodology was presented. Finally, in [19] the performance-oriented approach is adopted to investigate a warning system for preventing head-tail collisions in mixed traffic.

2.1.2.1. Definition of string stability

In the performance-oriented approach, string stability is characterized by the amplification in upstream direction of either distance error, velocity, or acceleration, the specific choice depending on the design requirements at hand.

A simple scenario can be used to explain the string stability, illustrated in Fig-

ure. 2.1. In this figure, a platoon of five vehicles, from left to right, is taking a brake action. The leading vehicle is denoted as 1st while the last vehicle is denoted as 5th. In the figure above, a speed vs. time coordinate graph for each of the five vehicles is shown. As time goes by, the leading vehicle decelerates linearly and we can see different response of the following vehicles in the platoon depending on whether the platoon is string stable or not. In Figure. 2.1(a), the vehicle platoon is string stable: the disturbance of the brake action of the leading vehicle is not amplified through the following vehicles and the deceleration of following vehicles is smooth with slight fluctuation of the speed. In Figure. 2.1(b), the platoon is considered not string stable (string unstable): the following vehicles decelerate even more than the leading vehicle. Though finally, the velocities of the following vehicles approach the leading vehicle's velocity, their response fluctuate significantly. Therefore, when velocity of vehicles fluctuates, the distance between consecutive vehicles is also suffering from great fluctuation. As a result, rear-end collisions between vehicles are more likely to taken place.

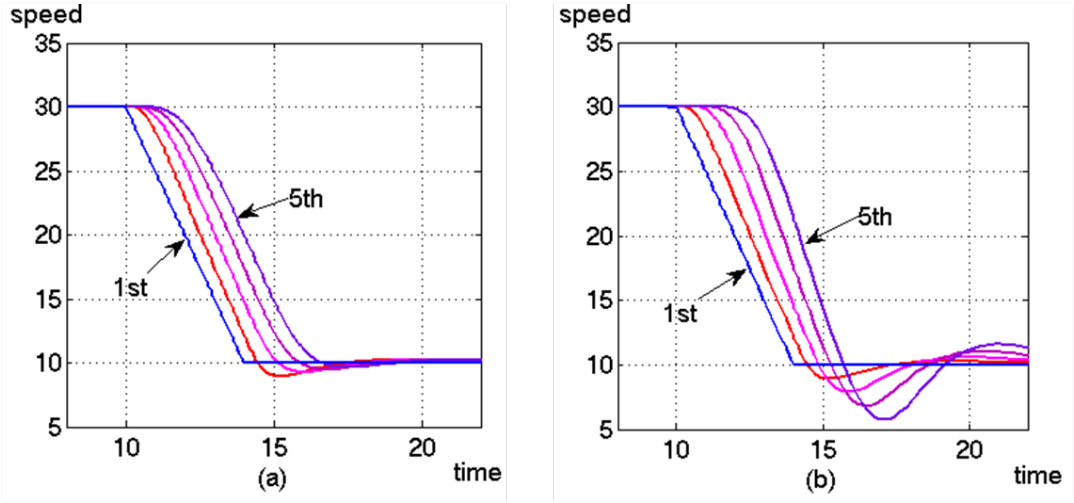


Figure 2.1 – *String stability illustration: (a) stable (b) unstable*

The more formalized and generalized definition of string stability was given by Swaroop [153]. The mathematical definitions for *string stability*, a *symptotically stability*, and l_p *string stability* were made. We use the definitions proposed by Swaroop [152, 153]. At first, we use the following notations: $\|f_i(\cdot)\|_\infty$ denotes

$\sup_{t\geq 0} \mid f_i(t) \mid$, and $\|f_i(0)\|_\infty$ denotes $\sup_i |f_i(0)|$. For all $p < \infty$, $\|f_i(\cdot)\|_p$ denotes $(\int_0^\infty |f_i(t)|^p dt)^{\frac{1}{p}}$ and $\|f_i(0)\|_p$ denotes $(\sum_1^\infty |f_i(0)|^p)^{\frac{1}{p}}$.

Consider an interconnected system:

$$\dot{x}_i = f(x_i,\ x_{i-1},\ \cdots,\ x_{i-r+1}) \tag{2.1}$$

where $i \in \mathbb{N}$, $x_{i-j} \equiv 0\ \forall i \leq j$, $x \in \mathbb{R}^n$, $f : \underbrace{\mathbb{R}^n \times \cdots \times \mathbb{R}^n}_{r\ times} \to \mathbb{R}^n$ and $f(0, \cdots, 0) = 0$.

Defnition 1 (***String stability***). The origin $x_i = 0$, $i \in \mathbb{N}$ of (2.1) is string stable, if given any $\epsilon > 0$, there exist a $\delta > 0$ such that :

$$\|x_i(0)\|_\infty < \delta \Rightarrow \sup_i \|x_i(\cdot)\|_\infty < \epsilon$$

Defnition 2 (***Asymptotically (exponential) stability***). The origin $x_i = 0$, $i \in \mathbb{N}$ of (2.1) is asymptotically (exponentially) string stable if it is string stable and $x_i(t) \to 0$ asymptotically (exponentially) for all $i \in \mathbb{N}$.

A more general definition of string stability is given in follow:

Defnition 3 (l_p ***String stability***). The origin $x_i = 0$, $i \in \mathbb{N}$ of (2.1) is l_p string stable if for any $\epsilon > 0$, there exist a $\delta > 0$ such that :

$$\|x_i(0)\|_p < \delta \Rightarrow \sup_i \left(\sum_1^\infty |x_i(t)|^p \right)^{\frac{1}{p}} < \epsilon$$

It is clear that Definition 1 can be obtained as l_∞ string stability of Definition 3. The generalized string stability implies uniform boundedness of the system states if the initial conditions are uniformly bounded.

2.1.2.2. String stability in vehicle following system

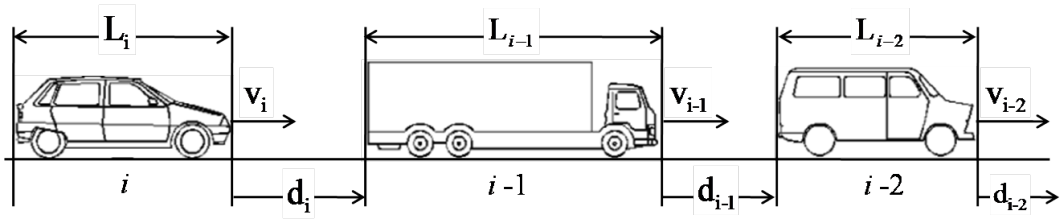


Figure 2.2 – *Vehicle platoon illustration*

In the case of vehicle following system, such as a vehicle platoon as shown in Fig. 2.2, for the ith vehicle, s_i is the location measured from an inertial reference, as shown in the same figure. We define the spacing error for the ith vehicle as:

$$e_i = s_i - s_{i-1} + d_r, i \tag{2.2}$$

where d_r, i is the desired spacing measured from vehicle $i-1$ to i, and it includes the preceding vehicle's length $L_i - 1$. A sufficient condition for string stability is that [152, 153]:

$$\|e_i\|_\infty \leq \|e_{i-1}\|_\infty \tag{2.3}$$

Let the signal of interest be denoted by z_i for ith vehicle, and let $\Gamma_i(j\omega)$ denote the frequency response function describing the relation between the scalar output z_{i-1} of a preceding vehicle $i-1$ and the scalar output z_i of the follower vehicle i. Then the interconnected system is considered string stable if

$$\sup_\omega |\Gamma_i(j\omega)| \leq 1, 2 \leq i \leq n \tag{2.4}$$

where n is the string length; the supremum of $\Gamma_i(j\omega)$ equals the scalar version of the norm. Since the H_∞ norm is induced by the L_2 norms of the respective signals, this approach requires the L_2 norm $||y_i(t)||_{L_2}$ to be non-increasing for increasing index i. Because of its convenient mathematical properties, the L_2 gain is mostly adopted; nevertheless, approaches that employ the induced L_∞ norm are also reported [38]. Regardless of the specific norm that is employed, the major limitation of the performance oriented approach is that only linear systems are considered, usually without considering the effect of nonzero initial conditions.

2.2. Markov Decision Processes

In recent years, a fast development of using machine learning techniques onto robot control problems is happening. Machine learning enables an agent to learn from example data or past experience to solve a given problem. In supervised learning, the learner is provided an explicit target for every single input, that is, the envi-

ronment tells the learner what its response should. In contrast, in reinforcement learning, only partial feedback is given to the learner about the learner's decisions. Therefore, under the framework of RL, the learner is a decision-making agent that takes actions in an environment and receives reward (or penalty) for its actions in trying to solve a problem. After a set of trial-and error runs, it should learn the best policy, which is the sequence of actions that maximizes the total reward [151].

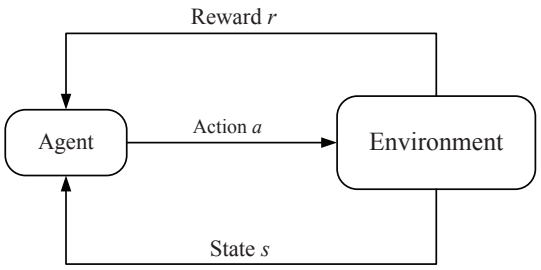


Figure 2.3 – *The mechanism of interaction between a learning agent and its environment in reinforcement learning*

Reinforcement learning is generally operated in a setting of interaction, shown in Figure 2.3: the learning agent interacts with an initially unknown environment, and receives a representation of the state and an immediate reward as the feedback. It then calculates an action, and subsequently undertakes it. This action causes the environment to transit into a new state. The agent receives the new representation and the corresponding reward, and the whole process repeats.

The environment in RL is generally formulated as a Markov Decision Process (MDP), and the goal is to learn to a control strategy so as to maximize the total reward which represents a long-term objective. In this chapter, we introduces the structural background of Markov Decision Process and reinforcement learning in robotics.

A *Markov Decision Process* describes a sequential decision-making problem in which an agent must choose the sequence of actions that maximizes some reward-based optimization criterion [123, 151]. Formally, an MDP is a tuple $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma\}$, where

- $\mathcal{S} = \{s_1, \ldots, s_N\}$ is a finite set of N *states* that represents the dynamic environment,

- $\mathcal{A} = \{a_1, \ldots, a_k\}$ is a set of k *actions* that could be executed by an agent,
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \longmapsto [0, 1]$ is a *transition probability function*, or *transition model*, where $\mathcal{T}(s, a, s')$ stands for the state transition probability upon applying action $a \in \mathcal{A}$ in state $s \in \mathcal{S}$ leading to state in state $s' \in \mathcal{S}$, i.e. $\mathcal{T}(s, a, s') = P(s' \mid s, a)$,
- $r : \mathcal{S} \times \mathcal{A} \longmapsto \mathbb{R}$ is a *reward function* with absolute value bounded by R_{max}; $r(s, a)$ denotes the immediate reward incurred when action $a \in \mathcal{A}$ is executed in state $s \in \mathcal{S}$,
- $\gamma \in [0, 1)$ is a *discount factor*.

Given an MDP $\mathcal{M}$, the agent-environment interaction in Figure 2.3 works as follows: let $t \in \mathbb{N}$ denote the current time, let $S_t \in \mathcal{S}$ and $A_t \in \mathcal{A}$ denote the random state of the environment and the action chosen by the agent at time t, respectively. Once the action is selected, it is sent to the system, which makes a transition:

$$(S_{t+1}, R_{t+1}) \sim P(\cdot \mid S_t, A_t). \tag{2.5}$$

In particular, S_{t+1} is random and $P(S_{t+1} = s' \mid S_t = s, A_t = a) = \mathcal{T}(s, a, s')$ holds true for any $s, s' \in \mathcal{S}, a \in \mathcal{A}$. Furthermore, $\mathbb{E}[R_{t+1} \mid S_t, A_t] = r(S_t, A_t)$. The agent then observes the next state S_{t+1} and reward R_{t+1}, chooses a new action $A_{t+1} \in \mathcal{A}$ and the process is repeated.

The Markovian assumption [151] implies that the sequence of state-action pairs specifies the transition model $\mathcal{T}$:

$$P(S_{t+1} \mid S_t, A_t, \cdots, S_0, A_0) = P(S_{t+1} \mid S_t, A_t). \tag{2.6}$$

State transitions can be deterministic or stochastic. In the deterministic case, taking a given action in a given state always results in the same next state; while in the stochastic case, the next state is a random variable.

The goal of the learning agent is to figure out a theory of choosing the actions so as to maximize the expected total discounted reward:

$$\mathcal{R} = \sum_{t=0}^{\infty} \gamma^t R_{t+1}. \tag{2.7}$$

If $\gamma < 1$ then the rewards received far in the future are exponentially less worthy than those received at the first stage.

2.3. Policies and Value Functions

The action selection of the agent is based on a special function called *policy*. A policy is defined as a mapping $\pi : \mathcal{S} \times \mathcal{A} \longmapsto [0, 1]$ that assigns to each $s \in \mathcal{S}$ a distribution $\pi(s, \cdot)$ over $\mathcal{A}$, satisfying $\sum_{a \in \mathcal{A}} \pi(a \mid s) = 1, \forall s \in \mathcal{S}$.

A *deterministic stationary policy* is the case that for all $s \in \mathcal{S}$, $\pi(\cdot \mid s)$ is concentrated on a single action, i.e. at any time $t \in \mathbb{N}$, $A_t = \pi(S_t)$. A *stochastic stationary policy* is a function that maps each state into a probability distribution over the different possible actions, i.e., $A_t \sim \pi(\cdot \mid S_t)$. The class of all stochastic stationary policies is denoted by Π.

Application of a policy works in the following way. First, a start state S_0 is generated. Then, the policy π suggests the action $A_0 = \pi(S_0)$ and this action is performed. Based on the transition function $\mathcal{T}$ and reward function r, a transition is made to state X_1, with a probability $\mathcal{T}(S_0, A_0, S_1)$ and a reward $R_1 = r(X_0, A_0, X_1)$ is received. This process continues, producing a sequence $S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, ...$, as shown in Figure 2.4.

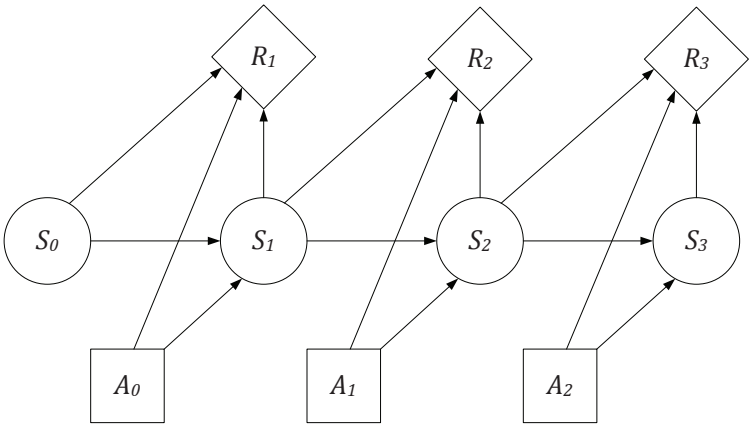


Figure 2.4 – *Decision network of a finite MDP*

Value functions are functions of states (or of state-action pairs) that estimate how good it is for the agent to be in a given state (or how good it is to perform a given action in a given state). The notion of "how good" here is defined in terms of future rewards that can be expected, or, to be precise, in terms of expected return. Of course the rewards the agent can expect to receive in the future depend on what actions it will take. Accordingly, value functions are defined with respect to particular policies [151].

Given a a policy π, the value function is defined as a function $V^\pi : \mathcal{S} \longmapsto \mathbb{R}$ that associates to each state the expected sum of rewards that the agent will receive if it starts executing policy π from that state:

$$V^\pi(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(S_t, A_t) \,|\, S_0 = s \right], \qquad \forall s \in \mathcal{S}. \tag{2.8}$$

S_t is the random variable representing the state at time t, A_t is the random variable corresponding to the action taken at that time instant and is such that $P(A_t = a \,|\, S_t = x) = \pi(x, a)$. $(S_t, A_t)_{t \geq 0}$ is the sequence of random state-action pairs generated by executing the policy π.

The value function of a stationary policy can also be recursively defined as:

$$\begin{aligned} V^\pi(s) &= \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(S_t, A_t) \,|\, S_0 = s \right] \\ &= \mathbb{E}_\pi \left[r(S_0, A_0) + \sum_{t=1}^{\infty} \gamma^t r(S_t, A_t) \,|\, S_0 = s \right] \\ &= r(s, \pi(s)) + \mathbb{E}_\pi \left[\sum_{t=1}^{\infty} \gamma^t r(S_t, A_t) \,|\, S_0 = s \right] \\ &= r(s, \pi(s)) + \gamma \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(S_t, A_t) \,|\, S_0 \sim \mathcal{T}(s, \pi(s), \cdot) \right] \\ &= r(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s, \pi(s), s') V^\pi(s'), \end{aligned} \tag{2.9}$$

where $\pi(s)$ is the action associated to state s.

If the uncertainty of a stochastic policy $\pi(s)$ is taken into account, $V^\pi(s)$ can also be specifically written as:

$$V^{\pi}(s) = \sum_{a \in \mathcal{A}(s)} \pi(s,a) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V^{\pi}(s') \right). \tag{2.10}$$

Similarly, the *action-value function* $Q^{\pi} : \mathcal{S} \times \mathcal{A} \longmapsto \mathbb{R}$ underlying a policy π is defined as

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(S_t, A_t) \,|\, S_0 = s, A_0 = a \right], \tag{2.11}$$

where S_t is distributed according to $\pi(S_t, \cdot)$ for all $t > 0$. Finally, we defined the *advantage function* associated with π as

$$A^{\pi} = Q^{\pi}(s,a) - V^{\pi}(s). \tag{2.12}$$

A policy that maximizes the expected total discounted reward over all states is called an *optimal policy*, denoted π^*. For any finite MDP, there is at least one optimal policy.

The *optimal value function* V^* and the *optimal action-value function* Q^* are defined by

$$\begin{aligned} V^*(s) &= \sup_{\pi} V^{\pi}(s), \qquad s \in \mathcal{S}, \\ Q^*(s,a) &= \sup_{\pi} Q^{\pi}(s,a), \quad s \in \mathcal{S}, a \in \mathcal{A}. \end{aligned} \tag{2.13}$$

Moreover, the optimal value- and action-value functions are connected by the following equations:

$$V^*(s) = \sup_{a \in \mathcal{A}} Q^*(s,a), \qquad s \in \mathcal{S}, \tag{2.14}$$

$$Q^*(s,a) = r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s' \,|\, s,a) V^*(s'), \qquad s \in \mathcal{S}, a \in \mathcal{A}. \tag{2.15}$$

It turns out that V^* and Q^* satisfy the so-called Bellman optimality equations [123]. In particular,

$$Q^*(s,a) = r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s' \mid s,a) \max_{b \in \mathcal{A}} Q^*(s',b), \tag{2.16}$$

$$V^*(s) = \max_{a \in \mathcal{A}} r(s,a) + V^*(s'). \tag{2.17}$$

We call a policy that satisfies $\sum_{a \in \mathcal{A}} \pi(a \mid s) Q(s,a) = \max_{a \in \mathcal{A}} Q(s,a)$ at all states $s \in \mathcal{S}$ *greedy* w.r.t. the function Q. It is known that all policies that are greedy w.r.t. Q^* are optimal and all stationary optimal policies can be obtained these way.

Here, we present the following important results concerning MDP [151]:

Theorem 1 (*Bellman Equations*). Let a Markov Decision Problem $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma\}$ and a policy $\pi : \mathcal{S} \times \mathcal{A} \longrightarrow [0,1]$ be given. Then, $\forall s \in \mathcal{S}, a \in \mathcal{A}$, V^π and Q^π satisfy

$$V^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s, \pi(s), s') V^\pi(s'), \tag{2.18}$$

$$Q^\pi(s,a) = r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V^\pi(s'). \tag{2.19}$$

Theorem 2 (*Bellman Optimality*). Let a Markov Decision Problem $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma\}$ and a policy $\pi : \mathcal{S} \times \mathcal{A} \longrightarrow [0,1]$ be given. Then, π is an optimal policy for $\mathcal{M}$ if and only if, $\forall s \in \mathcal{S}$,

$$\pi(s) \in \arg\max_{a \in \mathcal{A}} Q^\pi(s,a). \tag{2.20}$$

The transition probability $\mathcal{T}(s,a,s') = P(s' \mid s,a)$.

2.4. Dynamic Programming: Model-Based Algorithms

Dynamic programming (DP) is a method for calculation of an optimal policy π^* in order to solve a given Markov decision process.

Dynamic programming assumes complete knowledge of the Markov decision process, including the transition dynamics of the environment and the reward function [10]. Therefore, it is classified into *model-based* learning algorithms. On the

contrary are *model-free* learning algorithms, which do not require a perfect model of the environment, and will be introduced them later in this chapter.

Dynamic programming algorithms for solving MDPs can be categorized into one of the two aspects: value iteration (VI) and policy iteration (PI) [151]. Both of these approaches share a common underlying mechanism, the *generalized policy iteration* (GPI) principle [151], depicted in Figure 2.5. This principle consists of two interaction processes. The first step, *policy evaluation*, estimates the utility of the current policy π, that is, it computes the value V^{π}. This step gathers information about the policy for computing the second step, the *policy improvement* step. In this step, the values of the actions are evaluated for every state, in order ot find possible improvements, that is, possibly other actions in particular states that are better than the action the current policy proposes. This step computes an improved policy π' from the current policy π using the information in V^{π}. As long as both processes continue to update all states, the ultimate goal is to converge to the optimal value function and an optimal policy. Figure 2.6 presents a geometric metaphor for convergence of both the value function and the policy in GPI.

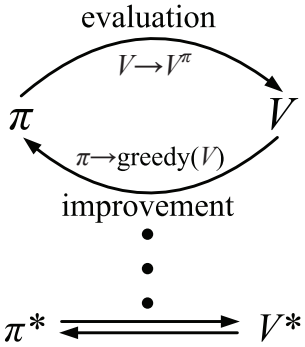


Figure 2.5 – *Interaction of policy evaluation and improvement processes*

2.4.1. Policy Iteration

Policy iteration iterates between the two processes of GPI. This is repeated until converging to an optimal policy. This method is depicted in Algorithm 1.

It consists in starting with a randomly chosen policy π_t and a random initialization of the corresponding value function V_k, for $k = 0$ and $t = 0$ (Steps 1 to 3), and iteratively repeating the *policy evaluation* and the *policy improvement* operations.

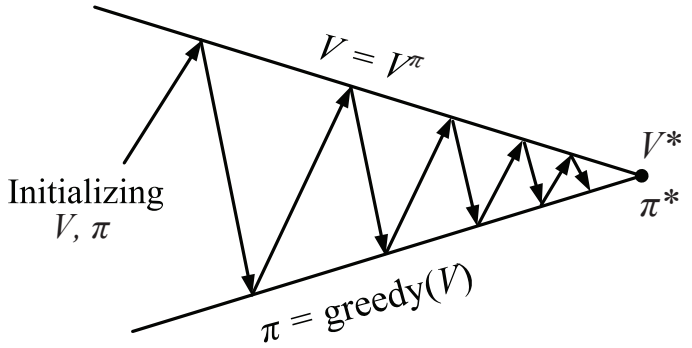


Figure 2.6 – *The convergence of both the value function and the policy to their optimals*

Algorithm 1: Policy Iteration [151]

Require: An MDP model $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma \rangle$;
`/* Initialization */`
$t = 0, k = 0$;
$\forall s \in \mathcal{S}$: Initialize $\pi_t(s)$ with an arbitrary action;
$\forall s \in \mathcal{S}$: Initialize $V_k(s)$ with an arbitrary value;
repeat
`/* Policy evaluation */`
repeat
$\forall s \in \mathcal{S} : V_{k+1}(s) = r(s, \pi_t(s)) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s, \pi_t(s), s') V_k(s')$;
$k \leftarrow k + 1$;
until $\forall s \in \mathcal{S} : |V_k(s) - V_{k-1}(s)| < \epsilon$;
`/* Policy improvement */`
$\forall s \in \mathcal{S} : \pi_{t+1}(s) = \arg\max_{a \in \mathcal{A}} [r(s, a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s, a, s') V_k(s')]$;
$t \leftarrow t + 1$;
until $\pi_t = \pi_{t-1}$;
$\pi^* = \pi_t$;
return An optimal policy π^*.

Policy evaluation (Steps 5 to 8) consists in calculating the action value of policy π_{t+1} by solving the solving the equations (2.19) for all the states $s \in \mathcal{S}$. An efficient iterative way to solve this equation is to initialize the value function of π_{t+1} with the value function V_k of the previous policy, and then repeat the operation:

$$\forall s \in \mathcal{S} : V_{k+1}(s) = r(s, \pi_t(s)) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s, \pi_t(s), s') V_k(s'), \tag{2.21}$$

until $\forall s \in \mathcal{S} : |V_k(s) - V_{k-1}(s)| < \epsilon$, for a predefined error threshold ϵ.

Policy improvement (Steps 9 to 10) consists in finding the greedy policy π_{t+1}

given the value function V_k:

$$\forall s \in \mathcal{S} : \pi_{t+1}(s) = \arg\max_{a \in \mathcal{A}} \left[r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V_k(s') \right]. \tag{2.22}$$

This process stops when $\pi_t = \pi_{t-1}$, in which case π_t is an optimal policy, i.e., $\pi^* = \pi_t$.

In sum, PI generates a direct sequence of alternating policies and value functions:

$$\pi_0 \to V^{\pi_0} \to \pi_1 \to V^{\pi_1} \to \cdots \to \pi^* \to V^* \to \pi^*$$

The policy evaluation processes occur in the transitions of $\pi_t \to V^{\pi_t}$; while the $V^{\pi_t} \to \pi_{t+1}$ conversions are realized by the policy improvement processes.

2.4.2. Value Iteration

One of the drawbacks of policy iteration is that a complete policy evaluation is involved in each iteration. Value iteration consists in overlapping the evaluation and improvement processes.

Instead of completely separating the evaluation and improvement processes, the *value iteration* approach breaks off evaluation after just one iteration. In fact, it immediately blends the policy improvement step into its iterations, thereby purely focusing on estimating directly the value function.

Value iteration, described in Algorithm 2, can be written as a simple backup operation:

$$\forall s \in \mathcal{S} : V_{k+1}(s) = \max_{a \in \mathcal{A}} \left[r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V_k(s') \right]. \tag{2.23}$$

This operation is repeated (Steps 3 to 6) until $\forall s \in \mathcal{S} : |V_k(s) - V_{k-1}(s)| < \epsilon$, in which case the optimal policy is simply the greedy policy with respect to the value function V_k (Step 7).

VI produces the following sequence of value functions:

Algorithm 2: Value Iteration [151]

Require: An MDP model $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma \rangle$;
$k = 0$;
$\forall s \in \mathcal{S}$: Initialize $V_k(s)$ with an arbitrary value;
repeat
$\forall s \in \mathcal{S} : V_{k+1}(s) = \max_{a \in \mathcal{A}} [r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V_k(s')]$;
$k \leftarrow k + 1$;
until $\forall s \in \mathcal{S} : |V_k(s) - V_{k-1}(s)| < \epsilon$;
$\forall s \in \mathcal{S} : \pi^*(s) = \arg\max_{a \in \mathcal{A}} [r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V_k(s')]$;
return An optimal policy π^*.

$$V_0 \rightarrow V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4 \rightarrow V_5 \rightarrow \cdots \rightarrow \pi^*$$

2.5. Reinforcement Learning: Model-Free Algorithms

Reinforcement learning is a one method of machine learning framework for solving sequential decision problems that can be modeled as MDPs [70]. Unlike dynamic programming that assumes the complete knowledge of a perfect model of the environment, RL is primarily concerned with how to obtain an optimal policy when such a model is not available. Therefore, reinforcement learning is model-free. In addition, RL adds to MDPs a focus on approximation and incomplete information, and the need for sampling and exploration to gather statistical knowledge about this unknown model.

