



THE HONG KONG
POLYTECHNIC UNIVERSITY

香港理工大學

Pao Yue-kong Library
包玉剛圖書館

Copyright Undertaking

This thesis is protected by copyright, with all rights reserved.

By reading and using the thesis, the reader understands and agrees to the following terms:

1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact lbsys@polyu.edu.hk providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.

THE HONG KONG POLYTECHNIC UNIVERSITY

Department of Electrical Engineering

**STUDIES ON LATERAL CONTROL AND LANE
CHANGING ALGORITHMS FOR APPLICATION
IN AUTONOMOUS VEHICLES**

HO MAN LUNG

**A thesis submitted in partial fulfilment of the
requirements for the Degree of Doctor of Philosophy**

February 2007



Pao Yue-kong Library
PolyU • Hong Kong

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

Ho Man Lung _____ (Name of Student)

To my parents and my wife

Abstract

The studies reported in this thesis focus on specific problems in the control of autonomous vehicles in a mixed traffic scenario whereby the road is shared by autonomous driving machines as well as human driven vehicles. Among the major concerns are the lateral and the lane change problems in Automated Highway Systems (AHS) and mixed mode traffic settings.

The studies in this thesis also contribute to theoretical knowledge in the area of soft computing. Firstly, a novel fused Neural Network (NN) controller based on task decomposition is proposed. The proposed NN controller structure has been applied to a class of benchmark systems that require two input variables such as displacement and orientation in order to demonstrate its effectiveness. It has been tested for lateral control of autonomous vehicles under simulated and experimental environments.

Secondly, an innovative encoding scheme coined as Fire Rules Chromosome (FRC) encoding scheme is proposed which can improve the convergence speed of a fuzzy controller optimized by Genetic Algorithms. Although it is a general purpose controller and can be applied to a variety of systems, here it has been developed as an efficient controller for lateral control. The robustness of this controller is studied by Monte-Carlo simulations.

Lane keeping and lane changing are the two tasks involved in vehicle lateral control. The aim of the lane keeping is to maintain the vehicle at the center of the road and also follow the reference lane. The simplified neural network controller and fuzzy

controller optimized by the proposed FRC scheme are implemented on a scaled vehicle for vehicle lateral control.

For lane changing, the concept of virtual curvature is proposed to assist an automatic lane change maneuver. The virtual curvature algorithm incorporates virtual road curvature with bicycle model for vehicle lane change guidance. The virtual road curvature, does not physically exist, is a user assigned radius of a curved lane changing path which connects the current lane to the adjacent lane. The lane changing path guidance is achieved by assigning a virtual road curvature to the bicycle model to transform existing physical reference path curvature to the desired lane change path curvature. This transforming effect is accomplished by the inherent property of the bicycle model. The method is inspired by the observation that any change in the road curvature affects the vehicle lateral dynamics

The lane change maneuver offers flexibility in vehicle navigation, coordination and obstacle avoidance. However, during the lane changing, the merging vehicle should cross lanes which imply that the vehicle should consider obstacles or vehicles on adjacent lanes – a situation that could lead to accidents if not properly handled. The problem of lane change abortion is also studied through a computation of a maximum lateral displacement required for the lane change abortion. Collision free abortion point is defined in the study of lane change abortion. The maximum allowable lateral acceleration and the vehicle speed are the two prime factors governing the collision free abortion point. The effect of the two factors stated above on the abortion point is discussed in the thesis.

A scaled prototype semi-autonomous vehicle is constructed for experimental

testing of suggested algorithms in the thesis. The scaled vehicle is a modified Radio Controller (RC) car which is driving and steering by the front wheels. Infrared and ultrasonic sensors are installed to measure distance, and encoder sensor to measure vehicle speed. An industrial computer combined with A/D card is mounted on the vehicle as the main control system. The vehicle lateral model is obtained and it is verified that the scaled vehicle dynamic model exhibits the same properties with standard vehicle model.

Thesis supervisor: Prof. A.B. Rad

Department of Electrical Engineering

The Hong Kong Polytechnic University

Acknowledgement

Thank you Prof. A. B. Rad, my supervisor, for having enough patience with me and gave me freedom to test and implement ideas with guidance over the years. He provided me numbers of valuable comment and inspiration through numerous of stimulating discussion to solve problems.

I would also like to thank Dr. P. T. Chan for his encouragement and sharp advice. I have gained solid foundation on applied research from him.

The financial support given to this project by the Hong Kong Polytechnic University is also gratefully acknowledged. Last but not least, thank I would like to thank my parents, my wife for all their support and love throughout this process.

M.L. Ho

Table of Contents

Content	Pages
LIST OF TABLES:	V
LIST OF FIGURES:	VI
LIST OF ABBREVIATIONS	IX
LIST OF SYMBOLS	X
CHAPTER 1. INTRODUCTION	1-1
1.1. <i>MOTIVATION OF RESEARCH</i>	1-3
1.2. <i>RESEARCH OUTLINE</i>	1-5
1.3. <i>ORGANIZATION OF THE THESIS</i>	1-6
1.4. <i>STATEMENT OF ORIGINALITY</i>	1-8
1.5. <i>PUBLICATIONS</i>	1-9
CHAPTER 2. LITERATURE REVIEW	2-1
2.1. <i>INTRODUCTION</i>	2-1
2.2. <i>LATERAL REFERENCING SYSTEM</i>	2-5
2.3. <i>LATERAL CONTROL ALGORITHM</i>	2-9
2.3.1. <i>Lane changing / obstacle avoidance</i>	2-11
2.4. <i>LONGITUDINAL CONTROL</i>	2-13
2.5. <i>SOFT COMPUTING METHODOLOGIES</i>	2-15
2.5.1. <i>Fuzzy systems</i>	2-16
2.5.2. <i>Genetic algorithms</i>	2-18
2.5.3. <i>Hybrid Fuzzy system and Genetic algorithms</i>	2-20
2.5.4. <i>Neural Network</i>	2-25
2.5.5. <i>Hybrid Neural Network and Genetic algorithms</i>	2-27
2.6. <i>CONCLUSION</i>	2-28
CHAPTER 3. PROTOTYPE OF A SEMI-AUTONOMOUS VEHICLE	3-1
3.1. <i>INTRODUCTION</i>	3-1
3.2. <i>HARDWARE DESIGN</i>	3-3
3.2.1. <i>Vehicle Design</i>	3-4
3.2.2. <i>Sensors</i>	3-5
3.1.2. <i>Actuators</i>	3-12
3.1.3. <i>Industrial Computer</i>	3-13
3.1.4. <i>Digital I/O Cards</i>	3-13
3.2. <i>SOFTWARE</i>	3-17

3.3.	<i>VEHICLE MODEL</i>	3-19
3.4.	<i>CONCLUSION</i>	3-24
CHAPTER 4.	A NOVEL FUSED NEURAL NETWORK CONTROLLER	4-1
4.1.	<i>INTRODUCTION</i>	4-1
4.2.	<i>FUSED NEURAL NETWORK DESIGN BASED ON TASK DECOMPOSITION</i>	4-3
4.3.	<i>NEURAL NETWORK OPTIMIZATION VIA GENETIC ALGORITHMS</i>	4-6
4.2.1	<i>Coding of connection weights into chromosome</i>	4-6
4.2.2	<i>Fitness function</i>	4-7
4.2.3	<i>Computation procedure for the neural network optimization</i>	4-8
4.3	<i>APPLICATION TO BENCHMARK PROBLEMS</i>	4-9
4.3.1	<i>Inverted Pendulum:</i>	4-10
4.3.2	<i>Ball-and-beam system:</i>	4-12
4.3.3	<i>Vehicle lateral dynamic</i>	4-12
4.4	<i>SIMULATION RESULTS</i>	4-13
4.4.1	<i>Inverted Pendulum</i>	4-15
4.4.2	<i>Ball-and-Beam Balance System</i>	4-17
4.4.3	<i>Lateral control</i>	4-18
4.4.4	<i>Controller robustness analysis</i>	4-20
4.5	<i>DISCUSSION</i>	4-27
4.6	<i>CONCLUSIONS</i>	4-29
CHAPTER 5.	GA/FS CONVERGENCE ENHANCEMENT BY FIRED RULES CHROMOSOME	5-1
5.1.	<i>INTRODUCTION</i>	5-1
5.2.	<i>FIRED RULES CHROMOSOME FOR GENETIC ALGORITHMS</i>	5-3
5.2.1.	<i>Fired Rules Chromosome scheme</i>	5-4
5.2.2.	<i>Optimization procedure with FRC</i>	5-6
5.3.	<i>FRC DEMONSTRATION EXAMPLE</i>	5-8
5.3.1.	<i>Search space reduction:</i>	5-9
5.3.2.	<i>Controller performance and GA convergence.</i>	5-10
5.4.	<i>APPLICATION TO VEHICLE LATERAL CONTROL</i>	5-13
5.4.1.	<i>Simulation result on vehicle lateral control</i>	5-15
5.4.2.	<i>Robustness evaluation by Monte-Carlo analysis</i>	5-16
5.5.	<i>DISCUSSION</i>	5-19
5.6.	<i>CONCLUSIONS</i>	5-19
CHAPTER 6.	EXPERIMENTAL RESULTS ON VEHICLE LATERAL CONTROL	6-1
6.1.	<i>INTRODUCTION</i>	6-1

6.2.	<i>LATERAL CONTROL EXPERIMENT SETUP</i>	6-3
6.2.1.	<i>Experimental setup</i>	6-4
6.2.2.	<i>Real time implementation of the lateral controller</i>	6-5
6.3.	<i>FUSED NEURAL NETWORK CONTROLLER EXPERIMENTAL RESULT</i>	6-6
6.4.	<i>FUZZY LATERAL CONTROLLERS EXPERIMENTAL RESULTS</i>	6-10
6.4.1.	<i>Fuzzy (FRC) controller experimental results</i>	6-11
6.4.2.	<i>Fuzzy (TRD) Controller experimental results</i>	6-14
6.5.	<i>DISCUSSION</i>	6-17
6.6.	<i>CONCLUSIONS</i>	6-19
CHAPTER 7.	LANE CHANGE ALGORITHM WITH VIRTUAL CURVATURE AND LANE CHANGE ABORTION ANALYSIS	7-1
7.1.	<i>INTRODUCTION</i>	7-1
7.2.	<i>LANE CHANGE PROBLEM</i>	7-5
7.3.	<i>ROAD PROFILE AND BICYCLE MODEL</i>	7-6
7.3.1.	<i>Road profile</i>	7-7
7.3.2.	<i>Bicycle model</i>	7-8
7.3.3.	<i>Concept of virtual road curvature</i>	7-9
7.4.	<i>LANE CHANGE ALGORITHM</i>	7-10
7.4.1.	<i>Illustration of the lane change algorithm</i>	7-10
7.4.2.	<i>Virtual road trajectory and curvature determination</i>	7-11
7.4.3.	<i>Lane change scheme</i>	7-15
7.5.	<i>LANE CHANGE ABORTION - COLLISION FREE ABORTION POINT ANALYSIS</i>	7-18
7.5.1.	<i>Phase I – Vehicle Orientation adjustment</i>	7-20
7.5.2.	<i>Phase II – Abortion Trajectory</i>	7-23
7.6.	<i>ABORTION TRAJECTORY ANALYSIS</i>	7-24
7.6.1.	<i>Determination of collision free abortion point</i>	7-27
7.7.	<i>SIMULATION AND EXPERIMENTAL RESULTS ON LANE CHANGE</i>	7-30
7.7.1.	<i>Simulation Result</i>	7-30
7.7.2.	<i>Experiment setup</i>	7-33
7.7.3.	<i>Experimental results</i>	7-36
7.8.	<i>CONCLUSIONS</i>	7-39
CHAPTER 8.	EXPERIMENTAL VERIFICATION OF LATERAL CONTROL ALGORITHMS	8-1
8.1.	<i>INTRODUCTION</i>	8-1
8.2.	<i>FUNDAMENTAL EXPERIMENT ANALYSIS</i>	8-2
8.2.1.	<i>Speed control</i>	8-3
8.2.2.	<i>Lateral control</i>	8-4

8.2.3.	<i>Lane change abortion</i>	8-8
8.3.	<i>INTEGRATED EXPERIMENT</i>	8-11
8.3.1.	<i>Experimental Result</i>	8-15
8.4.	<i>DISCUSSION</i>	8-19
8.5.	<i>CONCLUSION</i>	8-20
CHAPTER 9.	CONCLUSIONS	9-1
9.1.	<i>MAJOR CONTRIBUTIONS</i>	9-1
9.2.	<i>SUGGESTION FOR FUTURE WORK</i>	9-4

List of Tables:

TABLE 2-1: LATERAL CONTROLLER IN LITERATURES WITH DIFFERENT REFERENCE SYSTEMS.	2-11
TABLE 3-1: POLYNOMIAL COEFFICIENTS OF INFRARED SENSORS.	3-12
TABLE 3-2: VEHICLE MODEL PARAMETERS.	3-23
TABLE 4-1: GENETIC ALGORITHM PARAMETERS.	4-13
TABLE 4-2: SYSTEMS PARAMETERS.	4-14
TABLE 4-3: OPTIMIZED CONNECTION WEIGHTS.	4-14
TABLE 4-4: LINEAR CONTROLLER FEEDBACK GAIN MATRIX.	4-15
TABLE 4-5: INVERTED PENDULUM PARAMETERS VARIATION.	4-20
TABLE 5-1: GA PARAMETERS.	5-11
TABLE 5-2: RESULT OF AVERAGE OF 50 TEST SETS OF FRC AND TRD ENCODING.	5-12
TABLE 5-3 : GA PARAMETER.	5-15
TABLE 5-4. PERFORMANCE INDEXES OF BOTH FUZZY CONTROLLERS.	5-16
TABLE 6-1: CONTROLLER PERFORMANCE INDEX.	6-17
TABLE 7-1: COLLISION FREE ABORTION POINT ANALYSIS.	7-28
TABLE 7-2: VEHICLE PARAMETERS.	7-31

List of Figures:

FIGURE 1-1: THESIS SCOPE AND FLOW.	1-6
FIGURE 2-1: THESIS SCOPE AND ITS RESEARCH AREAS.	2-4
FIGURE 2-2: TYPICAL FUZZY SYSTEM ARCHITECTURE.	2-17
FIGURE 2-3 : INTEGER CODING OF THE GENERAL CHROMOSOMES FOR FUZZY SYSTEMS.....	2-23
FIGURE 2-4: OPTIMISATION OF FUZZY LOGIC CONTROLLER BY GENETIC ALGORITHMS.	2-24
FIGURE 3-1: SCHEMATIC OF HARDWARE CONFIGURATION.	3-3
FIGURE 3-2: PROTOTYPE SEMI-AUTONOMOUS VEHICLE.....	3-4
FIGURE 3-3: INFRARED SENSORS ALLOCATION.	3-5
FIGURE 3-4: SHARP GP2D12 INFRARED SENSOR.	3-6
FIGURE 3-5: TYPICAL NONLINEAR OUTPUT OF GP2D12.	3-6
FIGURE 3-6: SHARP GP2D05 OBSTACLE DETECTOR.	3-7
FIGURE 3-7: POLAROID 6500 SERIES RANGING MODULE.....	3-8
FIGURE 3-8: MICRO-CONTROLLER PROGRAM FLOW CHART FOR ULTRASONIC SENSOR.	3-9
FIGURE 3-9: MICRO-CONTROLLER PROGRAM FLOW CHART FOR SPEED SENSOR.....	3-10
FIGURE 3-10: (A) FUTABA S3001 SERVO MOTOR. (B) APEX PLUS SPEED CONTROLLER.	3-13
FIGURE 3-11: 8254 WIRING.	3-15
FIGURE 3-12: GENERATION OF PWM SIGNAL.....	3-17
FIGURE 3-13: PROGRAM FLOW CHAT.	3-19
FIGURE 3-14: CLASSICAL BICYCLE MODEL.	3-20
FIGURE 3-15: MODIFIED BICYCLE MODEL.	3-22
FIGURE 4-1: NEURAL NETWORK CONTROLLER.	4-5
FIGURE 4-2: CODING OF NETWORK WEIGHT INTO CHROMOSOME.	4-7
FIGURE 4-3 : THE UNIFIED CONTROLLER.	4-10
FIGURE 4-4: INVERTED PENDULUM.....	4-11
FIGURE 4-5: INVERTED PENDULUM SIMULATION RESULT.....	4-16
FIGURE 4-6: BALL-AND-BEAM SYSTEM SIMULATION RESULT.....	4-18
FIGURE 4-7: LATERAL CONTROL SIMULATION RESULT.	4-19
FIGURE 4-8: ROBUSTNESS ANALYSIS ON NN CONTROLLER FOR INVERTED PENDULUM.....	4-21
FIGURE 4-9: ROBUSTNESS ANALYSIS ON LC CONTROLLER FOR INVERTED PENDULUM	4-23
FIGURE 4-10: ROBUSTNESS ANALYSIS ON NN CONTROLLER FOR BALL-AND-BEAM.....	4-24
FIGURE 4-11: ROBUSTNESS ANALYSIS ON LC CONTROLLER FOR BALL-AND-BEAM.....	4-24
FIGURE 4-12: MONTE-CARLO EVALUATIONS ON NN CONTROLLER FOR VEHICLE LATERAL CONTROL. .	4-26
FIGURE 4-13: MONTE-CARLO EVALUATIONS ON LC CONTROLLER FOR VEHICLE LATERAL CONTROL. .	4-26

FIGURE 5-1 : CHANGE OF CHROMOSOME LENGTH IN DIFFERENT GENERATIONS.....	5-5
FIGURE 5-2: GA OPTIMIZATION FLOW WITH FRC ENCODING SCHEME.	5-7
FIGURE 5-3 : CODING OF THE FIRED RULES CHROMOSOMES.....	5-8
FIGURE 5-4 : FIRED RULES IN THE 30 TH GENERATION.	5-9
FIGURE 5-5: THE CONVERGENCE OF FRC (SOLID LINE) AND THE TRD (DOTTED LINE).....	5-12
FIGURE 5-6: PERFORMANCE OF FRC (SOLID LINE) AND THE TRD ENCODING (DOTTED LINE) IN THE FIRST TESTING.	5-12
FIGURE 5-7. INPUT MEMBERSHIP FUNCTIONS OF THE FUZZY LATERAL CONTROLLER.	5-13
FIGURE 5-8: LATERAL CONTROL SIMULATION RESULT.	5-16
FIGURE 5-9 : FRC MONTE-CARLO ANALYSIS RESULT.	5-17
FIGURE 5-10 : TRD MONTE-CARLO ANALYSIS RESULT.....	5-18
FIGURE 6-1: LATERAL CONTROL EXPERIMENT CONTROL LOOP.....	6-5
FIGURE 6-2: FUSED NEURAL NETWORK CONTROLLER.	6-7
FIGURE 6-3: FUSED NN CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 42CMS^{-1}	6-8
FIGURE 6-4: FUSED NN CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 60CMS^{-1}	6-9
FIGURE 6-5: FUSED NN CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 78CMS^{-1}	6-9
FIGURE 6-6: FRC FUZZY CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 42CMS^{-1}	6-12
FIGURE 6-7: FRC FUZZY CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 60CMS^{-1}	6-13
FIGURE 6-8: FRC FUZZY CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 78CMS^{-1}	6-13
FIGURE 6-9: TRD FUZZY CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 42CMS^{-1}	6-14
FIGURE 6-10: TRD FUZZY CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 60CMS^{-1}	6-15
FIGURE 6-11: TRD FUZZY CONTROLLER EXPERIMENTAL RESULT WITH SPEED= 78CMS^{-1}	6-16
FIGURE 7-1 TYPICAL ROAD AND ROAD CURVATURE PROFILE	7-7
FIGURE 7-2: ILLUSTRATION OF LANE CHANGE ALGORITHM	7-10
FIGURE 7-3 : LANE KEEPING SCHEME AT STAGE I AND III.	7-16
FIGURE 7-4: LANE CHANGE SCHEME AT STAGE II.	7-16
FIGURE 7-5: LANE CHANGE AND LANE CHANGE ABORTION.	7-19
FIGURE 7-6: LANE CHANGE ABORTION PROCESS.....	7-20
FIGURE 7-7: ABORTION DISTANCE REQUIRED IN THE FIRST SECTION OF ABORTION.	7-21
FIGURE 7-8: SAFETY REGION IN LANE CHANGE ABORTION.....	7-25
FIGURE 7-9: COLLISION FREE ABORTION REGION.	7-29
FIGURE 7-10 : CURVATURE OF VIRTUAL TRANSITION LANE	7-32
FIGURE 7-11 : LANE CHANGE SIMULATION	7-33
FIGURE 7-12: LANE CHANGE EXPERIMENT SETUP WITH VIRTUAL TRANSITION LANE.	7-34
FIGURE 7-13: LANE CHANGE EXPERIMENT SETUP WITH PHYSICAL TRANSITION LANE.....	7-35
FIGURE 7-14: EXPERIMENT RESULT WITH VIRTUAL TRANSITION LANE (A) AND PHYSICAL TRANSITION LANE (B) ($D = 4M$).	7-37

FIGURE 7-15: EXPERIMENT RESULT WITH VIRTUAL TRANSITION LANE (A) AND PHYSICAL TRANSITION LANE (B) ($D = 5M$)	7-38
FIGURE 8-1: VEHICLE SPEED CONTROL EXPERIMENTAL RESULT.....	8-4
FIGURE 8-2: LANE KEEPING EXPERIMENT SETUP.....	8-6
FIGURE 8-3: LANE KEEPING PERFORMANCE WITH SPEED = $40CMS^{-1}$	8-7
FIGURE 8-4: LANE KEEPING PERFORMANCE WITH SPEED = $70CMS^{-1}$	8-7
FIGURE 8-5: LANE KEEPING PERFORMANCE WITH SPEED = $100CMS^{-1}$	8-7
FIGURE 8-6: LANE CHANGE ABORTION EXPERIMENT SETUP.	8-9
FIGURE 8-7: EXPERIMENTAL ABORTION TRAJECTORIES UNDER DIFFERENT SPEED.	8-11
FIGURE 8-8: INTEGRATED EXPERIMENT SETUP.	8-14
FIGURE 8-9: SENSORS MEASUREMENT IN CASE 1.	8-16
FIGURE 8-10: CASE 1 EXPERIMENTAL RESULT.....	8-17
FIGURE 8-11: CASE 2 EXPERIMENTAL RESULT.....	8-18

List of Abbreviations

AHS	Automated Highway Systems
CG	centre of gravity
FS	Fuzzy Systems
FS/GA	Integrated Fuzzy and Genetic Algorithms
FRC	Fired Rules Chromosome
GA	Genetic Algorithms
ISE	Integral Square of Error
ISR	Interrupt Service Routine
ITAE	Integral Time of Absolute Error
LC	Linear Controller
NN	Neural Network
PWM	Pulse Width Modulation

List of Symbols

R_{ref}	Radius of the circular arcs
ρ_{ref}	Road curvature
L_W	Lane width
y	Vehicle lateral position
x	Vehicle longitudinal position
d_f	Displacement sensors in the front
d_r	Displacement sensors in the behind
l_f	Distance between CG and the front axle
l_r	Distance between CG and the rear axle
l_v	Wheelbase length is equals to l_f+l_r .
γ	Yaw motion above CG
v	Vehicle velocity
β	Vehicle side slip angle: angle between the vehicle center line and v
y_f	Lateral vehicle displacement at front from the reference lane
y_r	Lateral vehicle displacement at tail from the reference lane
σ_f	Steering angle,
a_{ij}	Vehicle parameters
b_{ij}	Vehicle parameters
T_{taken}	Time duration of using bicycle model for lateral position estimation
(T_s)	Vehicle starts time of Lane Change
T_f	Vehicle finished time of Lane Change

V	Vehicle speed
d	Longitudinal lane Change distance
a_{max}	Allowable lateral acceleration
κ	Curvature of the polynomial
θ	Vehicle angle,
$\dot{\theta}$	Rate of vehicle angle,
e	Lateral position error
\dot{e}	Rate of lateral position error
W_1 to W_6	Network connection weights
m_r	Mass of the rod of the inverted pendulum
l	Length of the rod of inverted pendulum
m_c	Mass of the cart of the inverted pendulum
θ_I	Rod angle of the inverted pendulum
x_I	Cart position of the inverted pendulum
θ_B	Beam angle of the Ball-and-Beam system
r_B	Ball position of the Ball-and-Beam system
V_w	Vehicle width
g	Gravitational acceleration
α	Ball-and-Beam system parameter
K_I	Feedback gain vector for inverted pendulum
K_b	Feedback gain vector for Ball-and-Beam system
K_v	Feedback gain vector for vehicle lateral control
S_r	Safety abortion region
P_{cf}	Collision free abortion point
d_a	longitudinal abortion distance
L_{zo}	Lateral position with zero vehicle orientation during abortion

L_{max}	Maximum lateral position during abortion
L_{ab}	Lateral position of the abortion point
L_{adj}	Lateral distance required to adjust vehicle orientation to zero

Chapter 1. Introduction

It can be well argued that the vision of Leonardo da Vinci and the genius of Issac Newton laid the foundation of one of the most intriguing machines that man has ever devised. However, the evolution of the *automobile* as we know it goes back to 1769 when a French engineer by the name of Nicolas Joseph Cugnot (1725-1804) tested the very first self-propelled vehicle powered by a steam engine. It took almost two and half centuries, more than 100,000 patents, contributions from thousands of engineers, and an insatiable passion to create the modern car. In the last 240 years, there have been significant improvements in the design, safety, and passenger comfort. In this process, driving a sleek and fast car has also become an obsession of the modern man!

The irony is that the term *automobile* is a misnomer. The car that we drive is very much a *manual-mobile*. Nowadays, some semi-autonomous versions have been out of the drawing boards into the assembly lines and on to roads. It is likely that in a decade or so, humans will experience the joy of a real *Automobile*. This thesis is a modest attempt to pave the way for the transition from a semi-autonomous (driver-assist) to completely autonomous (driverless) vehicle.

Like other significant technological developments, there have been some serious byproducts among which are environmental issues, traffic congestion, congested roads, and most importantly the loss of human life. Traffic congestion and accidents have impacts on human society in terms of travel delays, increases pollution, productivity and loss of life. The deployment on the Intelligent Transportation Systems (ITS) or Automated Highway Systems (AHS) is a solution towards

enhancing the traffic efficiency and safety by employing diverse advanced technologies of computing, communication, sensors, and control. The ultimate goal of the AHS is enabling fully automated highways such that driverless vehicles could be realized. The process towards full implementation of AHS requires sophisticated traffic management, intelligent and autonomous vehicles, state of art in sensor technology, control systems, and infrastructure enhancement. The whole process involves close collaboration of highway authorities with vehicle manufacturers.

The development of AHS is not a fantasy any more and there have been several test systems already in operation around the world. There are some positive signs that Hong Kong is preparing for AHS. The signposts with electronic displays located on the highways which provide traffic information ahead, GPS navigation systems, and driver warning systems to avoid congested traffic are among the steps that have been taken by the highway department. These are only starting steps but nevertheless timely and attribute to Advanced Traffic Management systems (ATMS) of ITS. These studies are outside the scope of this research.

Car manufacturers have heavily invested in safety and passenger comfort in the last three decades. *Lexus*, one of the largest Japan car manufacturers offers “Dynamic Radar Cruise Control” module which adjusts vehicle speed automatically to maintain a pre-set distance. The distance measurement is achieved by millimeter-wave radar. *Mercedes-Benz*, the car manufacturer from Germany, has launched the world first vehicle equipped with intelligent light system on E-Class saloons. The dipped beam headlamps with active curve illumination automatically adapts to the relevant traffic situation in order to enhance the driver visible range at night. *BMW*, also a car manufacturer from Germany, has introduced the “Dynamic Stability Control” (DSC)

system which stabilizes the vehicle by coordinating the engine driving torque and braking on individual wheels. To detect instability, DSC requires information of steering wheel movement, vehicle speed, transverse acceleration and yaw. The DSC system is already available on *BMW* full series sedan.

The aforementioned longitudinal and traction control systems on the market are all driver assistance systems and provide building blocks towards autonomous vehicle. To achieve the goal of AHS, the key is on the development of fully autonomous vehicle. In the thesis, the author primarily focuses on the problems associated with the lateral control of autonomous vehicles. This research investigates the integration of the hardware, the software, and the control of an in-house designed and built prototype vehicle.

In particular, the thesis addresses the following issues:

- ❖ To design a simplified neural network controller structure for vehicle lateral control and a class of systems with similar nature.
- ❖ To establish a convergence rate enhancement scheme for the integration of fuzzy logic and genetic algorithms.
- ❖ To design a lane change algorithm for automatic vehicle lane change maneuver and analyze the collision free abortion of the lane change process.
- ❖ To implement, test, and analyze different algorithm on a scaled prototype vehicle.

1.1. Motivation of research

Hong Kong's roads are known to have the highest vehicle densities in the world. As of March 2005, there were more than 500 000 licensed vehicles and less than 2500 kilometers of roads. According to the Annual Transport Digest, in 2005, there were

15062 traffic accidents across the territory. The report ranks the most contributing factors for traffic accidents to be too close spacing, careless lane changing, and turning inattentively. All these attribute to human error or negligence. In addition, the economic growth and the ever-increasing transportation needs in Hong Kong have led to the increased traffic congestion and traffic accidents. Coupled with a mixture of constraints such as complex terrain, small land area, land-use restriction and dense population, it is expected that traffic congestion to be a major obstacle for further economic growth in years to come.

It has been suggested that the implementation of autonomous vehicles should reduce the number of accidents by automating the decision process to maintain an appropriate spacing and environmental assessment for turning and lane changing. In addition to the automation of the decision making process, the advantage of machine over human driver is the reaction time. Machines react to relevant traffic situations in a few milliseconds but human takes hundreds of milliseconds even in the absence of fatigue. The fast reaction time of a self-driving machine implies the feasibility of increasing the highway capacity by reducing inter-vehicle spacing. Therefore the automation on vehicle control not only enhances the traffic safety but also benefits to traffic efficiency.

In this thesis, the author addresses selected problems in the interesting way. The focus is on lateral control and its associated problems. The control algorithms are implemented and tested on an in-house designed. A prototype vehicle was built with the aim of filling the gap between design and practical implementation. Computational methodologies –in particular, soft computing techniques are employed in the design of the vehicle control systems.

1.2. Research Outline

The scope and the flow of this thesis can be classified into 3 phases as shown in Figure 1-1. In Phase 1, the elementary components of a semi-autonomous vehicle in both hardware and software are attended. Soft computing techniques are used to design vehicle controllers and controllers for a class of benchmark systems. A fused neural network controller is designed for vehicle lateral control and pendulum balancing. The so-called Fired Rules Chromosome (FRC) encoding scheme for integration of fuzzy systems and genetic algorithm (FS/GA) is proposed to enhance the convergence rate and to retain the controller robustness.

In Phase 2, the problem of vehicle lane change maneuver is considered. The lane change problem is challenging due to the limitation on the data collection during the transition period. Lane change with virtual road curvature is suggested to tackle the problem of lane change. In addition to normal lane change process, the abortion process is also considered in this phase. The abortion point on the lane change process to ensure a collision free navigation is also proposed and analyzed. In Phase 3, the prototype vehicle is served as a platform for controllers testing experimentally on a test track. The performances of the controllers are verified under different test settings.

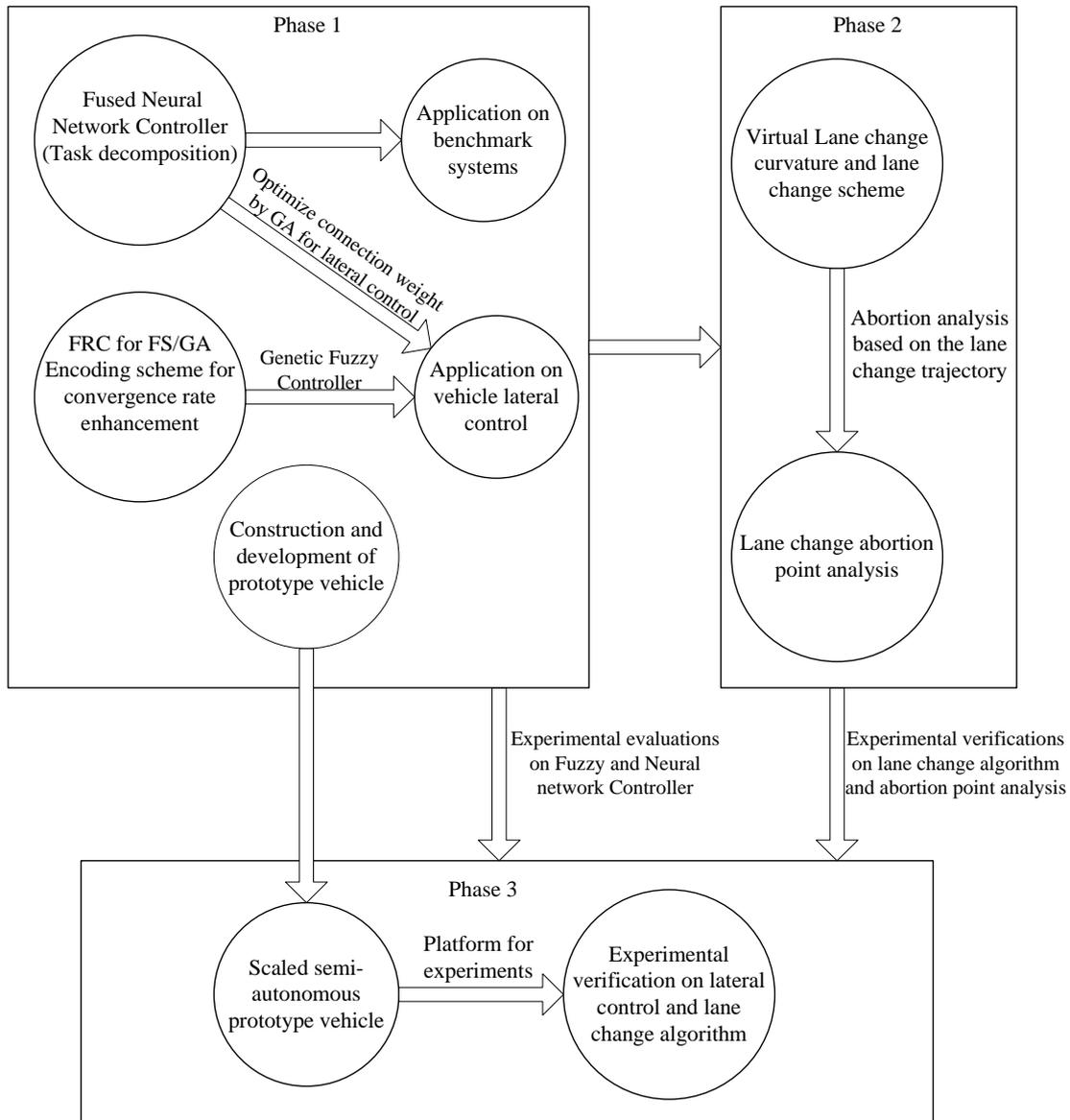


Figure 1-1: Thesis scope and flow.

1.3. Organization of the thesis

In Chapter 2, a brief history on the development of automated highway system, detailed literature on autonomous vehicle control, and various soft computing method are presented. Control algorithms proposed by researchers on vehicle lateral control are classified according to the type of the reference system employed in the design. This chapter lays the background for the rest of the thesis.

In Chapter 3, the design and construction of a scaled semi-autonomous prototype vehicle for experimental studies is presented. The scaled vehicle hardware construction such as sensors, computation unit, input-output interface are fully described. In addition, the theoretical lateral model of the scaled prototype vehicle is discussed in this chapter.

In Chapter 4, a fused neural network controller based on the task decomposition is presented. The controller can be applied to a class of systems which can be decomposed into two variables such as an angle and a displacement. The controller is optimized with genetic algorithm for vehicle lateral control and a class of bench mark systems including inverted pendulum and ball-and-beam system. Simulation results are presented in this chapter and compared with full state feedback linear controller.

In Chapter 5, a novel encoding scheme named as Fired Rules Chromosome (FRC) encoding scheme is proposed to enhance the rate of convergence on the optimization of fuzzy rule table with genetic algorithm. A fuzzy controller for vehicle lateral control is also presented. The convergence rate, performance index, and controller robustness of fuzzy controller optimized by GA with FRC encoding scheme are compared with fuzzy controller optimized by GA against a traditional method.

In Chapter 6, the vehicle lateral control experiments with controllers based on neural network controller (Chapter 4), and fuzzy controllers optimized by GA with FRC and traditional method (Chapter 5) are conducted. These controllers are implemented on the prototype vehicle (Chapter 3) under different speed settings. The performances of the three controllers are compared in terms of performance indices.

In Chapter 7, a thorough description on lane change algorithm with virtual curvature and lane change abortion analysis are presented. The proposed concept of virtual road curvature and lane change scheme guides vehicle from original lane to adjacent lane in the absent of infrastructure support. The virtual lane change algorithm is verified experimentally. The lane change abortion analysis studies the point of abortion which guarantee no collision occur. The collision free abortion point is evaluated under different speed and lateral acceleration limit.

In Chapter 8, experimental verifications on vehicle control algorithms are presented. The objective of this chapter is to assess the performance of the controllers and the analysis of results by various practical experiments. The lateral controller and lane change algorithm are integrated and tested under variable speed. The lane change abortion point stated in Chapter 7 is also verified experimentally in this chapter.

Chapter 9 outlines the achievements of the thesis and draws tentative conclusions and points out the future direction of this research work.

1.4. Statement of originality

The main contributions made by the author in this thesis are given in the following statements:

- Design of a fused neural network controller by task decomposition for the control a class of systems including two variables for manipulation such as vehicle lateral control (Chapter 4).
- Design of a Fired Rules Chromosome (FRC) encoding scheme for convergence rate enhancement in the application of fuzzy rule table optimization by genetic

algorithms (Chapter 5).

- Design of a lane change scheme with virtual road curvature to tackle the problem of automatic lane changing (Chapter 7).
- Analyze the collision free abortion point during lane change (Chapter 7).
- Implementation and verification of the proposed controllers and schemes in experiments on a scaled prototype vehicle (Chapter 8).

1.5. Publications

At the time of writing this thesis, one journal paper and four conference papers have been published. Also, there are three papers that have been submitted to international journals. The list of publications is as follows:

1. M.L. Ho, A. B. Rad and P.T. Chan, "Project based learning - design of a prototype semiautonomous vehicle", IEEE Control System Magazine, Vol. 24 No. 5, Oct. 2004. page 88-91.
2. M.L. Ho, P.T. Chan and A.B. Rad, "Lane Change Algorithm for Autonomous Vehicles via Virtual Curvature Method" Journal of Advanced Transportation (second review)
3. M.L. Ho, P.T. Chan and A.B. Rad, "A novel fused neural network controller for lateral control of autonomous vehicles", Mechatronics (submitted)
4. P.T. Chan, A.B. Rad, and M.L. Ho, "On a Fired Rules Chromosome Encoding Scheme", Fuzzy Sets and Systems (submitted)
5. Y.K. Lo, A.B. Rad, C.W. Wong and M.L. Ho, "Automatic Parallel Parking" IEEE Proceedings on Intelligent Transportation Systems, China, Vol. 2 , 12-15 Oct. 2003, page 1190-1193.
6. M.L. Ho, A.B. Rad, and P.T. Chan, "Evaluation of Automatic Cruise Control in Prototype Autonomous Vehicle" Regional Inter-University Postgraduate Electrical and Electronic Engineering conference (RIUPEEEEC) 2003, The Hong Kong

Polytechnic University, Hong Kong, 29-30 August, 2003, page 99-100.

7. M.L. Ho, P.T. Chan, A.B. Rad and C.H. Mak, "Truck Backing up neural network controller optimized by Genetic Algorithms", IEEE Congress on Evolutionary Computation 2003, Canberra, Australia, 8-12 December, 2003, page 944-951.

8. M.L. Ho, P.T. Chan, and A.B. Rad, "A Novel Lane Change Algorithm", Regional Inter-University Postgraduate Electrical and Electronic Engineering conference (RIUPEEEEC) 2005, City University of Hong Kong, Hong Kong, 14-15 July, 2005, page 689-692.

Chapter 2. Literature Review

2.1. Introduction

In order to set the scene for the rest of the thesis and to give background and insight to the core of the studies undertaken in this research, this chapter is aimed at giving a thorough yet selective account of the relevant work carried out by other researchers.

The earliest research and development on the safety and highway efficiency enhancements were initiated in 1960s by the Bureau of Public Roads of the Department of Commerce in the United State (Saxton 1993). In the past decades, there have been numerous research activities conducted by the Intelligent Transportation Systems (ITS) programs around the world especially in the United State, Europe and Japan. Fully automated vehicles for specialized applications were successfully deployed in the 1990s. The fundamental capability for passenger car automation was proven in Europe, Japan, and the United States. (Ioannou 1997).

The main research areas of ITS includes Advanced Traffic Management Systems (ATMS), Advanced Traveler Information Systems (ATIS), Commercial Vehicle Operation (CVO), Advanced Public Transportation Systems (APTS), and Advanced Vehicle Control Systems (AVCS). The ITS research areas focus on the use of information technologies and emerging electronic techniques to enhance road transportation efficiency and safety. The collaboration of the above mentioned systems collectively are referred to as Automated Highway Systems (AHS).

The AHS is an area under ITS which aims to improve the efficacy of highways and roads by adopting advanced technologies on infrastructure as well as vehicles. In the United State, a California statewide transportation research program known as Partners for Advanced Transit and Highways (PATH) was established in 1986. The PATH program research directions are towards AHS and it has been regarded as the pioneering program in the state of art in the field of AHS.

In Europe, a program initiated by vehicle industry called the PROMETHEUS (Program for European Traffic and Highest Efficiency and Unprecedented Safety) was formed in 1985. One year later, a research projects named DRIVE (Dedicated Road Infrastructure for Vehicle Safety in Europe) was established in parallel with the PROMETHEUS program but the research focus was on the highway authority problem rather than vehicle manufacturer (French *et al*, 1993). Recently the research projects in the Europe are fast growing. The Cybercars project developed in Europe shows successful realization of an autonomous vehicle in the urban area with dedicated lanes and it is extended to Cybercar-2 which aims to address high demand traffic by the enhancement on the communication technologies (Cybercar official website). Bouraoui *et al*. (2006) show the recent development of Cybercar on safety of intersections. PReVENT is another project in Europe which conducting research on active safety systems. There is wide range of projects under the PReVENT (PReVENT official website). Moehler (2006) presents lane keeping support system which is developed under the Safelane project of PReVENT.

Figure 2-1 shows the main research areas in ITS and the associated methodologies towards achieving more road safety for passengers and highway

capacity enhancement. The figure also indicates the research topics covered in this thesis.

There are two main concepts of research in AHS: dedicated lanes and mixed traffic. The dedicated lane concept is devoted to the use of platoons for vehicle longitudinal control. Horowitz and Varaiya (2000) presented a five layers AHS structure namely *network*, *link*, *coordination*, *regulation* and *physical* layers for dedicated lane approach. The control commands are sent from higher layers and the vehicles do not need to make intelligent decisions.

The mixed traffic approach treats each vehicle as an autonomous agent to make decision based on local sensors readings. Therefore the development in this paradigm aims at autonomous vehicle control for handling the challenging task in mixed traffic condition (i.e. designing an intelligent agent to be able to function safely and efficiently in a mixed traffic road). This thesis addresses the problem on the autonomous vehicle control.

The Advanced Vehicle Control Systems (AVCS) is the key element for both dedicated lane and mixed traffic concepts of AHS since both approaches require fully automated/autonomous vehicles. Nowadays, some AVCS systems are already implemented on commercial vehicles such as cruise control system and anti-lock braking system which are intended as driver assistance systems.

AVCS research enjoys the rich pool of classical and modern control systems approaches and computational intelligence methodologies at its disposal. In addition, communication and perception systems also play an important role in data collection, system monitoring and diagnosis.