For a RL problem, the agent and its environment could be modeled being in a state $s \in S$ and can perform actions $a \in A$, each of which may be members of either discrete or continuous sets and can be multi-dimensional. A state s contains all relevant information about the current situation to predict future states. An action a is used to control the state of the system. For every step, the agent also gets a reward R, which is a scalar value and assumed to be a function of the state and observation. It may equally be modeled as a random variable that depends on only these variables. In the navigation task, a possible reward could be designed based on the energy costs for taken actions and rewards for reaching targets. Reinforcement learning is designed to find a policy π from states to actions, that picks action a in given state s maximizing the cumulative expected reward. The policy π is ei-

ther deterministic or stochastic. The former always uses the exact same action for a given state in the form $a = \pi(s)$, the later draws a sample from a distribution over actions when it encounters a state, i.e., $a \sim \pi(s, a) = P(a|s)$. The reinforcement learning agent needs to discover the relations between states, actions, and rewards. Hence exploration is required which can either be directly embedded in the policy or performed separately and only as part of the learning process. Different types of reward functions are commonly used, including rewards depending only on the current state $R = R(s)$, rewards depending on the current state and action $R = R(s, a)$, and rewards including the transitions $R = R(s', a, s)$.

A detailed survey of reinforcement learning in robotics can be found in [76].

2.5.1. Objectives of Reinforcement Learning

The objectives of RL is to discover an optimal policy π^* that maps states or observations to actions so as to maximize the expected return J, which corresponds to the cumulative expected reward. A finite-horizon model only attempts to maximize the expected reward for the horizon H, i.e., the next H (time-)steps h:

$$J = \mathbb{E}\left\{\sum_{h=0}^{H} R_h\right\}. \tag{2.24}$$

This setting can also be applied to model problems where it is known how many steps are remaining.

Alternatively, future rewards can be discounted by a discount factor γ (with $0 \leq \gamma < 1$):

$$J = \mathbb{E}\left\{\sum_{h=0}^{\infty} \gamma^h R_h\right\}. \tag{2.25}$$

Two natural objectives arise for the learner. In the first, we attempt to find an optimal strategy at the end of a phase of training or interaction. In the second, the objective is to maximize the reward over the whole time the agent is interacting with the world.

compared to supervised learning, the agent must first discover its environment and is not told the optimal action it needs to take. To gain information about the

rewards and the behavior of the system, the agent needs to explore by considering previously unused actions or actions it is uncertain about. It needs to decide whether to play it safe and stick to well known actions with (moderately) high rewards or to dare trying new things in order to discover new strategies with an even higher reward. This problem is commonly known as the *exploration-exploitation trade-off*.

RL relies on the interaction between a learning agent and its environment (see Figure 2.3), the process is similar:

1. A learning agent interacts with its environment in discrete time steps;
2. At each time step t, the agent observes the environment, and receives a representation of state s_t and a reward r_t;
3. The agent infers an action a_t, and subsequently undertaken in the environment.
4. The agent observes the new environment, and receives a new state representation s_{t+1} and an associated reward r_{t+1}.

Based on how the agent chooses an action, RL can be distinguished between off-policy and on-policy methods. *Off-policy* algorithms learn independent of the employed policy, i.e., an explorative strategy that is different from the desired final policy can be employed during the learning process. *On-policy* algorithms collect sample information about the environment using the current policy. As a result, exploration must be built into the policy and determines the speed of the policy improvements. Such exploration and the performance of the policy can result in an exploration-exploitation trade-off between long- and short-term improvement of the policy. A simple exploration scheme known as ϵ-greedy, performs a random action with probability ϵ and otherwise greedily follows the state-action values.

2.5.2. Monte Carlo Methods

Monte Carlo methods use sampling in order to estimate the value function and discover the optimal policy [151]. The procedure can be used to replace the policy eval-

uation step of the dynamic programming-base methods above. Unlike DP, Monte Carlo methods do not assume complete knowledge of the environment. Monte Carlo methods are *model-free*, i.e., they do not need an explicit transition function. They require only experience – sample sequences of states, actions, and rewards from online or simulated interaction with an environment. Learning from online experience requires no prior knowledge of the environment's dynamics, yet can still attain optimal behavior. Learning from simulated experience requires a model, but the model need only generate sample transitions, not the complete probability distributions of all possible transitions that is required by dynamic programming methods.

Monte Carlo methods solve reinforcement learning problems based on averaging sample returns. They perform rollouts by executing the current policy on the system, hence operating on-policy. The frequencies of transitions and rewards are kept track of and used to form estimates of the value function. For example, in an episodic setting the state-action value of a given state action pair can be estimated by averaging all the returns that were received when starting from them.

2.5.3. Temporal Difference Methods

Temporal Difference (TD) Methods is a combination of Monte Carlo methods and dynamic programming methods [151]. Unlike Monte Carlo methods, TD learning methods do not have to wait until an estimate of the return is available (i.e., at the end of an episode) to update the value function. Instead, they use temporal errors and only have to wait until the next time step. The temporal error is the difference between the old estimate and a new estimate of the value function, taking into account the reward received in the current sample. These updates are done iteratively and, in contrast to dynamic programming methods, only take into account the sampled successor states rather than the complete distributions over successor states. Like the Monte Carlo methods, these methods are model-free, as they do not use a model of the transition function to determine the value function, and can learn directly from raw experience without a model of the environment's dynam-

ics. In this setting, the value function cannot be calculated analytically but has to be estimated from sampled transitions in the MDP.

Q-Learning [172] is a representative off-policy, model-free RL algorithm. It incrementally processes the transition samples. Q-value is updated iteratively by

$$Q'(s,a) \leftarrow Q(s,a) + \alpha \left(r(s,a) + \gamma \max_{b \in \mathcal{A}} Q(s',b) - Q(s,a) \right). \tag{2.26}$$

SARSA [137] is a representative on-policy, model-free RL algorithm. Different from Q-learning that uses $\max_{b \in \mathcal{A}} Q(s',b)$ for estimating future rewards, SARSA uses $Q(s',a')$ for a' the action executed in s' under the current policy that generates the transition sample $(s,a,,r,s',a')$. Mathematically, the update rule is:

$$Q'(s,a) \leftarrow Q(s,a) + \alpha \left(r(s,a) + \gamma Q(s',a') - Q(s,a) \right). \tag{2.27}$$

If each action is executed in each state an infinite number of times, and for all state-action pairs (s,a), the learning rate α is decayed appropriately, the Q-values will converge with probability 1 to the optimal Q^* [171]. Similar guarantee of convergence for SARSA can be found in [147] with a more strict requirement on the exploration of all states and actions.

More contents about reinforcement learning will be subsequently presented in Chapter 5.

2.6. Conclusion

This chapter has presented the most important criterion to evaluate the performance of intelligent vehicle platoon, the string stability. Then the Markov decision processes, which are the underlying structure of reinforcement learning. Several classical algorithms for solving MDPs were also briefly introduced. The fundamental concepts of the reinforcement learning was then brought.

Chapter 3

CACC system design

Sommaire

3.1. Introduction

With the increasing problems of traffic congestion and safety, the idea of using automated vehicles driving in automated highway is growing steadily more attractive. Longitudinal control is one of the basic functions of the vehicle automation. Longitudinal control system controls the longitudinal motion of the vehicle, such as velocity, acceleration or the its longitudinal distance from the front vehicle in the same lane, by using throttle and brake controllers [125], thus to realize the lane-keeping tasks for the automatic vehicles.

The longitudinal vehicle motion control has been pursued for several decades and at many different levels by researchers and automotive manufactures. From 1970s to 1980s, there appeared some researches in the control system design for vehicle engines and brake systems, as shown in [42, 51, 52, 101, 122]. Since then, some first generation of engine control systems appeared in [22, 49], and some results in the brake system control have obtained great success, such as the ABS (Anti-lock Brake System), which have been widely accepted in the automobile industry. Based on these results, and since 1990s, the researches in the longitudinal control combined with throttle and brake control has become steadily more attractive, and a variety of solutions have been proposed in [156, 92] and [60, 91]. In addition, in 1986, the California PATH (U.S.A.), one of the most fruitful organization in transportation researches, was established. Almost in the same time, the program of AHSS (Advanced Highway Safety System) in Japan, and the program of PROMETHEUS (PROgraMme for a European Traffic of Highest Efficiency and Unprecedented Safety) in Europe were carried out. These programs have contributed a considerable efforts and encouraging results in such a control system.

Nowadays, the standard CC system, which can automatically control the throttle to maintain the pre-set speed, is widely available on passenger cars. However, during the past decade, traffic congestion has become a severe problem in industrialized nations worldwide due to undesirable human driving behavior and limited transportation infrastructure. An effective solution to increase traffic throughput is to reduce the inter-vehicle distance, which is however parlous

for human drivers. To this end, ACC is being developed by researchers and automotive manufactures to improve traffic flow stability, throughput and safety [193][90][129][12][115][87][103]. In the case of absence of preceding vehicles, the ACC vehicle travels the same as a CC vehicle. Compared to simple CC, which is already equipped in certain commercial vehicles, ACC system is able to improve driver convenience, reduce workload, which however results in string instability in most cases.

As described in last chapter, the concept of string stability is generally characterized as the attenuation of the disturbances in the upstream platoon, e.g., brake or acceleration of leading vehicle. String stability of ACC system can be improved if the information through V2V transmission of the preceding vehicle is used in the feedback loop. This transmission is realized by a low latency communication medium. The most distinctive difference between ACC and CACC is that besides the preceding vehicle's speed and position used as inputs in ACC, the desired acceleration of the preceding vehicle transmitted through the wireless channel is also adopted as input in CACC controller. Therefore, CACC is treated as a solution to achieve a desired following distance with string stability. However, no generic approach for the design of CACC system is adopted. Most of the relative researches relied on the classic control theory [148][67][55]. In [166], a fault tolerance criterion for which CACC systems still is functional is defined. [142] has developed a generic safety checking strategy within the loop of vehicle-controller in a platoon of vehicles, which guarantees performance when the inter-vehicle distance in platoon is changed during a maneuver in emergency situation. Considering the tracking capability, fuel economy and driver desired response, a predictive model of CACC system is designed in [82]. Above all, decent performance of CACC in traffic throughput has been proved in many researches with a low time gap and increased traffic throughput, while maintaining safety, comfort and stability. Inspired by the concept of "platoon", our principal objective is to design a vehicle longitudinal control system which can enhance vehicle safety while at the same time improving traffic capacity. Thus, we need to envisage not only the control problems of a single vehicle but also the behaviors of a string of vehicles.

This chapter concentrates on the design problems of vehicle longitudinal control system design. At first, the notion of string stability is introduced in detail. Secondly, the longitudinal control system architecture of two different spacing policies are designed.To validate the proposed controllers, simulation tests will be carried out and their string stability will be analyzed. And some conclusion will be given in the end of this chapter.

3.2. Problem formulation

3.2.1. Architecture of longitudinal control

As we have introduced in chapter 1, the control architecture of an ITS is hierarchical and has 3 layers shown in Fig.1.5. In the information processing layer, the longitudinal control system, as one of the control strategy proposition, is in charge of the steady and transient longitudinal maneuvers. The architecture of longitudinal control system is illustrated in Fig. 3.1. In an intelligent vehicle structure, the module "CACC Controller" together with the module "Vehicle Dynamics" provides the prototype of CACC functionality. At the beginning of each simulation step, the module "CACC Controller" reads relative speed and inter-vehicle distance to the preceding vehicle from the "Radar" module. The host vehicle's acceleration and speed are read from the module "sensor" as inputs. In addition, CACC would read the desired acceleration of the preceding vehicle from the "Wireless Medium" by "Ad hoc network" module, which is not necessary for ACC. Meanwhile, the host vehicle transfers its own desired acceleration to the medium as well which is used for the CACC controllers of other vehicles. The desired time headway, desired distance at standstill and cruise speed are pre-set before the simulation starts. The time headway is the time it takes for ith vehicle to reach the current position of its preceding $i-1$th vehicle when continuing to drive with a constant velocity. Finally, the CACC controller renews the spacing error input and recalculate the new desired acceleration in next time step. The control objective is to realize a desired distance, taking into account a pre-defined maximum speed, referred to as the cruise speed. Note that the cruise speed is a maximum speed when the vehicle

operates in CACC mode. If there is no target vehicle, the system switches to a cruise control mode, in which case the cruise speed becomes the target speed.

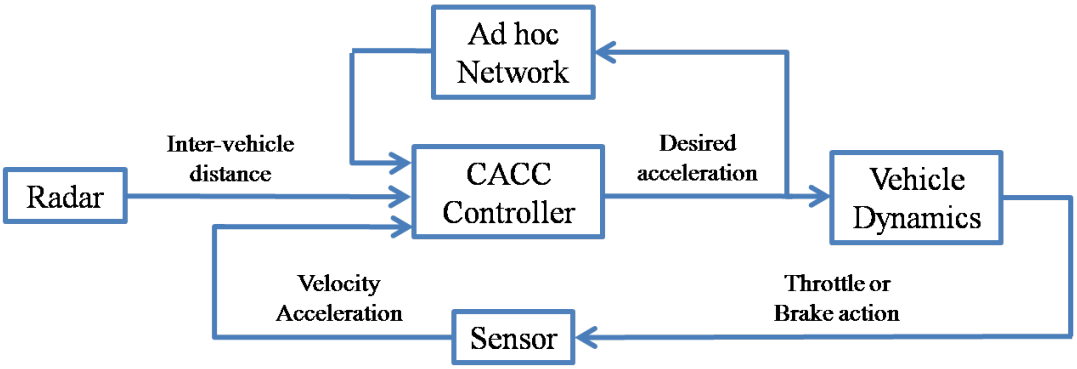


Figure 3.1 – *Architecture of CACC longitudinal control system*

3.2.2. Design objectives

As we have introduced in the previous section, the first-generation of longitudinal control systems like CC or ACC systems are primarily being developed from the point of view of increased driving comfort with some potential in increasing vehicle safety. However, the impacts of these longitudinal control systems on highway traffic have been inadequately studied [155, 140]. From the transportation planners' point of view, the automated vehicles equipped with the longitudinal control systems should heavily impact the traffic characteristics, including highway safety, efficiency and capacity because of their more uniform behavior compared with human drivers [198]. Before the longitudinal control systems are widely equipped on automated vehicles, their impacts on string behavior and flow characteristics need to be carefully investigated. Otherwise traffic congestion may become worse instead of being better.

As mentionned in previous chapter, the most important macroscopic behaviors of CACC vehicles is the string stability. The string stability of a vehicle platoon refers to the property in which spacing errors are guaranteed not to amplify as they propagate towards the tail of the string [154, 140]. This property ensures that any spacing error present at the head of the string does not amplify into a large error at the tail of the string. A general method to evaluate string stability is to

examine the transfer function from the spacing error of the proceeding vehicle to that of the following vehicle. If the infinite norm of this transfer function is less than 1, string stability is ensured [153, 169].

Based on the above discussions, the design of a CACC controller, which includes the specific spacing policy and the associated control laws, should be designed to achieve the following objectives:

- By using the wireless communication with related vehicles in a platoon, the host vehicle should follow its preceding vehicle while keeping a safe distance.
- The steady state of spacing error of each vehicle should be approximatively equal to zero for tracking purpose.
- The acceleration/deceleration and the velocity should be decreasing in upstream platoon which means the string stability is guaranteed.
- Instead of a centralized algorithm, a decentralized one should be proposed in order to reduce the computational cost.
- The control effort required by the control law should be within the vehicle's traction/braking capability.
- The passengers' comfort should be taken into account, in other words, sharp change of the acceleration should be averted.
- It should be used for a wide range of speed for vehicle operations in highway, which includes low and high speed scenarios.

3.3. CACC controller design

3.3.1. Constant Time Headway spacing policy

At present, the most common spacing policy used by researchers and vehicle manufactures is the Constant Time Headway (CTH) spacing policy [169]. Much research works have been done in the study of (C)ACC system with CTH spacing policy [90, 109, 65].

The desired distance is generally supposed to be an increasing function of host vehicle's velocity. The desired spacing of CTH spacing policy is given by

$$d_{r,i}(t) = r_i + +hv_i(t) \tag{3.1}$$

where $d_(r,i)(t)$ is the desired distance of ith vehicle from its front vehicle, r_i is the standstill distance and h is the constant time headway time.

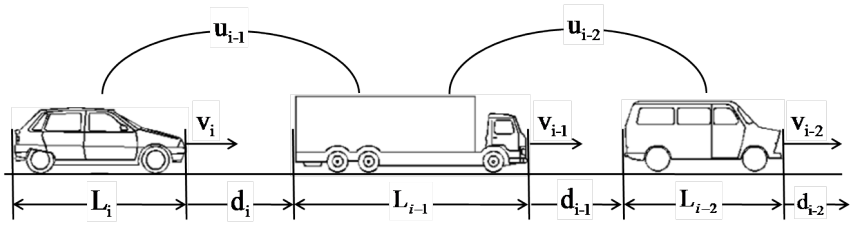


Figure 3.2 – *Vehicle platoon illustration*

Consider a platoon of vehicles shown in Fig. 3.2. A schematic of a homogeneous platoon of vehicles equipped with the CACC functionality is described, in which d_i, v_i, u_i and L_i represent the rear distance between the front bumper of ith vehicle and the rear bumper of $i-1$th vehicle, the velocity, the desired acceleration and the length of ith vehicle respectively. In this section, the homogeneity traffic is considered, i.e., vehicles with identical characteristics. With different types of vehicles in the platoon, the homogeneity can be obtained by low-level acceleration controllers so as to arrive at identical vehicle behavior. Vehicles in platoon utilize distance sensors to get the inter-vehicle distance and relative speed. Besides, the feedforward term u_i of the nearest front vehicle is transferred through the wireless V2V communication. Hence, the ACC functionality is still available if no communication is present.

Therefore, the spacing error e_i is then defined as

$$e_i(t) = d_i(t) - d_{r,i}(t) = (s_{i-1}(t) - s_i(t) - L_i) - (r_i + hv_i(t)) \tag{3.2}$$

The purpose of a CACC controller is to regulate the inter-vehicle distance $d_i(t)$

to the desired distance $d_{r,i}(t)$, i.e. zero spacing error.

$$a_0(t) = 0 \quad \forall t \geq 0 \Rightarrow \lim_{t\to\infty} e_i(t) = 0 \quad \forall 1 \leq i \leq n \tag{3.3}$$

The driving state of an intelligent vehicle includes its position, velocity, acceleration and spacing error. The first vehicle of the platoon, called leading vehicle, should be considered differently from the rest. It is manipulated either by human or following a virtual CACC-equipped vehicle.

3.3.2. Multiple V2V CACC system

Having formulated the control problem, a decentralized longitudinal control law of Two-Vehicle-Ahead (TVA) CACC system is designed in this section.

In actual situation, the host vehicle is influenced not only by its nearest front vehicle but also all the vehicles before it in the platoon, especially the first vehicle of the string, so called virtual leading vehicle, which plays an important role in the platoon that may determine the performance of the whole platoon. In order to imitate human behavior and make an optimized decision, the multiple V2V communication is favorable to be taken into account instead of conventional one-vehicle transmission. A longitudinal tracking control law is proposed in [56], in which the information of the front vehicle and the designed platoon leading vehicle are used as feedforward terms. The weights of the information differ from the different relative positions of vehicles in the platoon. The greater the distance between the host and the leading vehicle, the less weight is taken into account for the host vehicle. Then the expected velocity $v_{r,i}$ and acceleration $a_{r,i}$ are defined as:

$$v_{r,i} = (1 - p_i)v_{i-1}(t) + p_i v_l(t) \quad 3 \leq i \leq n \tag{3.4}$$

$$a_{r,i} = (1 - p_i)a_{i-1}(t) + p_i a_l(t) \quad 3 \leq i \leq n \tag{3.5}$$

where p_i is the influence weight of the ith vehicle compared to the platoon leader. n is the number of CACC-equipped vehicles in the platoon. Note that p_i

is dependent of index i because it differs from different relative position of the platoon. Compared to the basic CACC controller, the complex communication topology leads to a faster accelerator response.

However, the quality of the inter-vehicle communication between the host and the virtual leading vehicle can hardly be guaranteed. The transmission could be affected by the noise, weather, obstacles etc. For the vehicles with a large index in the platoon, the V2V communication between the host and the leading vehicle will degrade, losing data and slowing down the transmission, which leads to string instability and overshoots. Moreover the influence weight will be little for the host vehicle, even neglectable.

To illustrate the design procedure in the case of a more complex information topology and to investigate the possible benefits of this topology with respect to string stability, a novel Two-Vehicle-Ahead Cooperative Adaptive Cruise Control (TVACACC) controller is proposed in this section. Instead of using one input u_{i-1} from the $i-1$ vehicle, an additional input u_{i-2} is taken into account to improve the tracking capacity. Besides, the second vehicle is following the leading vehicle, which is considered as a conventional CACC controller with only one input u_1, while the rest vehicles of the platoon are TVACACC controllers. The influence weights are p_{i-1} and p_{i-2}. Therefore, the desired acceleration of the host vehicle is defined as follow. The benifits compared to the conventional CACC controller will be shown in the simulation section.

$$u_{i,input}(t) = p_{i-1}u_{i-1}(t) + p_{i-2}u_{i-2}(t) \quad 3 \leq i \leq n \tag{3.6}$$

3.3.3. System Response Model

Let us consider a platoon of n vehicles. As a basis for control design, the acceleration response can be approximated by a first-order system:

$$\tau \dot{a}_i + a_i = u_i \tag{3.7}$$

where a_i is the vehicle's actual acceleration, u_i is the desired acceleration, and τ is a constant time lag.

Then, the following vehicle dynamic model is adopted:

$$\begin{pmatrix} \dot{d}_i \\ \dot{v}_i \\ \dot{a}_i \end{pmatrix} = \begin{pmatrix} v_{i-1} - v_i \\ a_i \\ -\frac{1}{\tau}a_i + \frac{1}{\tau}u_i \end{pmatrix} \quad 2 \leq i \leq n \tag{3.8}$$

where a_i is the acceleration, u_i is the desired acceleration of of ith vehicle. In order to satisfy the tracking objective defined in equation 3.3, the error dynamics are formulated as:

$$\begin{pmatrix} e_{1,i} \\ e_{2,i} \\ e_{3,i} \end{pmatrix} = \begin{pmatrix} e_i \\ \dot{e}_i \\ \ddot{e}_i \end{pmatrix} \quad 2 \leq i \leq n \tag{3.9}$$

Combining equation 3.2 and 3.9, we obtain:

$$e_{2,i} = v_{i-1} - v_i - ha_i \tag{3.10}$$

$$e_{3,i} = a_{i-1} - a_i - h\dot{a}_i \tag{3.11}$$

$$\dot{e_{3,i}} = -\frac{1}{\tau}e_{3,i} + \frac{1}{\tau}u_{i-1} - \frac{1}{\tau}q_i \tag{3.12}$$

with a new input

$$q_i \doteq h\dot{u}_i + u_i \tag{3.13}$$

The input q_i is designed to regulate the inter-vehicle distance to $d_{r,i}$. In addition, the input $u_{i,input} = p_{i-1}u_{i-1} + p_{i-2}u_{i-2}$ should be compensated as well. Hence, the control law of q_i is designed as

$$q_i = K \begin{pmatrix} e_{1,i} \\ e_{2,i} \\ e_{3,i} \end{pmatrix} + p_{i-1}u_{i-1} + p_{i-2}u_{i-2} \quad 3 \leq i \leq n \tag{3.14}$$

Where $K = [k_p \quad k_d \quad k_{dd}]$ represents the controller coefficient vector. The two feedforward terms are obtained through ad hoc network with the front vehicles.

Due to the additional controller dynamic defined in equation 3.13, the platoon

model is augmented with one more state u_i, which can be obtained by using equation 3.13 and 3.14:

$$\begin{aligned}\dot{u}_i =& \frac{k_p}{h}e_i - \frac{k_d}{h}v_i - \frac{\tau h k_d + \tau k_{dd} - h k_{dd}}{\tau h}a_i - \frac{\tau + h k_{dd}}{\tau h}u_i \\ &+ \frac{k_d}{h}v_{i-1} + \frac{k_{dd}}{h}a_{i-1} + \frac{1}{2h}(u_{i-1} + u_{i-2})\end{aligned} \tag{3.15}$$

As a result, the 4th order closed-loop vehicle model is established. From the third to the last vehicle of the platoon, i.e., $3 \leq i \leq n$, the vehicle dynamics:

$$\begin{aligned}\begin{pmatrix}\dot{e}_i \\ \dot{a}_i \\ \dot{v}_i \\ \dot{u}_i\end{pmatrix} =& \begin{pmatrix}0 & -1 & -h & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} \\ \frac{k_p}{h} & -\frac{k_d}{h} & -\frac{\tau h k_d + \tau k_{dd} - h k_{dd}}{\tau h} & -\frac{\tau + h k_{dd}}{\tau h}\end{pmatrix}\begin{pmatrix}e_i \\ a_i \\ v_i \\ u_i\end{pmatrix} \\ &+ \begin{pmatrix}0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{k_d}{h} & \frac{k_{dd}}{h} & \frac{1}{2h}\end{pmatrix}\begin{pmatrix}e_{i-1} \\ a_{i-1} \\ v_{i-1} \\ u_{i-1}\end{pmatrix} + \begin{pmatrix}0 \\ 0 \\ 0 \\ \frac{1}{2h}\end{pmatrix}u_{i-2}\end{aligned} \tag{3.16}$$

or in short

$$\dot{X} = AX_i + BX_{i-1} + Cu_{i-2} \tag{3.17}$$

where the vehicle state is $X \doteq (e_i \quad v_i \quad a_i \quad u_i)^T$ and the matrices A, B, C are defined correspondingly.

Note that the virtual leading, the second and the rest vehicles of the platoon are of different state model. The second vehicle ($i = 2$) is assumed to follow a virtual vehicle ($i = 1$) where only the information from the virtual leading vehicle is applied, i.e., conventional CACC controller. Thus both vehicles state model are different from the rest of the platoon. The first vehicle may be formulated as follows.

$$\begin{pmatrix} \dot{e}_2 \\ \dot{a}_2 \\ \dot{v}_2 \\ \dot{u}_2 \end{pmatrix} = \begin{pmatrix} 0 & -1 & -h & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} \\ \frac{k_p}{h} & -\frac{k_d}{h} & -\frac{\tau h k_d + \tau k_{dd} - h k_{dd}}{\tau h} & -\frac{\tau + h k_{dd}}{\tau h} \end{pmatrix} \begin{pmatrix} e_2 \\ a_2 \\ v_2 \\ u_2 \end{pmatrix} + \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{k_d}{h} & \frac{k_{dd}}{h} & \frac{1}{h} \end{pmatrix} \begin{pmatrix} e_1 \\ a_1 \\ v_1 \\ u_1 \end{pmatrix} \tag{3.18}$$

The virtual leading vehicle, not having any information from other vehicles, can also be modeled, in which $\dot{e}_1(t) = e_1(t) = 0$is adapted, assuming that there is no error for the virtual leading vehicle.

$$\begin{pmatrix} \dot{e}_1 \\ \dot{a}_1 \\ \dot{v}_1 \\ \dot{u}_1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} \\ 0 & 0 & 0 & -\frac{1}{h} \end{pmatrix} \begin{pmatrix} e_1 \\ a_1 \\ v_1 \\ u_1 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{h} \end{pmatrix} q_i \tag{3.19}$$

According to the Lienard-Chipart stability criterion, it is shown that for bounded inputs u_{i-1} and u_{i-2}, the vehicle controller defined in euqation 3.16 and 3.18 will be stable if the following constraints are satisfied.

$$k_p > 0, \quad k_{dd} h + \tau > 0 \tag{3.20}$$

However, the single vehicle stability means that the spacing error is stable, which is not equivalent to the string stability mentioned in the previous section. The latter focuses on the decreasing of spacing error in the upstream direction. In fact, due to degradation of V2V communication, the headway time h and transmission delay θ may vary, which will greatly influence the string stability. This issue is discussed in the following chapter.