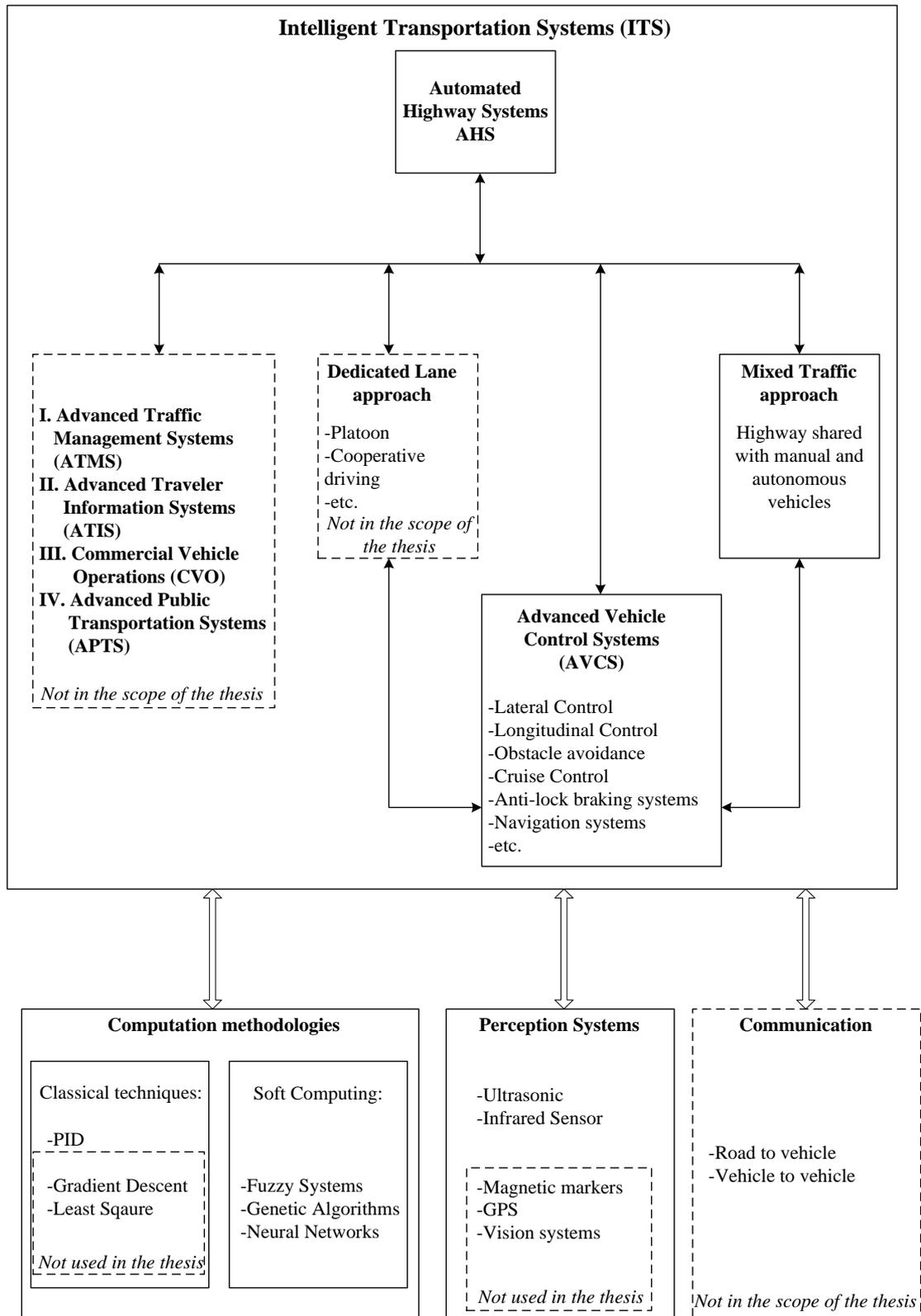


Figure 2-1: Thesis scope and ITS research areas.

The rest of this chapter is organized as follows: The literature on sensor systems for lateral measurements in the infrastructure supported highways are reviewed in Section 2.2. Sections 2-3 and 2-4 discuss the lateral and longitudinal control respectively. Section 2.5 introduces the background on soft computing methodologies including Fuzzy logic, Genetic Algorithms, and Neural Networks. Finally Section concludes this Chapter.

2.2. Lateral referencing system

The earliest lateral reference system was a single wire, laid along the center of the lane and excited with alternating current. This was used by the General Motor Corporation/Radio Corporation of America in 1960's. The lateral information was measured by amplitude-sensing techniques in the early stage and changed to phase-sensing approach which offered higher robustness to ferrous material on the highway (Fenton and Mayhan, 1991).

Nowadays, the reference system can be divided into *look-ahead* and *look-down* systems. The look-ahead systems replicate human driving behavior by measuring lateral distance ahead the vehicle. In contrast to look-ahead system, look-down system measures vehicle lateral position at a location within the vehicle boundaries. The automatic control approach can also be grouped into look-ahead and look-down systems. The classification is made according to the point of measurement of vehicle lateral position with respect to the reference path.

The following discusses the three most typical reference systems worldwide: Magnetic markers which are extensively used look-down reference systems in the PATH program, Frequency Selective Surface which is a look-ahead system developed

by the Ohio State University, and Machine vision approaches which extract lateral information from camera by image processing techniques.

Magnetic markers

Zhang and Parson (1998) suggested the use of discrete permanent magnets as reference markers for lateral control. The use of magnetic marker for lateral reference system has been investigated in depth by many researcher groups. Guldner *et al.* (1997) presented the use of magnetic marker for coding road information and outlined its implementation details. Different magnetic materials ceramic and rare earth magnet were installed on normal pavement and bridge respectively. Rare earth magnets are more expensive, but much more stronger than ceramic magnets, which allows to use 2.5cm long magnets as compared to 10cm long ceramic magnets. On bridge structures, drilling 10cm deep magnet holes at locations specified with high accuracy proved hazardous with respect to the structural integrity of the bridge and shorter rare earth magnets have to be installed.

The investigation of Guldner *et al.* (1997) also suggest the use of magnet binary coding to code permanent information such as road geometry, magnet type, merge/diverge, lane change permit, highway ID and Kilometer-post. The availability of roadway information to the vehicle provides a platform for the development of autonomous vehicle system.

Frequency Selective Surface

Farkas *et al.* (1997) developed a forward-looking chirp monopulse radar for vehicle lateral guidance. The radar chirp works with frequency selective surface (FSS) strip, which installed in the middle of the highway lane, to detect vehicle lateral

position. The detection of steering information is based on the sense of backscattered radar energy by the highway strip at a specific frequency. The radar operates from 10 to 11 GHz at *5ms* intervals. The configuration of the radar system enables the detection of road curvature which provides additional information for vehicle guidance. This look-ahead radar guidance system is part of the Ohio State University (OSU) Automated Highway System convoy. In addition to provide steering information, the radar also provides range information to the nearest front vehicle for automatic cruise control. The range information to the front vehicle is achieved by detecting the echo of the radar pulse.

Machine vision

The use of computer vision to detect and extract white painted lines along the road for vehicle guidance has been proposed by many researchers (Manigel and Leonhard (1992), Tsugawa (1994), Miura *et al.* (2002)). The visual guidance technique offers the advantage of minimal infrastructure modification that enhances the flexibility in vehicle guidance. In addition to vehicle guidance in highway, Chen *et al.* (2004) developed an autonomous vehicle named TerraMax with cameras, radar and sonars for off-road ground vehicles guidance. The TerraMax vehicle participated in the 2004 Darpa Grand Challenge which traveled *142* miles across the Mojave Desert. In addition to reference lane detection, vision based technology can be extended to the level of situation assessment. Eidehall *et al.* (2007) developed an active safety system named Emergency Lane Assist (ELA) which is using vision system for vehicle tracking and adjacent lanes monitoring. When dangerous situation is observed, the ELA applies torque to the steering wheel to prevent lane departure.

Vehicle guidance with vision sensor would overcome some practical limits of the

reference methods mentioned above. The obvious example is the guidance during lane changing. In order to provide continuous reference signal during lane change, the above methods (magnetic marker and FSS) require installation of extra markers or strip between lanes for lane change guidance. If extra markers or strip are not installed, some ad hoc techniques or position estimation should be conducted. For the machine vision approach, reference signal is extracted from image thus feedback signal is always available during lane change.

However, the limitation of computer vision is the dependency on weather condition and light. Researches conducted on this area are mainly on the image processing techniques to enhance the robustness of the visual sensors.

In this thesis, the lateral controller research is based on look-down reference system without preview information of road curvature. For practical experiment consideration, a scaled semi-autonomous vehicle was built and the vehicle details will be described in Chapter 3.

The establishment of look-down reference system in the scaled prototype system is achieved by installing infrared sensors on the vehicle and setting the lateral reference lane standing on the ground. The standing up reference lane served as a reflecting object for infrared sensor measurement. Since the infrared sensor measures lateral distance in the close vicinity of the vehicle, it can be classified as look-down reference system.

2.3. Lateral control algorithm

The objective of automatic vehicle lateral control is to steer a vehicle along a planned path. Much research has been reported in the past decades which are conducted by academic institutions, government funded programs, and industrial sector.

The research on the vehicle guidance and control starts at early 1960's (Fenton and Mayhan, 1991) and continuously has grown till now. In the past decades, many linear and nonlinear control algorithms such as linear quadratic, sliding mode, adaptive control, have been developed to design the control system for autonomous vehicles.

Fenton and Selim (1988) used optimal control technique with observer to design full state feedback from a measurement of lateral offset at the front of the vehicle. Hessburg *et al.* (1991) studied the feasibility with experimental vehicle of lateral control by using classical PID controller with magnetic sensor as reference system.

Guldern (1999) designed a robust lateral controller under vehicle parameter variations. Kato *et al* (2002) presented vehicle control algorithm for cooperative driving with communication. The lateral control algorithm presented by Kato *et al.* is based on dead reckoning function with differential global positioning system (DGPS). The vehicle is guided by a precise map that contains a series of points representing the path of the vehicle. The system is experimentally tested in Demo 2000. Hernandez and Kuo (2003) modified a look-down reference controller by using GPS to extract geometric information from digital map for tracking enhancement.

With the fast advancement of computing technology, soft computing techniques were employed in the control of autonomous vehicle. Fuzzy logic controller provides a platform for incorporating human experience and engineering judgment for control of vehicle steering.

Hessburg and Tomizuka (1991, 1993 and 1994) studied the use of fuzzy logic controller for vehicle lateral control. A manually tuned fuzzy controller was implemented on a full scale vehicle which consisted of three modules namely: feedback, preview, and gain scheduling. Kodagoda *et al.* (2002) designed a fuzzy controller from the perspective of variable structure systems theory. Table 2-1 shows lateral controllers in the literature with different techniques for different lateral reference system.

In addition to lateral controller design under normal road conditions, vehicle control has also been considered in the research of autonomous vehicle control under faulty scenarios. Lu (2005) proposed a laser scanning vehicle following scheme to handle the vehicle lateral control problem when there is a fault on sensors. Huang and Tomizuka (2005) also proposed a controller with linear time varying (LTV) design under the fault of rear sensor.

Controller type	Reference system	Reference
Full state feedback controller	Look ahead (FSS)	Fenton and Selim (1988)
PID	Lookdown (Magnetic Markers)	Hessberg <i>et al.</i> (1991)
Fuzzy controller	Lookdown + feedforward (Digital Map)	Hessburg and Tomizuka (1991, 1993 and 1994)
Robust nonlinear controller	Lookdown (Magnetic Markers)	Guldern (1999)
Linear controller	GPS	Kato <i>et al.</i> (2002)
Fuzzy controller		Kodagoda <i>et al.</i> (2002)
Nonlinear controller	Lookdown + GPS for feedforward	Hemandez and Kuo (2003)

Table 2-1: Lateral controller in literatures with different reference systems.

2.3.1. Lane changing / obstacle avoidance

Vehicles in the AHS capable of automatic lane changing enhance the flexibility in vehicle navigation, traffic coordination and obstacle avoidance. Automatic lane change can be performed either with or without infrastructure guidance (Tan *et al.*, 1998). For the infrastructure guided lane change, additional lateral reference markers are installed to connect two adjacent lanes.

Lane change maneuver without infrastructure support requires the vehicle to leave the reference markers, to cross over into the target lane using dead-reckoning, and resume tracking upon reaching the reference line in the target lane. The crossover trajectory is usually defined as an S-curve with smooth curvature transitions to avoid lateral jerks.

The requirement of additional markers installation in infrastructure guided lane change restricts the lane change locations which degrade the flexibility in vehicle coordination. Therefore lane change without infrastructure guidance is preferred. The problem of lane change without infrastructure guidance includes the determination of safety gap to initiate lane change, reference signal establishment, and controller design during the dead-reckoning period.

Nelson (1989) shows the use of 5th order polynomial as the continuous steering function for lane changing. The polynomial trajectory is computational simple, closed form expression that provides continuous curvature to connect the transition between two adjacent lanes. Hatipoglu *et al.* (2003) uses virtual yaw rate reference and switching controller to generate steering command so that accomplish a smooth lane change for the use in AHS. Shamir (2004) design the optimal trajectory for overtaking slower moving vehicle.

Julaet *et al.* (2000) present the determination of minimum longitudinal spacing that vehicle should initiate lane change to avoid any types of collision in lane change maneuver. Fu *et al.* (2004) present a GA tuned Fuzzy controller for tracking virtual path during lane change. Godbole *et al.* (1998) design a distributed control with communication to coordinate the motion of vehicles to ensure safety and efficiency during lane change.

In Chapter 7, a novel lane change algorithm with virtual curvature is proposed. The suggested algorithm use the inherit property of vehicle lateral model (the bicycle model) to assist lane change. The lane change scheme uses unified lateral controller approach which a single lateral controller is used for both lane keeping and lane

changing. Experimental study on scale autonomous vehicle is conducted to verify the performance of the lane change scheme. In addition to the study on lane changing, lane change abortion is also considered. The analysis on lane change abortion states the region of abortion point that ensures no collision occurs.

2.4. Longitudinal Control

Longitudinal control is the basis for different automated car initiatives like car platooning, Adaptive Cruise Control (ACC) and forward collision warning and avoidance systems. Vehicle longitudinal control governs the vehicle speed and inter-vehicle spacing. The performance of longitudinal controller contributes to the efficiency of the highway capacity and safety. Platooning and ACC are vehicle following modes dealing with some similar issues. In car platoon, the goal is to maintain close inter-vehicle spacing to increase the highway capacity. However, the main objective of the ACC is to maintain a safe distance to relieve the driver from spacing adjustment. The lower level controller of both vehicle following modes are very similar, the differences in control design philosophy are reflected more in the higher level controller (Vahidi and Eskandarian, 2003).

In dedicated lane concept of AHS, vehicles operate in platoon. Vehicles are supposed to follow the lead vehicle which is equipped with most of the control algorithms. Raza and Ioannou (1996, 1997) presented a well structured supervisory control design for different modes of operation. Platoon string stability is the key issue in the platooning research. Swaroop *et al.* (1994, 1998 and 1999) showed that the determination of vehicle acceleration in a platoon to ensure string stability such that the inter-vehicle spacing error does not grow toward to the end of the platoon. To determine vehicle acceleration, Shladover (1995) and Swaroop *et al.* (1994) also

demonstrated that the inter-vehicle communication was necessary for guaranteed string stability of a platoon. In addition to inter-vehicle communication, the use of rear-end sensors was also discussed in Zhang *et al.* (1999).

In the mixed traffic concept of AHS, vehicles operated as intelligent agents. The objectives of the longitudinal controller are safety and comfort to the passengers and the drivers of manual driven vehicles which share the traffic. The ACC emphasis has been on the improving safety and passenger comfort rather than increasing the road capacity. Gerdes and Hedrick (1997) applied a sliding mode control for vehicle speed and spacing adjustment. Lu *et al.* (2001) used optimal dynamic back-stepping control in deriving the desired acceleration. Liang and Peng (1998 and 1999) implemented an optimal control design to balance between various requirements in a following maneuver. Dai *et al.* (2005) used a fuzzy longitudinal controller with reinforcement learning for tuning fuzzy parameter to achieve vehicle longitudinal control.

Naranjo *et al.* (2006) reported on adaptive cruise control (ACC) with “Stop & Go” maneuvers. The controller adapts to a user-preset speed and reduces speed to keep a safe distance from the vehicle ahead in the same lane of the road. The extreme case is the stop and go operation in which the lead car stops and the vehicle at the rear must also do so.

Both platooning and ACC adjust the inter-vehicle spacing according to the desired space between the vehicles. The determination of the safe distance following is a matter of debate but most researchers adopt the standard distance suggested by road and safety authorities for manually driven cars. There are three basic classifications: constant spacing, constant time headway, and constant safety factor

(Shladover, 1995). The constant spacing systems are most suitable for using at very short spacing therefore it is adopted in the car platooning approach. The constant time headway vehicle following system defines the following distance by the controlled vehicle speed and desired time constant. In this design, the spacing varies proportionally with the speed of the controlled vehicle. The design mimics the human driver distance keeping algorithm. The constant safety factor systems may use any of several safety criteria, but typically they maintain spacing proportional to V^2 (Shladover, 1995).

The longitudinal control adopted in this thesis is a basic cruise controller which adjusts the vehicle speed according to the desired speed. The cruise controller acted as a lower level controller so that the longitudinal controller was decomposed into different levels for future development. For application of experimental studies in this thesis, a PID controller is implemented and tuned manually.

2.5. Soft Computing Methodologies

Computation methods that emulate human decision making and problem solving processes are collectively referred to as soft computing techniques (Aminzadeh, 1994). Fuzzy logic, neural network, probabilistic reasoning, expert systems, and genetic algorithms are main constituents of soft computing methodologies.

In this thesis, fuzzy logic (FL), neural network (NN) and genetic algorithms (GA) are employed in the deployment of control and decision making algorithms.

In this section, the focus is on the basic structure of hybrid soft computing

algorithms. In Section 2.4.1, the author discusses the architecture of GA for fuzzy rule optimization and related publications on FL/GA integration for vehicle control. Section 2.4.2 describes the use of GA in NN weight training.

2.5.1. Fuzzy systems

Classical control design solves a problem mathematically, however complex tasks are often difficult to be formulated completely in terms of mathematical functions or the mathematical functions are too complex to be solved efficiently with modest computing power. As the world becomes more complex, the ability to incorporate information becomes more important and critical to controller design (Zadeh 1973). Fuzzy systems (FS) capture imprecise human knowledge into computer with linguistic IF-THEN rules of FS. The knowledge capturing feature of FS helps the information incorporation in controller design. FS are also able to represent some complex nonlinear systems with simple structure instead of complex mathematical models. Effectively FS provides a systematic mechanism for transforming knowledge base into non-linear mapping.

A typical FS is generally divided into 3 independent units as shown in Figure 2.2. These modules are known as Fuzzification unit, inference mechanism, and Defuzzification unit. A fuzzy system is a nonlinear mapping between its inputs and outputs. Since the nonlinear mapping characteristic is described linguistically in the rule-base of the inference mechanism, the measured numerical data are converted into linguistic fuzzy set by the fuzzification process.

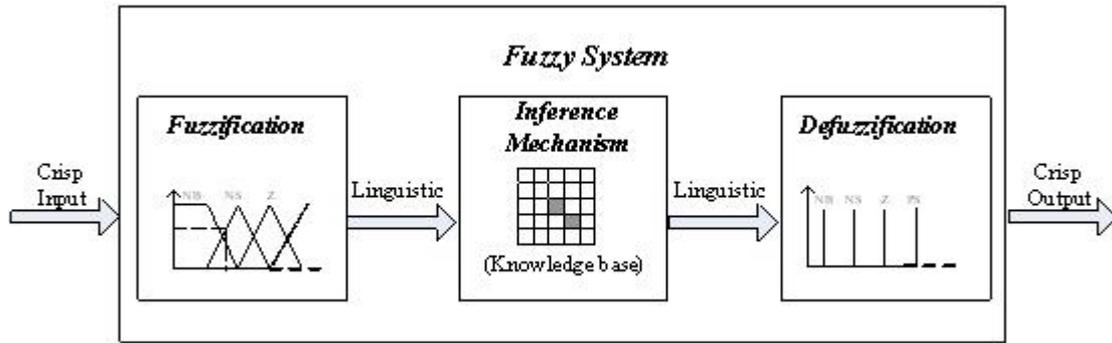


Figure 2-2: Typical fuzzy system architecture.

There are several typical types of fuzzy rule-base such as Mamdani, TSK, and Fuzzy base functions (FBF) and the major difference are in the consequent of the fuzzy rule. Mamdani fuzzy systems (Mamdani 1974): The consequent of fuzzy rules are fuzzy linguistic variable therefore the output is restricted to the defined linguistic variables. TSK fuzzy system (Takagi and Sugeno, 1985): The consequent is represented by a function of the input variables. Fuzzy base functions (FBF): The system is essentially the same as Mamdani except that the output consequents are not restricted to the output linguistic variables and is changed to real number.

The detailed mathematical manipulation of fuzzy systems is not included in this Chapter. Those readers interested in the details and definitions may refer to Passino and Yurkovich (1998).

Fuzzy controllers have been implemented in different areas for many years. In a recent literature, Sonbol and Fadali (2006) introduce the concept of fuzzy positive definite and fuzzy negative definite functions for the stability analysis of discrete TSK Types fuzzy systems. The approach uses the cascading of a system and a fuzzy Lyapunov function candidate to derive new conditions for stability and asymptotic

stability for discrete fuzzy systems. Lam and Leung (2007) present a fuzzy controller consists of two groups of fuzzy rules namely stability and performance rules which offers a property that tuning the performance rules will not alter the system stability. Aceves-Lopez and Aguilar-Martin, J. (2006) propose the natural logic controller (NLC) that it comes through a very important simplification of the Mamdani's fuzzy controller (MFC) allowing easy-design for single-input-single-output (SISO) regulation problems. The NLC approach minimizes fuzzy partition (only two fuzzy subsets per variable) and it uses the minimal fuzzy rule base (only two rules). Chen and Saif (2005) design a new fuzzy system containing a dynamic rule base. The dynamic nature of the rule base has a fixed number of rules and allows the fuzzy sets to dynamically change or move with the inputs.

2.5.2. Genetic algorithms

Genetic algorithms (GAs) are search algorithms based on the mechanism of natural selection and genetic reproduction. They use a simulated version of survival of the fittest in improving the features of potential solutions. GAs guide the selection of chromosomes such that chromosome with the best evaluations reproduce more often than those with bad evaluation.

Since GAs are derivative free optimization, they can solve non-linear, discontinuous, multi-objective optimization problems that are difficult using classical optimization techniques. These algorithms, using simple encoding and reproduction mechanisms, are able to solve some challenging problems in systems and control. GAs have extensively been studied and applied to non-linear, discontinuous, multi-objective optimisation problems (Teng and Wang (2004), Wang *et. al.* (2004), Herrera and Verdegay (1996), Sharma and Irwin (2003) and Chan *et. al.* (2003)).

The GAs manipulation process uses genetic operators to produce a new population of individuals (offspring) by manipulation the chromosome (which code the property of the solution candidate). The algorithms start working by evaluating many scenarios automatically in the evaluation module until finding an optimal answer. The multiple objectives are solved by formulating those objectives in proportion to the fitness function.

The coding methods, genetic operators, fitness evaluations and selection techniques form the skeleton of GAs. Coding means the formulation of parameters to be optimized into chromosome. Genetic operators are typically crossover and mutation which perform genes recombination between selected chromosomes and introduce variation into chromosomes respectively.

The fitness evaluation tests the performance of the solution candidate (chromosome) and grades it by an index function for performance comparison. The index function is named as fitness function. The optimization objective is embedded in the fitness function so that the index can reflect the achievement of the chromosome. The multiple objectives are solved by formulating those objectives in proportion to the fitness function.