3.3.4. TVACACC diagram

According to the previous subsection, The block diagram of this TVA-CACC system transferred in Laplace domain is depicted in Fig. 3.3. In this diagram the following definitions are used. The influence weights of $i-1$th and ith vehicles are selected to be equal: $p_{i-1} = p_{i-2} = 0.5$.

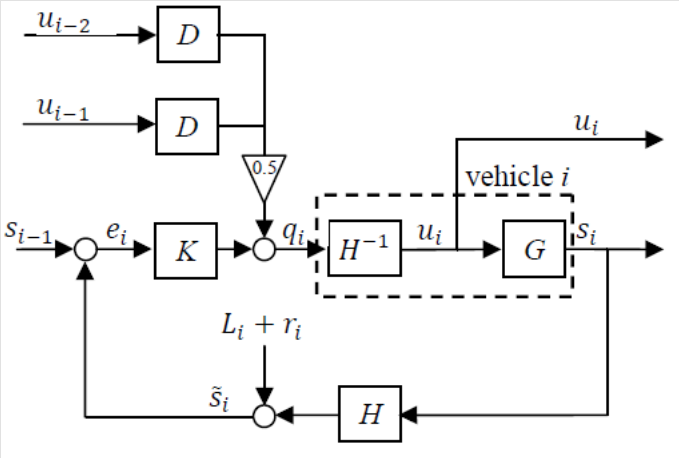


Figure 3.3 – *Block diagram of the TVACACC system*

- $\hat{S}_{i-1}(s)$: position of the front vehicle, the input by ranging sensor;
- $\hat{S}_i(s)$: position of the host vehicle;
- $\tilde{s}_i(s)$: virtual position of the host vehicle;
- $D(s)$: delay θ of wireless communication $\hat{u}_{i-1}$ due to queuing, connection and propagation;

$$D(s) = e^{-\theta s} \tag{3.21}$$

- $K(s) = \hat{q}_i(s)/\hat{e}_i(s)$: the controller defined in equation 3.14;

$$K(s) = k_p + k_d s + k_{dd} s^2 \tag{3.22}$$

- $H(s) = \hat{q}_i(s)/\hat{u}_i(s)$: the CTH spacing policy transfer function defined in equation 3.13;

$$H(s) = hs + 1 \tag{3.23}$$

- $G(s) = \hat{s}_i(s)/\hat{u}_i(s)$: the vehicle transfer function from acceleration to position with θ_G the time delay of the engine;

$$G(s) = \frac{e^{-\theta_G}}{s^2(\tau s + 1)} \tag{3.24}$$

3.4. String stability analysis

For a cascaded system, such as a platoon of automated vehicles, stability of each component system itself is not sufficient to guarantee a decent performance of all systems, such as the non-convergence of spacing error for two consecutive vehicles. This is the reason why our research object is a string of vehicles instead of only one vehicle. Therefore, besides the individual stability of each vehicle, another stability criterion known as the string stability is also required. The condition of individual vehicle stability of TVACACC system is already given in equation 3.20. In this subsection, the string stability of conventional ACC, CACC and the proposed TVACACC functionality will be shown theoretically.

Recall equation 2.4 in chapter 2, the condition for the string stability of vehicle platoon:

$$\sup_{\omega} |\Gamma_i(j\omega)| \leq 1, 2 \leq i \leq n \tag{3.25}$$

where $\Gamma_i(j\omega)$ is the frequency response function describing the relation between the scalar output z_{i-1} of a preceding vehicle $i-1$ and the scalar output z_i of the follower vehicle i. In our case, we choose the input of interest to be the acceleration $\Gamma_i(j\omega) = \hat{e}_i(j\omega)/\hat{e}_{i-1}(j\omega)$. While, if the system is string is unstable, $sup_\omega|\Gamma_i(j\omega)|$ will exceed 1. Still in that case, we would aim at keeping this norm as low as possible to minimize the disturbance amplification in upstream direction.

3.4.1. String stability of TVACACC

Parameters are chosen as $\tau = 0.1$, $k_p = 0.2$, $k_d = 0.7$, $k_{dd} = 0$ to avoid feedback of the and $h = 0.5s$, transmission delay θ=0.2s.

In order to improve the string stability of the platoon and to help intelligent

vehicles making a more conservative and reasonable decision, the TVACACC controller is proposed in this work. Regarding the second vehicle, there is only one vehicle before, the leading vehicle of the platoon. Therefore, the second vehicle receives only the information of first vehicle transmitted by V2V communication, which is different from the rest of the platoon. Thus the transfer function of second vehicle is obtained by replacing the input $u_{i,input} = p_{i-1}u_{i-1} + p_{i-2}u_{i-2}$ by u_{i-1}.

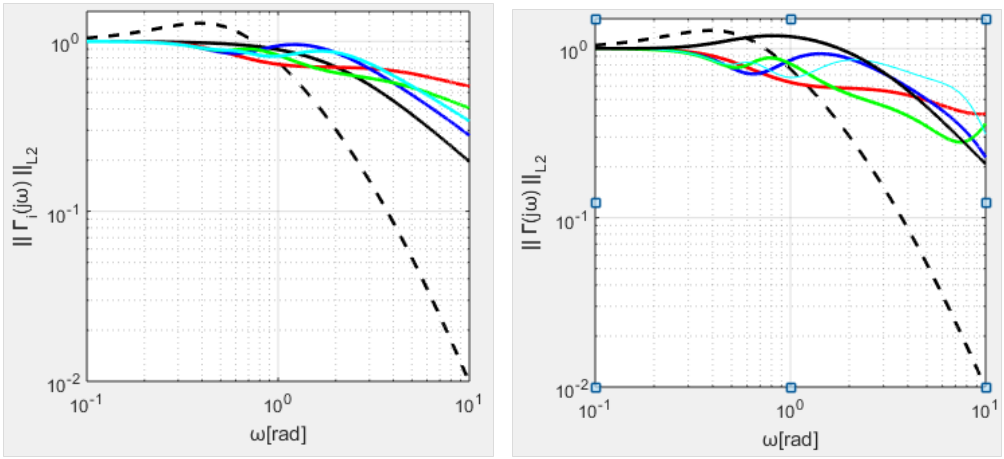
$$||\Gamma_2(j\omega)||_{L_2} = \frac{||a_2(s)||_{L_2}}{||a_1(s)||_{L_2}} = \frac{||D(s) + G(s)K(s)||_{L_2}}{||H(s)(1 + G(s)K(s)||_{L_2}} \tag{3.26}$$

The transfer function $||\Gamma_i(j\omega)||_{L_2}$ of the rest vehicles in the platoon is derived from equation 3.16.
For $3 \leq i \leq n$,

$$||\Gamma_i(j\omega)||_{L_2} = \frac{||a_i(s)||_{L_2}}{||a_{i-1}(s)||_{L_2}} = \frac{||D(s)(1 + \frac{D(s)}{\Gamma_{i-1}(s)} + G(s)K(s))||_{L_2}}{||2H(s)(1 + G(s)K(s))||_{L_2}} \tag{3.27}$$

Note that the transfer function for ith vehicle depends on the string stability of $i-1$th vehicle due to the two vehicle inputs. Thus the string stability differs for different vehicles in the platoon. Choosing the same parameters in the last paragraph, the transfer function from the second to the sixth vehicle of the platoon with transmission delay $\theta = 0.2s$, is shown in Fig. 3.4a. The transfer function response of the second vehicle, which receives only the information from the first vehicle, is represented by solid black line. The third to sixth vehicles are represented by colored line. Although the curves seem to be arbitrary, but it is shown that the norms is always smaller than 1, i.e., the string stability is guaranteed and the disturbance attenuates.

As mentioned above, communication degradation may happen while applying V2V communication. Assuming that the transmission delay increases to $\theta = 1s$ instead of $0.2s$, the transfer function $||\Gamma_i(j\omega)||_{L_2}$ is shown in Fig. 3.4b. The second vehicle of the platoon is the same as conventional CACC system, shown by black line, which is string unstable in transmission degradation situation. But the rest vehicles keep the string stability in this case. When the transmission delay increases, the vehicle platoon using conventional CACC system is unstable and worse than



(a) *Transmission delay 0.2s* (b) *Transmission delay 0.4s*

Figure 3.4 – *String stability comparison of ACC and two CACC functionality with different transmission delays: ACC (dashed black), Conventional CACC (black) and TVACACC in which the second vehicle (black) and the rest vehicles (colored)*

the normal situation. Instead, the string stability is maintained in the TVACACC case. The TVACACC model uses not only the input from vehicle $i-1$ but also $i-2$, so that the communication degradation from vehicle $i-1$ will have less influence, which leads to a better string stability.

3.4.2. Comparison of ACC, CACC AND TVACACC

The string stability of conventional CACC system is the same as the second vehicle in TVACACC system as mentioned above. Therefore, the transfer function is the same as equation 3.26.

$$||\Gamma_{CACC}(j\omega)||_{L_2} = \frac{||D(s) + G(s)K(s)||_{L_2}}{||H(s)(1 + G(s)K(s)||_{L_2}} \tag{3.28}$$

Moreover, ACC system is easily obtained by choosing the transmission delay block $D(s) = 0$, because there is no transmission between the host and its front vehicle. The transfer function of ACC is then derived as

$$||\Gamma_{ACC}(j\omega)||_{L_2} = \frac{||G(s)K(s)||_{L_2}}{||H(s)(1 + G(s)K(s)||_{L_2}} \tag{3.29}$$

In the case of transmission delay $\theta = 0.2s$, the frequency domain response of

ACC and conventional CACC systems are represented by dashed black and solid black line in Figure. 3.4a respectively. It is clearly shown that for an ACC system, the disturbance amplifies in the platoon upstream, resulting in worse influences on the rest vehicles in platoon. On the contrary, thanks to the V2V technology, the conventional CACC system as well as the proposed TVACACC system guarantee the string stability.

If the transmission delay degrades to $\theta = 1s$, the string stability is illustrated in Figure 3.4b. However in this situation, the same platoon using conventional CACC system is no longer string stable.We can see that the amplification of disturbance is almost the same compared to ACC system. ACC system is not changed as no V2V transmission is applied. Therefore, if transmission degradation occurs, the CACC functionality degrades and if the transmission delay continues to increases, the performance may be worse than ACC system.

Therefore, in the case of TVACACC system, an increased traffic flux and a decreased disturbance are obtained compared to the conventional CACC system. Besides, it performs better facing an increasing transmission delay than the existent CACC system.

3.5. Simulation tests

To validate the theoretical results and demonstrate its feasibility of the conventional and proposed CACC functionality, a series of simulations is carried out within a platoon of V2V communication equipped vehicles. It is shown whether the disturbance of the leading vehicle is attenuated upstream through the platoon, which is defined as string stability in chapter 2. Therefore, the vehicle's velocity and acceleration are selected as string stability performance measures. The results in both normal and degraded situations will be shown.

For validation of the theories of the proposed model in the previous sections, a stop-and-go scenario is chosen because it is the most dangerous situation of all possible situations in longitudinal control. The platoon is composed of six CACC equipped vehicles and they are assumed to share identical characters. The platoon starts in steady state with speed of $30m/s$ ($108km/h$). At $t = 10s$, the leading vehicle

of the platoon performs a brake with deceleration of $-5m/s^2$, and reaccelerates until regaining the initial velocity $30m/s$ with acceleration of $2m/s^2$ at $t = 30s$.

3.5.1. Comparison of ACC CACC and TVACACC

The Conventional CACC and ACC system are introduced in Figure. 3.5(a) and (c), to make a clear comparison to the TVACACC system. Each vehicle is following its front vehicle by respecting a safe distance with a headway time of $0.5s$. The transmission delay of the input u_i is set to be $0.2s$ for CACC system while there is no V2V communication in ACC system. It can be clearly seen that the simulation results correspond to the theoretical analysis shown in Figure. 3.4a. Under the designed condition, the platoon equipped with conventional CACC system is string stable. The influence of acceleration disturbance decreases in the upstream direction. However, the ACC system is not string stable under the same condition. The further the following vehicle is to the leading vehicle, the greater are the acceleration and deceleration responses. The string stability is a crucial criterion for CACC systems. It ensures that the following vehicles' safety and low fuel cost. On the contrary, the string instability results in larger acceleration and deceleration facing the stop-and-go scenario, which is the case of ACC system in this case. If there are more vehicles in the platoon, the last vehicle will suffer from a hard brake and acceleration to catch up the platoon, even beyond its physical limit which is not only harmful for the entire traffic flow, safety and comfort, but also might result in rear-end collision. That is the reason why conventional ACC system requires greater headway time to guarantee the string stability, which means lower traffic flux.

The simulation of the proposed TVA-CACC system in the same scenario is shown in Figure. 3.5(b) with the same parameters. It is obvious that the string stability is obtained, the same as conventional CACC system, i.e. the acceleration and deceleration disturbance decrease in the upstream direction. Moreover, the acceleration response is smaller which means better string stability. The result corresponds to the theoretical analysis in Figure. 3.4a. Therefore, with the proposed

system, a better traffic flux, a safer and more comfortable driving experience is obtained, compared to the conventional one-vehicle-ahead CACC system.

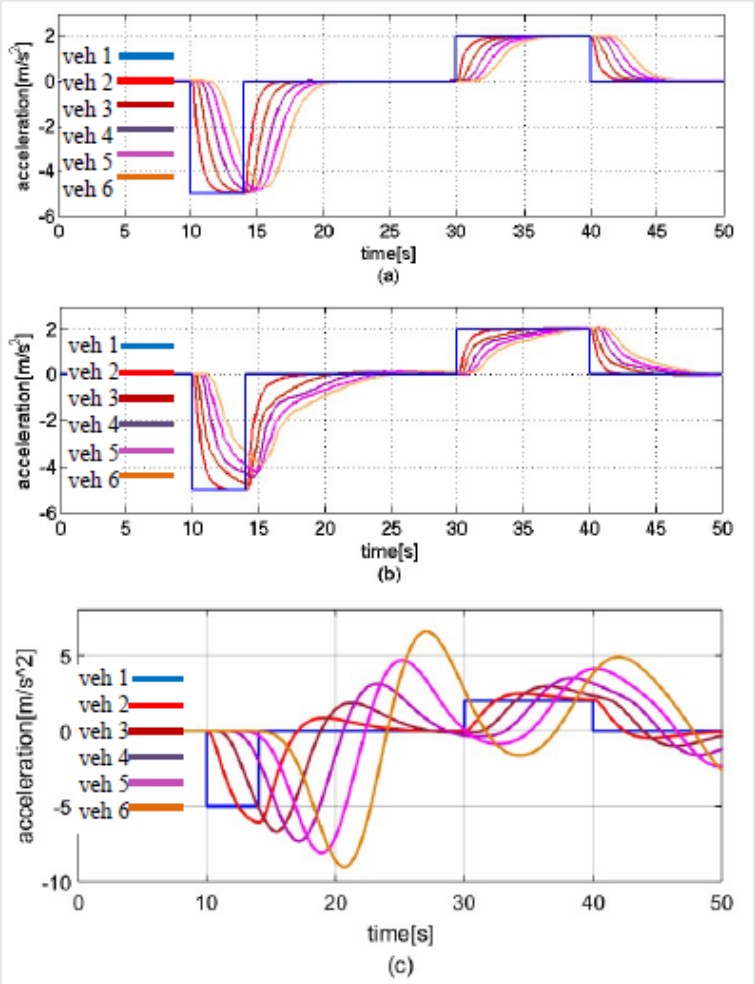


Figure 3.5 – *Acceleration response of a platoon in Stop-and-Go scenario using conventional CACC system (a), TVA-CACC system (b) and ACC system (c) with a communication delay of* 0.2*s*

3.5.2. Increased transmission delay

In this subsection, it is assumed that the CACC systems are suffering from transmission delay. Instead of a normal delay of 0.2*s*, the lagged transmission delay is 1s. In Figure. 3.6, it is clearly seen that the conventional CACC system is badly degraded, compared to the normal situation shown in Figure. 3.5, due to increased transmission delay. The acceleration response is overshoot and increases in the upstream direction which means the system is string unstable. The experimental

results correspond to the theoretical analysis of string stability. One solution is regain the string stability is to increase the headway time, which however, decrease the traffic flow. On the contrary, in the case of TVA-CACC system, the acceleration disturbance still attenuates in upstream direction, i.e., the string stability is maintained in the degraded situation. However, the acceleration response of the same vehicle slightly increases which means the string is less stable than it is in the normal transmission situation. And if is transmission is even more delayed, the proposed CACC system cannot guarantee its string stability. The threshold according to simulation is about 2.5s.

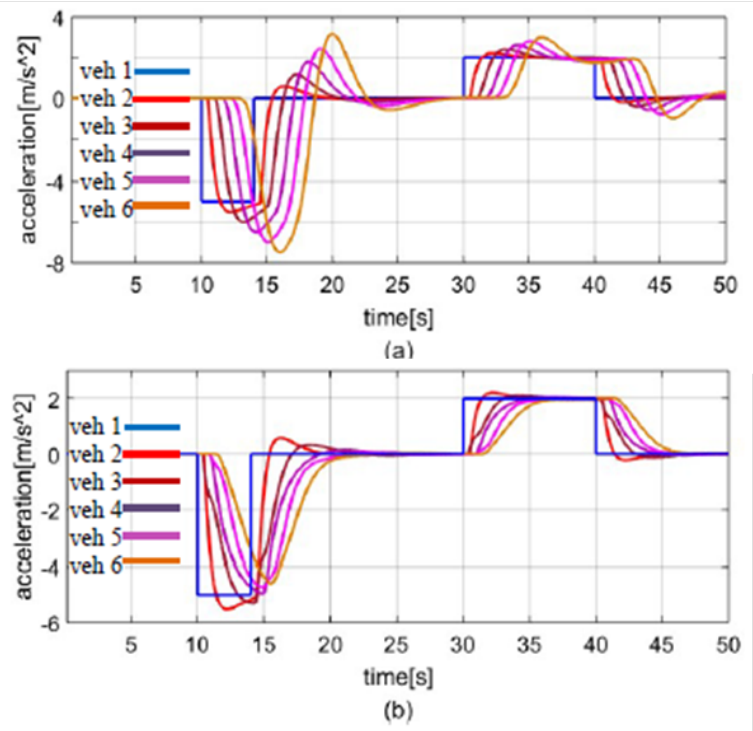


Figure 3.6 – *Acceleration response of a platoon in Stop-and-Go scenario using conventional CACC system (a) and TVACACC system (b) with a communication delay of 1s*

3.6. Conclusion

In this chapter, we concentrated on the vehicle longitudinal control system design.

The spacing policy and its associated control law were designed with the constrains of string stability. The CTH spacing policy is adopted to determine the desired spacing from the preceding vehicle. It was shown that the proposed TVA-

CACC system could ensure both the string stability. In addition, through the comparisons between the TVACACC and the conventional CACC and ACC systems, we could find the obvious advantages of the SSP system in improving traffic capacity especially in the high-density traffic conditions.

The above proposed longitudinal control system was validated to be effective through a series of simulations.

Chapter 4

Degraded CACC system design

Sommaire

4.1. Introduction

Wireless communication systems are applied in the control systems to substitute the cables, simplify the hardware system. But unlike the system with the cables, the data transmission (state parameters or control signals) is not that reliable or predictable. Transmission through the air often get interfered by the noise, obstacle etc. and randomly cause the transmission delay and data fault or loss. All these kinds of error will somehow influence the performance of the control system. The extent depends on the extent of the error and the system itself. In some of the systems which have the long settling time would be hardly influenced by little of the transmission fault. But nowadays more and more control systems are required for faster, more accurate and stable. The fault discussed above can slow the process, make it inaccurate and unstable. Performance will be degraded such as bigger settling time and overshoot, or sometimes the whole system will be harmed for example the system become unstable.

Previous work has more focused on the stability problems that the transmission error would cause to the control system by finding the control methods of counteract the effect of delay and loss. For the time delay system, reference[133] has given a thorough analysis and summarized some of the optimal control methods to deal with the delay. In the system with data delay, the stability of the new system becomes the most important for the system. [14, 4] have applied the linkage of time-delay $e^{-\tau s}$ to the close-loop transfer function of the system and used different ways to deal with the quasi-polynomials according to the rules of stability. As to the control system with packet loss, the stability of the system doesn't change no matter if there is or not loss because the structure of the system never changes [111]. So the research on the system with the data-loss system is more emphasized on the performance degradation of the system [111]. [139, 186] used state space representation to describe the control system with the data loss. Methods are proposed to deal with data loss. [111] proposed a compensation method to counteract the effect to the performance of the loss of control signal. [63] has proposed the optimal control method under both the conditions that the communication network

is acknowledgement-support (TCP) and no acknowledgement-support (UDP). But when come across the systems without all these kinds of control strategy, the degradation becomes important to be aware. [46] has applied the concept of dependability to the close-loop control system with variable delays and message losses. He used Mobius tools to simulate the control system. The influence of the delay and data loss has been simulated using the Monte Carlo method. Some evaluations of criteria of reliability were proposed (failure by overshoot, failure by stability).

Intelligent vehicles use wireless communications to make important driving decisions, pass, speed control ...[190]. This chapter focuses on the degradation of control performance of the CACC systems. By analyzing the process of the system, the method of estimation of the degradation will be proposed according to the system characters, data delay and data loss. The second section of this chapter deals with the discrete sampling control system analyzing and the model construction. The third section discusses the degradation of performance that caused by the data delay. The forth section is about the degradation of performance that caused by data loss.

4.2. Transmission degradation

Wireless V2V communication is a key factor to realize CACC systems. Unlike the on-board radar sensors equipped in ACC systems which are used to measure the inter-vehicle distance and relative velocity, the extended V2V wireless communication is less reliable due to high latency and packet loss, which makes CACC functionality dependent to the quality of transmission [112]. In case of communication degradation, one possible solution is that the CACC inherently degrades to ACC, thus resulting in significantly larger time headway for string-stable behavior. In [119] proved that the minimum headway time increases 10 times to keep the string stable, which dramatically decreases the traffic throughput.

Previous researches focused mostly on the tracking control law of the CACC systems and the transmission delay is assumed to be a constant. Nevertheless, the data loss rate, which is an important factor that would degrade the transmission, is

hardly concerned. This subsection focuses on the signal degradation of the discrete sampling control system due to data loss.

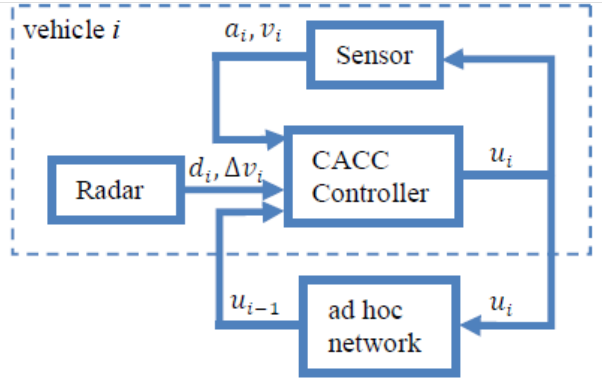


Figure 4.1 – *Structure of a vehicle's control system*

DSRC is a highly efficient wireless V2X communication technology. Among all, the IEEE 802.11p protocol is drawing much attention, which is analytically studied in a typical highway environment [19]. It is shown that this protocol with Quality of Service (QoS) support provides a relatively good guarantee for higher priority applications. As the PID controller is widely used in many researches, the influence of data transmission problems are widely studied on these systems. Both time-fixed and time-varying delay and data loss are discussed using mathematical and statistical simulation method respectively. The overshoot and the accommodation time are considered to be two main criterion of the performance, which can help evaluating the dependability of the system. Different conditions of control input can cause different levels of overshoot, thus consequently different accommodation time. The experiments in [195] and [194] show that the performance level is decided by the several beginning sampling points. Once the second sampling point is not lost, the performance will be at the first level with the overshoot 17% 18%. If the second sampling point is lost while the third one is not, the performance will be at the second level with the overshoot 54% 60%. Therefore, the overshoot of signal may occur due to data loss. Simulations results will show how CACC systems degrade if data loss appears in the input of desired acceleration of previous vehicle. Compared to V2V communication, the information gathered by laser ranging sensor is much more stable and reliable. The delay is about 8ms and the accuracy

is $\pm 2mm$. That's why the improvement of the quality of V2V communication is crucial for ITS.

4.3. Degradation of CACC

In previous section,a novel TVACACC system is proposed. The difference of the proposed system with the conventional CACC system is that in the feedforward term, the effect of two preceding vehicles' input u_{i-1} and u_{i-2} are included into the control loop. These inputs are implemented through wireless V2V communication. Consequently, if the wireless link fails or when the preceding vehicle is not equipped with CACC, CACC would degrade to ACC, leading to a significant increase in the minimum string-stable time headway. To implement an alternative fallback scenario that more gracefully degrades the CACC functionality, it is proposed to estimate the actual acceleration $\hat{a}_{i-1}$ of the preceding vehicle, which can then be used as a replacement of the desired acceleration u_{i-1} in case no communication updates are received.

4.3.1. Estimation of acceleration

4.3.1.1. Filter Kalman

Kalman filtering, is an algorithm that uses a series of measurements observed over time, containing statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more precise than those based on a single measurement alone, by using Bayesian inference and estimating a joint probability distribution over the variables for each timeframe. It is a useful tool to estimate the acceleration of previous vehicle in case of transmission lost. Therefore, a brief introduction of Kalman filter is introduced in this section.

Consider a continuous time-invariant model,

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t) \tag{4.1}$$

$$y(t) = Hx(t) + v(t) \tag{4.2}$$

where

- x, u and y are the state, system input and observation vector respectively;
- A is the state transition matrix;
- B is the control-input model;
- w is the input noise which is assumed to be drawn from a zero mean multivariate normal distribution with covariance Q;

$$w \sim \mathcal{N}(0, Q) \tag{4.3}$$

- H is the observation model which maps the true state space into the observed space;
- v is the observation noise which is assumed to be zero mean Gaussian white noise with covariance R;

$$v \sim \mathcal{N}(0, R) \tag{4.4}$$

- P is the error covariance matrix;
- w, v and x_0 are uncorrelated.

The Kalman filter is a recursive estimator, which means that only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state. In contrast to batch estimation techniques, no history of observations and/or estimates is required. The notation $\hat{x}$ represents the estimate value of x.

The Kalman filter is conceptualized as two distinct phases: "Predict" and "Update". The predict phase uses the state estimate from the previous time step to produce an estimate of the state at the current time step. This predicted state estimate is also known as the a prior state estimate because, although it is an estimate of the state at the current time step, it does not include observation information from the current time step. In the update phase, the current a prior prediction is

combined with current observation information to refine the state estimate. This improved estimate is termed the a posterior state estimate.

Predict phase

- Predicted state estimate $\hat{x} = Ax + Bu$
- Predicted estimate covariance $P = APA^T + Q$

Update phase

- Optimal Kalman gain $K = PH^T(HPH^T + R)^{-1}$
- Updated state estimate $\hat{x} = \hat{x} + K(y - H\hat{x})$
- Updated estimate covariance $P = (I - KH)P$

4.3.1.2. Dynamic model

To describe an object's longitudinal motion, the acceleration model in [146] is adopted, which is used to describe the longitudinal vehicle dynamics. Note that rigorous analysis of longitudinal vehicle behavior in everyday traffic, and the dynamic vehicle model may lead to other choices; this is, however, outside the scope of this paper. The singer acceleration model is defined by the following linear time-invariant system:

$$\dot{a}(t) = -\alpha a(t) + u(t) \tag{4.5}$$

where a is the acceleration of the host vehicle, u is the model input, α is a constant time due to maneuver, the choice of which will be briefly exemplified at the end of Section IV. The input u is chosen as a zero-mean uncorrelated random process (i.e., white noise) to represent throttle or brake action that may cause the host vehicle to accelerate or decelerate. To determine the variance of u, the object vehicle is assumed to obey physical limits with a maximum acceleration a_{max} and a maximum deceleration a_{min} with a probability P_{max} and P_{min} respectively. And the probability of zero acceleration is P_0, whereas other acceleration values are uniformly distributed with probability P_r, such that the sum of probabilities equals

to 1. Consequently, the mean of vehicle's acceleration $\bar{a}$ is equal to

$$\bar{a} = P_{max}a_{max} + P_{min}a_{min} + \int_{a_{min}}^{a_{max}} xP_r \, dx \tag{4.6}$$

Thus the acceleration variance is

$$\sigma_a^2 = (a_{max} - \bar{a})^2 P_{max} + (a_{min} - \bar{a})^2 P_{min} + (a_0 - \bar{a})^2 P_0 + \int_{a_{min}}^{a_{max}} (x - \bar{a})^2 P_r \, dx \tag{4.7}$$

It is shown in [146] that in order to satisfy $p(a)$, the covariance $C_u(\tau)$ of the white noise input u in 4.5 is

$$C_u(\tau) = 2\alpha\sigma_a^2\delta(\tau) \tag{4.8}$$

where δ is the unit impulse function. As a result, the random variable a in equation 4.5, satisfying a probability density function $p(a)$ with variance σ_a^2 is described, with with a white noise input u(t) satisfying equation 4.8.