The selections include parent selection and generation selection. Roulette wheel selection will be used to choose parent chromosomes where the corresponding angle of the wheel sector is proportional the chromosome fitness. For the generation selection, the best members of each generation are copied into succeeding generation.

The following steps summarize the procedure of GA optimization:

1. Initializes chromosomes randomly to form an initial population
2. Tests the performance of the chromosome in simulation. Then, calculate the fitness value and assign chromosome fitness
3. Select parent chromosomes from the population to reproduce children chromosomes. The selection is based on Roulette wheel selection.
4. Reproduce children chromosomes by either crossover or mutation.
5. Tests the performance of the children chromosome in simulation. Then, calculate the fitness value and assign chromosome fitness
6. Select the best fit chromosomes from children and parent chromosome to form new generation.
7. Repeat Steps 2 to 6 until either maximum number of generations is reached or desired fitness value is evaluated.

2.5.3. Hybrid Fuzzy system and Genetic algorithms

Many researchers have suggested that the fuzzy systems (FS) and the genetic algorithm (GA) complement each other and have advocated their integration (Zadeh (1979), Goldberg (1989), Davis (1991), Karr (1991), Casillas *et al.* (2005), and Juang (2005)). Different types of GA optimized fuzzy controllers are proposed for vehicle lateral controller (Fu *et al.* (2004a and 2004b) and Cai *et al.* (2003)). GA can be used in various ways to optimise a fuzzy system. It can be used to tune the membership functions (Teng and Wang (2004)), scaling factors (Casillas *et al.* (2005)), rule-base (Chan *et al.* (2000) and Wang *et al.* (2004)) or all design parameters simultaneously. The reader may refer to an edited volume by Herrera and Verdegay (1996) for a comprehensive collection of the papers on integration of fuzzy systems and GA.

Wang et al. (2004) proposed a GA-based output-feedback direct adaptive fuzzy-neural controller for uncertain nonlinear dynamical systems. They established weighting vector of the controller by the Lyapunov design approach and tuned the weighting factors via GA. Sharma and Irwin (2003) proposed a fuzzy chromosome encoding method to represent real number parameters in a genetic algorithm context. The fuzzy coding provided the value of parameters on the basis of the optimum number of selected fuzzy sets and their effectiveness in terms of degree of membership.

Chang et al. (2003) proposed a discrete-time Takagi-Sugeno (TS) fuzzy-model-based controller using GA search space. Juang (2005) proposed a combination of online clustering and Q-value based GA learning scheme for fuzzy system design with reinforcements. The precondition part of a fuzzy system was constructed online by an aligned clustering-based approach whereas the consequent part was designed by a Q-value based genetic reinforcement learning.

Teng and Wang (2004) proposed a GA-based algorithm to construct a fuzzy system for approximating an unknown system determined automatically number of membership functions and parameters of membership functions. The algorithm also tried to discard the dummy input variables.

Casillas *et. al.* (2005) introduced a genetic tuning process to use linguistic hedges to slightly modify the symbolic representations to keep good interpretability of the fuzzy system. They also proposed refining the scaling factors to modify the membership functions of a linguistic fuzzy model.

Chan and Rad (2000) proposed the use of symmetric rule-base and mirror action for chromosomes to improve convergence by exponentially reducing the search space. Chan et al. (2003) also proposed the use of supervisory control to guarantee stability and improve convergence of the integrated FS/GA.

This section presents the integration of fuzzy system and genetic algorithm (FS/GA). The rule-table of the fuzzy system will be optimized by GA. The fuzzy rules are encoded into chromosome for genetic evolution. Consider Mamdani type fuzzy system, the antecedents are represented by linguistic variables. The following shows the typical procedure to code a Mamdani type fuzzy rules-table into chromosome by integer coding. Consider a two-input/one-output fuzzy system and assume that each has seven fuzzy sub-sets. The linguistic variables of output, namely NB, NM, NS, ZE, PS, PM, PB are coded as 1, 2, ..., 7 respectively. Figure 2-3 shows a chromosome with length of 49; i.e. the fuzzy system has 49 rules. The first gene of the chromosome codes the first rule as:

Rule 1: IF x_1 is NB and x_2 is PB THEN y is NB.

where the first gene is set to “1” representing “NB” which is the consequent part of Rule 1.

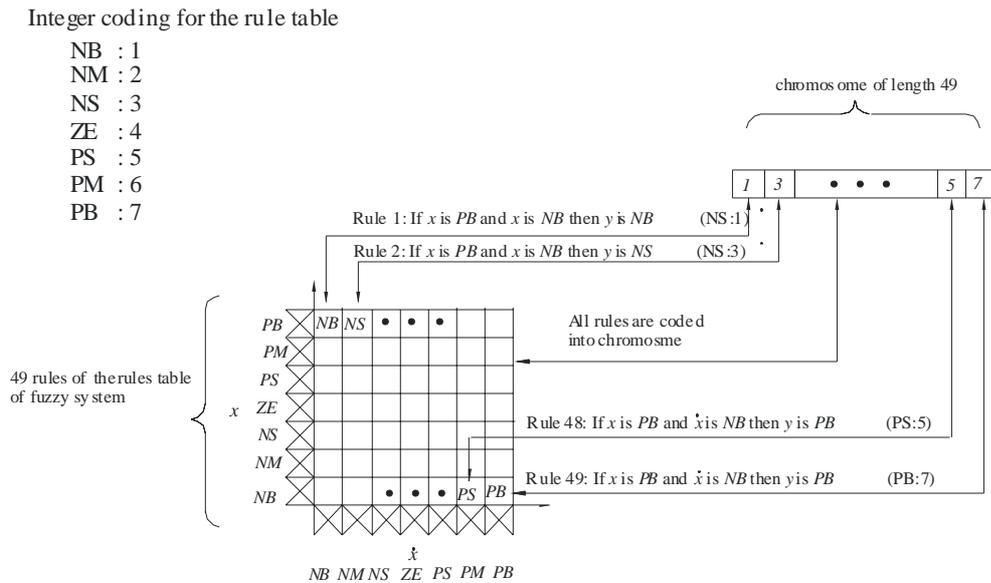


Figure 2-3 : Integer coding of the general chromosomes for fuzzy systems.

Figure 2-4 shows a traditional architecture of the FS/GA. There are three main modules namely evaluation, selection and reproduction. Given an optimisation problem, GA encodes the fuzzy parameters into chromosomes, each of which presents a possible solution to the problem. GA tests the initial rule-base for the given problem. It calculates the fitness values of the Fuzzy System. Next, selection, reproduction, evaluation modules are used to generate the next population.

The Evaluation Module contains an evaluation function that measures the fitness of chromosomes of the problem to be solved. The Selection Module selects individuals for the next generation.

The Reproduction Module contains techniques for creating new offspring. GA use the three modules (selection, reproduction and evaluation) based on the fitness function for an iterative evolution. The reproduction mechanism, together with crossover and mutation (Goldberg (1989)), cause the best schema (segment of good

features) to proliferate in the population, i.e. to produce high-quality schema in a single chromosome. Repeated selection and crossover cause the continuous evolution of the gene pool and the generation of individuals that survive better in a competitive environment, emulating the ‘survival of the fittest’ mechanism in nature.

Finally GA decodes the solution to the problem from the best chromosome. The algorithms could either be terminated after a fixed number of generation or after a chromosome with a certain high fitness value is allocated.

The integration of FS/GA automates the process of FS parameters determination. In this thesis, Mamdani type rule tables of fuzzy controllers for vehicle lateral control and process control are presented in Chapter 5. The GA optimizes the rule table based on the difference between the desired output and process output. Therefore the rule table evolves towards the desired system performance.

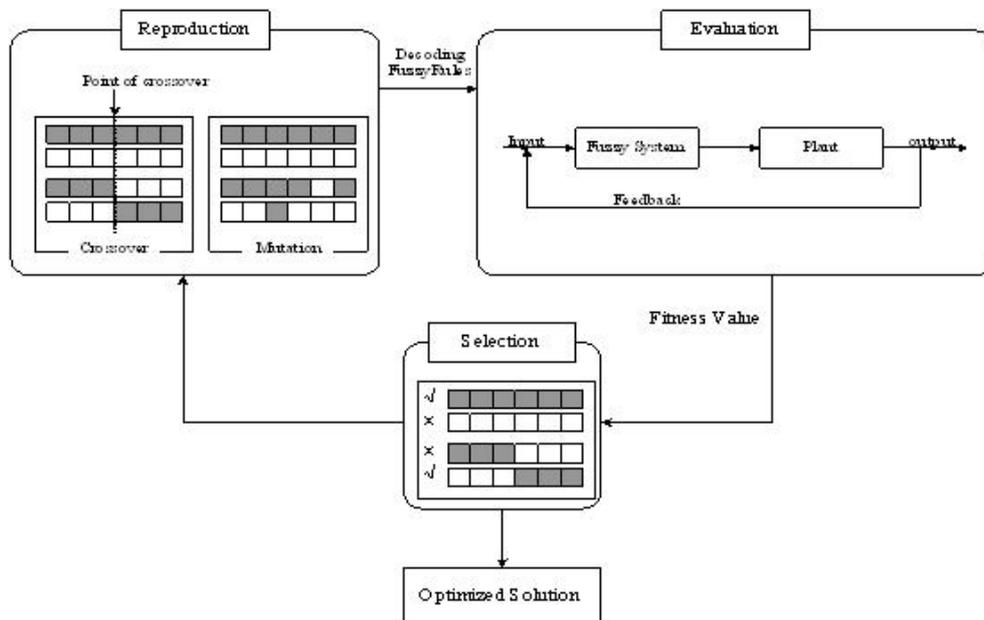


Figure 2-4: Optimisation of fuzzy logic controller by genetic algorithms.

GA optimization is a time consuming process due to the evaluation of the entire

possible solution candidate in the population. The enhancement in convergence rate would also benefit to the reduction in optimization time. In Chapter 5 a novel fire rules chromosome (FRC) encoding scheme is proposed to enhance the convergence rate. The suggested FRC scheme would alter the chromosome length dynamically throughout the optimization process.

2.5.4. Neural Network

Multilayer neural network controllers can be used to control nonlinear systems. This is due to the nonlinear mapping characteristic of these powerful processing networks. Nevertheless, neural network structure and number of neurons is application dependent and there is no generally accepted theory or methodology for the design of neural network Tsoukalas and Uhrig (1997).

Increasing the number of neurons and layers would enhance the network nonlinear mapping capability. However, excessive numbers of neurons increase the network complexity and reduce the effectiveness of network optimization. The objective of this chapter is introducing a simplified neural network structure for known physical structure systems.

The design of neural network structure and number of neurons is application dependent and there is no generally accepted theory or methodology for the design of neural network Tsoukalas and Uhrig (1997). In general neural network structure can be designed through a trial and error procedure. Increasing the number of neurons and layers would enhance the network nonlinear mapping capability. However, excessive numbers of neurons increase the network complexity and reduce the effectiveness of network optimization.

A multilayer neural network refers to a system that consists of numbers of highly interconnected neurons. These neurons are organized into a sequence of layers with full or random connection between layers. Each of the connection between neurons has an adjustable weight, the connection weight. For a typical fully connected multilayer neural network, the increase in number of neuron and layer would exponentially increase the number of connection weight.

For a fully connected neural network with n layers, the number of connection weights can be calculated as follow. Assuming there are i input buffers, j neurons in each hidden layer and k outputs in network. The total number of connection weight (W) is calculated by equation (2.1). If the network parameters are follow $n = 3$, $i = 4$, $j = 2$, and $k = 1$ then the total number of connection weights (W) is 10.

$$W = j(i + (n - 3)j + k) \quad (2.1)$$

Kehtarnavaz *et al.* (1998) implement a neural network module for autonomous vehicle following. The module collects rang and heading angle data of the lead vehicle to perform following function. The speed and steering controller are implemented by 2 independent networks and the numbers of hidden layer are 21 and 12 layers respectively.

Tahirovic *et al.* (2005) use neural network for vehicle longitudinal guidance. The neural network learns from human driver driving style and it was tested in an Audi test vehicle. Zhenhai and Bo (2005) present the use of neural network to tune an adaptive PID controller for vehicle lane keeping. The network tunes PID parameters based on pre-given target trajectory. Kumarawadu and Lee (2006) proposed adaptive

Radial Basis Function (RBF) networks to counteract uncertainty for both longitudinal and lateral control. The lateral and longitudinal controllers are conventional PD type controller with constant gain setting. The RBF network outputs are fused with the PD controllers output for uncertainty compensation.

2.5.5. Hybrid Neural Network and Genetic algorithms

In addition to network establishment by trial and error method, automatic generation of neural network structure is an active topic in neural network research. Genetic Algorithms (GA) is an effective search method for optimization and it is widely applied in network structure determination. In the literature, the use of GA for neural network optimization has been explored by many researchers. Yen and Lu (2000) present hierarchical GA for evolving both network topology and parameters. Tsai, Chou, and Liu (2006) tuning both neural network structure and parameters by modified GA.

Once the network structure is fixed, the remaining task is the training process. Error backpropagation is one of the supervised learning techniques for neural network training. It was introduced by Werbos (1974) and further enhanced by Rumelhart, Hinton, and Williams (1986). The backpropagation training method updates the connection weights based on the gradient of the network error with respect to the network's modifiable weights. The connection weight update rules base on simple gradient descent for error minimization.

The application of GA in neural network not only applies in the network structure determination but also in the network parameter optimization. Whitley and Hanson (1989) presented the use of GA in training network parameter of known

structure. Miller *et al.* (1989) have used the genetic algorithm to evolve the optimal connected matrix to form a neural network. Whitehead and Choate (1996) use GA to find the center and widths of a radial basis function network for time series prediction.

Let us consider the computation burden in network training, with GA. It encodes all the neural network connection weights into chromosome for network optimization. For the error backpropagation training method, the error gradient is calculated with respect to all the connection weights involved in the network. Thus the complexity of the training process and convergence rate of the network optimization techniques mentioned above are heavily depends on the number of connection weight.

In Chapter 4, a fused neural network controller is proposed which based on task decomposition. The proposed fused neural network controller structure decomposes the task of vehicle lateral control into displacement and angular adjustment. Therefore the network structure is simplified into 2 independent networks so that the number of connection weights is reduced in compare with fully connected neural network as shown in (2.1).

To adopt GA for connection weight optimization of neural network, connection weights are encoded into chromosome with real value. Then the GA optimization process is similar to FS/GA discussed in Section 2.5.3. The main different is the encoding parameters are network connection weights instead of fuzzy rule-table.

2.6. Conclusion

In this Chapter, some background information on the AHS, literature survey on

the autonomous vehicle control, including lateral control, longitudinal control, and soft computing methodologies are presented. The integration methodologies of FS/GA and NN/GA have been discussed. This chapter provides underlying materials on problems to be tackled in the later chapters.

The review on the AHS background information provides the foundation on the vehicle control system design. The constraint and requirements are dissimilar in different automation concepts (i.e. the dedicated lane and mixed traffic) and lateral reference system (i.e. look-ahead and look-down reference). The understanding on the overall system constraint facilitates system design and towards realistic implementation.

The literature on vehicle control and soft computing methodologies contain host of control techniques to tackle vehicle control problems on both aspects of lateral and longitudinal control. The techniques implemented are advance but complicated in controller development. Furthermore the traditional integration method does not consider the optimization rate. The proposed simplified neural network and encoding scheme for FS/GA convergence rate enhancement offer simple but efficient technique in contrast to the technique on the literature.

In the next Chapter, the construction of the scaled prototype semi-autonomous vehicle and the mathematical vehicle lateral model will be presented.

Chapter 3. Prototype of a Semi-autonomous Vehicle

3.1. Introduction

The early studies on scaled vehicles were associated to wind tunnels experiments. Later the use of scaled vehicles was extended to automobiles in the area of crash reconstruction, road tire interaction, suspension analysis and dynamics, and roll dynamics. In the past decade, scaled vehicles have been used for controller implementation and verification. Brennan (1999) designed a scaled vehicle system with a treadmill as roadway known as Illinois Roadway Simulator (IRS) for vehicle controller design. In IRS, the vehicle maneuvers on a roadway (treadmill surface) which is moving under the vehicle so that simulate a vehicle traveling on a highway. For the position sensing system and control signal delivery, IRS calculates vehicle position based on mechanically connected potentiometers and delivers control signal from desktop computer through transmitters to the vehicle.

Brennan (1999) states three reasons to use scaled prototype autonomous vehicle in stead of the use of full size vehicle. First, the capital for equipping a full size vehicle and conducting research on it require huge amount of grants. However, conducting research on scaled vehicle can greatly reduce the cost in both construction and maintenance on the prototype vehicle as well as the infrastructure development. Second, the development time on scaled vehicle is much less than a full size vehicle. The time required to modify a scaled vehicle is insignificant compared to a full size vehicle. Third, scaled vehicle testing platform offers safer testing environment than full size vehicle for controller testing. In scaled testing, there are no drivers or

pedestrians are put at risk during testing of vehicle controllers.

A scaled semi-autonomous prototype was constructed as a test bed for experimental studies of different controllers designed in the course of this project. The use of scaled vehicle as a test bed can be traced back to 1960's. The autonomous prototype vehicle presented in this chapter is equipped with onboard sensors, actuators, interface and processing units. The vehicle dynamics parameters (lateral position, speed, and steering angle) are measured by onboard sensors. The measured signal is sampled by the onboard computer for control signal generation. The hardware construction and configuration will also be presented in this chapter.

Computer control system is an essential component in automatic vehicle control. The flow of computer program should be systematic so that performs sampling and processing of sensor information in real time. The onboard computer of the prototype vehicle samples sensor information and generates control signal within the timer interrupt service routine. The timer interrupt service routine ensures the data sampling in fixed period of time so that the vehicle system is feasible for real time application.

Understanding the theoretical lateral model of the prototype autonomous vehicle is beneficial to controller development and design. To identify the vehicle lateral model, the structure of bicycle model is selected for system identification. The identified bicycle model is presented in this chapter and it is extensively used in the studies presented in the thesis.

This chapter is organized as follow: Section 3.2 describes the hardware configuration and the interfacing methods for data sampling and signal generation.

Section 3.2 shows the architecture of the software program. Section 3.3 discusses the lateral model of the prototype autonomous vehicle. Finally Section 3.4 concludes this chapter.

3.2. Hardware design

This Section presents the detailed information of hardware design including vehicle construction, sensors, actuator, computer interface and the circuit design. Figure 3-1 shows the schematic of hardware configuration. The hardware configuration consists of 3 parts: sensors, industrial PC and actuators. Sensors collect vehicle dynamics information (speed, lateral and front distance). The industrial computer acquires the sensor information through Digital I/O card or A/D card. The implemented control algorithms process the data collected and generate control signal to the actuators for lateral and longitudinal control.

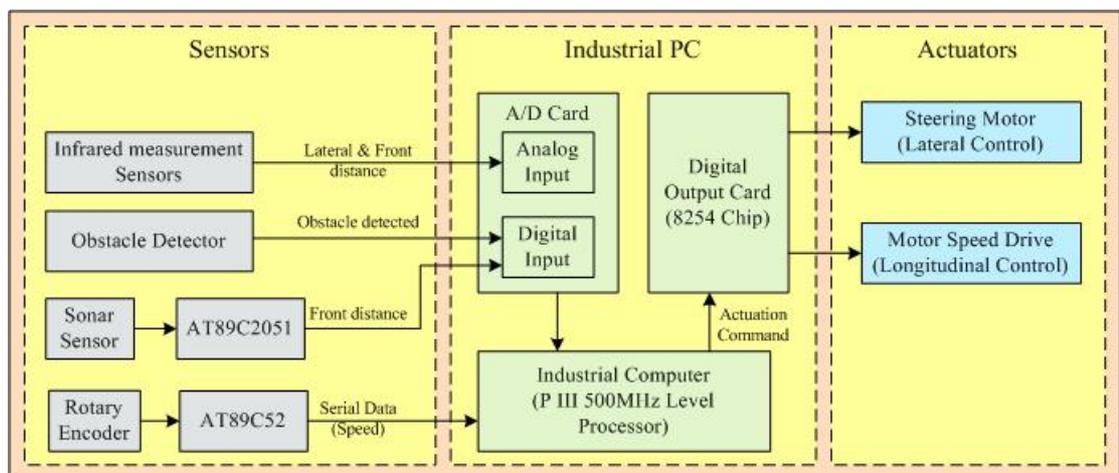


Figure 3-1: Schematic of hardware configuration.

The following subsections show the detail specification of the hardware being

selected for the prototype vehicle. The associated microcontroller programs for driving sensors will also be discussed in the following sections.

3.2.1. Vehicle Design

The prototype vehicle (Figure 3-2) is a modified radio controlled (RC) model car with size of 1/10th of a full size vehicle and the vehicle is front wheel steering and driving. The existing hardware (chassis, steering mechanism, gear box, and wheels) of the RC car is retained as the mechanical building blocks of the vehicle. A platform on the top of the vehicle chassis was built for electronic circuitries, sensors and control units installation. Figure 3-3 illustrates the allocation of the hardware and infrared sensors. The shaded blocks are infrared sensors (GP2D12) for distance measurement. There are 5 infrared sensors mounted on the vehicle which located at the front and both sides of the vehicle. Sensors located at the side measure lateral displacement from the reference wall. The lateral distance measured near the vehicle front and tail are named as \bar{y}_f and \bar{y}_r respectively. For the sensor located at the front of the vehicle measures longitudinal distance from the vehicle to obstacle and the measured distance is named as \bar{y}_d .

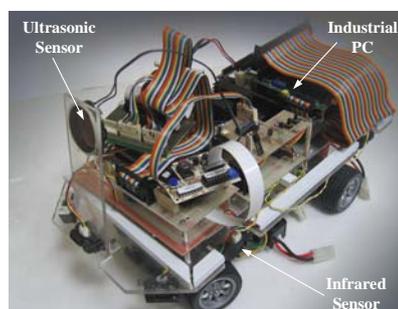


Figure 3-2: Prototype semi-autonomous vehicle.