Using the acceleration model 4.5, the corresponding equation of motion can be described in the state space as

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{4.9}$$

$$y(t) = Cx(t) \tag{4.10}$$

where $x^T = [s \;\; v \;\; a]$ in which s, v, a represent the host vehicle's position, velocity and acceleration respectively. The vector $y^T = [s \;\; v]$ is the output of the model, which is in practical measured by vehicle onboard sensor. The matrix A, B, and C are defined as

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\alpha \end{pmatrix}, \;\; B = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \;\; C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \tag{4.11}$$

Note that the state equation 4.9 closely resembles the vehicle dynamics model in equation 3.7 when replacing α by $1/\tau$.

The model 4.9 is used as a basis for the estimation of the object vehicle acceleration by means of a Kalman filter. To design this observer, the state-space model 4.9 is extended so as to include a process noise term $w(t)$, representing model uncertainty, and a measurement noise term $v(t)$, yielding

$$\dot{x}(t) = Ax(t) + w(t) \tag{4.12}$$

$$y(t) = Cx(t) + v(t) \tag{4.13}$$

The input $u(t)$ in equation 4.9, which was assumed to be white noise, is included in 4.12 by choosing $w(t) = Bu(t)$. $v(t)$ is a white noise signal with covariance matrix $R = E[v(t)v^T(t)]$, as determined by the noise parameters of the onboard sensor used in the implementation of the observer, Furthermore, using equation 4.8, the continuous-time process noise covariance matrix $Q = E[w(t)w^T(t)]$ is equal to

$$Q = BE[w(t)w^T(t)]B_a^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2\alpha\sigma_a^2 \end{pmatrix} \tag{4.14}$$

With the given Q and R matrix, the following continuous-time observer is obtained:

$$\dot{\hat{x}}(t) = A\hat{x} + K(y - C\hat{x}) \tag{4.15}$$

where $\hat{x}$ is the estimate of the object vehicle state $x^T = [s \;\; v \;\; a]$, K is the continuous-time Kalman filter gain matrix, and y is the measurement vector, consisting of position s and velocity v of the object vehicle. This observer provides a basis for the design of the fallback control strategy, as explained in the following subsection.

4.3.2. DTVACACC

The fallback CACC strategy, which is hereafter referred to as Degraded Two-Vehicle-Ahead CACC (DTVACACC), aims to use the observer 4.15 to estimate the acceleration a_{i-1} of the preceding vehicle, when the communication between the host and its nearest front vehicle is degraded. However, the measurement y in

equation 4.15, containing the absolute vehicle position and velocity, is not available. Instead, the onboard sensor of the host vehicle provides inter-vehicle distance and relative velocity. Consequently, the estimation algorithm needs to be adapted, as described below.

When the transmission of a_{i-1} is lost or badly degraded, the observer 4.15 is described in the Laplace domain by a transfer function $T(s)$, which takes the actual position s_{i-1} and velocity v_{i-1} of the preceding vehicle, contained in the measurement vector y, as input. The output of $T(s)$ is the estimate $\hat{a}_{i-1}$ of the preceding vehicle's acceleration, being the third element of the estimated state. This yields the estimator

$$\hat{a}_{i-1} = T(s) \begin{pmatrix} s_{i-1} \\ v_{i-1} \end{pmatrix} \tag{4.16}$$

where $\hat{a}_{i-1}(s)$ denotes the Laplace transform of $\hat{a}_{i-1}(t)$, and $s_{i-1}(s)$ and $v_{i-1}(s)$ are the Laplace transforms of $s_{i-1}(t)$ and $v_{i-1}(t)$ respectively. Moreover, the estimator transfer function $T(s)$ is is derived from equation 4.15:

$$T(s) = \hat{C}(sI - A - KC)^{-1}K \tag{4.17}$$

where $\hat{C} = [0 \;\; 0 \;\; 1]$.

The second step involves a transformation to relative coordinates,using the relation that

$$s_{i-1}(s) = d_i(s) + s_i(s) \tag{4.18}$$

$$v_{i-1}(s) = \Delta v_i(s) + v_i(s) \tag{4.19}$$

where $\Delta v_i(s)$ denotes the Laplace transform of the relative velocity $\Delta v_i(t) = \dot{d}_i(t)$. Substituting 4.18 and 4.19 into 4.16, we obtain

$$\hat{a}_{i-1}(s) = T(s) \begin{pmatrix} d_i(s) \\ \Delta v_i(s) \end{pmatrix} + T(s) \begin{pmatrix} s_i(s) \\ v_i(s) \end{pmatrix} \tag{4.20}$$

As a result, the acceleration estimator is, in fact, split into a relative coordinate estimator $\Delta\hat{a}_i(s)$ and an absolute coordinate estimator $\hat{a}_i(s)$, i.e., $\hat{a}_{i-1}(s) = \Delta\hat{a}_i(s) + \hat{a}_i(s)$.

$$\Delta\hat{a}_i(s) := T(s) \begin{pmatrix} d_i(s) \\ \Delta v_i(s) \end{pmatrix} \tag{4.21}$$

$$\hat{a}_i(s) := T(s) \begin{pmatrix} s_i(s) \\ v_i(s) \end{pmatrix} \tag{4.22}$$

where $\Delta\hat{a}_i(s)$ is the Laplace transform of the estimated relative acceleration $\Delta\hat{a}_i(t)$ and $\hat{a}_i(s)$ is the Laplace transform of the estimated local acceleration.

Finally, $\hat{a}_i(s)$ in 4.22 can be easily computed with

$$\begin{aligned} \hat{a}_i(s) = T(s) \begin{pmatrix} s_i(s) \\ v_i(s) \end{pmatrix} &= \begin{pmatrix} T_{as}(s) & T_{av}(s) \end{pmatrix} \begin{pmatrix} s_i(s) \\ v_i(s) \end{pmatrix} \\ &= (\frac{T_{as}(s)}{s^2} + \frac{T_{av}(s)}{s}) a_i(s) := T_{aa}(s) a_i(s) \end{aligned} \tag{4.23}$$

Using the fact that the local position $s_i(t)$ and velocity $v_i(t)$ are the result of integration of the locally measured acceleration $a_i(t)$, thereby avoiding the use of a potentially inaccurate absolute position measurement by means of a global positioning system. The transfer function $T_{aa}(s)$ acts as a filter for the measured acceleration a_i, yielding the "estimated" acceleration $\hat{a}_i$. In other words, the local vehicle acceleration measurement a_i is synchronized with the estimated relative acceleration Δa_i by taking the observer phase lag of the latter into account.

The control law of the fallback DTVACACC system is now obtained by replacing the preceding vehicle's input u_{i-1} in equation 3.6 by the estimated acceleration $\hat{a}_{i-1}$. As a result, the control law is formulated in the Laplace domain as

$$u_i(s) = H^{-1}(s)(K(s)e_i(s) + T(s) \begin{pmatrix} d_i(s) \\ \Delta v_i(s) \end{pmatrix} + T_{aa}(s)a_i(s)) \tag{4.24}$$

which can be implemented using the radar measurement of the distance d_i and the relative velocity Δv_i, and the locally measured acceleration a_i and velocity v_i, the latter being required to calculate the distance error e_i. The corresponding block diagram of the closed-loop DTVACACC system as a result of this approach

is shown in Figure. 4.2, which can be compared with Figure. 3.3, showing the TVACACC scheme.

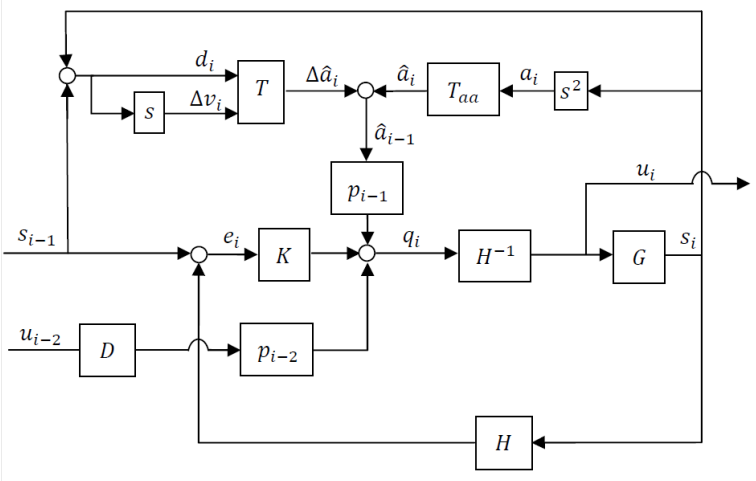


Figure 4.2 – *Block diagram of the DTVACACC system*

4.3.3. String stability analysis

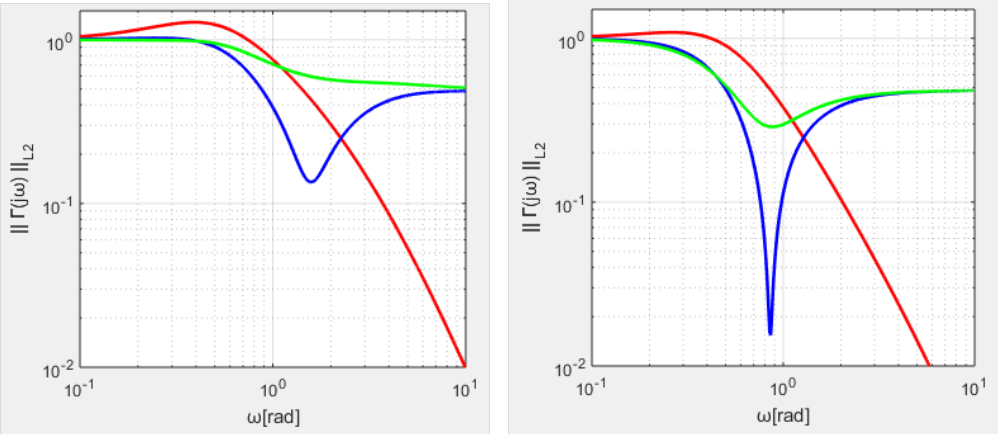
To analyze the DTVACACC string stability properties, the output of interest is chosen to be the acceleration. Recall that parameters are chosen as the same as we defined in previous chapter. $\tau = 0.1$, $k_p = 0.2$, $k_d = 0.7$, $k_{dd} = 0$ to avoid feedback of the and $h = 0.5s$, transmission delay θ=0.2s. Besides, the novel parameters for DTVACACC is defined as $a_{max} = 3m/s^2$, $a_{min} = -5m/s^2$, $P_{max} = P_{min} = 0.01$, $P_0 = 0.1$, $P_r = 0.11$, $\alpha = 1.25$, $\sigma_d^2 = 0.029$ and $\sigma_{\Delta v}^2 = 0.029$. As a result, with the closed-loop configuration given in Figure. 4.2, the transfer function is obtained:

$$\begin{aligned} ||\Gamma_{DTVACACC}(j\omega)||_{L_2} &= \frac{||a_i(s)||_{L_2}}{||a_{i-1}(s)||_{L_2}} \\ &= \frac{||G(s)K(s) + 0.5s^2 T_{aa} G(s) + 0.5D(s)/\Gamma_2(j\omega)||_{L_2}}{||H(s)(1 + G(s)K(s)||_{L_2}} \end{aligned} \tag{4.25}$$

where Γ_2 is the transfer function of second vehicle in the platoon which receives only one input from the leading vehicle. Therefore, it uses the conventional CACC system. The transfer function is the same as equation 3.26

$$||\Gamma_2(j\omega)||_{L_2} = \frac{||a_2(s)||_{L_2}}{||a_1(s)||_{L_2}} = \frac{||D(s) + G(s)K(s)||_{L_2}}{||H(s)(1 + G(s)K(s)||_{L_2}} \tag{4.26}$$

The platoon of vehicles is string stable if the infinite norm of the transfer function is less than 1, i.e., $||\Gamma_{DTVACACC}(j\omega)||_{L_\infty} \leq 1$. Furthermore, if the system is string unstable, $||\Gamma_{DTVACACC}(j\omega)||_{L_\infty}$ will exceed 1; still, in that case, we would aim at making this norm as low as possible to minimize disturbance amplification. The L_2 norm is here used to make a comparison between different CACC systems. The frequency response magnitudes $||\Gamma_{DTVACACC}(j\omega)||_{L_\infty}$ from 4.25, $||\Gamma_{TVACACC}(j\omega)||_{L_\infty}$ from 3.27, $||\Gamma_{ACC}(j\omega)||_{L_\infty}$ from 3.29 as a function of the frequency ω, are shown in Figure. 4.3a and 4.3b for different headway time $h = 0.5s$ and $h = 2s$, respectively.



(a) *headway time* $h = 0.5s$ (b) *headway time* $h = 2s$

Figure 4.3 – *Frequency response magnitude with different headway time, in case of (blue) TVACACC, (green) DTVACACC, and (red) ACC*

Recall the string stability criterion defined in equation 2.4, $||\Gamma_i(j\omega)||_{L_\infty} = \sup_\omega ||\Gamma_i(j\omega)|| \leq 1$. From the frequency response magnitudes, it follows that for $h = 0.5s$, only TVACACC system results in string-stable behavior that $||\Gamma_{TVACACC}(j\omega)||_{L_\infty} = 1$; whereas both DTVACACC and ACC system is not string stable, $||\Gamma_{DTVACACC}(j\omega)||_{L_\infty} = 1.0192$ and $||\Gamma_{ACC}(j\omega)||_{L_\infty} = 1.2782$. But even if the system is unstable, we try to find the lowest response to keep the disturbance amplification as small as possible. Therefore it is clear that the DTVACACC system helps to improve the performance compared to ACC system, in case of no communication from $i - 1$th vehicle.

As for $h = 1.3s$, both TVACACC and DTVACACC yield string stability. Clearly, ACC is still not string stable in either case. Here, $||\Gamma_{TVACACC}(j\omega)||_{L_\infty} = ||\Gamma_{DTVACACC}(j\omega)||_{L_\infty} = 1$, $||\Gamma_{ACC}(j\omega)||_{L_\infty} = 1.0859$. This is logical because increasing headway time helps to improve the string stability, which however results in large inter-vehicle distance and low traffic flow capicity.

4.3.4. Model switch strategy

Until now, either full wireless communication under nominal conditions or a persistent loss of communication has been considered. However, in practice, the loss of the wireless link is often preceded by increasing communication latency, represented by the time delay θ. Intuitively, it can be expected that above a certain maximum allowable latency, wireless communication is no longer effective, upon which switching from TVACACC to DTVACACC is beneficial in view of string stability. This section proves this intuition to be true and also calculates the exact switching value for the latency, thereby providing a criterion for activation of DTVACACC.

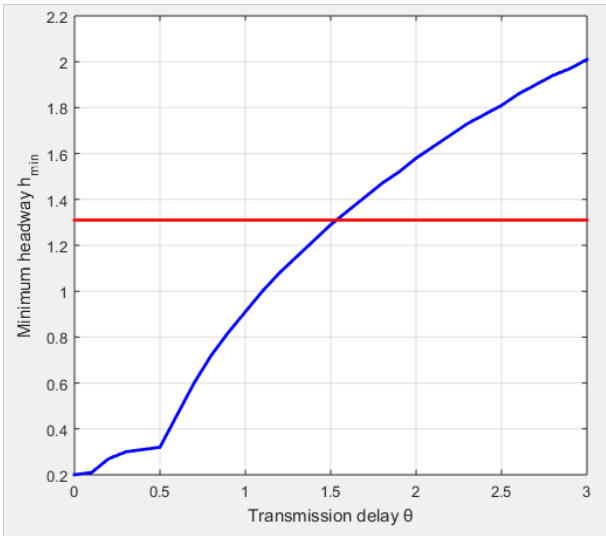


Figure 4.4 – *Minimum headway time (blue) $h_{min,TVACACC}$ and (red) $h_{min,DTVACACC}$ versus wireless communication delay θ*

From analysis of string stability of DTVACACC system in equation 4.25, it is shown that the magnitude of the transfer function changes its string stability when different headway time is chosen. This infinite norm value $||\Gamma_{TVACACC}(j\omega)||_{L_\infty}$ is

reduced by increasing headway time h, of which the effect is increasing the $H(s)$ in denominator. Consequently, for TVACACC, a minimum string-stable headway time $h_{min,TVACACC}$ must exist, which depends on the delay θ. Along the same line of thought, it can be shown that for DTVACACC, a minimum string-stable headway time also exists, which is obviously independent of the communication delay. Figure. 4.4 shows $h_{min,TVACACC}$ and $h_{min,DTVACACC}$ as a function of θ. Here, the minimum headway time is obtained by searching for the smallest h for each $\hat{\mathrm{I}}$, such that $||\Gamma_i(j\omega)||_{L_\infty} = 1$ for each system. This figure clearly shows a breakeven point θ_b of the delay θ, i.e., $h_{min,DTVACACC} = h_{min,TVACACC}(\theta_b)$, which is equal to $\theta_b = 1.53s$ for the current controller and acceleration observer. The figure also indicates that for $\theta \leq \theta_b$, it is beneficial to use TVACACC in view of string stability, since this allows for smaller time gaps, whereas for $\theta \geq \theta_b$, DTVCACC is preferred. This is an important result, since it provides a criterion for switching from TVACACC to DTVCACC and vice versa in the event that there is not (yet) a total loss of communication, although it would require monitoring the communication time delay when CACC is operational. As a final remark on this matter, it should be noted that the above analysis only holds for a communication delay that slowly varies, compared with the system dynamics. Moreover, it does not cover the situation in which data samples (packets) are intermittently lost, rather than delayed.

4.4. Simulation

To test the performance of the proposed model, a stop-and-go scenario is chosen because it is the most dangerous situation of all possible situations in longitudinal control. The platoon consists of several CACC equipped vehicles and they are assumed to be identical. The platoon starts at a constant speed of 30m/s. At t=50s, the leading vehicle of the platoon brakes with a deceleration of $-5m/s^2$, and reaccelerates until regaining the initial velocity 30m/s with an acceleration of $2m/s^2$ at t=70s. The results in different headway time will be shown. The numerous parameters are described in the table below.

The conventional ACC and TVACACC systems are introduced to make a clear comparison with the DTVACACC system. The first vehicle's acceleration is repre-

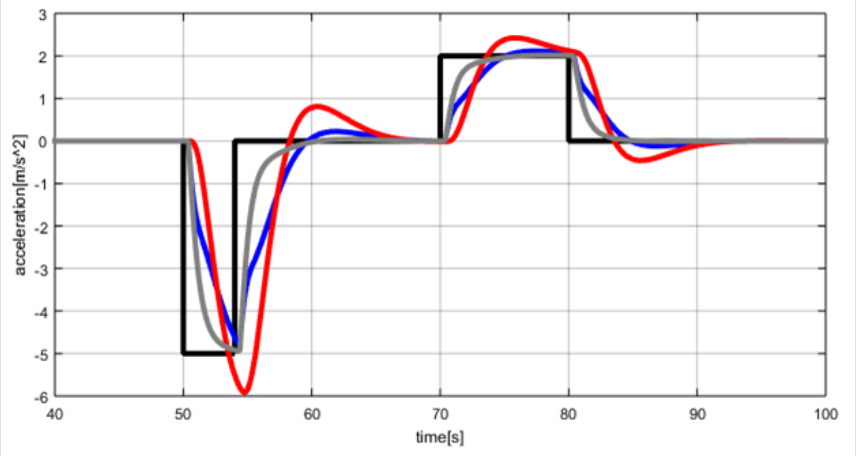


Figure 4.5 – *Acceleration response of the third vehicle in Stop-and-Go scenario using conventional ACC system (red), TVACACC system (gray) and DTVACACC system (blue) with a communication delay of* 1*s and headway* 0.5*s*

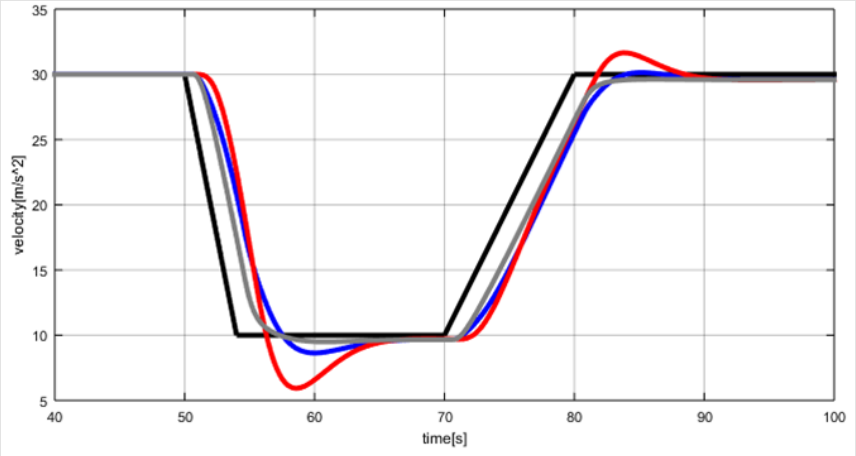


Figure 4.6 – *Velocity response of the third vehicle in Stop-and-Go scenario using conventional ACC system (red), TVACACC system (gray) and DTVACACC system (blue) with a communication delay of* 1*s and headway* 0.5*s*

sented in black line. And the third vehicle is chosen to investigate the difference between the conventional ACC system (red), TVACACC system (gray) and DTVA-CACC system (blue), shown in Figure. 4.5. We can see that each vehicle is following its preceding vehicle by respecting a safe distance with a headway time of 0.5*s*. However, the string stability criterion, is obviously not satisfied in existent ACC system as the absolute values of deceleration and acceleration are much greater than the first vehicle.The DTVACACC system is not string stable either. However we can see that the response is less overshoot. The vehicle is keeping a lower ac-

celeration and deceleration for the following objective. It is reasonable to conclude that the proposed acceleration estimate approach by Kalman filter helps to improve the string stability in case of V2V communication degradation. Similar results are obtained for velocity responses. The ACC system always responses greater than the leading vehicle. If a platoon consists of a large number of vehicles, the velocity, acceleration and spacing error will become extremely great in the upstream direction under the determined condition by using the existent ACC system, which is uncomfortable and dangerous. It is obvious that the proposed DTVACACC system outperforms ACC again, but still worse than the TVACACC system. Because the transmission of the front vehicle $i-1$th is degraded or lost.

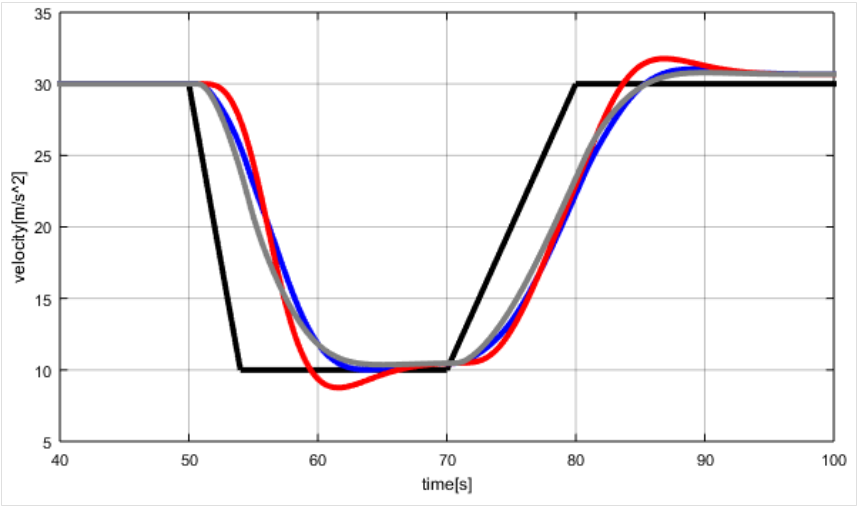


Figure 4.7 – *Velocity response of the third vehicle in Stop-and-Go scenario using conventional ACC system (red), TVACACC system (gray) and DTVACACC system (blue) with a communication delay of* 1*s and headway* 1.5*s*

As we have discussed above, increasing the headway time can improve the string stability. Therefore, different headway time of 1.5s and 3S are chosen to determine the improvement of the performance. If $h = 1.5$ shown in Figure. 4.7, the DTVACACC system is now string stable while ACC is still not. Then if we continue to increase headway time $h = 3s$ shown in Figure. 4.8, all three systems obtain the string stability. ACC system needs the largest headway time to keep the platoon string stable, then DTVACACC and finally TVACACC, which is the same as our theoretical analysis. The string instability is not only wasting energy, but also making the situation dangerous. Imagine a platoon of twenty vehicles, the last

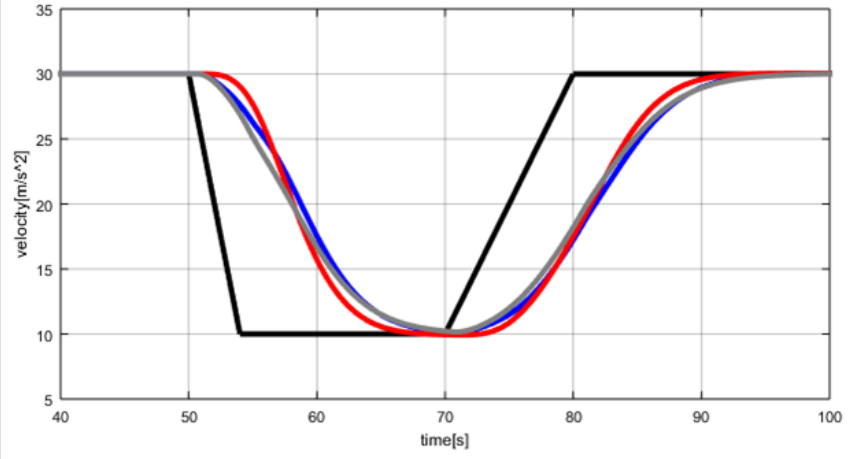


Figure 4.8 – *Velocity response of the third vehicle in Stop-and-Go scenario using conventional ACC system (red), TVACACC system (gray) and DTVACACC system (blue) with a communication delay of* 1*s and headway* 3*s*

vehicle will suffer from a hard brake and acceleration, even beyond its physical limit which may result in rear-end collision.

4.5. Conclusion

In this chapter, we concentrated on the degradation of CACC system.

To accelerate practical implementation of CACC in everyday traffic, wireless communication faults must be taken into account. To this end, a graceful degradation technique for CACC was presented, serving as an alternative fallback scenario to ACC. The idea behind the proposed approach is to obtain the minimum loss of functionality of CACC when the wireless link fails or when the preceding vehicle is not equipped with wireless communication means. The proposed strategy, which is referred to as DTVACACC, uses an estimation of the preceding vehicle's current acceleration as a replacement to the desired acceleration, which would normally be communicated over a wireless link for this type of CACC. In addition, a criterion for switching from TVACACC to DTVACACC was presented, in the case that wireless communication is not (yet) lost, but shows increased latency. It was shown that the performance, in terms of string stability of DTVACACC, can be maintained at a much higher level compared with an ACC fallback scenario. Both theoretical as well as experimental results showed that the DTVACACC system outperforms the

ACC fallback scenario with respect to string stability characteristics by reducing the minimum string-stable time gap to less than half the required value in case of ACC.