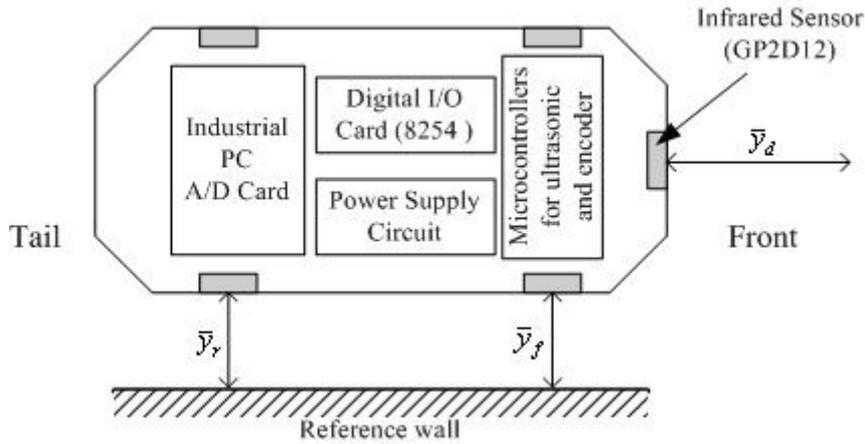


Figure 3-3: Infrared sensors allocation.

3.2.2. Sensors

3.2.2.1. Infrared measurement sensor

The infrared sensors (Sharp GP2D12) were installed on both side of the vehicle for measuring vehicle lateral position from the reference path. The Sharp GP2D12 infrared sensor (Figure 3-4) takes continuous measurement after power on and reports distance as analog voltage. The range of distance measurement is 10cm to 80cm.

However, the mapping of analog voltage to the distance being measured is nonlinear. Figure 3-5 shows the typical nonlinear output of GP2D12 that provided by the manufacturer. The solid and dashed lines shown in the figure are the sensor measurement output voltage on objects with reflectivity 90% and 18% respectively. Since both solid and dashed lines are close to each other, it shows that the sensor output is robust to object reflectivity.

To convert the sensor output voltage to distance measured (\bar{y}_d , \bar{y}_f , and \bar{y}_r), a

fifth order polynomial was used. The coefficients of the polynomial were obtained from experiment data via least square method.



Figure 3-4: Sharp GP2D12 infrared sensor.

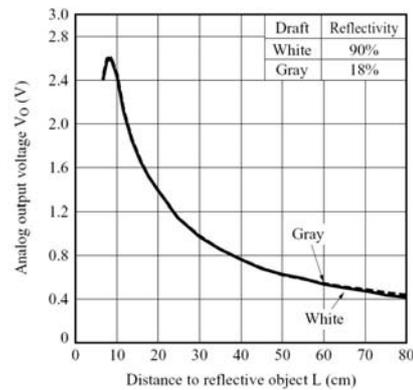


Figure 3-5: Typical nonlinear output of GP2D12.

3.1.1.1. Obstacle detector

The obstacle detector (Sharp GP2D05) detects the object close to vehicle surrounding. The GP2D05 (Figure 3-6) output is either logic 1 or 0 to indicate the present or absent of obstacle respectively. The range threshold can be adjusted from 10cm to 80cm by a variable resistor on the back of the detector. To initiate measurement, the detector input should bring to low and maintain at low for 56ms. When the measurement finished the input should float high for at least 1ms to reset the detector.

The detectors are installed at the corners of the vehicle for detecting obstacles or vehicle at neighborhood lanes.



Figure 3-6: Sharp GP2D05 obstacle detector.

3.1.1.2. Ultrasonic Sensor

The aforementioned infrared sensors have measurement range 10cm - 80cm but it is not sufficient for longitudinal distance measurement. Therefore ultrasonic sensor was installed at the front end of the vehicle for longitudinal distance measurement. The ultrasonic sensor is Polaroid 6500 series ranging module (Figure 3-7). The module effectively is a sonar driver which transmitting ultrasonic pulses on request (initiation signal). After initiation of transmitting pulses, the modules will generate a feedback signal when an echo is detected.

The distance measurement is achieved by counting the time different between the initiation signal and the echo feedback. A microcontroller program is designed to taking measurement automatically.



Figure 3-7: Polaroid 6500 series ranging module

The sensor has two basic modes of operation: single-echo and multiple-echo modes. Single-echo mode operation reports the first echo feedback after the initiation of measurement. The pins INT and ECHO on the sonar driver are used to initiate measurement and output echo feedback respectively. The measurement is initiated by pulling the INT pin to high and maintain at high until the echo feedback. The time between the INT going high and the ECHO output going high was proportional to the distance being measured.

The micro-controller AT89C2051 was used to handle the measurement process of the ultrasonic sensor. Figure 3-8 shows the micro-controller flow chart for the ultrasonic sensor. The timer is initialized by setting overflow at $18ms$ which is the time required to measure $2m$. The micro-controller starts timer when measurement is initiated. The timer stops only when either echo received or Timer overflow. Finally, the micro-controller reports the time counted in the timer. The computer reads the output port of the micro-controller and calculates the distance measured by the sonar sensor.

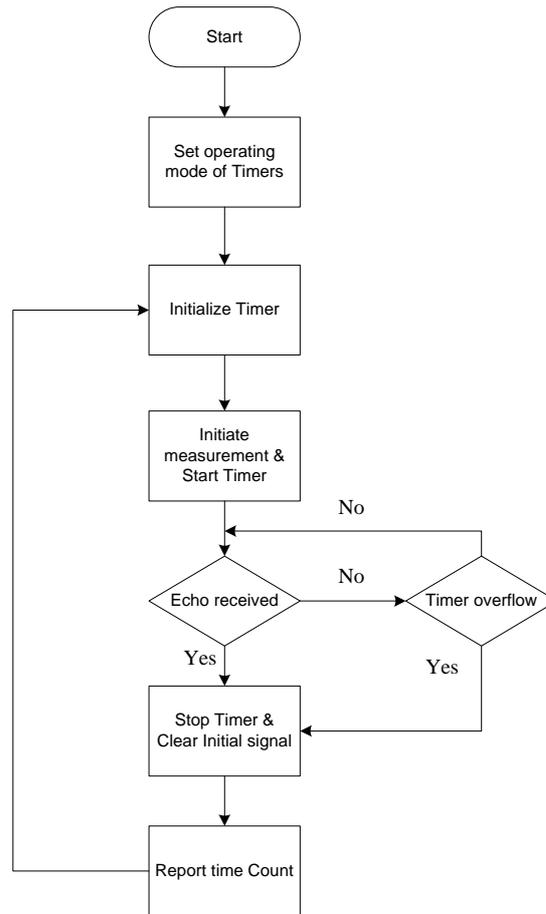


Figure 3-8: Micro-controller program flow chart for ultrasonic sensor.

3.1.1.3. Speed sensor

To measure the speed of the vehicle, a rotary encoder was installed at the rear draft of the prototype vehicle. The encoder converts rotary movement to electrical pulses. The vehicle speed is measured by counting the time required to receive a fixed number of pulses generated from the encoder.

A micro-controller AT89c52 was used to receiving the signal from encoder. The AT89c52 provides 3 timers/counters which suitable for the application of pulses counting, time counting, and serial communication.

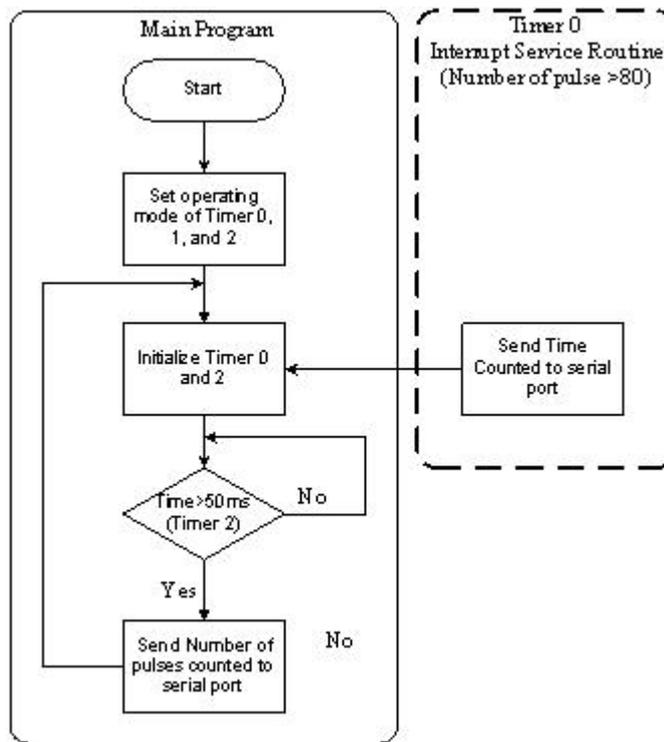


Figure 3-9: Micro-controller program flow chart for speed sensor.

The 3 Timers of AT89c52 are named as Timer 0, Timer 1, and Timer 2. In this application, Timer 0, Timer 1, and Timer 2 serve as a counter to record the number of pulses from encoder output, a rate generator for serial communication, and a timer for time counting respectively. Figure 3-9 shows the micro-controller program flow chart for the speed sensor. The two-dashed line blocks separate the normal main program flow and the timer 0 interrupt service routine (ISR).

The main program starts with setting the operating mode of the three timers. Then the Timer 0 and Timer 2 are initialized by setting overflow at 80 pulses and 50ms count respectively. If the time counted in Timer 2 is greater than 50ms, the number of pulses counted in timer 0 will be send to serial port. Therefore, it report the number of pulses counted within 50ms. When the data is sent to serial port, both

timers (Timers 0 and 2) are reset (initialized). Therefore the functional blocks initialization, time comparison, and sending data form the loop of main program.

Timer 0 ISR is interrupt driven. The interrupt is initiated when the Timer 0 overflow. When the ISR is activated, the time counted in timer 2 will be send to serial port. Thus it reports the time required to count 80 pulses. Then both timers (Timer 0 and Timer 2) are reset after data sent.

The program architecture ensures that the micro-controller reports speed information at every 50ms under slow vehicle speed. At high speed operation, the micro-controller reports speed information when maximum number of pulse is counted. Finally, the computer converts the data sent from the micro-controller to speed.

3.1.1.4. Initial Calibration on Sensors

Initial testing and calibration was carried out on the hardware for implementation. First, we focused on the sensors, including infrared and speed sensor. As discussed in section 3.2.2.1, for infrared sensors, the relationship between the output voltage and the actual distance was nonlinear. In order to measure the distance accurately, a large number of repeatable experimental data were collected and a fifth-order polynomial model was fitted for each infrared sensor. The polynomial coefficients were derived via least-squares method and are tabulated in Table 3-1.

Sensor position	v^5	v^4	v^3	v^2	v^1	v^0
Left front	-3.977E-13	1.600E-9	-2.545E-6	2.032E-3	-8.500E-1	1.719E+2
Left rear	-4.069E-13	1.642E-9	-2.628E-6	2.118E-3	-8.949E-1	1.815E+2
Right front	-3.065E-13	1.267E-9	-2.095E-6	1.759E-3	-7.820E-1	1.684E+2
Right rear	-3.301E-13	1.353E-9	-2.214E-6	1.836E-3	-8.056E-1	1.7079E+2

Table 3-1: Polynomial coefficients of infrared sensors.

As mentioned in previous section, the encoder was installed on the shaft of the rear wheels and hence off-road testing on the complete speed sensor module could not be conducted for this front wheel-driving vehicle. A square wave signal of 2k Hz frequency (equivalent to 70 cms^{-1}) was generated from a signal generator to replace the encoder output to test the accuracy of the micro controller assembly code.

3.1.2. Actuators

The existing RC car technology was used in both lateral and longitudinal actuators. The lateral control actuator is Futaba S3001 (Figure 3-10a) servo motor in the front wheel steering. The longitudinal control actuator is Apex Plus speed controller (Figure 3-10b). The inputs of both actuators of lateral and longitudinal control are Pulse-width modulation (PWM) signal with 50Hz as fundamental frequency. The generation of the PWM signal will be shown in the Section 3.1.1.1.

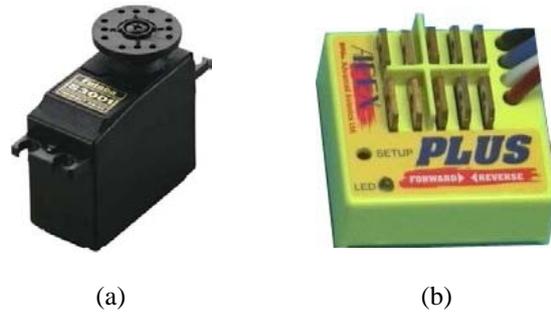


Figure 3-10: (a) Futaba S3001 servo motor. (b) Apex Plus speed controller.

3.1.3. Industrial Computer

The computing unit used in this prototype vehicle is a compact industrial computer. The computer is *ADVANTECH* PCM-9370 with on-board embedded Transmeta Crusoe TM 5400-500 Processor (Pentium III level) and 64MB RAM. This Industrial computer is selected for prototype vehicle because the size is compact ($145mm \times 102mm$) and the power supply requirement is single +5V, which is feasible to install on this prototype vehicle.

The use of industrial computer as the computing unit offers the advantage of easy interfacing with the existing standard data acquisition modules such as analog to digital conversion card. However, the main disadvantage of using industrial computer as on board computing unit in scaled vehicle is high power consumption.

3.1.4. Digital I/O Cards

3.1.4.1. Analog to Digital Conversion card

The Analog to Digital Conversion (A/D) Card used is *ADVANTECH*

PCM-3718HG which has 16 single-ended or 8 differential analog inputs and 16 channels digital input/output (I/O). The A/D conversion resolution is 12 bits. It interface with computer through PC/104 slot and the power required by the A/D card is also +5V therefore compatible with the industrial computer.

Since the A/D card not only offers analog to digital channels but also digital I/O channels, in addition to analog output sensor, digital output sensors are connected to the A/D cards. The infrared sensors (GP2D12) output are connected to the A/D channels which configured as single-ended inputs for distance measurement. For digital output sensors, the microcontroller outputs for ultrasonic measurement and the obstacle detectors are connected to the digital I/O ports of the A/D card.

3.1.1.1. Digital Output card with 8254 chips

The steering servo motor and the hardware speed controller require PWM signal with 50Hz fundamental frequency as their control signal. In order to generate the PWM signal, 8254 digital output card is used. A single 8254 chips can generate 2 PWM signals with the same fundamental frequency under specific circuitry and program.

The 8254 chip consists of 3 counters and each counter can be programmed in different mode of operation. Each counter has 2 inputs and 1 counter output (OUT). The 2 inputs are clock signal (CLK) and gate signal (GATE). In general, the counters

are working as a programmable timer which will alter the output signal (OUT) when desired time is counted. The counting of time is defined by number of CLK pulses counted. Different operation modes define the counter OUT behavior with respect to the GATE input and the CLK counted. The detail description on the different operating modes and functions of input signal can be referred to the manufacturer datasheet.

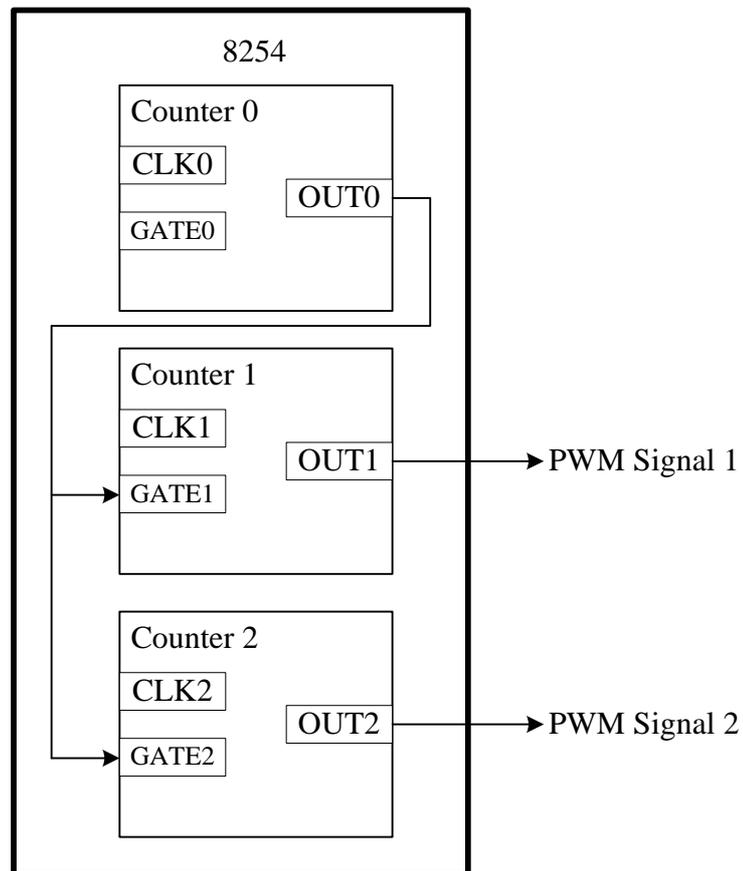


Figure 3-11: 8254 wiring.

For the application of PWM generation, the input and outputs of the 3 counters are connected as shown in Figure 3-11. The CLK are internally connected to the system clock and does not shown in the figure therefore the counters share the same frequency of clock signal. The output of Counter 0 (OUT0) is connected to gate signal input of Counter 1 (GATE1) and 2 (GATE2). The outputs OUT1 and OUT2 are served as the PWM signal outputs.

Counter 0 is programmed as a rate generator (Mode 2) to generate pulses with 50Hz. The detail of Mode 2 operation is not described here, for readers have interest please refer to manufacturer datasheet. Since OUT0 is connected to GATE1 and GATE2, the GATE of Counter 1 and 2 is effectively triggered by pulses with 50Hz.

For Counter 1 and Counter 2, they are programmed as “hardware retriggerable one shot” (Mode 1) operation. The following briefly describes the mode 1 operation. At mode 1 operation, the OUT signal is initially at high and it will reset to low and remains at low when GATE is triggered (rising edge). Meanwhile, the GATE also triggers the counter to start counting. When desired number of CLK is counted the OUT will set to high and thus completes a counting cycle.

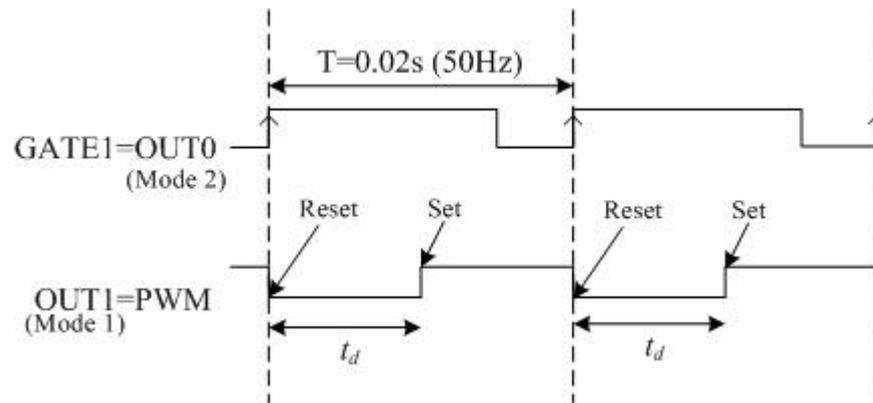


Figure 3-12: Generation of PWM signal.

Figure 3-12 illustrates the timing diagram of PWM signal generation by 8254 chips. Although the figure only shows the output of Counter 1, the operation is the same in Counter 2. The time of GATE1 being positive triggered is indicated by vertical dash lines. Refer to Figure 3-11 OUT0 is connected to GAET1 thus GATE1 is the signal generated by Counter 0. The OUT1 is reset (low) by the positive trigger of GATE1 and set (high) after counting the desired time (t_d). This output pattern is effectively a PWM signal, if the t_d is less than the period of 50Hz (0.02s).

3.2. Software

The real-time control experiments were implemented in C code. To sample the sensor information and deliver control signal to actuators in a fix time interval, onboard timer interrupt service routine (ISR) was used. The onboard timer generated interrupt at every 50ms and thus the ISR is initiated every 50ms.

When timer interrupt is initiated, the ISR starts with sampling analog data from the A/D card and then sampling digital data from the digital input ports. After data sampling, the sensor information is passed to control algorithms so that generate control commands for lateral and longitudinal control. The control commands are converted to control signals and deliver to actuators at the end of the ISR. The above describes the general flow of the ISR. In the following paragraphs give descriptions on the program flow of control algorithms.

Figure 3-13 describes the flow chart of the control program. After receiving the information from the sensors, the control algorithm would be applied. If the vehicle was in a emergency situation, such as approaching an obstacle, a stop command would be initiated. The longitudinal and speed control is always taking action. If Lane changing maneuver is required, the control loop will switch to lane change algorithm. The lane keeping control will be activated under normal driving condition.

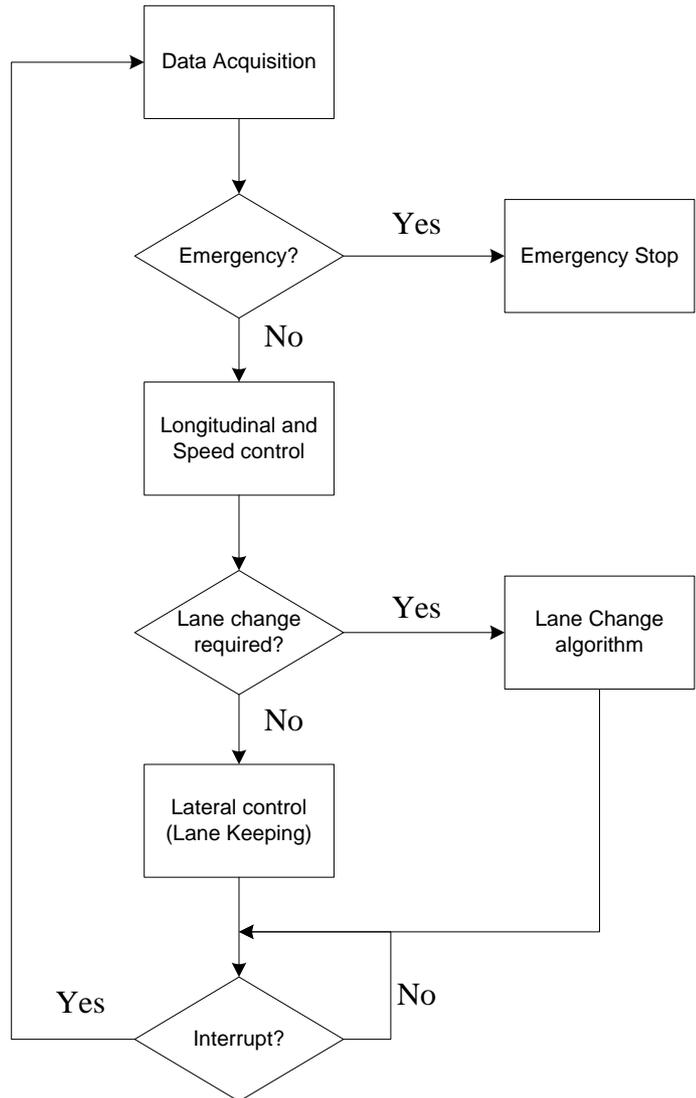


Figure 3-13: Program flow chat.