Chapter 5

Reinforcement Learning approach for CACC

Sommaire

5.1. Introduction

Endowing vehicles with human-like abilities to perform specific skills in a smooth and natural way is one of the important goals of ITS. Reinforcement learning (RL) is the key tool that helps us to create vehicles that can learn new skills by themselves, just similarly to our human beings. Reinforcement learning is realized by interacting with an environment. In RL, the learner is a decision-making agent that takes actions in an environment and receives an reinforcement signal for its actions in trying to accomplish a task. The signal, well known as reward (or penalty), evaluates an action's outcome, and the agent seeks to learn to select a sequence of actions, i.e. a policy, that maximize the total accumulated reward over time. Reinforcement learning can be formulated as a Markov Decision Process. Model-based RL algorithms can be used if we know the state transition function $T(s, a, s')$.

The whole learning scenario is a process of trial-and-error runs. We apply a Boltzmann probability distribution to tackle the problem of the exploration-exploitation trade-off, that is, the dilemma between should we exploit the past experiences and select the actions that as far as we know are beneficial, or should we explore some new and potentially more rewarding states. Under the circumstances, the policies are stochastic.

Analytic methods to ACC and CACC control problems are often different because of nonlinear dynamics and high-dimensional state spaces. Generally speaking, linearization is not sufficient to help solving this problem, thus it would be preferred to investigate new approaches, particularly RL, in which the knowledge of the Markov decision process (MDP) that sustains it is not necessary. In this chapter, a new RL approach to the CACC system that uses policy search is designed, i.e., directly modifying the parameters of a control policy based on obtained rewards. The policy-gradient method is adopted, because unlike other RL methods, it converges very well to high-dimensional systems. The advantages of the policy-gradient methods are obvious [114]. Among all the most important methods, it is indispensable that the policy representation can be chosen such that it is useful for the task, i.e., the domain knowledge can easily be incorporated. The proposed ap-

proach, in general, leads to fewer parameters in the learning process compared to other RL methods. Besides, there are already many different algorithms for policy-gradient estimation in the literature, most of which are based on strong theoretical foundations. Finally, we use policy-gradient methods for model free problem and it can therefore be applied to problems without analytically knowing task and reward models as well. Consequently, in this chapter, we propose a policy-gradient algorithm for CACC, where the algorithm repeatedly estimates the gradient of the value with respect to the parameters, based on the information observed during policy trials, and then updates the parameters in the upstream direction.

5.2. Related Work

Most of the researches on CACC systems have concerned about the classic control theory to develop autonomous controllers. However, recent projects based on machine-learning approach has been launched promising theoretical and practical results for the resolution of control problems in uncertain and partially observable environments, and it would be desirable to apply it on CACC. One of the first project efforts to use machine learning for autonomous vehicle control was Pomerleau's autonomous land vehicle in a neural network (ALVINN) [121], in which it consisted of a computer vision system, based on a neural network, that learns to correlate observations of the road to the correct action to take. The results are tested on autonomous controller which drove a real vehicle by itself for more than 30 miles. In [175, 179, 177], a RL based self-learning algorithm is designed in the cases where former experiences are not available to learning agents in advance and they are obliged to find a robust policy via interacting with the environment. Experiments are realized on the autonomous navigation tasks for mobile robots.

To our knowledge, Yu [185] was the first researcher that gave the idea to use RL for steering control. According to it, using RL approach allows control designers to remove the requirement for extra supervision and also to provide continuous learning abilities. RL is one of the machine-learning approach which has shown as the adaptive optimal control of a process $\mathcal{P}$, where the controller (called agent) interacts with $\mathcal{P}$ and learns to control it. To this end, the agent learns behavior

through trial-and-error interactions with $\mathcal{P}$. The agent then perceives the state of $\mathcal{P}$, and it select an action which maximizes the cumulative return that is based on a real-valued reward signal, which comes from $\mathcal{P}$ after each action. Thus, RL relies on modifying the control policy, which associates an action a to a state s, based on the state of the environment. Vehicle following has also been investigated in [110], using RL and vision sensor. Through RL, the control system indirectly learns the vehicle-road interaction dynamics, the knowledge of which is essential to stay on the road in high-speed road tracking.

In [43], the author has been directed toward obtaining a vehicle controller using instance-based RL. To this objective, the stored instances of past observations are used as values estimates for controlling autonomous vehicles that were extended to automobile control tasks. Simulations in an extensible environment and a hierarchical control architecture for autonomous vehicles have been realized. Particularly, the controllers proposed from this architecture were evaluated and improved in the simulator until difficult traffic scenarios are taken into account in a variety of (simulated) highway networks. However, this approach is, limited to the memory storage, which can be very rapidly developed when it deals with a realistic application.

More recently, in [107], an adaptive control system using gain scheduling learned by RL is proposed. In this research, they somehow kept the nonlinear nature of vehicle dynamics. This proposed controller performs better than a simple linearization of the longitudinal model, which is not be suitable for the entire operating range of the vehicle. The performance of the proposed approach at specific operating points shows accurate tracking ability of both velocity and positions in most cases. However in the case of adaptive controller deployed in a convoy or a platoon, the tracking performance is less desirable. In particular, the second car attempts to track the leader, resulting in slight oscillations. This oscillation is passed onto the following vehicles, but in the upstream direction of the platoon, the oscillations decrease, implying string stability. Thus, this approach is more convenient for platooning control than the CACC, because sometimes in this later case, it engenders slight oscillations.

Thus, although some researches are dedicated to the longitudinal control using RL, no researcher has particularly used RL for controlling CACC. In this chapter, we will try to fix this problem.

5.3. Neural Network Model

An artificial neural network (ANN) [136, 105] is organized in layers and each layer is composed of a bunch of "neuron" nodes. A neuron is a computational unit that can reads inputs, processes them and generates an output, see Figure 5.1 as an example.

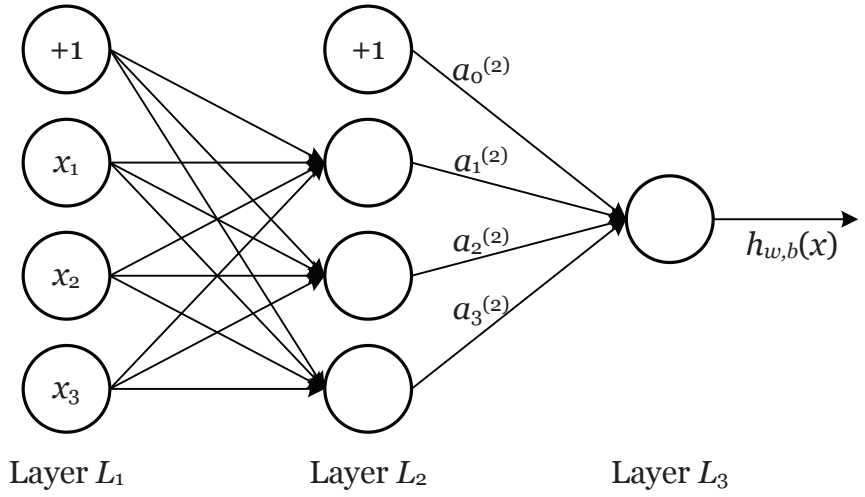


Figure 5.1 – *A neural network example*

The whole network is constructed by interconnecting many neurons. In this figure, one circle represents one neuron. The leftmost layer of the network is called the *input layer*, and the rightmost layer the *output layer*. The middle layer of nodes is called the *hidden layer*, since its values are not observed in the training set. The input layer and output layer serve respectively as the inputs and outputs of the neural network. The neurons labeled "+1" are called *bias units*. A bias unit has no input and always outputs +1. Hence, this neural network has 3 input units (excluding the bias unit), 3 hidden units (excluding the bias unit), and 1 output unit.

We use n_l to denote the number of layers and label each layer l as L_l. In Figure 5.1, $n_l = 3$, layer L_1 is the input layer, and layer L_{n_l} the output layer.

The links connecting two neurons are named *weights*, representing the connection strength between the neurons. The parameters inside the neural network are $(W, b) = (W^{(1)}, b^{(1)}, W^{(2)}, b^{(2)})$, where we write $W_{ij}^{(l)}$ to denote the weight associated with the connection between unit j in layer l, and unit i in layer $l+1$. Also, $b_i^{(l)}$ is the bias associated with unit i in layer $l+1$. Thus, we have $W^{(1)} \in \mathbb{R}^{3\times 3}$ and $W^{(2)} \in \mathbb{R}^{1\times 3}$.[1]

Each neuron in the network contains an *activation function* in order to control its output. We denote the activation of unit i in layer l by $a_i^{(l)}$. For the input layer L_1, $a_i^{(1)} = x_i$, the i-th input of the whole network. For the other layers, $a_i^{(l)} = f(z_i^{(l)})$. Here, $z_i^{(l)}$ denote the total weighted sum of inputs to unit i in layer l, including the bias term (e.g., $z_i^{(2)} = \sum_{j=1}^{n} W_{ij}^{(1)} x_j + b_i^{(1)}$), so that $a_i^{(l)} = f(z_i^{(l)})$.

Given a fixed setting of the parameters (W, b), the neural network outputs a real number that is defined as the hypothesis $h_{W,b}(x)$. Specifically, the computation that this neural network represents is given by:

$$\begin{aligned} a_1^{(2)} &= f(W_{11}^{(1)} x_1 + W_{12}^{(1)} x_2 + W_{13}^{(1)} x_3 + b_1^{(1)}), \\ a_2^{(2)} &= f(W_{21}^{(1)} x_1 + W_{22}^{(1)} x_2 + W_{23}^{(1)} x_3 + b_2^{(1)}), \\ a_3^{(2)} &= f(W_{31}^{(1)} x_1 + W_{32}^{(1)} x_2 + W_{33}^{(1)} x_3 + b_3^{(1)}), \\ h_{W,b}(x) = a_1^{(3)} &= f(W_{11}^{(2)} a_1^{(2)} + W_{12}^{(2)} a_2^{(2)} + W_{13}^{(2)} a_3^{(2)} + b_1^{(2)}). \end{aligned} \tag{5.1}$$

For a more compact expression, we can extend the activation function $f(\cdot)$ to apply to vectors in an element-wise fashion, i.e., $f([z_1, z_2, z_3]) = [f(z_1), f(z_2), f(z_3)]$, then we can write the equations above as:

[1] $b_i^{(l)}$ can also be interpreted as the connecting weight between the bias unit in layer l who always outputs +1 and the neuron unit i in layer $l+1$. Thus, $b_i^{(l)}$ may be replaced by $W_{i0}^{(l)}$. In this way, $W^{(1)} \in \mathbb{R}^{3\times 4}$ and $W^{(2)} \in \mathbb{R}^{1\times 4}$.

$$
\begin{aligned}
a^{(1)} &= x, \\
z^{(2)} &= W^{(1)} a^{(1)} + b^{(1)}, \\
a^{(2)} &= f(z^{(2)}), \\
z^{(3)} &= W^{(2)} a^{(2)} + b^{(2)}, \\
h_{W,b}(x) &= a^{(3)} = f(z^{(3)}).
\end{aligned}
\tag{5.2}
$$

$x = [x_1, x_2, x_3]^\top$ is a vector of values from the input layer. This computational process, from inputs to outputs, is called *forward propagation*. More generally, given any layer l's activation $a^{(l)}$, we can compute the activation $a^{(l+1)}$ of the next layer $l+1$ as:

$$
\begin{aligned}
z^{(l+1)} &= W^{(l)} a^{(l)} + b^{(l)}, \\
a^{(l+1)} &= f(z^{(l+1)}).
\end{aligned}
\tag{5.3}
$$

In this dissertation, we will choose $f(\cdot)$ to be the sigmoid function $f : \mathbb{R} \mapsto]-1, +1[$:

$$
f(z) = \frac{1}{1 + \exp(-z)}. \tag{5.4}
$$

Its derivative is given by

$$
f'(z) = f(z)(1 - f(z)). \tag{5.5}
$$

The advantage of putting all variables and parameters into matrices is that we can greatly speed up the calculation speed by using matrix-vector operations.

Neural networks can also have multiple hidden layers or multiple output units. Taking Figure 5.2 as an example, this network has two hidden layers L_2 and L_3 and two output units in layer L_4.

The forward propagation applies to all architectures of feedforward neural networks, i.e., to compute the output of the network, we can start with the input layer

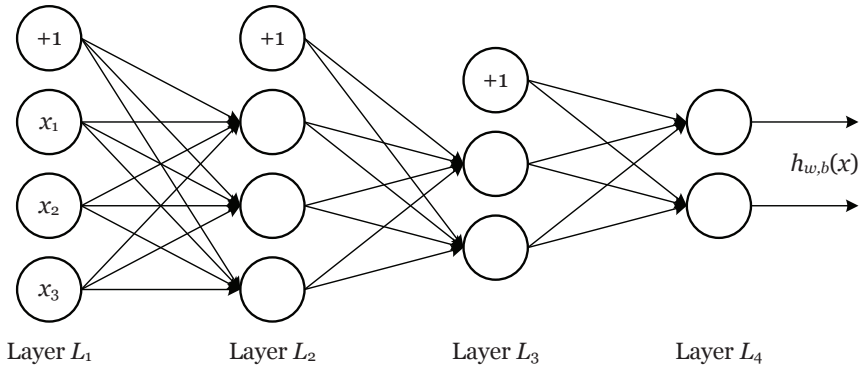


Figure 5.2 – *A neural network example with two hidden layers*

L_1, and successively compute all the activations in layer L_2, then layer L_3, and so on, up to the output layer L_{n_l}.

5.3.1. Backpropagation Algorithm

Suppose we have a fixed training set $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\}$ of m training examples. We can train our neural network using batch gradient descent. In detail, for a single training example (x, y), we define the cost function with respect to that single example to be:

$$J(W, b; x, y) = \frac{1}{2}\|h_{W,b}(x) - y\|^2. \tag{5.6}$$

This is a squared-error cost function. Given a training set of m examples, we then define the overall cost function $J(W, b)$ to be:

$$\begin{aligned} J(W, b) &= \left[\frac{1}{m}\sum_{i=1}^{m} J(W, b; x^{(i)}, y^{(i)})\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}\left(W_{ji}^{(l)}\right)^2 \\ &= \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\|h_{W,b}(x^{(i)}) - y^{(i)}\|^2\right)\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}\left(W_{ji}^{(l)}\right)^2. \end{aligned} \tag{5.7}$$

s_l denotes the number of nodes in layer l (not counting the bias unit). The first term in the definition of $J(W, b)$ is an average sum-of-squares error term. The second term is a regularization term that tends to decrease the magnitude of the weights, and helps prevent overfitting. Regularization is applied only to W but not

to b. λ is the regularization parameter which controls the relative importance of the two terms. Note that $J(W, b; x, y)$ is the squared error cost with respect to a single example; while $J(W, b)$ is the overall cost function that includes the regularization term.

The goal of the backpropagation is to minimize $J(W, b)$ as a function of W and b. To train the neural network, we first initialize each parameter $W_{ij}^{(l)}$ and each $b_i^{(l)}$ to a small random value near zero, and then apply an optimization algorithm such as batch gradient descent. It is important to initialize the parameters randomly, rather than to all 0's. If all the parameters start off at identical values, then all the hidden layer units will end up learning the same function of the input. More formally, $W_{ij}^{(1)}$ will be the same for all values of i, so that $a_1^{(2)} = a_2^{(2)} = a_3^{(2)} = \dots$ for any input x. The random initialization serves the purpose of symmetry breaking.

One iteration of gradient descent updates the parameters W, b as follows:

$$\begin{aligned} W_{ij}^{(l)} &= W_{ij}^{(l)} - \alpha \frac{\partial}{\partial W_{ij}^{(l)}} J(W, b), \\ b_i^{(l)} &= b_i^{(l)} - \alpha \frac{\partial}{\partial b_i^{(l)}} J(W, b). \end{aligned} \tag{5.8}$$

The parameter α is the learning rate. It determines how fast W and b move towards their optimal values. If α is very large, they may miss the optimal and diverge. If α is tuned too small, the convergence may need a long time.

The key step in Equation (5.8) is computing the partial derivatives terms of the overall cost function $J(W, b)$. Derived from Equation (5.7), we can easily obtain:

$$\begin{aligned} \frac{\partial}{\partial W_{ij}^{(l)}} J(W, b) &= \left[\frac{1}{m} \sum_{i=1}^{m} \frac{\partial}{\partial W_{ij}^{(l)}} J(W, b; x^{(i)}, y^{(i)}) \right] + \lambda W_{ij}^{(l)}, \\ \frac{\partial}{\partial b_i^{(l)}} J(W, b) &= \frac{1}{m} \sum_{i=1}^{m} \frac{\partial}{\partial b_i^{(l)}} J(W, b; x^{(i)}, y^{(i)}). \end{aligned} \tag{5.9}$$

One of the main tasks of the backpropagation algorithm is to compute the partial derivatives terms $\frac{\partial}{\partial W_{ij}^{(l)}} J(W, b; x^{(i)}, y^{(i)})$ and $\frac{\partial}{\partial b_i^{(l)}} J(W, b; x^{(i)}, y^{(i)})$ in Equation (5.9).

The backpropagation algorithm for one training example is shown as follows:

1. Perform a forward propagation, computing the activations for layers L_2, L_3, and so on up to the output layer L_{n_l}.

2. For each output unit i in the output layer n_l, set

$$\delta_i^{(n_l)} = \frac{\partial}{\partial z_i^{(n_l)}} \left(\frac{1}{2} \|y - h_{W,b}(x)\|^2 \right) = -(y_i - a_i^{(n_l)}) \cdot f'(z_i^{(n_l)}). \tag{5.10}$$

3. For $l = n_l - 1, n_l - 2, n_l - 3, \ldots, 2$:

   for each node i in layer l, set

$$\delta_i^{(l)} = \left(\sum_{j=1}^{s_{l+1}} W_{ji}^{(l)} \delta_j^{(l+1)} \right) f'(z_i^{(l)}). \tag{5.11}$$

4. Compute the desired partial derivatives, which are given as:

$$\begin{aligned} \frac{\partial}{\partial W_{ij}^{(l)}} J(W, b; x, y) &= a_j^{(l)} \delta_i^{(l+1)}, \\ \frac{\partial}{\partial b_i^{(l)}} J(W, b; x, y) &= \delta_i^{(l+1)}. \end{aligned} \tag{5.12}$$

Given a training example (x, y), we first run a forward propagation to compute all the activations throughout the network, including the output value of the hypothesis $h_{W,b}(x)$. Then, for each node i in layer l, we compute an error term $\delta_i^{(l)}$ that measures how much that node was "responsible" for any errors in our output. For an output node, we can directly measure the difference $\delta_i^{(n_l)}$ between the network's activation and the true target value, and for hidden units, we compute $\delta_i^{(l)}$ based on a weighted average of the error terms of the nodes that uses $a_i^{(l)}$ as an input.

In practice, we use matrix-vectorial operations to reduce the computational cost. We use "$\circ$" to denote the element-wise product operator [2]. By definition, if $C = A \circ B$, then

$$(C)_{ij} = (A \circ B)_{ij} = (A)_{ij} \cdot (B)_{ij}.$$

[2] Also called the Hadamard product.

The algorithm for one can then be written:

1. Perform a forward propagation, computing the activations for layers L_2, L_3, up to the output layer L_{n_l}, using the equations defining the forward propagation steps.

2. For the output layer n_l, set

$$\delta^{(n_l)} = -(y - a^{(n_l)}) \circ f'(z^{(n_l)}). \tag{5.13}$$

3. For $l = n_l - 1, n_l - 2, n_l - 3, \ldots, 2$, set

$$\delta^{(l)} = \left((W^{(l)})^\top \delta^{(l+1)}\right) \circ f'(z^{(l)}). \tag{5.14}$$

4. Compute the desired partial derivatives:

$$\begin{aligned} \nabla_{W^{(l)}} J(W, b; x, y) &= \delta^{(l+1)} \left(a^{(l)}\right)^\top, \\ \nabla_{b^{(l)}} J(W, b; x, y) &= \delta^{(l+1)}. \end{aligned} \tag{5.15}$$

In steps 2 and 3 above, we need to compute $f'(z_i^{(l)})$ for each value of i. Assuming $f(z)$ is the sigmoid activation function, we would already have $a_i^{(l)}$ stored away from the forward propagation throughout the whole network. Thus, using the Equation (5.5) for $f'(z)$, we can compute this as $f'(z_i^{(l)}) = a_i^{(l)}(1 - a_i^{(l)})$.

After getting all the partial derivatives that we desire, we can finally implement the gradient descent algorithm. One iteration of batch gradient descent is processed as follows:

1. Set $\Delta W^{(l)} := 0$, $\Delta b^{(l)} := 0$ (matrix/vector of zeros) for all l.

2. For $i = 1$ to m,

   (a) Use backpropagation to compute $\nabla_{W^{(l)}} J(W, b; x, y)$ and $\nabla_{b^{(l)}} J(W, b; x, y)$.

   (b) Set $\Delta W^{(l)} := \Delta W^{(l)} + \nabla_{W^{(l)}} J(W, b; x, y)$.

   (c) Set $\Delta b^{(l)} := \Delta b^{(l)} + \nabla_{b^{(l)}} J(W, b; x, y)$.

3. Update the parameters:

$$\begin{aligned} W^{(l)} &= W^{(l)} - \alpha \left[\left(\frac{1}{m} \Delta W^{(l)} \right) + \lambda W^{(l)} \right], \\ b^{(l)} &= b^{(l)} - \alpha \left[\left(\frac{1}{m} \Delta b^{(l)} \right) \right]. \end{aligned} \tag{5.16}$$

$\Delta W^{(l)}$ is a matrix of the same dimension as $W^{(l)}$, and $\Delta b^{(l)}$ is a vector of the same dimension as $b^{(l)}$.

To train the neural network, we can repeatedly take steps of gradient descent to reduce our cost function $J(W, b)$.

5.4. Model-Free Reinforcement Learning Method

In our work, We study reinforcement learning approach for longitudinal control problems of intelligent vehicles. The leading vehicle is taking random decisions and sequentially the following vehicles choose actions over a sequence of time steps, in order to maximize a cumulative reward. We model the problem as a Markov Decision Process: a state space $\mathcal{S}$, an action space $\mathcal{A}$, a transition dynamics distribution $P(s_{t+1} \mid s_t, a_t)$ satisfying the Markov property $P(s_{t+1} \mid s_1, a_1, ..., s_t, a_t) = P(s_{t+1} \mid s_t, a_t)$, for any trajectory $s_1, a_1, s_2, a_2, ..., s_T, a_T$ in state-action space, and a reward function $r : \mathcal{S} \times \mathcal{A} \longrightarrow \mathbb{R}$. A stochastic policy $\pi(s_t, a_t) = P(a_t \mid s_t)$ is used to select actions and produce a trajectory of states, actions and rewards $s_1, a_1, r_1, s_2, a_2, r_2, ..., s_T, a_T, r_T$ over $\mathcal{S} \times \mathcal{A} \times \mathbb{R}$.

An on-policy method learns the value of the policy that is used to make decisions. The value functions are updated using results from executing actions determined by some policy. An off-policy methods can learn the value of the optimal policy independently of the agent's actions. It updates the estimated value functions using hypothetical actions, those which have not actually been tried.

We focus on model-free RL methods that the vehicle drives an optimal policy without explicitly learning the model of the environment. Q-learning [172] algorithm is one of the major model-free reinforcement learning algorithms.

Q-Learning algorithm is an important off-policy model-free reinforcement learning algorithm for temporal difference learning. It can be proven that given

sufficient training under any ϵ-soft policy, the algorithm converges with probability 1 to a close approximation of the action-value function for an arbitrary target policy. Q-Learning learns the optimal policy even when actions are selected according to a more exploratory or even random policy.

The update of state-action values in Q-learning is defined by

$$Q(s_t, a_t) := Q(s_t, a_t) + \alpha \left[r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \right]. \tag{5.17}$$

The parameters used in the Q-value update process are:

α - the learning rate, set between 0 and 1. Setting it to 0 means that the Q-values are never updated, hence nothing is learned. Setting a high value such as 0.9 means that learning can occur quickly.

γ - discount factor, also set between 0 and 1. This models the fact that future rewards are worth less than immediate rewards. Mathematically, the discount factor needs to be set less than 0 for the algorithm to converge.

In this case, the learned action-value function, Q, directly approximates Q^*, the optimal action-value function, independent of the policy being followed. This dramatically simplifies the analysis of the algorithm and enabled early convergence proofs. The policy still has an effect in that it determines which state-action pairs are visited and updated. However, all that is required for correct convergence is that all pairs continue to be updated. Under this assumption and a variant of the usual stochastic approximation conditions on the sequence of step-size parameters, Q_t has been shown to converge with probability 1 to Q^*. The Q-learning algorithm is shown below.

5.5. CACC based on Q-Learning

One of the strengths of Q-learning is that it is able to compare the expected utility of the available actions without requiring a model of the environment. Q-learning can handle problems with stochastic transitions and rewards.

Algorithm 3: One-step Q-learning algorithm [172]

```
Initialize Q(s,a) arbitrarily;
repeat
    Initialize s;
    repeat
        Choose a from s using policy derived from Q;
        Take action a, observe r, s';
        Q(s,a) ← Q(s,a) + α [r + γ max_a' Q(s',a') − Q(s,a)];
        s ← s';
    until s is terminal
until all episodes end.
```

This section explains the design of an autonomous CACC system that integrates both sensors and inter-vehicle communication in its control loop to keep a secure longitudinal vehicle-following behavior. To this end, we will use the policy-gradient method that we described in the previous section to learn a vehicle control by direct interaction with a complex simulated driving environment. In this section, we will present the driving scenario simulated, show the learning simulations in detail, and evaluates the performance of the resulting policies.

The learning task concerned in this chapter is the same as previous chapters, corresponding to a Stop-and-Go scenario. This type of scenario is the most interesting, because it usually occurs on urban roads. It has been used by many researchers for the development of autonomous controllers and the evaluation of their efficiency and effects on the traffic flow. In this case, the learning vehicle's objective is to learn to follow the leading vehicle while keeping a specific defined range of 2 s.

5.5.1. State and Action Spaces

Since reinforcement learning algorithms can be modeled as an MDP, we need first to define the state space $\mathcal{S}$ and action space $\mathcal{A}$.

For the definition of the states, the following three state variables are considered:

- *headway time* H_ω: Headway time (also called the "range") is defined as the

distance in time from the front vehicle and is calculated as follows:

$$H_\omega = \frac{S_{Leader} - S_{Follower}}{V_{Follower}} \tag{5.18}$$

where S_{Leader} and $S_{Follower}$ are the position of leading vehicle and following vehicle respectively, $V_{Follower}$ is the velocity of the following vehicle. This measurement is widely adopted for inter-vehicle spacing that has the advantage of being dependent on the current velocity of the following vehcile. This state representation is also interesting, because it is independent of the velocity of its front vehicle which is good for a heterogeneous platoon. Thus, a behavior learned using these states will generalize to all the possible front vehicle velocities.

- *headway time derivative* ΔH_ω: Headway time derivative (also called the "range rate") contains valuable information about the relative velocity between the two vehicles and is expressed by

$$\Delta H_\omega = H_{\omega_t} - H_{\omega_{t-1}} \tag{5.19}$$

It shows whether the following vehicle is moving closer to or farther from the front vehicle since the previous update of the value. Both the headway and the headway can be derived by using a simulated laser sensor. Although continuous values are considered, we limit the range of the state space by bounding the value of these variables to specific intervals that is valuable experience to learn vehicle following behavior. Thus, the possible values of headway is bounded from 0 to $10s$, whereas the headway derivative is bounded from $-0.1s$ to $0.1s$.

- *Front-vehicle's acceleration* a_{i-1}: The acceleration of the front vehicle, which can be obtained through wireless V2V communication, is another important state variable of our system. The same as two previous state variables, the acceleration values are bounded to a particular interval, ranging from $-3m/s^2$ to $5m/s^2$.

Finally, the action space is composed of the following three actions: 1) a braking action (B); 2) a gas action (G); and 3) a non-operation action ($NO-OP$). The state and action space of our framework can formally be described as follows:

$$\mathcal{S} = \{H_\omega,\ \Delta H_\omega,\ a_{i-1}\} \tag{5.20}$$

$$\mathcal{A} = \{B,\ G,\ NO-OP\} \tag{5.21}$$

5.5.2. Reward Function

The progress of the learning phase depends on the reward function used by the agent, because this function is mostly used by the learning algorithm to direct the agent in areas of the state space where it will gather the maximum expected reward. It is used to evaluate how good or how bad the selected action is. Obviously, the reward function must be designed to be positive reward values to actions that get the agent toward the safe inter-vehicle distance to the preceding vehicle (see Figure 5.3).

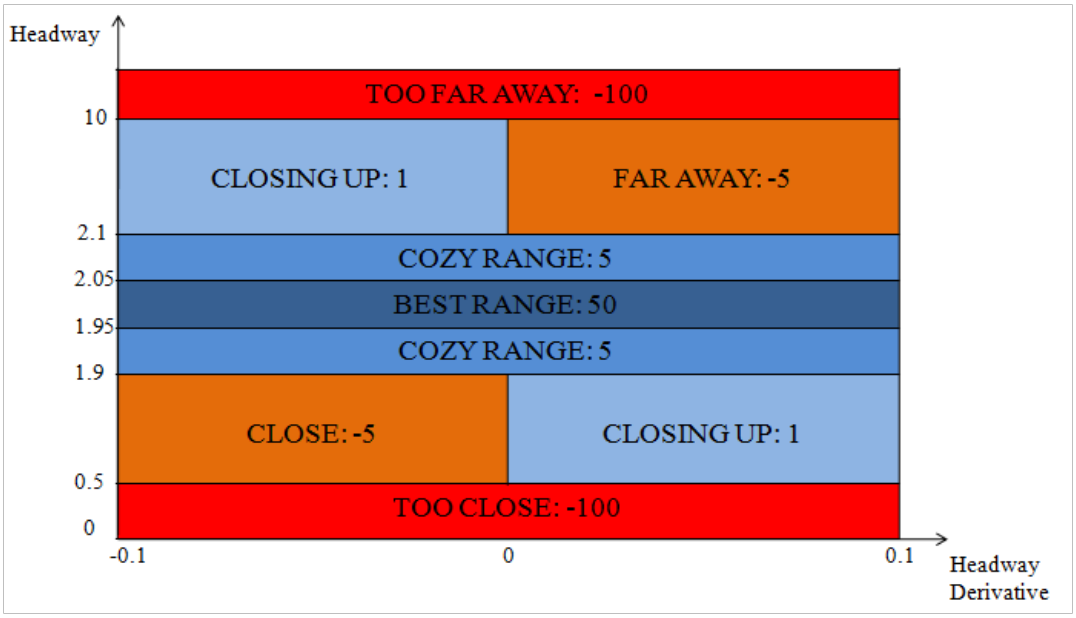


Figure 5.3 – *Reward of CACC system in RL approach*

As the secure inter-vehicle distance should be around the pre-defined value of

2 s (a common value in industrialized countries' legislation), we choose a large positive reward given when the vehicle enters the zone that extends at $\pm 0.1s$ from the headway goal of 2 s. Moreover, we also define a even smaller zone at $\pm 0.05s$ from the safe distance, where the agent receives the most important reward. The desired effect of such a reward function is to advise the agent to stay as close as possible to the safe distance. On the contrary, we give negative rewards to the vehicle when it is located very far from the safe distance or when it is too close to the preceding vehicle. To reduce learning times, we also use a technique called reward shaping, which directs the exploration of the agent by giving small positive rewards to actions that make the agent progress along a desired trajectory through the state space (i.e., by giving positive rewards when the vehicle is very far but gets closer to its front vehicle).

5.5.3. The Stochastic Control Policy

A reinforcement learning agent learns from the consequences of its state-action pairs rather than from being explicitly taught, and it selects its actions on basis of its past experiences and also by new choices. If we may visit each state-action (s, a) a sufficient large number of times, we could obtain the state values via, for example, Monte Carlo methods. However, it is not realistic, and even worse, many state-action pairs would not be visited once. It is important to deal with the exploration-exploitation trade-off.

In our work, we transplant a Boltzmann distribution to express a stochastic control policy. The learning agent tries out actions probabilistically based on their Q-values. Given a state s, the stochastic policy outputs an action a with probability:

$$\pi(s, a) = P(a \mid s) = \frac{e^{\frac{Q(s,a)}{T}}}{\sum_{b \in \mathcal{A}} e^{\frac{Q(s,b)}{T}}}. \tag{5.22}$$

where T is the temperature that controls the stochasticity of action selection. If T is high, all the action Q-values tend to be equal, and the agent choose a random action. If T is low, the action Q-values differ and the action with the highest Q-value is preferred to be picked. Thus, $P(a|s) \propto e^{\frac{Q(s,a)}{T}} > 0$ and $\sum_a P(a|s) = 1$.

We do not fix the temperature to a constant, since random exploration throughout the whole self-learning process takes too long to focus on the best actions.

At the beginning, all $Q(s,a)$ are generated inaccurately, so a high T is set to guarantee the exploration that all actions have a roughly equal chance of being selected. As time goes on, a large amount of random exploration have been done, and the agent could gradually exploit its accumulating knowledge. Thus, the agent decreases T, and the actions with the higher Q-values become more and more likely to be picked. Finally, as we assume Q is converging to Q^*, T approaches zero (pure exploitation) and we tend to only pick the action with the highest Q-value:

$$P(a|s) = \begin{cases} 1, & \text{if } Q(s,a) = \max_{b \in \mathcal{A}} Q(s,b) \\ 0, & \text{otherwise} \end{cases} \tag{5.23}$$

In sum, the agent starts with high exploration and converts to exploitation as time goes on, so that after a while we are only exploring (s,a)'s that have worked out at least moderately well before.

5.5.4. State-Action Value Iteration

The Q-value function expresses the mapping policy from the perceived state of environment to the executing action. One Q-value $Q(s_t, a_t)$ corresponds with one specific state and one action in this state. Like many RL researches, they have a large-scale state and action spaces. Traditionally, all the state or action values are store in a Q-table. However, this is not practical and computationally expensive for large-scale problems. In our method, We propose to predict all state Q-values by using a three-layer neural network, as shown in Figure 5.4.

The inputs are the state features that the robot perceives in the surrounding environment, and the outputs correspond to all the action Q-values. Therefore, according to Equation (5.20) and (5.21), the network has 3 neurons in the input layer, and 3 in the output layer. Moreover, 8 neurons are designed in the hidden layer.

The bias units are set to 1. The weight $W^{(1)} \in \mathbb{R}^{8\times 4}$ is used to connect the input layer and the hidden layer, and similarly, the weight $W^{(2)} \in \mathbb{R}^{3\times 9}$ links the

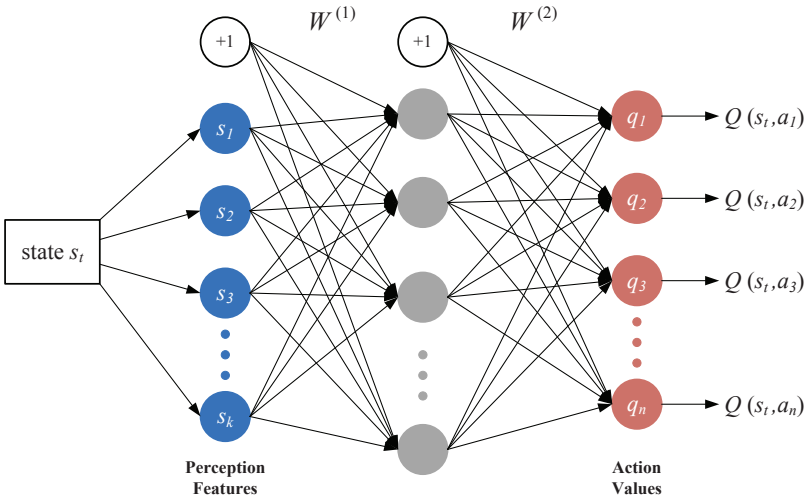


Figure 5.4 – *A three-layer neural network architecture*

hidden layer and the output layer. The sigmoid function is used for calculating the activation in the hidden and output layers.

We denote $Q(s_t)$ a vector of all action-values in the state s_t, and use $Q(s_t, a_t)$ to specify the Q-value of taking a_t in s_t. Thus,

$$Q(s_t) = \begin{bmatrix} Q(s_t, a_1) \\ Q(s_t, a_2) \\ Q(s_t, a_3) \end{bmatrix}.$$

The action value iteration is realized by updating the neural network by the means of its weights. In the previous chapter, the neural network was applied for supervised learning where the label for each training state-action pair was explicitly provided. Differently, the neural network in the reinforcement learning does not has label outputs. Q-learning is a process of value iteration and the optimal value after each iteration serves as the target value for neural network training. The update rule is

$$Q_{k+1}(s_t, a_t) = Q_k(s_t, a_t) + \alpha \left[r_t + \gamma \max_{a \in A} Q_k(s_{t+1}, a) - Q_k(s_t, a_t) \right]. \tag{5.24}$$

where the initial action values Q_0 of al the state-action pairs are generated ran-

domly between 0 and 1. $Q_{k+1}(s_t, a_t)$ is treated as the target value of the true value $Q_k(s_t, a_t)$ in the $(k+1)^{\text{th}}$ iteration.

In the vector $Q_k(s_t)$, only $Q_k(s_t, a_t)$ is updated to $Q_{k+1}(s_t, a_t)$, and the rest elements stay unchanged. Sometimes, $Q_{k+1}(s_t, a_t)$ may exceed the range $[0, 1]$, then we need to rescale $Q_{k+1}(s_t)$ to make sure all its components are in $[0, 1]$. We denote $\tilde{Q}_{k+1}(s_t)$ the rescaled action value. To make it clear, the update of Q-value is realized along the road $Q_k \to Q_{k+1} \to \tilde{Q}_{k+1}$.

The network error is a vector of form:

$$\delta_{k+1} = \tilde{Q}_{k+1}(s_t) - Q_k(s_t). \tag{5.25}$$

We employ the stochastic gradient descent (SGD) to train the neural network online. The goal is to minimize the cross-entropy cost function J defined as:

$$J = -\left[\sum_{i=1}^{N_A} (\tilde{Q}_{k+1})_i \cdot \log(Q_k)_i + (1 - (\tilde{Q}_{k+1})_i)(1 - \log(Q_k)_i)\right]. \tag{5.26}$$

where N_A is the number of actions used for training. In our navigation tasks, $N_A = 5$.

The action Q-values are nonlinear functions of weights of the network. SGD optimizes J and updates weights by using one or a few training examples according to:

$$W^{(i)} \leftarrow W^{(i)} - \alpha \frac{\partial J}{\partial W^{(i)}}. \tag{5.27}$$

Each iteration outputs new weights $W^{(i)}$ and a new cost J' is calculated. This update repeats until it arrives at a maximum times of iteration or $|J' - J| < \epsilon$.

5.5.5. Algorithm

A longitudinal control problem via NNQL can be divided into two processes. The first one is the training process to endow the vehicle with the self-learning ability, and the second one is the tracking process to use the trained policy to execute an independent tracking task.

5.5.5.1. Training Process of NNQL

Training the vehicle is done by exposing it to a bunch of learning episodes and each episode has a different environment. The variety helps the vehicle to encounter as many situations as possible, which could accelerate the learning speed.

The key of training efficiency is greatly related to how to make use of the accumulated sequence of state-action pairs and their Q-values. A bunch of previous work [124, 69, 179] used one-step Q-learning to update one Q-value at a time. When the vehicle is at a new state, only the new Q-value will be updated and the previous action values will be discarded. Others used batch learning [134] that updates all the Q-values once they are all collected. This also poses some advantages. First, without online update, we cannot guarantee that the collected Q-values have their optimal target values. Moreover, waiting all the values being obtained is always time-wasting. We propose to update online not only the current Q-value but also gather the previous values to train together.

The learning algorithm is given in Algorithm 4.

Algorithm 4: Training algorithm of NNQL

Initialize the NN weights $W^{(1)}$ and $W^{(2)}$ randomly;
for all episodes **do**
  Initialize the leading vehicle state;
  Read the sensor inputs;
  Observe current state s_1;
  $t \leftarrow 1$;
  for all moving steps **do**
    Compute all action-values $\{Q(s_t, a_i)\}_i$ in state s_t via NN;
    Select one action a_t according to the stochastic policy $\pi(s, a)$ in (5.22), and then execute;
    Observe new state s_{t+1} and state property p_{t+1};
    Obtain the immediate reward r_t;
    Update the Q-value function from $Q(s_t, a_t)$ to $\tilde{Q}(s_t, a_t)$ via (5.24);
    Apply feature scaling for $\tilde{Q}$ to the range $[0, 1]$;
    Apply SGD to train $(input, target)$ and to update the weights $W^{(1)}$ and $W^{(2)}$;
    $t \leftarrow t + 1$;
  end for
end for

5.5.5.2. Tracking Problem Using NNQL

After training the vehicle, the resulting policy is still stochastic but closed to deterministic that used by the vehicle for future tracking problems in various environments.

The tracking problem algorithm is shown in Algorithm 5.

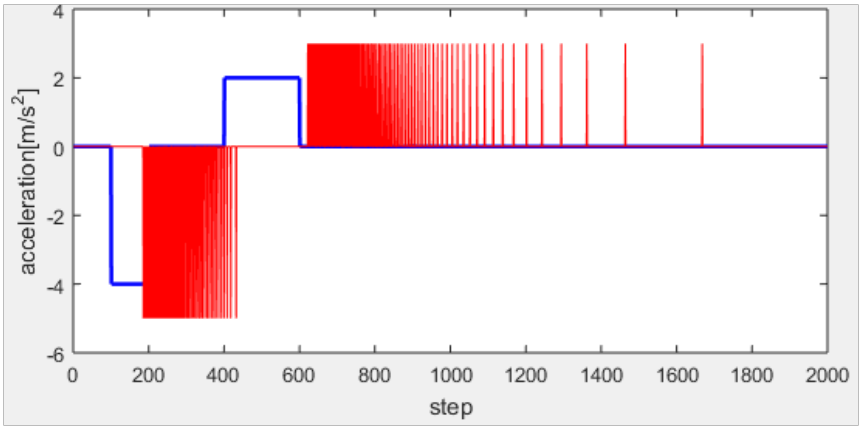
Algorithm 5: Tracking problem using NNQL

1: Load the trained NN weights $W^{(1)}$ and $W^{(2)}$;
2: Initialize the leading vehicle state randomly;
3: Load the vehicle initial state;
4: $t \leftarrow 1$;
5: **for** all moving steps **do**
6: Observe current state s_t and state property p_t;
7: Compute all action Q-values $\{Q(s_t, a_i)\}_i$ via neural network;
8: Pick the moving action a_t according to greedy policy, and then move;
9: **end for**

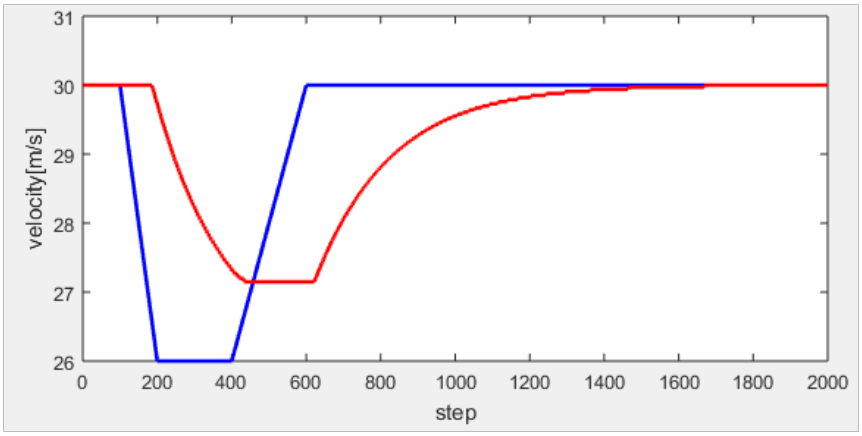
5.6. Experimental Results

Due to the stochastic property of the policy gradient algorithms, a hundred learning simulations that result in a hundred different control policies have been executed. After the learning phase, the policy that obtained the highest reward sum is chosen and is tested in the Stop-and-Go scenario that was used for learning. The results are presented in the Figures. In the figures it is shown respectively, their accelerations, the velocities of both vehicles, the headway time and the inter-vehicle distance in the simulation.

The headway response of the follow vehicle, as shown in Figure. 5.6b, indicates that, when the front vehicle is braking, the follower is able to keep a safe distance by using the learned policy. During this period, the headway of the follower oscillates close to the desired value of $2s$ (approximately from $1.95s$ to $2.05s$). Note that this oscillatory problem is due to the small number of discrete time steps that we defined in this simulation. From time steps 200 to 400, however, we can see that CACC operates the vehicle away from the desired headway that it gets closer to its front vehicle. This behavior can be resulted due to the fact that, at this time step,

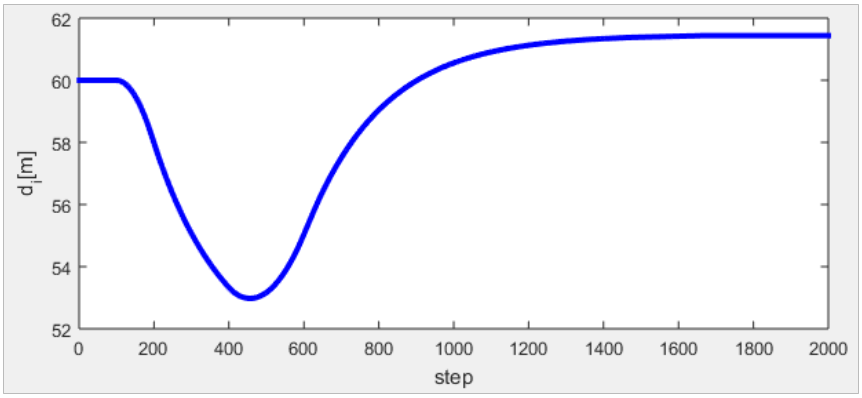


(a) *Acceleration response*

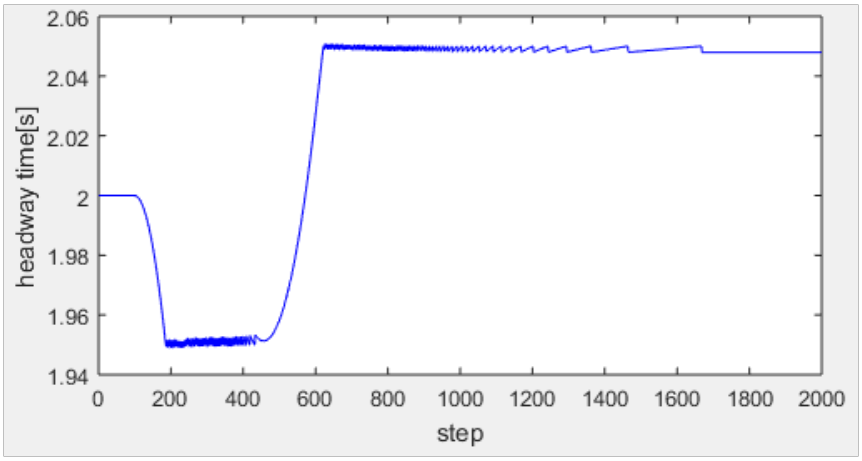


(b) *Velocity response*

Figure 5.5 – *Acceleration and velocity response of tracking problem using RL*



(a) *Inter-vehicle distance*



(b) *Headway time*

Figure 5.6 – *Inter-vehicle distance and headway time of tracking problem using RL*

the front vehicle has stopped accelerating. Thus, to select actions of the following vehicle, its controller observes a constant velocity (acceleration of 0) of the front vehicle and accordingly selects actions. In reality, at this time, the following vehicle is still rolling a faster than the front vehicle (as shown in the velocity profile in Fig. 5.5b). As consequence, the following vehicle has a tendency to get closer to the front vehicle, because it uses "no-op" actions, although it should still be braking for a small amount of time. The RL approach for CACC obtained is also interesting when looking at the acceleration response of the following vehicle. Obviously, in Fig. 5.5a it is shown that CACC does not need to use as much braking as the leader (around -4 m/s2), i.e. string stability is obtained. This is because of the defined actions, where only a deceleration of $-5m/s^2$ is considered.

Macroscopically, the performance is desirable, because it has shown that there is no amplification of velocity, which would result, within a vehicle platoon, in a complete halt of the traffic flow further down the stream. Thus, the string stability is kept and the presence of the acceleration signal of the leader enables the learning of a better control policy.

5.7. Conclusion

In this chapiter, we have proposed a novel design approach to obtain an autonomous longitudinal vehicle controller. To achieve this objective, a vehicle architecture with its CACC subsystem has been designed. With this architecture, we have also described the specific definitions for an efficient autonomous vehicle control policy through RL and the simulator in which the learning engine is embedded. The policy-gradient algorithm estimation is used to optimizer the policy and has used a back propagation neural network for achieving the longitudinal control. Then, experimental results, through Stop-and-Go scenario, have shown that this proposed RL approach results in efficient behavior for CACC.

Conclusions and Perspectives

Conclusions

In this thesis, we addressed the issue of CACC performance.

In chapter 1 a generally introduction to intelligent road transportation systems was presented. Firstly, the current traffic problems and situation were introduced. Then several historical researches worldwide were presented. In order to reduce the accidents caused by human errors, autonomous vehicles are being developed by research organizations and companies all over the world. Researches in autonomous vehicle development was introduced in this chapter as well. Secondly, ITS, AHS and intelligent vehicle were introduced, which are considered as the most promising solutions to the traffic problems. Thirdly, CACC as an extension of ACC systems by enabling the communication among the vehicles in a platoon, was then presented. CACC systems prevent the driver from repetitive jobs like adjusting speed and distance to the preceding vehicle. Fourthly, V2X communication, an important technology in developing ITS, was introduced. The VANETs are formed enabling communications among these agents, so that autonomous vehicles can be upgraded into cooperative systems, in which a vehicle's range of awareness can be extended. Finally, the technology of machine learning was introduced, which can be applied on intelligent vehicles.

Chapter 2 has presented the most important criterion to evaluate the performance of intelligent vehicle platoon, the string stability. Then the Markov decision processes were described in detail, which are the underlying structure of reinforcement learning. Several classical algorithms for solving MDPs were also briefly introduced. The fundamental concepts of the reinforcement learning was then brought.

Chapter3 concentrated on the vehicle longitudinal control system design. The spacing policy and its associated control law were designed with the constrains of string stability. The CTH spacing policy was adopted to determine the desired spacing from the preceding vehicle. It was shown that the proposed TVACACC system

could ensure both the stability of individual vehicle and the string stability. In addition, through the comparisons between the TVACACC and the conventional CACC and ACC systems, we could find the obvious advantages of the proposed system in improving traffic capacity especially in the high-density traffic conditions. The above proposed longitudinal control system was validated to be effective through a series of simulations in stop-and-go scenario.

In chapter4, a degradation approach for TVACACC was presented, used as an alternative fallback strategy to ACC. The concept of the proposed approach is to remain the minimum loss of functionality of TVACACC when the wireless communication is failed or when the preceding vehicle is not intelligent, which is not equipped with wireless communication units. The proposed degraded system, which is referred to as DTVACACC, uses the Kalman Filter to estimate the preceding vehicle's current acceleration to replace the desired acceleration, which is normally be communicated over a wireless V2V communication for the conventional CACC system. What's more, a switch criterion from TVACACC to DTVACACC was presented, in the case that wireless communication is not (yet) lost completely, but is suffering from increased transmission delay. Theoretical results have shown that the performance, in terms of string stability of DTVACACC, can be kept at a much higher level compared with an ACC fallback strategy. Both theoretical as well as experimental results have shown that the DTVACACC system outperforms the ACC fallback scenario by reducing the minimum string-stable time gap to less than half the required value in case of ACC.

Finally in chapter 5, we have proposed a novel approach to obtain an autonomous longitudinal vehicle cACC controller. To achieve this ovjective, a vehicle architecture with its CACC subsystem has been presented. Using this architecture, the specific requirements for an efficient autonomous vehicle control policy through RL and the simulator in which the learning engine is embedded are described. The policy-gradient algorithm estimation has been applied and we have used a back propagation neural network for achieving the longitudinal control. Then, through experimental results, through Stop-and-Go Scenario simulation, it is shown that this design approach can result in efficient behavior for CACC.

Future work

Much work can still be achieved to improve the performance of vehicle longitudinal controller proposed in this thesis.

- Further experimental validation of the proposed framework, TVACACC on real platoon is part of future research. Moreover, a various headway time and communication delay is required due to different factors, such as road condition and weather.

- The approach to estimate the front vehicle's acceleration in case of losing the V2V communication can be improved. In this thesis, we used typical filter Kalman for estimation based on the inter-vehicle distance and relative speed. Other technology of estimation can be applied to improve the performance of CACC systems.

- The state and action of vehicle in RL is not precisely defined. More factors of vehicle state and action should be taken into account. Issues of the oscillatory behavior of our vehicle control policy can be solved by using continuous actions. This approach would require further study to efficiently realize this method, because it causes additional complexity to the learning process.

- Some elements to our simulation of RL approach can also be improved, with the ultimate goal of having an even more realistic environment through which we can make our learning experiments. In fact, an important aspect to concern, as we did in chapter 3, would be to simulate a more accurate simulator for sensory and communication systems, which means sensor and communication delay, data loss and noise. These factors would make the learning process more complex, but the results would be much closer to real-life environments.

- Our controller can also be completed by extending an autonomous lateral control system. Again, this issue can be tackled using RL, and a potential solution is to use a reward function in the form of a potential function over the width of a lane, which is similar to the current force feedback given by

the existing lane-keeping assistance system. This reward function will surely direct the driving agent toward learning an adequate lane-change policy.

Résumé Étendu en Français

Introduction

Cette thèse est consacrée à la recherche de l'application de la théorie du contrôle intelligent dans les futurs systèmes de transport routier. A cause du développement de la société humaine, la demande de transport est beaucoup plus élevé que toute autre période de l'istoire. Plus flexibles et plus confortables, les voitures privées sont préférées par beaucoup de gens. En outre, le développement de l'industrie automobile réduit le coût de posséder une voiture, ainsi le nombre de voitures a augmenté rapidement dans le monde entier, surtout dans les métropoles. Toutefois, l'augmentation du nombre de voitures rend notre société à souffrir de la congestion du trafic, pollution des gaz et accidents. Ces effets négatifs nous exigent de trouver des solutions. Dans ce contexte, la notion de Systèmes de Transport Intelligents (ITS) est proposée. Les scientifiques et les ingénieurs travaillent depuis des décennies pour appliquer des technologies multidisciplinaires aux transports, afin d'avoir des systèmes plus stables, plus efficaces, plus d'économie d'effort, et environnemental amicale.

Une pensée est le système (semi-)autonome. L'idée principale est d'utiliser des applications pour aider ou remplacer l'opération humaine et la décision. Les systèmes d'Assistance Avancés au Conducteur (ADAS) sont conçus pour aider les conducteurs en les alertant lorsque le danger s'est produit (changement de la voie, avertissement de collision directe), fournissant de plus d'informations pour la prise de décision (plan d'itinéraire, évitement de la congestion) et libérant des manœuvres répétitives (régulateur de vitesse adaptatif, parking). Dans les systèmes semi-automatiques, le processus de conduite nécessite le conducteur humain: le conducteur doit définir certains paramètres dans le système, et il peut décider de suivre l'assistance consultative ou pas. Récemment, avec l'amélioration des technologies de détection et d'intelligence artificielle, les entreprises et les instituts se sont engagés dans la recherche et le développement de la conduite autonome. Dans certaines scénarios, par exemple des autoroutes et des routes principales, à l'aide de

capteurs et la carte très précis, les mains-off et pieds-off expériences de conduite seraient réalisées. L'élimination de l'erreur humaine rendra le transport routier beaucoup plus sécurisé et l'optimisation de l'espace entre véhicules améliorera l'utilisation de la capacité routière. Toutefois, les voitures ont encore besoin de l'anticipation du conducteur dans certains scénarios avec une situation de trafic compliquée ou des informations limitées. La structure intérieure des véhicules autonomes ne serait pas différente que celle des voitures actuelles, parce que le volant et les pédales sont toujours nécessaires. L'étape suivante de la conduite autonome est la conduite sans conducteur, c'est-à-dire la voiture est totalement conduit par lui-même. Le siège dédié au conducteur disparaîtrait et les gens à bord se concentreraient sur leur propre personnel. L'économie de l'auto-partage des voitures sans conducteur seraient énormes: à l'avenir, les gens préféreraient une voiture sans conducteur lorsqu'ils ont besoin d'une voiture privée. Ainsi, les congestions et les pollutions pourraient être soulagées.