3.3. *Vehicle Model*

Modelling of the vehicle lateral dynamics is accomplished by fixing a coordinate system to the center of gravity of the vehicle. Roll, pitch, bounce and deceleration dynamics are neglected to simplify the vehicle dynamics to two degrees of freedom: the lateral position and yaw angle. The model is further simplified by assuming that

each axle shares the same steering angle and that consequently each wheel produces the same wheel angle steering forces. The resulting dynamics model is known as the bicycle model, because the dynamics conceptually model a bicycle whose motion is constrained to in-plane motion (Brennan and Alleyne, 2001).

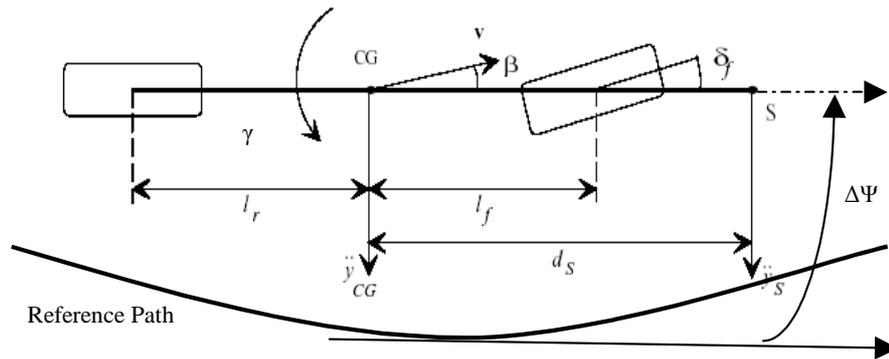


Figure 3-14: Classical bicycle model.

$$\frac{d}{dt} \begin{bmatrix} \beta \\ \gamma \\ \Delta\Psi \\ y_s \end{bmatrix} = \begin{bmatrix} -\frac{\mu(c_r + c_f)}{Mv} & -1 + \frac{\mu(c_r l_r - c_f l_f)}{Mv^2} & 0 & 0 \\ \frac{\mu(c_r l_r - c_f l_f)}{J} & -\frac{\mu(c_r l_r^2 + c_f l_f^2)}{Jv} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ v & d_s & v & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \gamma \\ \Delta\Psi \\ y_s \end{bmatrix} + \begin{bmatrix} \frac{\mu c_f}{Mv} & 0 \\ \frac{\mu c_f l_f}{J} & 0 \\ 0 & -v \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_f \\ \rho_{ref} \end{bmatrix} \quad (3.1)$$

The bicycle model is one of the most widely used today for purposes of vehicle lateral control. The classical bicycle model description and the associated equation are shown in Figure 3-14 and (3.1) respectively. Although the bicycle model is relatively simple, it has been verified to be a good approximation for full-size vehicle dynamics with small steering angle. The model variables denote:

- V vehicle velocity vector with $v = |v| > 0$; speed v is assumed measurable,
- B side slip angle between vehicle longitudinal axis and velocity vector v at CG,
- Γ vehicle yaw angle with respect to a fixed inertial coordinate system,
- \ddot{y}_{CG} lateral acceleration at CG,
- \ddot{y}_S lateral acceleration at sensor S ,

δ_f	front wheel steering angle.
$\Delta\Psi$	angular displacement error with the tangent to the reference path,
ρ_{ref}	reference road curvature.
M	total vehicle mass,
I_ψ	total vehicle inertia about vertical axis at CG,
$l_f(l_r)$	distance of front(rear) axle from CG with $l = l_f + l_r$,
d_s	“look-ahead” distance between sensor location S and CG,
$c_f(c_r)$	front(rear) tire cornering stiffness,
M	road adhesion as a factor of effective tire cornering stiffness $c_f^* = \mu c_f$ ($c_r^* = \mu c_r$)

Guldner *et. al.* (1997) analyzed the vehicle system in detail, including time domain, frequency domain and eigenvalue domain analysis, and proposed a control design direction which was to add a second sensor to measure lateral vehicle displacement from the lane reference at tail bumper. Assuming that the two sensors are mounted at d_f for front of and d_r for behind of center of gravity (CG) to measure the displacement y_f and y_r of the vehicle from the reference road. The new state space model (3.3) can be transformed form (3.1) using the transform matrix P shown in (3.2). The Figure 3-15 shows the modified bicycle model.

$$\begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ v & d_f & v & 0 \\ 0 & 0 & -(d_f + d_r) & 1 \\ v & -d_r & v & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \gamma \\ \Delta\Psi \\ y_s \end{bmatrix} = P \begin{bmatrix} \beta \\ \gamma \\ \Delta\Psi \\ y_s \end{bmatrix} \quad (3.2)$$

The new state space model is

$$\frac{d}{dt} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a_{21} & a_{22} & -a_{21} & a_{24} \\ 0 & 0 & 0 & 1 \\ a_{41} & a_{42} & -a_{41} & a_{44} \end{bmatrix} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ 0 & b_{32} \\ b_{41} & b_{42} \end{bmatrix} \begin{bmatrix} \delta_f \\ \rho_{ref} \end{bmatrix} \quad (3.3)$$

$$a_{21} = \frac{\mu(c_f + c_r)}{M(d_f + d_r)} - \frac{d_f \mu(c_r l_r - c_f l_f)}{J(d_f + d_r)}$$

$$a_{22} = \frac{\mu(c_r l_r - c_f l_f) - d_r \mu(c_f + c_r)}{Mv(d_f + d_r)} + \frac{d_f [d_r \mu(c_r l_r - c_f l_f) - \mu(c_r l_r^2 + c_f l_f^2)]}{Jv(d_f + d_r)}$$

$$a_{24} = -\frac{\mu(c_r l_r - c_f l_f) + d_f \mu(c_f + c_r)}{Mv(d_f + d_r)} + \frac{d_f [d_f \mu(c_r l_r - c_f l_f) + \mu(c_r l_r^2 + c_f l_f^2)]}{Jv(d_f + d_r)}$$

$$a_{41} = \frac{\mu(c_f + c_r)}{M(d_f + d_r)} + \frac{d_r \mu(c_r l_r - c_f l_f)}{J(d_f + d_r)}$$

$$a_{42} = \frac{\mu(c_r l_r - c_f l_f) - d_r \mu(c_f + c_r)}{Mv(d_f + d_r)} - \frac{d_r [d_r \mu(c_r l_r - c_f l_f) - \mu(c_r l_r^2 + c_f l_f^2)]}{Jv(d_f + d_r)}$$

$$a_{44} = -\frac{\mu(c_r l_r - c_f l_f) + d_f \mu(c_f + c_r)}{Mv(d_f + d_r)} - \frac{d_r [d_f \mu(c_r l_r - c_f l_f) + \mu(c_r l_r^2 + c_f l_f^2)]}{Jv(d_f + d_r)}$$

$$b_{21} = \mu c_f \left(\frac{1}{M} + \frac{d_f l_f}{J} \right) \qquad b_{41} = \mu c_f \left(\frac{1}{M} - \frac{d_r l_r}{J} \right)$$

$$b_{22} = -v^2 \qquad b_{32} = v(d_f + d_r)$$

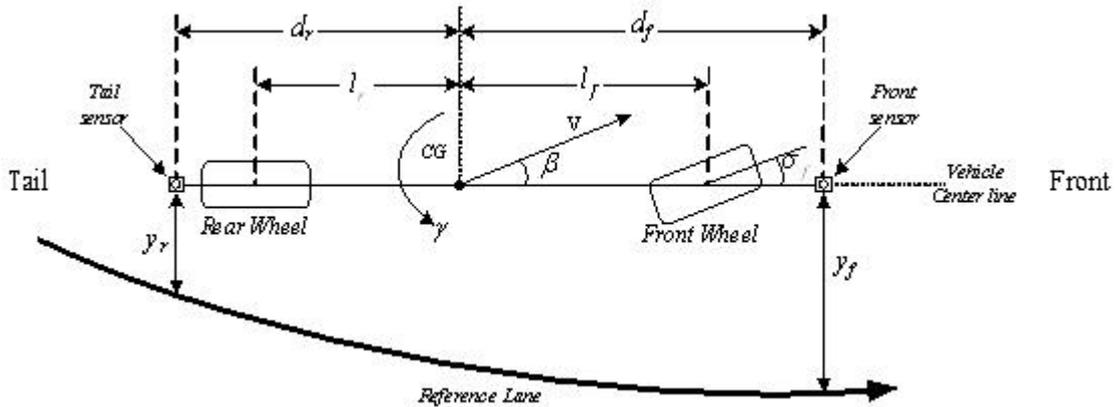


Figure 3-15: Modified bicycle model.

The lateral model of the scaled prototype vehicle presented in this chapter has been identified by Cai (2003). The identification is achieved by assuming the vehicle

model behavior similar to bicycle model. To collect data for identification, close loop approach is selected. The lateral control loop is close by a proportional controller. The steering angle and lateral position are recorded with respect to a periodic square wave reference input for system identification.

Grey model method was then used to define the lateral model structure which reduces to unknown parameters to 8 (a_{21} , a_{22} , a_{24} , a_{41} , a_{42} , a_{44} , b_1 and b_2). Finally the unknown parameters are identified by Prediction Error Method using Matlab System identification toolbox.

Longitudinal velocity (ms^{-1})	State-space model (A)				State-space model (B)	Poles	Zeros
1.3	0	1	0	0	0	0.00	-9.67
	251.64	-73.24	-251.64	44.96	63.77	0.00	$\pm 13.13i$
	0	0	0	1	0	-26.57	20.88
	239.56	-2.89	-239.56	-24.04	-6.67	-70.72	-121.80
1.1	0	1	0	0	0	0.00	-14.93
	251.64	-96.35	-251.64	54.85	63.77	0.00	$\pm 6.56i$
	0	0	0	1	0	-39.13	16.87
	239.56	-3.92	-239.56	-35.59	-6.67	-92.81	-150.69
0.9	0	1	0	0	0	0.00	26.77
	251.64	-92.97	-251.64	43.55	63.77	0.00	-9.93
	0	0	0	1	0	-46.42	15.51
	239.56	-5.79	-239.56	-41.26	-6.67	-87.81	-163.91
0.7	0	1	0	0	0	0.00	-48.29
	251.64	-130.13	-251.64	61.70	63.77	0.00	-5.51
	0	0	0	1	0	-64.67	13.35
	239.56	-4.90	-239.56	-60.25	-6.67	-125.67	-190.41
0.5	0	1	0	0	0	0.00	-67.34
	251.64	-193.00	-251.64	105.58	63.77	0.00	-3.95
	0	0	0	1	0	-83.08	11.91
	239.56	-0.897	-239.56	-82.33	-6.67	-192.25	-213.49

Table 3-2: Vehicle model parameters.

From (3.3), it can be seen that some parameters are speed dependent therefore different sets of data are recorded under different speed. The identified parameters under different speed by Cai (2003) are listed in Table 3-2.

From parameters listed in Table 3-2 and (3.3), the vehicle lateral model of the prototype vehicle can be generalized as shown in (3.4). The generalized model provides a platform for controller development and design on the prototype autonomous vehicle.

$$\frac{d}{dt} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 251.64 & \frac{-90}{v} & -251.64 & \frac{50}{v} \\ 0 & 0 & 0 & 1 \\ 239.56 & \frac{-1.92}{v} & -239.56 & \frac{-40}{v} \end{bmatrix} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 63.77 & -v^2 \\ 0 & 0.2v \\ -6.67 & -v^2 \end{bmatrix} \begin{bmatrix} \delta_f \\ \rho_{ref} \end{bmatrix} \quad (3.4)$$

where v is the longitudinal speed of vehicle, with the unit of ms^{-1} .

3.4. Conclusion

This chapter presents the design and construction of a scaled prototype autonomous vehicle. The use of scaled vehicle provides a platform for controller testing which is safer than actual implementation on a full size vehicle. The construction of the vehicle is discussed in detail including hardware configuration, sensor selection, sensor calibrations, and signal generation. The software architecture presented ensures the feasibility of real time application.

Chapter 4. A Novel Fused Neural Network Controller

4.1. Introduction

It was noted in Chapter 2 that many researchers have advocated the design of hybrid controllers. These controllers exploit one or more methodologies and encapsulate their merits into a coherent design in order to meet the stringent requirements of complex control systems. However, such attempts usually lead to very complex controller design. Hence, the design of a simple and functional hybrid controller is indeed a challenge. The proposed algorithm in this Chapter is an attempt to address this challenge.

Here, we report a simple fused neural network controller which is optimized using Genetic algorithms. The controller can be applied to a class of systems that require two input parameters for their stabilization. Consider the following three systems: an inverted pendulum, a ball-and-beam apparatus, and a vehicle lateral control problem. These systems share common dynamics and their control systems should consider two variables of displacement and the angle. The proposed controller architecture is designed according to task decomposition and although was designed for vehicle lateral control, it is well suited to systems with similar dynamics such as inverted pendulum and the ball-beam balancing control systems. The proposed network contains two sub-networks to tackle different tasks. The sub-networks outputs are fused together by a simple summation. The controller is optimized by Genetic algorithms for improved performance. Due to the simple structure of the controller, the complexity and search space of the parameters for optimization are also

reduced.

Researches have been conducted on the use of neural network controller for stabilizing classical benchmark control problem in last two decades. The inverted pendulum control problem is widely used for testing neural network control algorithms. Anderson (1989) and Jameson (1990) presented a neural network controller training with reinforcement learning technique for stabilizing an inverted pendulum. Dominic *et al.* (1991) extended the work of Anderson (1989) to genetic reinforcement learning for neural network to controller inverted pendulum. Dadios and Williams (1998) addressed the problem of inverted pendulum with Multilayer neural network and trained with backpropagation.

For the ball-and-beam system, Wang et al. (2004) tackled the problem by neural network controller and using genetic algorithm to search for optimum weight setting in offline. Eaton *et al.* (2000) used neural network controller to solve the “fuzzy ball-and-beam problem” suggested by Zadeh (1996). The “fuzzy ball-and-beam problem” is much more challenging than conventional ball-and-beam system. Since the beam is covered by a special sticky material so that the friction on the beam is not uniform.

The organization of the chapter is as follows: Section 4.2 describes the proposed neural network controller structure and the connection weights training method. The dynamics of the three systems that can be controlled by decomposition is explained in Section 4.3. Section 4.4 presents the performance and robustness analysis of the proposed controller for different benchmark problems. Section 4.5 discusses the results of the simulations conducted. Finally, Section 4.6 concludes this chapter.

4.2. Fused neural network design based on task decomposition

The concept of decomposition is an attractive tool to tackle systems that require two input parameters –say an angle and a displacement as control variables. The control problems of vehicle lateral control, inverted pendulum, and ball-and-beam are among systems that have the above characteristics. For vehicle lateral control, the controller has to keep the vehicle position within the lane while maintaining the vehicle orientation parallel to a reference lane. The two benchmark systems, namely inverted pendulum and ball-and-beam, require maintaining a cart or a ball to a desired position while keeping the pole or beam angle at zero degrees. The above systems have a common dynamics phenomenon that while maintaining the displacement, the angle change simultaneously.

Neural network designed by task decomposition would reduce the number of connection weights and hence reduce the network complexity. In this section, the neural network controller simplification process will be described for those systems have aforementioned common dynamics characteristic. The task decomposition techniques start with identifying control objectives from the control system and assign it into individual subtasks.

When the subtasks are identified, obtain the intuitive control laws for the subtasks based on the understanding on the control system. From the intuitive control law, control variables related to the control subtasks can be obtained. For the network architecture, independent sub-networks should be created for individual subtasks and using the related control variable as network inputs. Finally, the outputs of the independent sub-network are fused together as the controller output.

The neural network simplification process based on task decomposition can be summarized in the following steps:

1. Decomposing control objective into subtasks.

e.g. Inverted pendulum:

Subtask 1 - Maintain pendulum (θ) up right

Subtask 2 - Control cart position (d) .

2. Setting up intuitive control law for the subtasks.

e.g. Inverted pendulum:

Control law 1 - Increase/decrease forces on the system when θ deviates from zero.

Control law 2 - Increase/decrease forces on the system when d deviates from the set-point.

3. Identifying control variables for the subtasks from the intuitive control law obtained in step 2.

e.g. Inverted pendulum:

Since the action of the control law 1 depends only on pendulum angle, control variables required in the first subtask are pendulum angle and its rate of change. For the second subtask, since the action of the control law 2 depends only on the cart position, control variables required in the first subtask are pendulum angle and its rate.

4. Creating independent sub-networks for the subtasks and using the corresponding control variables obtained in step 3 as sub-network inputs.
5. Fusing the sub-networks output together as the controller output.

In accordance with the decomposition technique described above, the unified neural network controller for the inverted pendulum, ball-and-beam system, and

vehicle lateral control is divided into two independent sub-networks. Figure 4-1 shows the network structure. The network has four inputs ($\theta, \dot{\theta}, d, \dot{d}$) and one control output (u) to the system. The parameters θ and d are the angle and the displacement respectively (The dot over a variable means the rate of change of that variable). There are two main sub-networks in the network and each sub-network targets on one sub-task of the systems. The first sub-network deals with the angle and aims to maintain it at zero. The second network deals with the displacement of the control system. Finally the sub-networks output are fused together by a simple summation to form the final controller output.

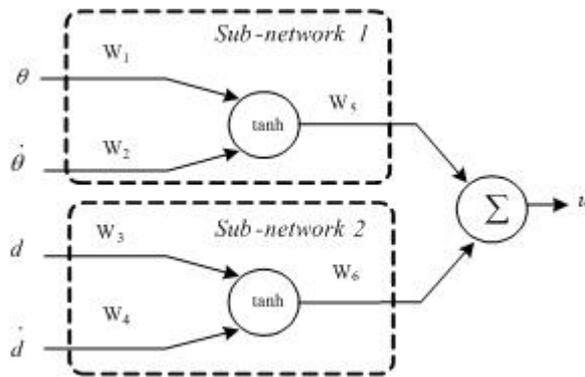


Figure 4-1: Neural network controller.

The proposed three layers network as shown in Figure 4-1 has two inputs, one output and two neurons in the hidden layer. The total number of connection weight is six. Referring to the multilayer feedforward neural network structure described in Chapter 2, the number of connection weights (W) for a fully connected network with same structure as the proposed network is ten. In this case, 40% of connection weights are reduced from a fully connected network.

The connection weight reduction is achieved by dividing variables into groups and set into sub-networks. It can be seen that each sub-network are independent

therefore the inputs of sub-network one are not connected to sub-network two and vice versa. Since the connection weights are reduced simultaneously. Equations (4.1-4.3) show the characteristic of the network.

$$Z_1 = \tanh(\theta \cdot W_1 + \dot{\theta} \cdot W_2) \quad (4.1)$$

$$Z_2 = \tanh(d \cdot W_3 + \dot{d} \cdot W_4) \quad (4.2)$$

$$u = Z_1 \cdot W_5 + Z_2 \cdot W_6 \quad (4.3)$$

where Z_1 and Z_2 are outputs from sub-network 1 and sub-network 2 respectively and W_{1-6} are the connection weights shown in Figure 4-1. The next section shows the network optimization procedure with genetic algorithms.

4.3. Neural network Optimization via Genetic algorithms

It was explained in Chapter 2 that a neural network controller could be optimized by genetic algorithm (GA). The optimization procedure follows the typical GA optimization cycle as depicted in Chapter 2. The reader may refer to Chapter 2 for the general description of GA.

4.2.1 Coding of connection weights into chromosome

The first step of GA is to encode the optimization variables into chromosomes. This incorporates two processes of encoding mechanism and representation method. The encoding mechanism describes the allocation method of the variables in the chromosome. The representation method facilitates the conversion details from optimization variables into genes. In the case of neural network optimization, the network can be optimized by GA in several ways such as network structure, activation, and connection weights. The choice of encoding mechanism and representation

method depends on the nature of optimization and it is important in designing a GA optimization process.

The network structure and activation function of the proposed neural network controller are both fixed; therefore, only connection weights are encoded into chromosome. Figure 4-2 shows the coding mechanism of connection weight into chromosome. In the figure, the chromosome length is six and each gene carries a connection weight of the network. Since real value does not lose precision due to binary quantization, the six connection weights are coded into six real value genes instead of conventional binary coding.

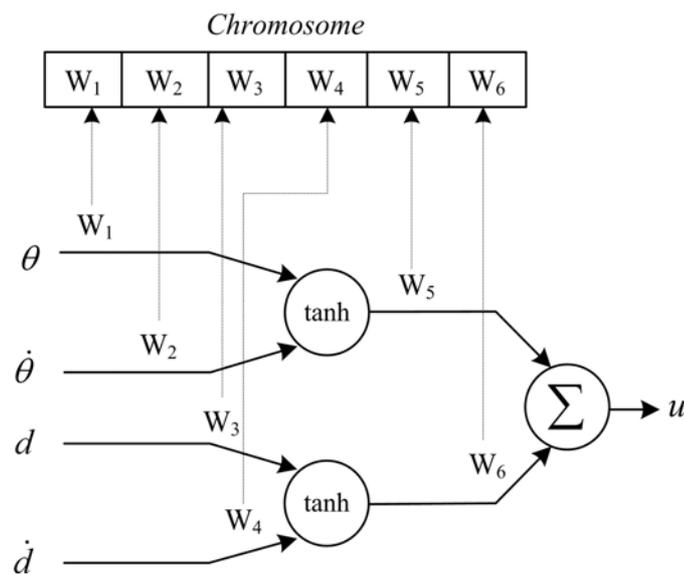


Figure 4-2: Coding of network weight into chromosome.

4.2.2 Fitness function

Fitness function determines the optimization objective. Since the control variables of the controlled system are displacement and angle, the fitness function is parameterized by these two parameters. The performance index used is the sum of the

integral time absolute error (*ITAE*) of displacement and orientation error, and is defined as shown in (4.4):

$$fitness = \frac{1}{\left[1 + \sum (Abs[(setpoint - x_2) + (x_1)] \times t)\right]} \quad (4.4)$$

where x_2 and x_1 are displacement and angle respectively.

4.2.3 Computation procedure for the neural network optimization

The following steps summarize the procedure of the GA optimization process for the neural network controller.

1. Fix the search range of individual genes in the chromosomes.
2. Initialize 50 chromosomes randomly to form an initial population
3. Decode the chromosomes into neural network controller and test its performance in simulation. Then, calculate the fitness value from (4.4) and assign chromosome fitness
4. Select parent chromosomes from the population to reproduce children chromosomes. The selection is based on Roulette wheel selection.
5. Reproduce children chromosomes by either crossover or mutation. One point crossover is used to recombines the genetic material in two parent chromosomes to make two children. Mutation is used to introduce innovative materials to the population.
6. Decode children chromosome into neural network controller to evaluate fitness
7. Select 50 best fit chromosomes from children and parent chromosome to form new generation.
8. Repeat Steps 3 to 7 until either maximum number of generations is reached or desired fitness value is evaluated.