Une autre penseé est le système coopératif. De toute évidence, pour le transport routier actuel les notifications sont conçu pour les conducteurs humains, tels que les feux de circulation et les panneaux latéraux. Les véhicules autonomes actuels sont équipés avec des caméras dédiées à la détection de ces signes. Toutefois, les notifications humaines n'est pas assez efficace pour les véhicules autonomes, car l'utilisation de la caméra est limitée par la portée et la visibilité, et des algorithmes doivent être conçus pour reconnaître ces signes. Si l'interaction entre les véhicules et l'environnement est activée, les notifications peuvent être transmises via les communications Vehicule-to-X (V2X). Ainsi les véhicules peuvent être remarqués dans la plus grande distance même au-delà de la vue, et les informations transmises sont plus précises que celles détectées par les capteurs. Quand le taux de communication des voitures sans conducteur est assez élevé, il ne serait plus nécessaire d'avoir des feux de circulation physiques et des panneaux. Le panneau de trafic personnel virtuel peut être communiquées aux véhicules individuels par le gestionnaire du trafic. Dans les systèmes coopératifs, un individu n'a pas besoin d'acquérir l'nformation tout par ses propres capteurs, mais avec l'aide des autres

par la communication. Par conséquent, l'intelligence autonome peut être étendue à l'intelligence coopérative.

La recherche présentée dans cette thèse 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 cible des systèmes coopératifs. Stratégies de contrôle sont conçues pour définir la méthode dont les véhicules interagissent les uns avec les autres.

Contributions Principales

Un nouveau système décentralisé de Régulateur de Vitesse Coopératif Adaptif à deux véhicules (TVACACC) est proposé dans ce document thèse. Il est montré que le contrôleur proposé avec deux entrées d'accéleration souaitée permet de réduire la distance entre véhicules, en utilisant une politique d'espacement dépendante de la vitesse. De plus, une approche de la stabilité dan le domaine fréquenciel est théoriquement analysée. En utilisant la communication multiple sans fil entre les véhicules, comparée au système conventionnel, une meilleure stabilité de chaîne est démontrée, qui entraîne une perturbation plus faible. La caravane des véhicules dans le scénario Stop-and-Go est simulé avec la communication de V2V dégradée. Il est montré que le système proposé donne un comportement stable de chaîne.

Une technique de dégradation gracieuse est proposé pour CACC, qui constitue un scénario alternatif de ACC. L'idée de l'approche proposée est d'obtenir la perte minimale de fonctionnalité de CACC lorsque la communication sans fil échoue ou le véhicule précédent n'est pas équipé de module de communication sans fil. La stratégie proposée, appelée TVACACC Dégradée (DTVACACC), utilise une estimation de l'accélération actuelle du véhicule précédent en remplacement de l'accélération souhaitée, qui est normalement communiquée par la communication sans fil.

Une nouvelle approche de conception pour obtenir un contrôleur de véhicule longitudinal autonome est proposé. Pour atteindre cet objectif, une architecture de véhicule CACC a été présenté. Avec cette architecture, nous avons décrit

les exigences spécifiques pour un contrôle autonome efficace des véhicules par l'Apprentissage de Renforcement (RL) et le simulateur dans lequel le moteur d'apprentissage est intégré. Une estimation d'algorithme de gradient de politique a été introduit et a utilisé un réseau neuronal de rétro-propagation pour le contrôle longitudinal.

Conlusions et Perspectives

Dans cette thèse, nous avons abordé le recherche de la performance du CACC. Au chapitre 1, une introduction aux systèmes intelligents de transport routier a été présenté. Tout d'abord, les problèmes de circulation et la situation actuelle ont été introduits. Ensuite, plusieurs recherches historiques ont été présentées dans le monde entier. Pour but de réduire les accidents causés par les erreurs humaines, les véhicules autonomes sont en cours de développement par des organismes de recherche et des entreprises partout dans le monde. Le développement des véhicules a également été introduit dans ce chapitre. Deuxiémement, ITS, AHS et le véhicule intelligent ont été introduits, qui sont considérés comme des solutions prometteuses aux problèmes de trafic. Troisièmement, le CACC en tant que prolongement du ACC systèmes en permettant la communication entre les véhicules d'une caravane, était alors présenté. Les systèmes CACC empêchent le conducteur de faire des tâches répétitives, en maintenant la vitesse et la distance inter-véhicules plus optimisées par rapport au ACC et CC systèmes. Quatrièmement, la communication V2X, une technologie importante dans le développement des ITS, a été introduite. Les VANET sont formés permettant la communication entre les agents, de sorte que les véhicules autonomes mise au point en systèmes coopératifs, dans lesquels la gamme de sensibilisation d'un véhicule est prolongée. Enfin, la technologie de l'apprentissage a été introduite, qui peut être appliqué sur les véhicules intelligents.

Le chapitre 2 a présenté le critère le plus important pour évaluer la performance d'une caravane de véhicules intelligents, la stabilité de chaîne. Puis la Décision du Markov Processus (MDP) a été décrite en détail, qui est la structure de l'Apprentissage de Renforcement (RI). Plusieurs algorithmes classiques pour ré-

soudre les MDP ont également été brièvement Introduits. Les concepts fondamentaux du RI ont été apportés.

Le chapitre 3 se concentre sur la conception du système de contrôle longitudinale du véhicule. La politique d'espacement et sa loi de contrôle associée ont été conçues avec les contraintes de stabilité de chaîne. La politique d'espacement CTH a été adoptée pour déterminer l'espacement souhaité du véhicule précédent. Il a été démontré que le système proposé TVACACC pourrait assurer à la fois la stabilité du véhicule individuelle et la stabilité de chaîne. En outre, à travers les comparaisons entre TVACACC, CACC conventionnel et ACC, nous avons prouvé les avantages évidents du système proposé dans l'amélioration de la capacité de trafic, en particulier dans les conditions de trafic à forte densité. Le système de contrôle longitudinal proposé a été validé par une série de simulations dans le scénario stop-and-go.

Au chapitre 4, une technique gracieuse de dégradation du CACC a été présentée, comme un scénario alternatif de rechange à ACC. L'idée de l'approche proposée est d'obtenir la perte minimale de fonctionnalité de CACC lorsque la liaison sans fil échoue ou lorsque le véhicule précédent n'est pas équipé d'une communication sans fil. La stratégie proposée, appelée DTVACACC, utilise le filtre Kalman pour estimer l'accélération actuelle du véhicule précédent en remplacement de l'accélération souaitée, qui est normalement communiquée par un lien sans fil pour ce type de CACC. En outre, un critère pour passer de TVACACC à DTVACACC a été présentée, dans le cas où la communication sans fil n'est pas (encore) perdue, mais montre un délai accru. Il a été démontré que la performance, en termes de la stabilité de chaîne de DTVACACC, peut être maintenu à un niveau beaucoup plus élevé qu'un système ACC. Les résultats théoriques et expérimentaux ont montré que le système DTVACACC surpasse ACC avec des caractéristiques de stabilité de chaîne en réduisant l'intervalle de temps minimum une moitié de la valeur requise dans le cas de ACC.

Enfin, dans le chapitre 5, nous avons proposé une nouvelle approche d'pprentissage pour obtenir un régulateur longitudinal de vitesse de véhicule. Pour parvenir à cette condition, une architecture de véhicule dans CACC a été

présentée. Avec cette architecture, nous avons également décrit les exigences spécifiques d'un véhicule autonom, la politique de contrôle par RL et le simulateur dans lequel le moteur d'apprentissage est intégré. Une méthode d'estimation d'algorithme, le gradient de politique, a été introduite et utilisé dans un réseau neuronal de rétro-propagation pour réaliser le contrôle longitudinal. Alors, les résultats expérimentaux, grâce à la simulation, ont montré que cette approche de conception peut entraîner un comportement efficace pour les CCAC.

Beaucoup de travail peut encore être fait pour améliorer le contrôleur de véhicule proposé dans cette thèse.

Validation expérimentale supplémentaire du cadre proposé, TVACACC sur une caravane de véhicules réels fait partie de la recherche future. En outre, une intervalle de temps et le retard de communication variés peut être prises en compte en raison de différents facteurs, par exemple la condition routière météorologique.

L'approche pour estimer l'accélération du véhicule précédent en cas de perte de la communication V2V peut être améliorée. Dans cette thèse, nous avons utilisé un filtre Kalman typique pour l'estimation basée sur la distance inter-véhicule et la vitesse relative. D'autres techniques d'estimation peuvent être appliquées pour améliorer le système CACC dégradé.

L'état et l'action du véhicule dans RL n'est pas précisément défini. Plus de facteurs de l'état du véhicule et de l'action doit être prise en compte. Problèmes relatives au comportement oscillatoire de notre politique de contrôle des véhicules peut être améliorés par des actions continues. Ce cas nécessiterait une étude plus approfondie pour cette approche, car elle apporte une complexité supplémentaire à l'apprentissage processus.

Certains éléments de notre simulation de l'approche RL peuvent également être améliorés, avec l'objectif ultime d'un environnement encore plus réaliste. En fait, un aspect important à considérer, comme nous l'avons fait au chapitre 3, serait d'intégrer un simulateur plus précis pour les systèmes sensoriels et de communication, ce qui signifie capteur et communication en retard, avec perte de données et bruit. Cette condition rendrait le processus de l'apprentissage plus complexe,

mais l'environnement qui en résulterait ressemblerait beaucoup plus aux conditions réelles.

Notre contrôleur peut également être complété par un système de contrôle latéral autonome. Encore une fois, cette approche peut être faite en utilisant RL. Une solution possible est d'utiliser une fonction de récompense sous la forme d'une fonction potentielle sur la voie, semblable à la rétroaction de la force actuelle donnée par la voie existante de système d'assistance. Cette fonction de récompense dirigera sûrement l'agent de conduite vers une politique de changement de voie adéquate.

Bibliography

[1] Pieter Abbeel, Adam Coates, and Andrew Y Ng. Autonomous helicopter aerobatics through apprenticeship learning. *The International Journal of Robotics Research*, 2010. (Cited page 33.)

[2] Pieter Abbeel, Adam Coates, Morgan Quigley, and Andrew Y Ng. An application of reinforcement learning to aerobatic helicopter flight. *Advances in Neural Information Processing Systems*, 19:1, 2007. (Cited page 33.)

[3] J. Abele, C. Kerlen, S. Krueger, H Baum, and et al. Exploratory study on the potential socio-economic impact of the introduction of intelligent safety systems in road vehicles. Final report, SEISS, Teltow, Germany, January 2005. (Cited page 14.)

[4] M. Alit, Z. Hou, and M. Noori. Stability and performance of feedbackcontrol sysytems with time delays. *Computer & Structures*, 66(2-3):241–248, 1998. (Cited page 82.)

[5] Card Andrew H. Hearing before the subcommitteee on investigations and oversight of the committee on science, space and technology. US. House of Representatives, 103 congress, First Session, PP. 108-109, US. Printing Office, November 1993. (Cited page 19.)

[6] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469–483, 2009. (Cited page 33.)

[7] Bassam Bamieh, Fernando Paganini, and Munther A Dahleh. Distributed

control of spatially invariant systems. *IEEE Transactions on Automatic Control*, 47(7):1091–1107, 2002. (Cited page 39.)

[8] E Barbieri. Stability analysis of a class of interconnected systems. *Journal of Dynamic Systems, Measurement, and Control*, 115(3):546–551, 1993. (Cited page 39.)

[9] Lakshmi Dhevi Baskar, Bart De Schutter, J Hellendoorn, and Zoltan Papp. Traffic control and intelligent vehicle highway systems: a survey. *Intelligent Transport Systems, IET*, 5(1):38–52, 2011. (Cited page 20.)

[10] Dimitri P Bertsekas. *Dynamic Programming and Optimal Control*, volume 1. Athena Scientific Belmont, Massachusetts, 1996. (Cited page 49.)

[11] RJ Betsold. Intelligent vehicle/highway systems for the united states-an emerging national program. In *Proceedings of JSK International Symposium-Technological Innovations for Tommorrow's Automobile Traffic and Driving Information Systems*, pages 53–59, 1989. (Cited page 18.)

[12] Gennaro Nicola Bifulco, Luigi Pariota, Fulvio Simonelli, and Roberta Di Pace. Development and testing of a fully adaptive cruise control system. *Transportation Research Part C: Emerging Technologies*, 29:156–170, 2013. (Cited page 61.)

[13] C Bonnet. Chauffeur 2 final report. *Deliverable D24, Version*, 1, 2003. (Cited page 26.)

[14] M. Bozorg and E. Davison. Control of time delay processes with uncertain delays:time delay stability margins. *Journal of Process Control*, 16:403–408, 2006. (Cited page 82.)

[15] Alberto Broggi, Paolo Medici, Paolo Zani, Alessandro Coati, and Matteo Panciroli. Autonomous vehicles control in the vislab intercontinental autonomous challenge. *Annual Reviews in Control*, 36(1):161–171, 2012. (Cited page 21.)

[16] F Broqua. 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*, 1991. (Cited page 24.)

[17] T. F. Buckley, P.H. Jesty, K. Hobley, and M. West. Drive-ing standards: a safety critical matter. In *Proceedings of the Fifth Annual Conference on Computer Assurance, Systems Integrity, Software Safety and Process Security*, pages 164–172, Gaithersburg, USA, June 1990. (Cited page 18.)

[18] Marc Carreras, Junku Yuh, Joan Batlle, and Pere Ridao. A behavior-based scheme using reinforcement learning for autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering*, 30(2):416–427, 2005. (Cited page 34.)

[19] Animesh Chakravarthy, Kyungyeol Song, and Eric Feron. Preventing automotive pileup crashes in mixed-communication environments. *IEEE Transactions on Intelligent Transportation Systems*, 10(2):211–225, 2009. (Cited pages 40 and 84.)

[20] Michelle Chandler. Google, baidu, tesla gunning self-driving car development. `www.investors.com/news/technology`, 2016. (Cited page 22.)

[21] S. Cheon. An Overview of Automated Highway Systems (AHS) and the Social and Institutional Challenges They Face. Report 624, University of California Transportation Center, 2002. (Cited page 19.)

[22] D. Cho and JK Hedrick. Automotive powertrain modeling for control. *Journal of Dynamic Systems, Measurement, and Control*, 111:568–576, 1989. (Cited page 60.)

[23] Kai-Ching Chu. Optimal dencentralized regulation for a string of coupled systems. *IEEE Transactions on Automatic Control*, 19(3):243–246, 1974. (Cited page 39.)

[24] Luis C Cobo, Kaushik Subramanian, Charles L Isbell, Aaron D Lanterman, and Andrea L Thomaz. Abstraction from demonstration for efficient re-

inforcement learning in high-dimensional domains. *Artificial Intelligence*, 216:103–128, 2014. (Cited page 33.)

[25] COMMISSION OF THE EUROPEAN COMMUNITIES. On the intelligent car initiative: Raising awareness of ict for smarter, safer and cleaner vehicles. Report COM(2006) 59 final, COMMISSION OF THE EUROPEAN COMMUNITIES, Februry 2006. (Cited page 17.)

[26] Mark Cummins and Paul Newman. Probabilistic appearance based navigation and loop closing. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 2042–2048. IEEE, 2007. (Cited page 33.)

[27] Ruth Curtain, Orest V Iftime, and Hans Zwart. System theoretic properties of a class of spatially invariant systems. *Automatica*, 45(7):1619–1627, 2009. (Cited page 39.)

[28] H Dahmani, M Chadli, A Rabhi, and A El Hajjaji. Road curvature estimation for vehicle lane departure detection using a robust takagi–sugeno fuzzy observer. *Vehicle System Dynamics*, 51(5):581–599, 2013. (Cited page 23.)

[29] S Darbha, KR Rajagopal, et al. Information flow and its relation to the stability of the motion of vehicles in a rigid formation. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 1853–1858. IEEE, 2005. (Cited page 39.)

[30] Alex Davies. 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/`, 2014. (Cited page 22.)

[31] Alex Davies. Baidu's self-driving car has hit the road. `www.wired.com`, 2015. (Cited page 22.)

[32] Dik De Bruin, Joris Kroon, Richard Van Klaveren, and Martin Nelisse. Design and test of a cooperative adaptive cruise control system. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 392–396. IEEE, 2004. (Cited page 23.)

[33] S. S. Dorle, D. M. Deshpande, A. G. Keskar, and M. Chakole. Vehicle classification and communication using zigbee protocol. *3rd International Conference on Emerging Trends in Engineering and Technology (ICETET)*, pages 106–109, 2010. (Cited page 29.)

[34] P. Eamsomboon, K.Phongsak, A. G. Keskar, and C. Mitrpant. The performance of wi-fi and zigbee networks for inter-vehicle communication in bangkok metropolitan area. *8th International Conference on ITS Telecommunications*, pages 408–411, 2008. (Cited pages 29 and 31.)

[35] Eurostat. Freight transport statistics. `http://ec.europa.eu/eurostat/statistics-explained/index.php/Freight_transport_statistics#Further_Eurostat_information`, 2016. (Cited page 9.)

[36] Eurostat. Passenger transport statistics. `http://ec.europa.eu/eurostat/statistics-explained/index.php/Passenger_transport_statistics`, 2016. (Cited page 9.)

[37] Eurostat. Road safety statistics at regional level. `http://ec.europa.eu/eurostat/statistics-explained/index.php/Road_safety_statistics_at_regional_level`, 2016. (Cited page 10.)

[38] J Eyre, D Yanakiev, and I Kanellakopoulos. A simplified framework for string stability analysis of automated vehicles. *Vehicle System Dynamics*, 30(5):375–405, 1998. (Cited page 43.)

[39] P Fancher. Intelligent cruise control field operational test. Technical report, University of Michigan Transportation Research Institute, 1998. (Cited page 24.)

[40] K. Fehrenbacher. Ford's "talking cars" could reduce crashes, fuel use. *gigaom.com*, 2010. (Cited page 29.)

[41] Lino Figueiredo, Isabel Jesus, JA Tenreiro Machado, J Ferreira, and JL Martins De Carvalho. Towards the development of intelligent transportation systems.

In *Intelligent Transportation Systems*, volume 88, pages 1206–1211, 2001. (Cited pages 18 and 24.)

[42] DK Fisher. Brake system component dynamic performance measurement and analysis. *SAE paper*, 700373:1157–1180, 1970. (Cited page 60.)

[43] Jeffrey Roderick Norman Forbes. *Reinforcement learning for autonomous vehicles*. PhD thesis, UNIVERSITY of CALIFORNIA at BERKELEY, 2002. (Cited page 104.)

[44] M. Freyssenet. Worldwide automobile production from 2000 to 2015 (in million vehicles). `http://www.oica.net/category/production-statistics/`, 2016. (Cited page 8.)

[45] Andreas Geiger, Martin Lauer, Frank Moosmann, Benjamin Ranft, Holger Rapp, Christoph Stiller, and Jens Ziegler. Team annieway's entry to the 2011 grand cooperative driving challenge. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1008–1017, 2012. (Cited page 27.)

[46] R. Ghostine, J. Thiriet, and J. Aubry. Variable delays and message losses: Influence on the reliability of a control loop. *Reliability Engineering & System Safety*, 96(1):160–171, 2011. (Cited page 83.)

[47] A González-Villaseñor, AC Renfrew, and PJ Brunn. A controller design methodology for close headway spacing strategies for automated vehicles. *International Journal of Control*, 80(2):179–189, 2007. (Cited page 40.)

[48] R. Goonewardene, A. Baburam, F.H. Ali, and E. Stipidis. Wireless ad-hoc networking for intelligent vehicles. *available on line:http://www.ee.ucl.ac.uk/lcs/previous/LCS2002/LCS069.pdf*, 2011. (Cited pages 29 and 31.)

[49] JW Grizzle, JA Cook, and WP Milam. Improved cylinder air charge estimation for transient air fuel ratio control. In *American Control Conference*. Citeseer, 1994. (Cited page 60.)

[50] Erico Guizzo. How google's self-driving car works. *IEEE Spectrum Online, October*, 18, 2011. (Cited page 21.)

[51] R. R. Guntur and H. Ouwerkerk. Adaptive brake control system. *Proceedings of the Institution of Mechanical Engineers*, 186:855–880, 1972. (Cited page 60.)

[52] RR Guntur and JY Wong. Some Design Aspects of Anti-Lock Brake Systems for Commercial Vehicles. *Vehicle System Dynamics*, 9(3):149–180, 1980. (Cited page 60.)

[53] Levent Guvenc, Ismail Meriç Can Uygan, Kerim Kahraman, Raif Karaahmetoglu, Ilker Altay, Mutlu Senturk, Mumin Tolga Emirler, Ahu Ece Hartavi Karci, Bilin Aksun Guvenc, Erdinç Altug, et al. Cooperative adaptive cruise control implementation of team mekar at the grand cooperative driving challenge. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1062–1074, 2012. (Cited page 27.)

[54] Donghoon Han and Kyongsu Yi. A driver-adaptive range policy for adaptive cruise control. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 220(3):321–334, 2006. (Cited page 24.)

[55] Shi-Yuan Han, Yue-Hui Chen, Lin Wang, and Ajith Abraham. 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, 2013. (Cited page 61.)

[56] Shi-Yuan Han, Yue-Hui Chen, Lin Wang, and Ajith Abraham. 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, 2013. (Cited page 66.)

[57] H. Hartenstein, B. Bochow, A. Ebner, M. Lott, M. Radimirsch, and D. Vollmer. Position-aware ad hoc wireless networks for inter-vehicle communications: the fleetnet project. *Proceeding on MobiHoc '01 Proceedings of the 2nd ACM International Symposium on Mobile ad hoc Networking & Computing*, pages 259–262, 2001. (Cited page 29.)

[58] H. Hedd, J. Rioult, M. Cuvelier, S. Ambellouis, M. S. Venant, and A. Rivenq. Technical evaluation of an electronic millimeter wave pre-view mirror. *IEEE Vehicular Technology Conference*, 5:2025–2032, 2000. (Cited page 29.)

[59] H. Hedd, J. Rioult, M. Klinger, A. Menhaj, , and C. Gransart. Microwave radio coverage for vehicle-to-vehicle and in-vehicle communication. *8th World Congress on Intelligent Transport Systems*, 2001. (Cited page 29.)

[60] McMahon D. Narendran V. Swaroop D. Hedrick, J.K. Longitudinal vehicle controller design for ivhs systems. In *American Control Conference*, pages 3107 –3112, June 1991. (Cited page 60.)

[61] G. Held. *Inter- and Intra- Vehicle Communications*. Auerbach Publishers Inc., 2007. (Cited page 29.)

[62] R. Horowitz, C.W. Tan, and X. Sun. An efficient lane change maneuver for platoons of vehicles in an automated highway system. Report UCB-ITS-PRR-2004-16, UC Berkeley, California PATH, May 2004. (Cited page 40.)

[63] O. Imera, S. YÃ$\frac{1}{4}$kselb, and T. Basar. Optimal control of lti systems over unreliable communication links. *Automatica*, 42:1429–1439, 2006. (Cited page 82.)

[64] Intel. Building an intelligent transportation system with the the internet of things (iot). `http://www.intel.cn/content/www/cn/zh/internet-of-things`, 2015. (Cited page 11.)

[65] PA Ioannou, F. Ahmed-Zaid, and D. Wuh. A time headway autonomous intelligent cruise controller: Design and simulation. Research Report UCB-ITS-PWP-94-07, California PATH, April 1994. (Cited page 64.)

[66] ISO 15628:2013. Intelligent transport systems – dedicated short range communication (DSRC) – dsrc application layer. `http://www.iso.org/iso/home/store/catalogue_ics/`, 2013. (Cited page 23.)

[67] V. Milanes E. Onieva J. Perez, A. Gajate and M. Santos. Design and implementation of a neuro-fuzzy system for longitudinal control of autonomous

vehicles. In *Fuzzy Systems (FUZZ), 2010 IEEE International Conference on*, pages 1–6. ieee, 2010. (Cited page 61.)

[68] Janet. *Strategic Plan for IVHS in the United States*. IVHS, AMERICA, 1992. (Cited page 18.)

[69] Mohammad Abdel Kareem Jaradat, Mohammad Al-Rousan, and Lara Quadan. Reinforcement based mobile robot navigation in dynamic environment. *Robotics and Computer-Integrated Manufacturing*, 27(1):135–149, 2011. (Cited pages 34 and 121.)

[70] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, pages 237–285, 1996. (Cited page 53.)

[71] J. Kenny. Dedicated short-range communications (dsrc)standards in the united states. *Proceedings of IEEE*, 99(7):1162–1182, 2011. (Cited page 28.)

[72] Maziar E Khatir and Edward J Davison. Decentralized control of a large platoon of vehicles using non-identical controllers. In *American Control Conference, 2004. Proceedings of the 2004*, volume 3, pages 2769–2776. IEEE, 2004. (Cited page 40.)

[73] Roozbeh Kianfar, Bruno Augusto, Alireza Ebadighajari, Usman Hakeem, Johan Nilsson, Arif Raza, Reza S Tabar, Naga V Irukulapati, Cristofer Englund, Paolo Falcone, et al. Design and experimental validation of a cooperative driving system in the grand cooperative driving challenge. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):994–1007, 2012. (Cited page 27.)

[74] J. Kim, W. Han, W. Choi, Y. Hwang, T. Kim, J. Jang, J. Um, and J. Lim. Performance analysis on mobility of ad-hoc network for inter-vehicle communication. *Proceedings of the Fourth Annual ACIS International Conference on Computer and Information Science*, pages 528–533, 2005. (Cited pages 29 and 31.)

[75] Steffi Klinge and Richard H Middleton. String stability analysis of homogeneous linear unidirectionally connected systems with nonzero initial condi-

tions. In *Signals and Systems Conference (ISSC 2009), IET Irish*, pages 1–6. IET, 2009. (Cited page 39.)

[76] Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, page 0278364913495721, 2013. (Cited page 54.)

[77] J Zico Kolter and Andrew Y Ng. Policy search via the signed derivative. In *Robotics: Science and Systems*, 2009. (Cited page 34.)

[78] Petar Kormushev, Sylvain Calinon, and Darwin G Caldwell. Robot motor skill coordination with em-based reinforcement learning. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 3232–3237. IEEE, 2010. (Cited page 34.)

[79] Kirsten Korosec. Tesla: This is our most significant step towards safe self-driving cars. `http://fortune.com/2016/02/09/tesla-self-parking/`, 2016. (Cited page 22.)

[80] M Koshi. 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*, pages 9–19, 1989. (Cited page 18.)

[81] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *arXiv preprint arXiv:1504.00702*, 2015. (Cited page 34.)

[82] Shengbo Li, Keqiang Li, Rajesh Rajamani, and Jianqiang Wang. Model predictive multi-objective vehicular adaptive cruise control. *IEEE Transactions on Control Systems Technology*, 19(3):556–566, 2011. (Cited page 61.)

[83] Chi-Ying Liang and Huei Peng. Optimal adaptive cruise control with guaranteed string stability. *Vehicle System Dynamics*, 32(4-5):313–330, 1999. (Cited page 39.)

[84] Bin-Feng Lin, Yi-Ming Chan, Li-Chen Fu, Pei-Yung Hsiao, Li-An Chuang, Shin-Shinh Huang, and Min-Fang Lo. Integrating appearance and edge features for sedan vehicle detection in the blind-spot area. *IEEE Transactions on Intelligent Transportation Systems*, 13(2):737–747, 2012. (Cited page 23.)