4.3 Application to benchmark problems

In order to evaluate the proposed controller, it is applied to two benchmark problems of inverted pendulum and the ball-beam systems. These two systems have similar physical characteristics to lateral control problem of an autonomous vehicle. These two benchmark problems are generally used to assess the performance of newly developed controllers in literature. Chan *et al.* (2002), Lee and Wong (1996), and Wang (1998) implemented a newly designed fuzzy controller on inverted pendulum in simulation for performance evaluation. Anderson (1989) used an inverted pendulum to verify the proposed neural network controller. Hao *et al.* (1993), and Jiang *et al.* (1995) using ball-and-beam system for neural network controller performance analysis.

These systems have two main variables to describe the states of the system, an angle and a displacement. At steady state the angle is maintained around zero to stabilize the system. However, while adjusting the displacement, the angle is changed simultaneously. Figure 4-3 shows the proposed unified controller structure for the mentioned systems which can be decomposed into angular and displacement control. The controller is a single layer feed-forward neural network with four inputs and a single output. The vehicle lateral control problem consisted of the angle between the vehicle center line and the reference lane and the lateral displacement. Inverted pendulum control problem can be represented as the cart displacement to the origin and the rod angle. The ball-and-beam system can be represented by the ball position and the beam angle. The representation of the lateral control problem, the inverted pendulum and the ball-and-beam system into different sub-controllers simplify the overall controller structure and imply shorter search space and less parameter to be

optimized.

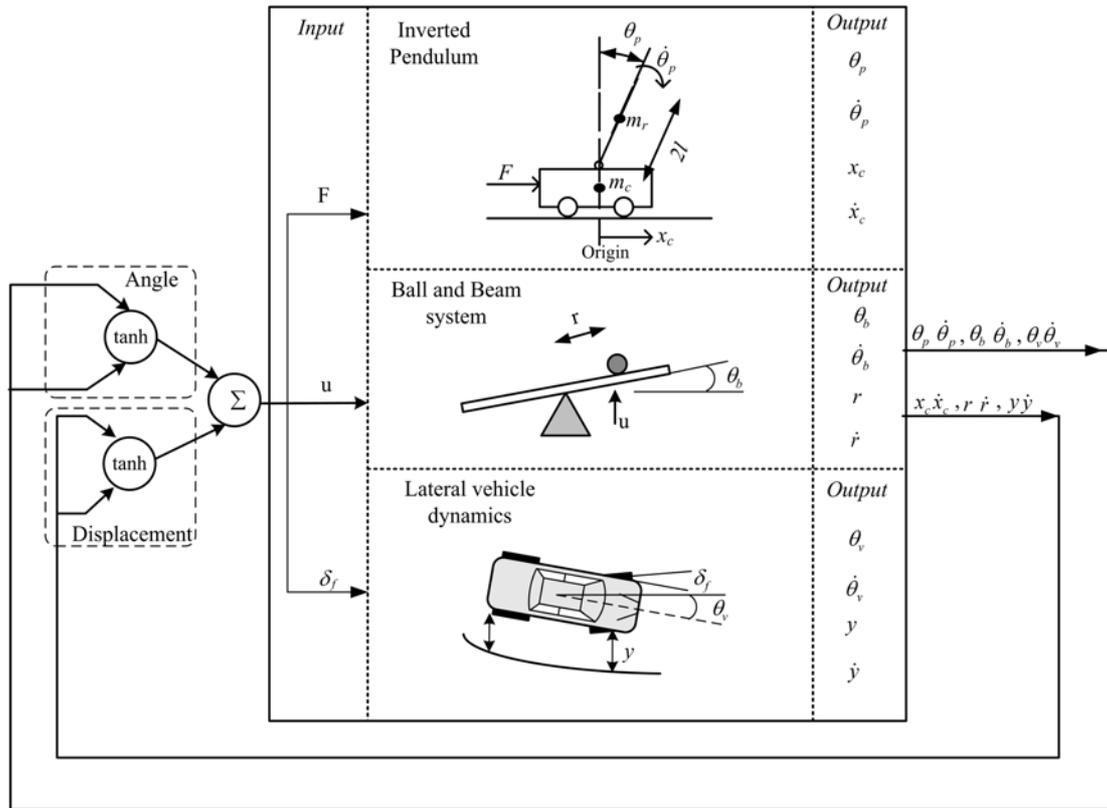


Figure 4-3 : The unified controller.

In this section, full state feedback linear controller (LC) will be designed and compared with the NN controller in terms of robustness. The selection of the system close-loop ($s = p_i, i=1,2,3,4$) bases on the NN controller performance so that the LC has similar and comparable performance with the NN controller for robustness comparison. Then the robustness of controllers will be evaluated by varying model parameters. The following subsections present the system models for NN controller optimization and LC controller design.

4.3.1 Inverted Pendulum:

An inverted pendulum mounted on a cart is shown in Figure 4-4. The inverted pendulum is unstable in that it may fall in any direction unless a suitable control force is applied. The objective of the inverted pendulum is to maintain the cart at the origin

from any initial position while maintaining the rod at the vertical position ($\theta = 0$). Therefore controller tackles the cart displacement and rod angular displacement simultaneously.

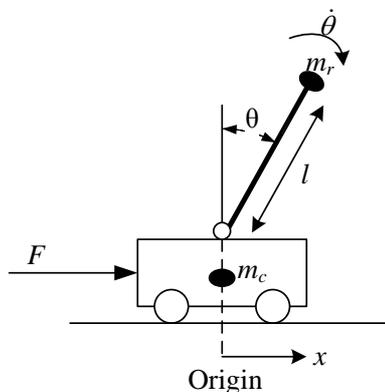


Figure 4-4: Inverted pendulum.

The linearized equations of inverted pendulum are shown in (4.5) and (4.6) Ogata (1990). The state space equation for simulation, NN controller optimization, and LC controller design is shown in (4.7).

$$\ddot{\theta}_l = \frac{(m_c + m_r)g\theta_l - F}{m_c l} \quad (4.5)$$

$$\ddot{x}_l = \frac{F - mg\theta_l}{m_c} \quad (4.6)$$

$$\dot{X}_l = A_l X_l + B_l F$$

$$\begin{bmatrix} \dot{\theta}_l \\ \ddot{\theta}_l \\ \dot{x}_l \\ \ddot{x}_l \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{m_c + m_r}{m_c l} g & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{m_r}{m_c} g & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_l \\ \dot{\theta}_l \\ x_l \\ \dot{x}_l \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{m_c l} \\ 0 \\ \frac{1}{m_c} \end{bmatrix} F_l \quad (4.7)$$

where $X_l = [\theta_l \quad \dot{\theta}_l \quad x_l \quad \dot{x}_l]^T$ is the system state vector, m_c and m_r are mass of the cart and the rod respectively. l is the length of the rod. x_l is the horizontal displacement of the cart with reference to the origin. θ_l is the clockwise angular displacement of the rod. F_l is the control input to the system.

4.3.2 Ball-and-beam system:

The objective of the ball-and-beam system is to maintain the ball at a desired position ($r_B = r_d$) by controlling and stabilizing the beam angle to zero ($\theta_B=0$). Thus controller handles the ball displacement and beam angular displacement at the same time. Equation (4.8) represents the nonlinear dynamics of the ball-and-beam system.

(4.8) is used for NN controller optimization.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_2 \\ \alpha(x_1 x_4^2 - g \sin x_3) \\ x_4 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u_B \quad (4.8)$$

where $[x_1, x_2, x_3, x_4] = [r_B, \dot{r}_B, \theta_B, \dot{\theta}_B]$. r_B and θ_B are the displacement of the ball from the center and the angular displace of the beam from the horizon respectively.

Variables α and β are physical parameter and gravitational acceleration respectively.

Linearized ball-and-beam system state space equation around the equilibrium point

($r_B = 0$) shown in equation (4.9) (Wang (1997)) is used for LC feedback gain matrix

determination.

$$\begin{aligned} \dot{X}_b &= A_b X_b + B_b u \\ \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\alpha g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u \end{aligned} \quad (4.9)$$

4.3.3 Vehicle lateral dynamic

The Bicycle model described in Chapter 3 will be used for both controller design and simulation. The bicycle model equation is shown in (4.10). The detail

description on the model parameters are shown in Chapter 3. The state of the bicycle

model is manipulated to match with the NN controller input signals θ , $\dot{\theta}$, d , and \dot{d} .

The equation for manipulation is shown in (4.11).

$$\begin{bmatrix} \dot{y}_f \\ \ddot{y}_f \\ \dot{y}_r \\ \ddot{y}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a_{21} & a_{22} & -a_{21} & a_{24} \\ 0 & 0 & 0 & 1 \\ a_{41} & a_{42} & -a_{41} & a_{44} \end{bmatrix} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ 0 & b_{32} \\ b_{41} & b_{22} \end{bmatrix} \begin{bmatrix} \delta_f \\ \rho_{ref} \end{bmatrix} \quad (4.10)$$

$$\begin{bmatrix} \theta \\ \dot{\theta} \\ d \\ \dot{d} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0.5 & 0 & 0.5 & 0 \\ 0 & 0.5 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} \quad (4.11)$$

4.4 Simulation Results

In this Section, simulation studies of lateral control problem, inverted pendulum and ball and beam system are reported. The parameters setting of GA are listed in Table 4-1. There are 50 chromosomes in each generation. Maximum number of generation is 400. A new generation is reproduced either by mutation or crossover with rates of 0.3 and 0.7 respectively. Fitness function for fitness evaluation is based on the integral time of absolute error.

Population size	50
Maximum generation	400
Crossover rate	0.7
Mutation rate	0.3
Fitness function	1/(1+ITAE)

Table 4-1: Genetic algorithm parameters.

Connection weight search space for the inverted pendulum of W_1 , W_2 , W_3 , and W_4 are [-20, 20] and [-200, 200] for W_5 and W_6 . For the ball-and-beam system, the connection weight search space of W_1 , W_2 , W_3 and W_4 are [-100, 100] and [-10, 10] for W_5 and W_6 . For lane keeping problem, the connection weight search space of W_1 ,

W_2 , W_3 , and W_4 are $[-1, 1]$ and $[-30, 30]$ for W_5 and W_6 . The assignment of the search space is combined with knowledge of the controlled system dynamics and the neural network structure.

Inverted pendulum	m_c	2.0kg	l	0.5m
	m_r	0.1kg	g	9.8ms ⁻²
Ball and beam system	α	0.7143	g	9.81
Bicycle model	a_{21}	251.64	a_{44}	-60.25
	a_{22}	-130.13	b_{21}	63.77
	a_{24}	61.70	b_{22}	-4.9
	a_{41}	239.56	b_{32}	14
	a_{42}	-4.9	b_{41}	-6.67

Table 4-2: Systems parameters.

The parameters and coefficients in the bicycle model and physical model for the inverted pendulum and the ball-and-beam system are shown in Table 4-2. Table 4-3 shows the optimized connection weights after optimization.

Inverted pendulum	W_1	-12.7475	W_4	-6.7588
	W_2	-3.21193	W_5	-112.502
	W_3	8.245037	W_6	-99.423
Ball and Beam system	W_1	16.4417	W_4	2.1535
	W_2	-9.7559	W_5	-14.112
	W_3	17.4231	W_6	15.6517
Lane keeping problem	W_1	0.9846	W_4	0.2148
	W_2	0.1342	W_5	-16.1873
	W_3	0.1303	W_6	12.4779

Table 4-3: Optimized connection weights.

After the optimization of NN controllers, LC controllers for the three systems are designed. Since the rank of the controllability matrixes of the three systems mentioned (i.e. (4.7), (4.9), and (4.10)) are 4, the systems are state controllable and

arbitrarily pole placement are possible. The feedback gain matrix determination is based on ackermann's formula. The feedback gain matrixes K_I , K_b , and K_v for inverted pendulum, ball-and-beam, and vehicle lateral control respectively and the associated close-loop system poles are summarized in Table 4-4.

Model	Close-loop poles		K_1	K_2	K_3	K_4
Inverted Pendulum	-7.45,-1.65 -10, -10	K_I	-377.430	-87.992	-125.11	-117.785
Ball and Beam	-3.2±j2.4, -10, -10	K_b	-226.308	-135.785	244	26.4
Bicycle Model	-1.2±j0.9, -10, -10	K_v	11.1067	-2.6691	-11.093	-0.334

Table 4-4: Linear controller feedback gain matrix.

4.4.1 Inverted Pendulum

The controller testing condition is follow. The cart is initially located at 0.2m away from the origin with vertically positioned pole therefore the initial state of the inverted pendulum $[\theta, x]$ is $[0, 0.2]$. Figure 4-5 shows the performance of both NN and LC controllers. Figure 4-5a shows displacement of the cart controlled by NN controller (solid line) and LC controller (dash line). The LC control signal is generated by $F_I = -K_I X_I$. Where X_I is the state vector described in equation (4.7).

Since the design of the LC controller is based on the NN controller, they have similar performance in terms setting time and overshoot. The solid and dashed line is the cart displacement controlled by NN and LC controller respectively. The results show that both controllers maneuver the cart to the origin after 3.5s.

Figure 4-5b shows the angle of the pole. The solid and dashed line is the pole angle controlled by NN and LC controller respectively. The pole angles controlled by

both controllers are stabilized at 3.2s. However, the pole swing controlled by NN controller is larger than the swing controlled LC controller.

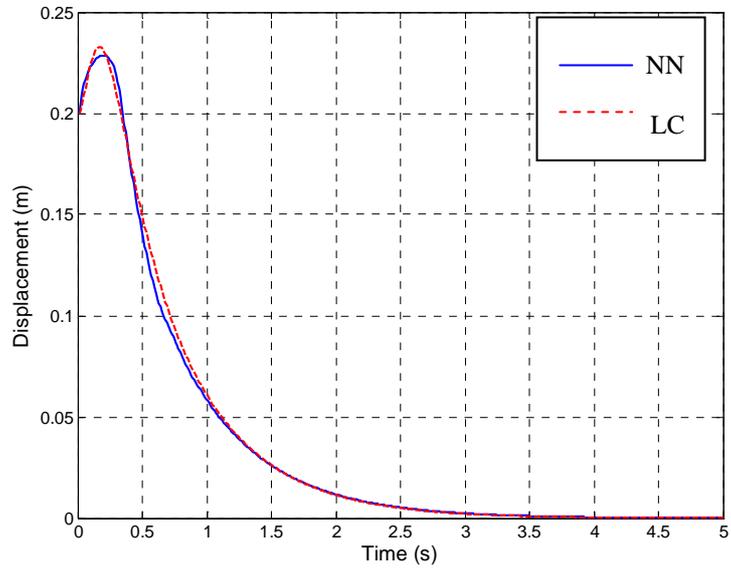


Figure 4-5 a (cart displacement.)

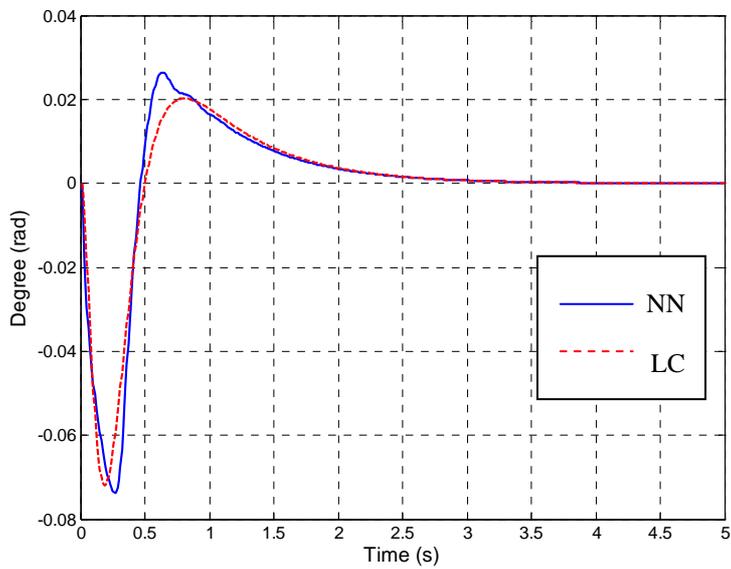


Figure 4-5b (pole angle.)

Figure 4-5: Inverted pendulum simulation result.

4.4.2 Ball-and-Beam Balance System

The initial state of the ball-and-beam system $[\theta_B, r_B]$ is $[0, 0.2]$ for the simulation of NN and LC. The control signal of the LC is generated by $u_B = -K_b X_b$ where X_b is the state vector of the Ball-and-Beam system described in equation (4.9). Figure 4-6 shows the performance of both controllers on Ball-and-Beam system.

Figure 4-6a shows the ball displacement controlled by NN controller (solid line) and LC (dash line). The results show that both controllers stabilized the ball to the origin (i.e. 0 m) after 2s. Figure 4-6b shows the beam angle. The solid and dashed line is the beam angle controlled by NN controller and LC respectively. The beam angle controlled by both NN controller and LC are stabilized at 2s.

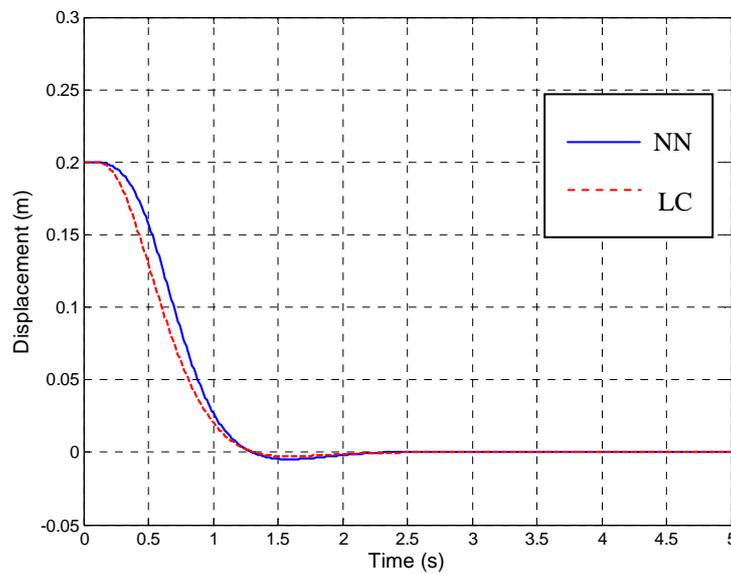


Figure 4-6 a (the ball displacement)

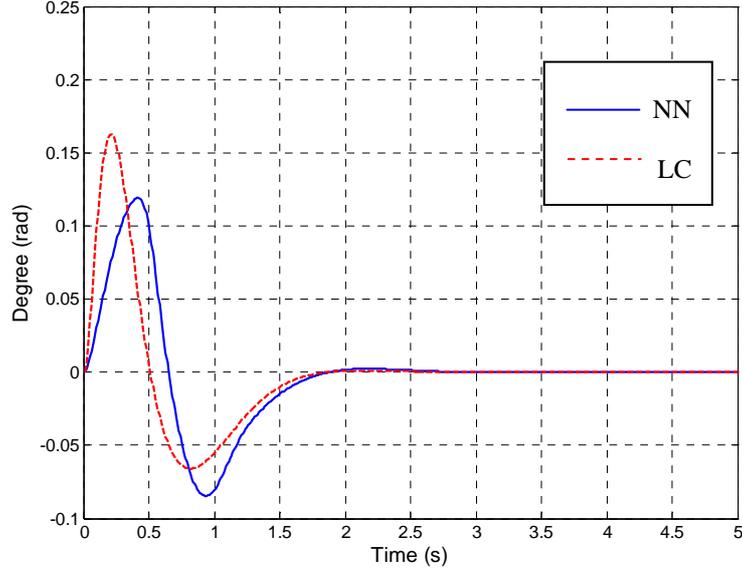


Figure 4-6b (the beam angle)

Figure 4-6: Ball-and-beam system simulation result.

4.4.3 Lateral control

Figure 4-7 shows the lateral control performance in a straight road scenario. The road curvature (ρ_{ref}) is set at zero through out the simulation such that to simulate a straight road condition. The initial state of the vehicle was located at 0 m away from the and parallel to the reference lane i.e. zero angle. The distance set-point from the reference lane was 0.2 m. In order to track with lateral set-point, the LC generate control signal by equation (4.12).

$$\delta_f = [K_{v1} \quad -K_{v2} \quad K_{v3} \quad -K_{v4}] [0.2 - y_f \quad \dot{y}_f \quad 0.2 - y_r \quad \dot{y}_r]^T \quad (4.12)$$

where 0.2 is the lateral set-point.

In Figure 4-7a, the vehicle lateral position controlled by both controllers reaches to set-point after time 2.5s and stabilized at time 5s. In Figure 4-7b, the angle between

vehicle center line and the road is calculated by the different between front and tail displacement. The maximum angle is 0.09 rad at time 0.5 s and it resumes to 0 rad at time 6 s .

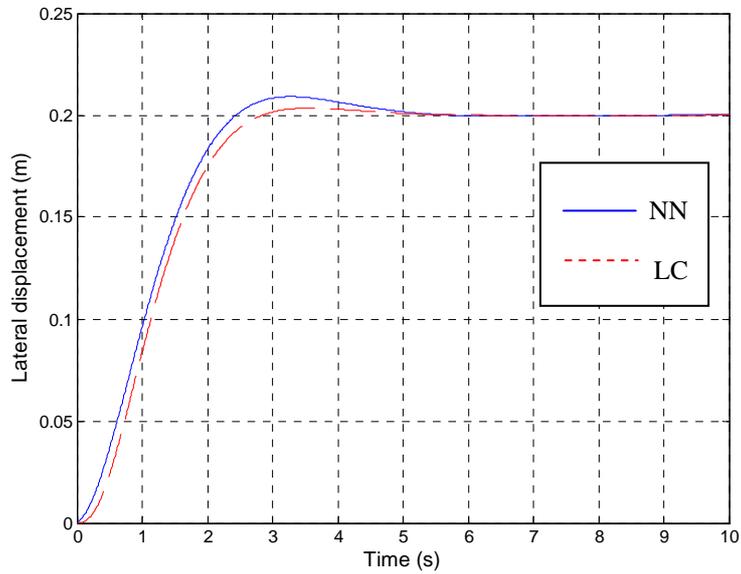


Figure 4-7a (Vehicle lateral displacement).