[85] Bing Liu and Abdelkader El Kamel. V2x-based decentralized cooperative adaptive cruise control in the vicinity of intersections. *IEEE Transactions on Intelligent Transportation Systems*, 17(3):644–658, 2016. (Cited page 27.)

[86] Xiao-Yun Lu, J Karl Hedrick, and Mike Drew. Acc/cacc-control design, stability and robust performance. In *American Control Conference, 2002. Proceedings of the 2002*, volume 6, pages 4327–4332. IEEE, 2002. (Cited page 23.)

[87] Li-hua Luo, Hong Liu, Ping Li, and Hui Wang. Model predictive control for adaptive cruise control with multi-objectives: comfort, fuel-economy, safety and car-following. *Journal of Zhejiang University SCIENCE A*, 11(3):191–201, 2010. (Cited page 61.)

[88] Minzhi Luo, Abdelkader El Kamel, and Guanghong Gong. Uml-based design of intelligent vehicles virtual reality platform. In *2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 115–120. IEEE, 2011. (Cited page 26.)

[89] Duncan Mackinnon. High capacity personal rapid transit system developments. *IEEE Transactions on Vehicular Technology*, 24(1):8–14, 1975. (Cited page 24.)

[90] G. Marsden, M. McDonald, and M. Brackstone. Towards an understanding of adaptive cruise control. *Transportation Research Part C*, 9(1):33–51, 2001. (Cited pages 61 and 64.)

[91] DH McMahon, VK Narendran, D. Swaroop, JK Hedrick, KS Chang, and PE Devlin. Longitudinal vehicle controllers for IVHS: Theory and experiment. In *Proceedings of the 1992 American Control Conference*, pages 1753–1757, Chicago, 1992. (Cited page 60.)

[92] Hedrick J. K. Shladover S. E. McMahon, D. H. Vehicle modelling and control for automated highway systems. In *American Control Conference*, pages 297 –303, May 1990. (Cited page 60.)

[93] SM Melzer and BC Kuo. Optimal regulation of systems described by a countably infinite number of objects. *Automatica*, 7(3):359–366, 1971. (Cited page 39.)

[94] Richard H Middleton and Julio H Braslavsky. String instability in classes of linear time invariant formation control with limited communication range. *IEEE Transactions on Automatic Control*, 55(7):1519–1530, 2010. (Cited page 40.)

[95] Vicente Milanés, Steven E Shladover, John Spring, Christopher Nowakowski, Hiroshi Kawazoe, and Mitsutoshi Nakamura. Cooperative adaptive cruise control in real traffic situations. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):296–305, 2014. (Cited page 38.)

[96] Harvey J Miller and Shih-Lung Shaw. *Geographic information systems for transportation: principles and applications*. Oxford University Press on Demand, 2001. (Cited page 23.)

[97] LUO Minzhi, Abdelkader EL KAMEL, and GONG Guanghong. Simulation of natural environment impacts on intelligent vehicle based on a virtual reality platform. *IFAC Proceedings Volumes*, 45(24):116–121, 2012. (Cited page 26.)

[98] J. Misener, R. Sengupta, and H. Krishnan. Cooperative collision warning:enabling crash avoidance with wireless technology. *12th World Congress on ITS*, 3:1–11, 2005. (Cited page 29.)

[99] Michael Montemerlo, Jan Becker, Suhrid Bhat, Hendrik Dahlkamp, Dmitri Dolgov, Scott Ettinger, Dirk Haehnel, Tim Hilden, Gabe Hoffmann, Burkhard Huhnke, et al. Junior: The stanford entry in the urban challenge. *Journal of field Robotics*, 25(9):569–597, 2008. (Cited pages 20 and 33.)

[100] Brendan Morris, Anup Doshi, and Mohan Trivedi. Lane change intent pre-

diction for driver assistance: On-road design and evaluation. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 895–901. IEEE, 2011. (Cited page 23.)

[101] JJ Moskwa and JK Hedrick. Automotive engine modeling for real time control application. In *American Control Conference*, pages 341–346, 1987. (Cited page 60.)

[102] Katharina Mülling, Jens Kober, Oliver Kroemer, and Jan Peters. Learning to select and generalize striking movements in robot table tennis. *The International Journal of Robotics Research*, 32(3):263–279, 2013. (Cited page 34.)

[103] Gerrit Naus, Jeroen Ploeg, Rene van de Molengraft, and Maarten Steinbuch. Explicit mpc design and performance-based tuning of an adaptive cruise control stop-&-go. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 434–439. IEEE, 2008. (Cited page 61.)

[104] Gerrit JL Naus, Rene PA Vugts, Jeroen Ploeg, Marinus JG van de Molengraft, and Maarten Steinbuch. String-stable cacc design and experimental validation: A frequency-domain approach. *IEEE Transactions on Vehicular Technology*, 59(9):4268–4279, 2010. (Cited page 40.)

[105] Andrew Ng. Sparse autoencoder. *CS294A Lecture notes*, 72, 2011. (Cited page 105.)

[106] Andrew Y Ng, Adam Coates, Mark Diel, Varun Ganapathi, Jamie Schulte, Ben Tse, Eric Berger, and Eric Liang. Autonomous inverted helicopter flight via reinforcement learning. In *Experimental Robotics IX*, pages 363–372. Springer, 2006. (Cited page 33.)

[107] Luke Ng. Reinforcement learning of dynamic collaborative driving. 2008. (Cited page 104.)

[108] T. Nothdurft, P. Hecker, S. Ohl, F. Saust, M. Maurer, A. Reschka, and J. R. Böhmer. Stadtpilot: first fully autonomous test drives in urban traffic. In *2011 14th International IEEE Conference on Intelligent Transportation Systems*, pages 919–924, Washington, USA, October 2011. (Cited page 18.)

[109] L. Nouveliere and S. Mammar. Experimental vehicle longitudinal control using second order sliding modes. In *Proceedings of the 2003 American Control Conference*, volume 6, pages 4705 – 4710, Denver, Colorado, june 2003. (Cited page 64.)

[110] Se-Young Oh, Jeong-Hoon Lee, and Doo-Hyun Choi. A new reinforcement learning vehicle control architecture for vision-based road following. *IEEE Transactions on Vehicular Technology*, 49(3):997–1005, 2000. (Cited page 104.)

[111] R. Okano, T. Ohtani, and A. Nagashima. Networked control systems by pid controllerimprovement of performance degradation caused by packet loss. *6th IEEE International Conference onIndustrial Informatics*, pages 1126–1132, 2008. (Cited page 82.)

[112] Sinan oncu, Nathan van de Wouw, WP Maurice H Heemels, and Henk Nijmeijer. String stability of interconnected vehicles under communication constraints. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 2459–2464. IEEE, 2012. (Cited page 83.)

[113] U. Ozguner, B. Baertlein, C. Cavello, D. Farkas, C. Hatipoglu, S. Lytle, J. Martin, F. Paynter, K. Redmill, S. Schneider, E. Walton, and J. Young. The osu demo '97 vehicle. In *1997 IEEE Conference on Intelligent Transportation System*, pages 502–507, Boston, MA, November 1997. (Cited page 18.)

[114] Jan Peters and Stefan Schaal. Reinforcement learning of motor skills with policy gradients. *Neural networks*, 21(4):682–697, 2008. (Cited page 102.)

[115] J Piao and M McDonald. Advanced driver assistance systems from autonomous to cooperative approach. *Transport Reviews*, 28(5):659–684, 2008. (Cited page 61.)

[116] Louis A Pipes. An operational analysis of traffic dynamics. *Journal of Applied Physics*, 24(3):274–281, 1953. (Cited page 24.)

[117] Jeroen Ploeg, Bart Scheepers, Ellen Van Nunen, Nathan Van de Wouw, and Henk Nijmeijer. Design and experimental evaluation of cooperative adap-

tive cruise control. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 260–265. IEEE, 2011. (Cited page 38.)

[118] Jeroen Ploeg, Steven Shladover, Henk Nijmeijer, and Nathan van de Wouw. Introduction to the special issue on the 2011 grand cooperative driving challenge. *Intelligent Transportation Systems, IEEE Transactions on*, 13(3):989–993, 2012. (Cited page 27.)

[119] Jeroen Ploeg, Nathan Van De Wouw, and Henk Nijmeijer. Lp string stability of cascaded systems: Application to vehicle platooning. *IEEE Transactions on Control Systems Technology*, 22(2):786–793, 2014. (Cited page 83.)

[120] Sharon L Poczter and Luka M Jankovic. The google car: Driving toward a better future? *Journal of Business Case Studies (Online)*, 10(1):7, 2014. (Cited page 38.)

[121] Dean A Pomerleau. Neural network vision for robot driving. In *The Handbook of Brain Theory and Neural Networks*. Citeseer, 1996. (Cited page 103.)

[122] BK Powell and JA Cook. Nonlinear low frequency phenomenological engine modeling and analysis. In *Proceeding of American Control Conference*, volume 1, pages 332–340, 1987. (Cited page 60.)

[123] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994. (Cited pages 44 and 48.)

[124] Junfei Qiao, Zhanjun Hou, and Xiaogang Ruan. Application of reinforcement learning based on neural network to dynamic obstacle avoidance. In *Information and Automation, 2008. ICIA 2008. International Conference on*, pages 784–788. IEEE, 2008. (Cited page 121.)

[125] R. Rajamani. *Vehicle dynamics and control*. Springer, New York, 2006. (Cited pages 40 and 60.)

[126] R Rajamani, SB Choi, JK Hedrick, and B Law. Design and experimental implementation of control for a platoon of automated vehicles. In *Proceedings of the ASME Dynamic Systems and Control Division (1998)*, 1998. (Cited page 25.)

[127] R. Rajamani, Han-Shue Tan, Boon Kait Law, and Wei-Bin Zhang. Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons. *IEEE Transactions on Control Systems Technology*, 8(4):695–708, July 2000. (Cited page 38.)

[128] Rajesh Rajamani. *Vehicle dynamics and control*. Springer Science & Business Media, 2011. (Cited pages 23 and 24.)

[129] Rajesh Rajamani and Chunyu Zhu. Semi-autonomous adaptive cruise control systems. *IEEE Transactions on Vehicular Technology*, 51(5):1186–1192, 2002. (Cited pages 40 and 61.)

[130] Kisiel Ralph. Electronics overload may limit option choices; some features may draw too much battery power. http://reviews.cnet.com/8301-13746_7-10123235-48.html, December 2008. Automotive News. (Cited page 19.)

[131] Nathan D Ratliff, David Silver, and J Andrew Bagnell. Learning to search: Functional gradient techniques for imitation learning. *Autonomous Robots*, 27(1):25–53, 2009. (Cited page 34.)

[132] A. Reschka, J.R. Bohmer, F. Saust, B. Lichte, and M. Maurer. Safe, dynamic and comfortable longitudinal control for an autonomous vehicle. In *2012 IEEE Intelligent Vehicles Symposium*, pages 346–351, Reschka, Andreas, June 2012. (Cited page 18.)

[133] J-P. Richard. Time-delay systems:an overview of some recent advances and open problems. *Automatica*, 39:1667–1694, 2003. (Cited page 82.)

[134] Martin Riedmiller. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *Machine Learning: ECML 2005*, pages 317–328. Springer, 2005. (Cited page 121.)

[135] Martin Riedmiller, Thomas Gabel, Roland Hafner, and Sascha Lange. Reinforcement learning for robot soccer. *Autonomous Robots*, 27(1):55–73, 2009. (Cited page 34.)

[136] Raúl Rojas. *Neural networks: a systematic introduction*. Springer, 1996. (Cited page 105.)

[137] Gavin A Rummery and Mahesan Niranjan. On-line q-learning using connectionist systems. 1994. (Cited page 57.)

[138] Steve Russell. DARPA grand challenge winner: Stanley the robot! *Popular Science*, 2006. (Cited page 20.)

[139] B. Sadjadi. Stability of networked control systems in the presence of packet losses. *42nd IEEE Conference on Decision and Control*, 1:676–681, 2003. (Cited page 82.)

[140] K. Santhanakrishnan and R. Rajamani. On spacing policies for highway vehicle automation. *IEEE Transactions on Intelligent Transportation Systems*, 4(4):198–204, December 2003. (Cited pages 40 and 63.)

[141] Stefan Schaal and Christopher G Atkeson. Learning control in robotics. *Robotics & Automation Magazine, IEEE*, 17(2):20–29, 2010. (Cited page 33.)

[142] Elham Semsar-Kazerooni and Jeroen Ploeg. Performance analysis of a cooperative adaptive cruise controller subject to dynamic time headway. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1190–1195. IEEE, 2013. (Cited page 61.)

[143] Shahab Sheikholeslam and Charles A Desoer. Control of interconnected nonlinear dynamical systems: The platoon problem. *IEEE Transactions on Automatic Control*, 37(6):806–810, 1992. (Cited page 38.)

[144] Shahab Sheikholeslam and Charles A Desoer. Longitudinal control of a platoon of vehicles with no communication of lead vehicle information: a system level study. *IEEE Transactions on Vehicular Technology*, 42(4):546–554, 1993. (Cited page 40.)

[145] Steven E Shladover. Review of the state of development of advanced vehicle control systems (avcs). *Vehicle System Dynamics*, 24(6-7):551–595, 1995. (Cited page 24.)

[146] Robert A Singer. Estimating optimal tracking filter performance for manned maneuvering targets. *IEEE Transactions on Aerospace and Electronic Systems*, (4):473–483, 1970. (Cited pages 87 and 88.)

[147] Satinder Singh, Tommi Jaakkola, Michael L Littman, and Csaba Szepesvári. Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine Learning*, 38(3):287–308, 2000. (Cited page 57.)

[148] Thomas Stanger and Luigi del Re. A model predictive cooperative adaptive cruise control approach. In *2013 American Control Conference*, pages 1374–1379. IEEE, 2013. (Cited page 61.)

[149] Srdjan S Stankovic, Milorad J Stanojevic, and Dragoslav D Siljak. Decentralized overlapping control of a platoon of vehicles. *IEEE Transactions on Control Systems Technology*, 8(5):816–832, 2000. (Cited page 40.)

[150] R. Sukthankar, J. Hancock, and C. Thorpe. Tactical-level simulation for intelligent transportation. *Mathematical and computer modelling*, 27(9-11):229–242, 1998. (Cited page 20.)

[151] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: An introduction*. MIT press, 1998. (Cited pages xi, 44, 45, 47, 49, 50, 51, 53, 55, and 56.)

[152] D. Swaroop. *String stability of interconnected systems: An application to platooning in automated highway systems*. PhD thesis, University of California at Berkeley, 1994. (Cited pages 38, 41, and 43.)

[153] D. Swaroop and JK Hedrick. String stability of interconnected systems. *IEEE Transactions on Automatic Control*, 41(3):349–357, 1996. (Cited pages 40, 41, 43, and 64.)

[154] D. Swaroop, JK Hedrick, CC Chien, and P. Ioannou. A Comparision of Spacing and Headway Control Laws for Automatically Controlled Vehicles 1. *Vehicle System Dynamics*, 23(1):597–625, 1994. (Cited pages 24, 40, and 63.)

[155] D. Swaroop and K. R. Rajagopal. Intelligent cruise control systems and traffic flow stability. *Transportation Research Part C: Emerging Technologies*, 7(6):329 – 352, 1999. (Cited pages 39 and 63.)

[156] HS Tan and M. Tomizuka. An adaptive sliding mode vehicle traction controller design. In *Proceedings of the American Control Conference*, volume 2, pages 1856–1861, San Diego, CA, 1990. (Cited page 60.)

[157] Brad Templeton. Cameras or lasers? `www.templetons.com/brad/robocars/cameras-lasers.html`, 2013. (Cited page 22.)

[158] Chuck Thorpe, Todd Jochem, and Dean Pomerleau. The 1997 automated highway free agent demonstration. In *Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on*, pages 496–501. IEEE, 1997. (Cited page 25.)

[159] Sebastian Thrun, Mike Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, et al. Stanley: The robot that won the darpa grand challenge. *Journal of Field Robotics*, 23(9):661–692, 2006. (Cited pages 20 and 33.)

[160] M. Tomizuka and JK Hedrick. Automated vehicle control for ivhs systems. In *IFAC Conference, Sydney, Australia*, pages 109–112, 1993. (Cited page 19.)

[161] Cem Unsal. *Intelligent navigation of autonomous vehicles in an automated highway system: Learning methods and interacting vehicles approach*. PhD thesis, Virginia Polytechnic Institute and State University, 1998. (Cited page 11.)

[162] Chris Urmson, Joshua Anhalt, Drew Bagnell, Christopher Baker, Robert Bittner, MN Clark, John Dolan, Dave Duggins, Tugrul Galatali, Chris Geyer, et al. Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics*, 25(8):425–466, 2008. (Cited pages 20 and 33.)

[163] Ardalan Vahidi and Azim Eskandarian. Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE Transactions on Intelligent Transportation Systems*, 4(3):143–153, 2003. (Cited page 22.)

[164] Bart Van Arem, Cornelie JG Van Driel, and Ruben Visser. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *Intelligent Transportation Systems, IEEE Transactions on*, 7(4):429–436, 2006. (Cited pages 25 and 26.)

[165] Ellen van Nunen, RJAE Kwakkernaat, Jeroen Ploeg, and Bart D Netten. Cooperative competition for future mobility. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1018–1025, 2012. (Cited page 27.)

[166] Ellen van Nunen, Jeroen Ploeg, Alejandro Morales Medina, and Henk Nijmeijer. Fault tolerancy in cooperative adaptive cruise control. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1184–1189. IEEE, 2013. (Cited page 61.)

[167] Joel Vander Werf, Steven Shladover, Mark Miller, and Natalia Kourjanskaia. Effects of adaptive cruise control systems on highway traffic flow capacity. *Transportation Research Record: Journal of the Transportation Research Board*, (1800):78–84, 2002. (Cited page 26.)

[168] P. Varaiya. Smart cars on smart roads: problems of control. *IEEE Transactions on Automatic Control*, 38(2):195–207, 1993. (Cited page 17.)

[169] J. Wang and R. Rajamani. The impact of adaptive cruise control systems on highway safety and traffic flow. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 218(2):111–130, 2004. (Cited pages 40 and 64.)

[170] Junmin Wang and Rajesh Rajamani. Should adaptive cruise-control systems be designed to maintain a constant time gap between vehicles? *IEEE Transactions on Vehicular Technology*, 53(5):1480–1490, 2004. (Cited page 24.)

[171] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992. (Cited page 57.)

[172] Christopher John Cornish Hellaby Watkins. *Learning from delayed rewards*. PhD thesis, University of Cambridge England, 1989. (Cited pages xi, 57, 112, and 114.)

[173] J. M. Wille, F. S., and M. Maurer. Stadtpilot: driving autonomously on braunschweig's inner ring road. In *2010 IEEE Intelligent Vehicles Symposium*, pages 506–511, San Diego, USA, June 2010. (Cited page 18.)

[174] M. Williams. Prometheus-the european research programme for optimising the road transport system in europe. In *IEEE Colloquium on Driver Information,*, pages 1/1–1/9, London, UK, December 1988. (Cited pages 18 and 24.)

[175] Chen Xia and Abdelkader El Kamel. An intelligent method of mobile robot learning in unknown environments. In *International Conference on Computer Science and Information Technology (ICCSIT)*, page C048, Barcelona, Spain, 2014. (Cited pages 34 and 103.)

[176] Chen Xia and Abdelkader El Kamel. Mobile robot navigation using neural network based q-learning. In *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, page paper 197, Bali, Indonesia, 2014. (Cited page 34.)

[177] Chen Xia and Abdelkader El Kamel. Mobile robot navigation using neural network based q-learning. In *3rd International Conference on Control, Robotics and Informatics (ICCRI)*, page M0014, Hong Kong, 2014. (Cited page 103.)

[178] Chen Xia and Abdelkader El Kamel. Online reinforcement learning from accumulated experience based on a nonlinear neural policy. *Expert Systems with Applications*, page submitted, 2015. (Cited page 34.)

[179] Chen Xia and Abdelkader El Kamel. A reinforcement learning method of obstacle avoidance for industrial mobile vehicles in unknown environments using neural network. In *2014 International Conference on Industrial Engineering*

and Engineering Management (IEEM), pages 671–675, 2015. (Cited pages 34, 103, and 121.)

[180] Chen Xia and Abdelkader El Kamel. Neural inverse reinforcement learning in autonomous navigation. *Robotics and Autonomous Systems*, 2016. (Cited page 34.)

[181] Lingyun Xiao and Feng Gao. A comprehensive review of the development of adaptive cruise control systems. *Vehicle System Dynamics*, 48(10):1167–1192, 2010. (Cited pages 23 and 24.)

[182] Jin Xu, Guang Chen, and Ming Xie. Vision-guided automatic parking for smart car. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 725–730, 2000. (Cited page 22.)

[183] Qing Xu and Raja Sengupta. Simulation, analysis, and comparison of acc and cacc in highway merging control. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 237–242. IEEE, 2003. (Cited page 23.)

[184] X. Yan, H. Zhang, and C. Wu. Research and development of intelligent transportation systems. In *2012 11th International Symposium on Distributed Computing and Applications to Business, Engineering Science*, pages 321–327, Wuhan, China, October 2012. (Cited page 18.)

[185] Gening Yu and Ishwar K Sethi. Road-following with continuous learning. In *Intelligent Vehicles' 95 Symposium., Proceedings of the*, pages 412–417. IEEE, 1995. (Cited page 103.)

[186] M. Yu, L. Wang, T. Chu, and G. Xie. Stabilization of networked control systems with data packetdropout and network delays via switching system pproach. *43rd IEEE Conference on Decision and Control*, 2004. (Cited page 82.)

[187] Yue Yu, Abdelkader El Kamel, and Guanghong Gong. Hla-based design for intelligent vehicles simulation system. In *CESA 2012*, pages 139–144, 2012. (Cited page 26.)

[188] Yue Yu, Abdelkader El Kamel, and Guanghong Gong. Modeling and simulation of overtaking behavior involving environment. *Advances in Engineering Software*, 67:10–21, 2014. (Cited page 26.)

[189] Yue Yu, Abdelkader El Kamel, Guanghong Gong, and Fengxia Li. Multi-agent based modeling and simulation of microscopic traffic in virtual reality system. *Simulation Modelling Practice and Theory*, 45:62–79, 2014. (Cited page 26.)

[190] J. Zhao. *Contribution to Intelligent Vehicle Platoon Control*. PhD thesis, Ecole Central de Lille, Lille France, 2010. (Cited page 83.)

[191] Jin Zhao and Abdelkader El Kamel. Multimodel fuzzy controller for lateral guidance of vehicles. In *CSCS'09*, 2009. (Cited page 26.)

[192] Jin Zhao, Gaston Lefranc, and A. El Kamel. Lateral control of autonomous vehicles using multi-model and fuzzy approaches. In *IFAC 12th LSS Symposium, Large Scale Systems: Theory and Applications*, Villeneuve D'Ascq, France, July 2010. (Cited page 26.)

[193] Jin Zhao, M. Oya, and A. El Kamel. A safety spacing policy and its impact on highway traffic flow. In *2009 IEEE Intelligent Vehicles Symposium*, pages 960–965, Xi'an, China, June 2009. (Cited page 61.)

[194] Tian Zheng, Abdelkader El Kamel, and Shaoping Wang. Control performance degradation in the sampling control system considering data delay and loss. In *CESA 2012*, pages 215–221, 2012. (Cited page 84.)

[195] Tian Zheng, Abdelkader El Kamel, and Shaoping Wang. Data loss and delay distribution of wireless sensor networks. In *ASCC 2013*, page xxx, 2013. (Cited page 84.)

[196] Jing Zhou and Huei Peng. Range policy of adaptive cruise control vehicles for improved flow stability and string stability. *IEEE Transactions on Intelligent Transportation Systems*, 6(2):229–237, 2005. (Cited page 24.)

[197] Chris Ziegler. Volvo will run a public test of self-driving cars with 100 real people in 2017. www.theverge.com, 2015. (Cited page 22.)

[198] P. Zwaneveld and B. van Arem. Traffic effects of automated vehicle guidance systems. In *Fifth World Congress on Intelligent Transportation Systems*, Seoul, Korea, October 1998. (Cited page 63.)

Analyse de Performance de Réegulateur de Vitesse Adaptatif Coopératif

Résumé: Cette thèse est consacrée à l'analyse de performance du Régulateur de Vitesse Adaptatif Coopératif (CACC) pour un train de véhicules intelligents pour les objectifs principaux de la réduction de congestion du trafic et l'amélioration de la sécurité routière. Ensuite, une approche de domaine fréquenciel de la stabilité de chaîne est présentée, qui est généralement définie comme la perturbation du premier véhicule n'amplifie pas pour les véhicules suivants.

Premièrement, la politique d'espacement , Intervalle Constante de Temps (CTH) pour un train de véhicule est introduite. Basé sur cette politique d'espacement, un nouveau système décentralisé de Deux-Véhicules-Devant CACC (TVACACC) est proposé, dans lequel l'accélération souhaitée de deux véhicules précédents est prise en compte. Ensuite, la stabilité de chaîne du système proposé est théoriquement analysé. Il est démontré que grâce à l'aide de la communication multiple sans fil parmi les véhicules, une meilleure stabilité la chaîne est obtenue par rapport au système conventionnel. Un train de véhicules dans Stop-and-Go scénario est simulé avec la communication normale et dégradée, y compris le délai de transmission élevé et la perte de données. Le système proposé donne un comportement stable de chaîne, correspondant Ã l'analyse théorique.

Deuxièmement, une technique de dégradation gracieuse pour CACC a été présenté, comme une stratégie alternative lorsque la communication sans fil est perdu ou mal dégradé. La stratégie proposée, qui est appelée DTVACACC, utilise le filtre de Kalman pour estimer l'accélération actuelle du véhicule précédent remplaçant l'accélération souhaitée. Il est démontré que la performance, en termes de stabilité de chaîne de DTVACACC, peut être maintenue à un niveau beaucoup plus élevé.

Enfin, une approche d'Apprentissage par Renforcement (RL) pour système CACC est proposé. L'algorithme politique-gradient est introduit pour réaliser le contrôle longitudinal. Ensuite, la simulation a montré que cette nouvelle approche de RL est efficace pour CACC.

Mots-clés: Systèmes de Transport Intelligents, Véhicules Autonomes, Régulateur de Vitesse Adaptatif Coopératif, Analyse de Performance, Contrôle Longitudinal, Degradation de Transmission, Apprentissage par Renforcement.

Cooperative Adaptive Cruise Control Performance Analysis

Abstract: This PhD thesis is dedicated to the performance analysis of Cooperative Adaptive Cruise Control (CACC) system for intelligent vehicle platoon with the main aims of alleviating traffic congestion and improving traffic safety. Then a frequency-domain approach of string stability is presented, which is generally defined as the disturbance of leading vehicle not amplifying through upstream of the platoon. At first, the Constant Time Headway (CTH) spacing policy for vehicle platoon is introduced. Based on this spacing policy, a novel decentralized Two-Vehicle-Ahead CACC (TVACACC) system is proposed, in which the desired acceleration of two front vehicles is taken into account. Then the string stability of the proposed system is theoretically analyzed. It is shown that by using the multiple wireless communication among vehicles, a better string stability is obtained compared to the conventional system. Vehicle platoon in Stop-and-Go scenario is simulated with both normal and degraded communication, including high transmission delay and data loss. The proposed system yields a string stable behavior, in accordance with the theoretical analysis. Secondly, a graceful degradation technique for CACC was presented, as an alternative fallback strategy when wireless communication is lost or badly degraded. The proposed strategy, which is referred to DTVACACC, uses Kalman filter to estimate the preceding vehicle's current acceleration as a replacement of the desired acceleration. It is shown that the performance, in terms of string stability of DTVACACC, can be maintained at a much higher level. Finally, a Reinforcement Learning (RL) approach of CACC system is proposed. The policy-gradient algorithm is introduced to achieve the longitudinal control. Then simulation has shown that this new RL approach results in efficient performance for CACC.

Keywords: Intelligent Transportation Systems, Autonomous Vehicles, Cooperative Adaptive Cruise Control, Performance Analysis, Longitudinal Control, Transmission Degradation, Reinforcement Learning.