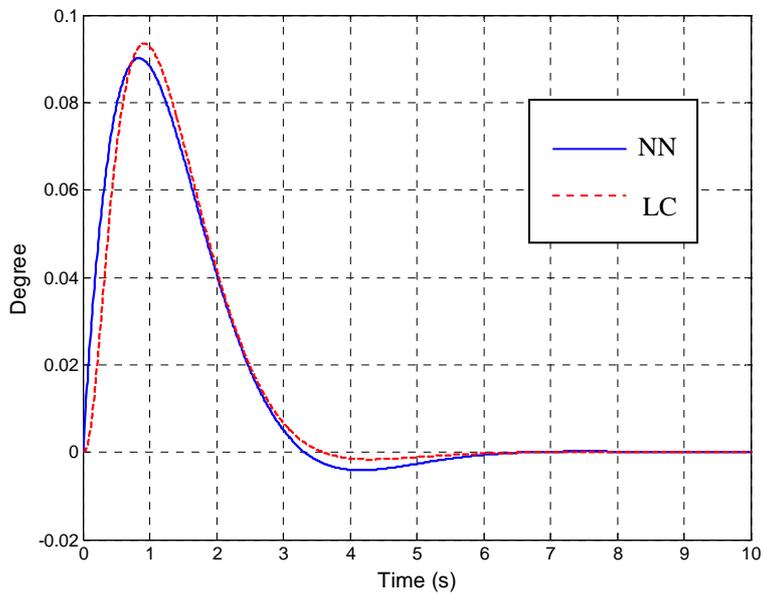


Figure 4-7b (Vehicle orientation)

Figure 4-7: Lateral control simulation result.

4.4.4 Controller robustness analysis

The robustness of the LC and the NN controller are compared subject to system parameters variations. In this section controllers performance under system parameters variation will be presented.

4.4.4.1 Inverted pendulum

There are 3 parameters in the model of inverted pendulum (4.7) namely mass of cart (m_c), mass of rod (m_r), and length of rod (l). The parameters are varied $\pm 40\%$ for controller robustness analysis. There are 6 simulation cases study for each controller and there is only one parameter is varied in each simulation. The 6 cases are shown in Table 4-5. The shaded cells in Table 4-5 indicate the parameter is varied from the nominal value.

Case	Parameters		
	m_c	m_r	l
1	1.2 kg	0.1 kg	0.5 m
2	2.8 kg	0.1 kg	0.5 m
3	2 kg	0.06 kg	0.5 m
4	2 kg	0.14 kg	0.5 m
5	2 kg	0.1 kg	0.3 m
6	2 kg	0.1 kg	0.7 m

Table 4-5: Inverted pendulum parameters variation.

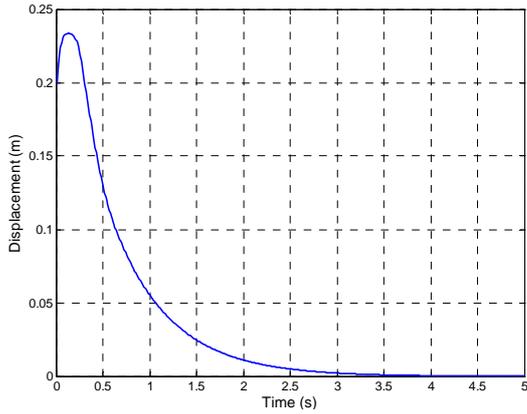


Figure 4-8a (Case 1: $m_c=1.2\text{kg}$)

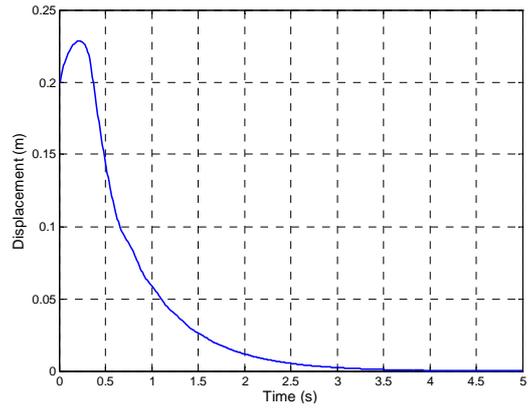


Figure 4-8b (Case 2: $m_c=2.8\text{kg}$)

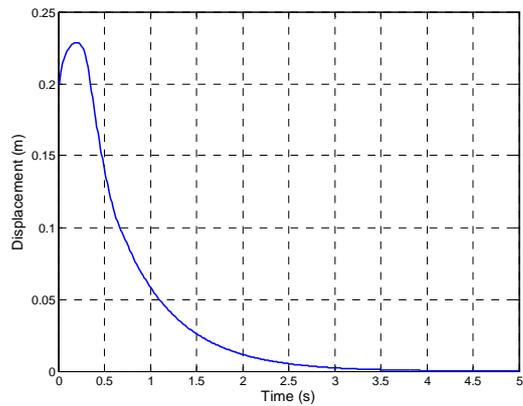


Figure 4-8c (Case 3: $m_r=0.06\text{kg}$)

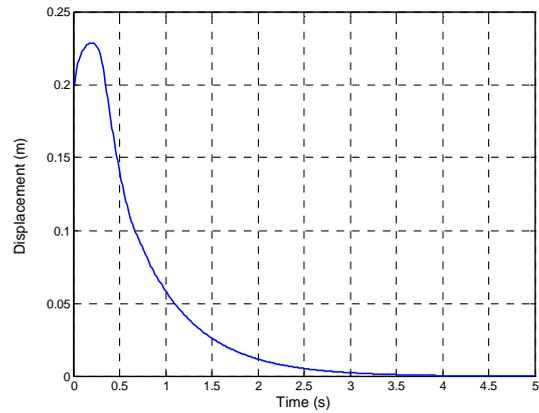


Figure 4-8d (Case 4: $m_r=0.14\text{kg}$)

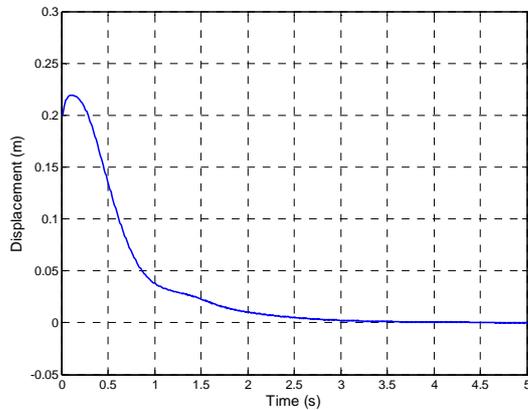


Figure 4-8e (Case 5: $l=0.3\text{m}$)

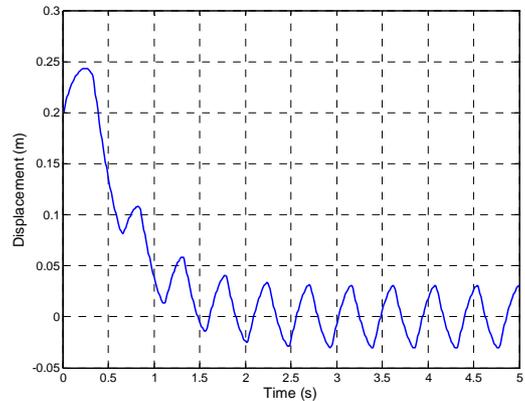


Figure 4-8f (Case 6: $l=0.7\text{m}$)

Figure 4-8: Robustness analysis on NN controller for inverted pendulum

Figure 4-8 presents the 6 cases of the NN controller simulation results. Figure 4-8a to Figure 4-8f show the results on cart displacement of Case 1 to 6 respectively. The results presented in Figure 4-8a to Figure 4-8d are the simulation results under the change of mass and the results show that the variation in the mass of cart (m_c) and

rod (m_r) does not affect the performance of the NN controller. In Case 6 (Figure 4-8f), the controller performance is deteriorated due to the enlargement in the length of the rod (l). The cart position is maintained at the origin with $\pm 0.025m$ deviation and 2Hz oscillation. Based on the simulation result, the proposed NN controller is robust to $\pm 40\%$ parameters variations and it is able to stabilize the inverted pendulum in all of the 6 cases.

For the LC, Figure 4-9 presents the 6 cases of the LC simulation results. Figure 4-9a to Figure 4-9f show the results of Case 1 to 6 respectively. The results also show that the variation in m_c and m_r does not affect the performance of the LC. In Case 6 (Figure 4-9f), the controller fails to control the system and the system is oscillating with monotonic increasing. According to the simulation results, the LC is robust to the $\pm 40\%$ variation in m_r and m_c . However, it is not robust to the variation in the length of the rod (l).

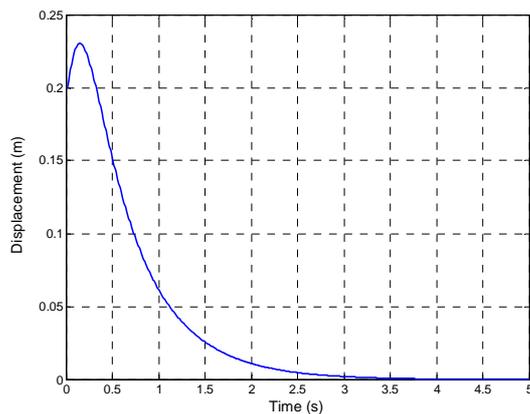


Figure 4-9a ($m_c=1.2kg$)

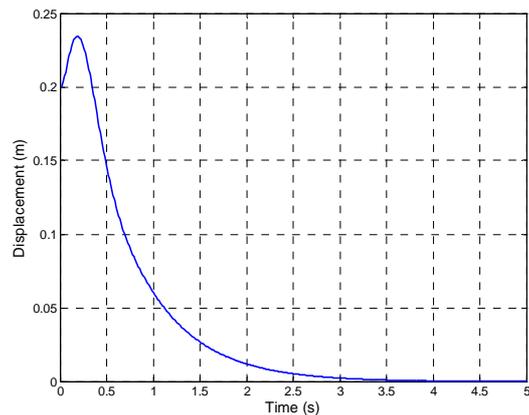


Figure 4-9b ($m_c=2.8kg$)

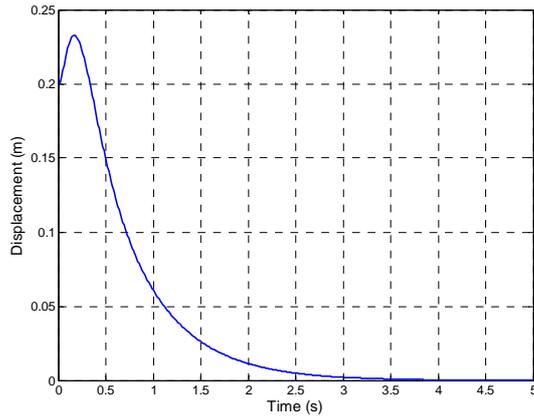


Figure 4-9c ($m_r=0.06$ kg)

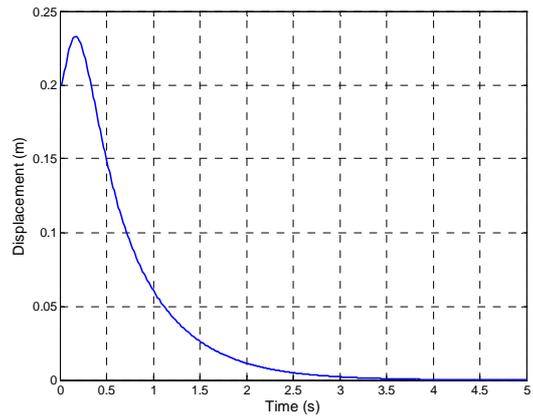


Figure 4-9d ($m_r=0.14$ kg)

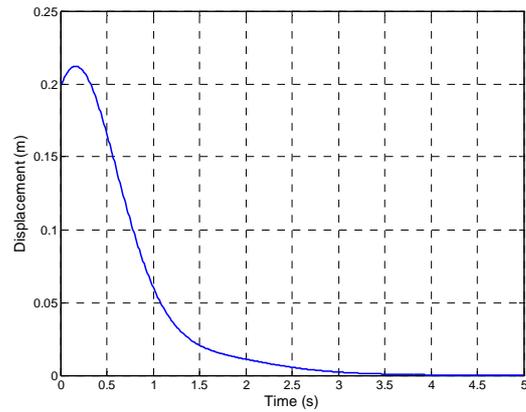


Figure 4-9e ($l=0.3m$)

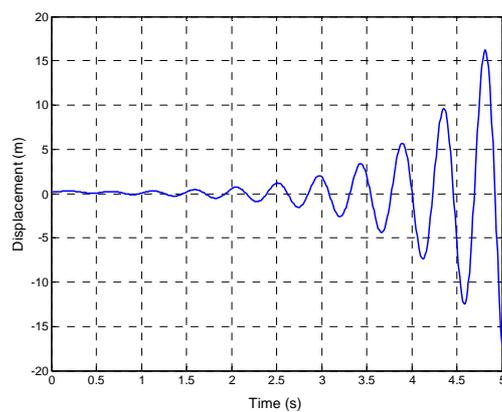


Figure 4-9f ($l=0.7m$)

Figure 4-9: Robustness analysis on LC controller for inverted pendulum

4.4.4.2 Ball-and-Beam Balance System

From equation (4.8), the ball-and-beam system consists of 2 parameters α and g . Since g is gravitational acceleration, it is assumed as constant. Then the remaining parameter is varied $\pm 40\%$ for robustness evaluation. The value of α with -40% and $+40\%$ variations are 0.42853 and 1.002 respectively. The NN and LC controller performances of ball displacement subject to $\pm 40\%$ variation are shown in Figure 4-10 and Figure 4-11 respectively.

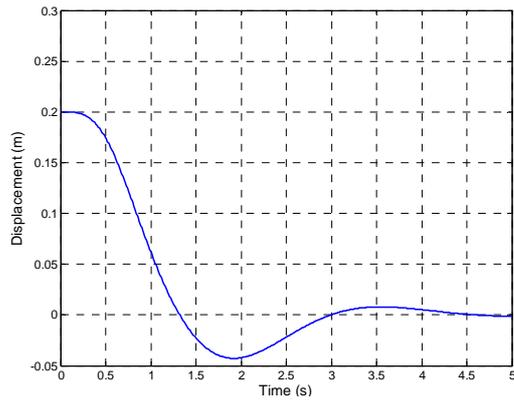


Figure 4-10a ($\alpha=0.42853$)

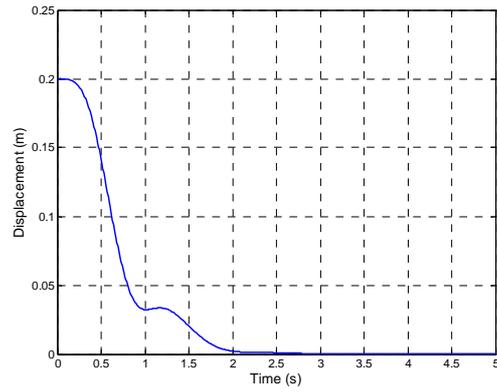


Figure 4-10b ($\alpha=1.002$)

Figure 4-10: Robustness analysis on NN controller for ball-and-beam

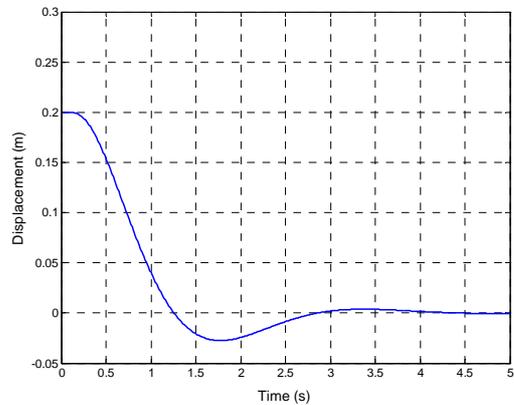


Figure 4-11a ($\alpha=0.42853$)

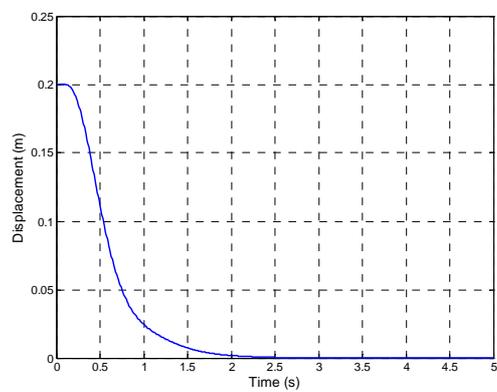


Figure 4-11b ($\alpha=1.002$)

Figure 4-11: Robustness analysis on LC controller for ball-and-beam

Figure 4-10a and Figure 4-10b present the NN controller performance with $\alpha=0.42853$ and $\alpha=1.002$ respectively. For LC, Figure 4-11a and Figure 4-11b show the LC performance with $\alpha=0.42853$ and $\alpha=1.002$ respectively. The simulation results suggest that both NN controller and LC are able to stabilize the ball at the origin and thus both controllers are robust to $\pm 40\%$ parameter variable on the ball-and-beam system.

4.4.4.3 *Vehicle lateral control*

The controller robustness for vehicle lateral control problem is examined by the Monte-Carlo evaluation method (Ray and Stengel (1993)). There are 8 parameters in the bicycle model (4.10) and single parameter is randomly selected and varied $\pm 20\%$ of uniform uncertainty in each evaluation. 1,000 Monte-Carlo evaluations are performed on the NN controller and LC. The results of Monte-Carlo evaluations on NN controller and LC are shown in Figure 4-12 and Figure 4-13 respectively.

Figure 4-12 shows the 1,000 step responses with $\pm 20\%$ randomly selected parameters uncertainty of the bicycle model. Envelopes (thicker lines) shown in figure are constructed based on the nominal step responses of the vehicle with NN controller. The step response is multiplied by factors 1.2 and 0.8 to define $\pm 20\%$ deviation limit respectively. From Figure 4-12, most of the 1,000 evaluations violate the deviation limit but all the 1,000 evaluations converge to the lateral set-point at 20cm . Thus the NN controller is robust to $\pm 20\%$ parameter variation on the bicycle model.

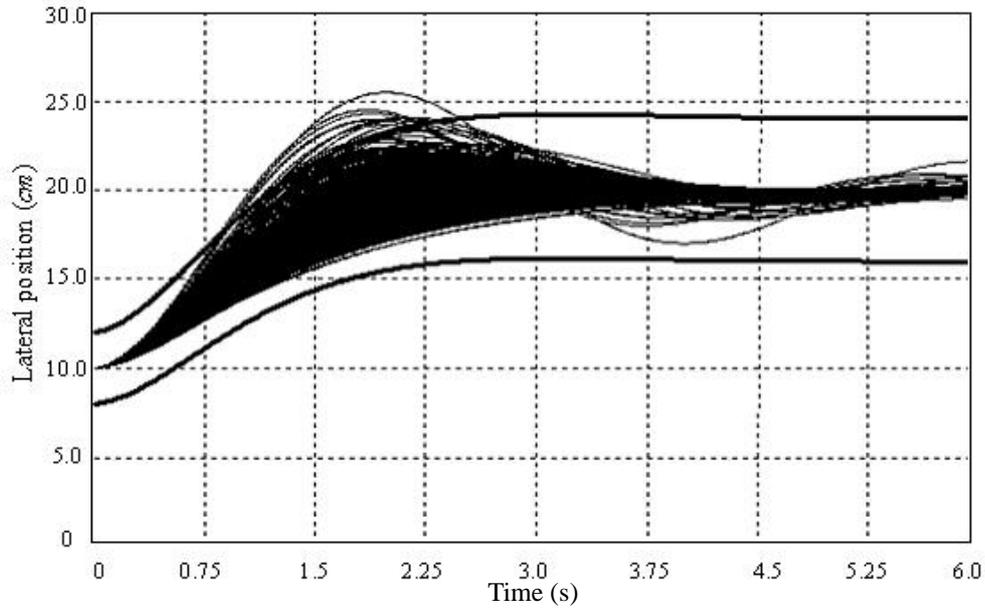


Figure 4-12: Monte-Carlo evaluations on NN controller for vehicle lateral control.

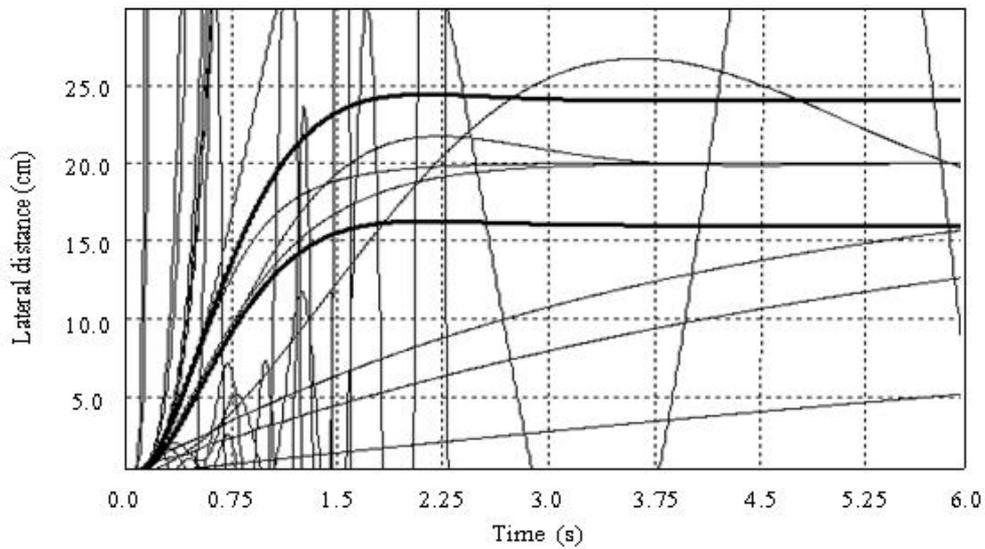


Figure 4-13: Monte-Carlo evaluations on LC controller for vehicle lateral control.

1,000 Monte-Carlo evaluations have been conducted on the LC. However, the LC performance is unstable in the Monte-Carlo evaluations and only a few evaluations converge to the lateral set-point. Therefore there are only 20 Monte-Carlo evaluations on the LC are presented in Figure 4-13. From the result of Monte-Carlo evaluation, it is concluded that the LC controller is not robust to $\pm 20\%$ parameter

variation on the bicycle model.

Simulation results on the robustness analysis of the NN and LC controller for inverted pendulum, ball-and-beam system, and vehicle lateral control are presented in this section. The results suggest that the robustness of the NN controller is comparable with the LC controller in the classical benchmark systems. For the vehicle lateral control, the robustness of the proposed NN controller is superior to the LC controller.

4.5 Discussion

In the above sections, linear controllers (LC) with full state feedback are designed for the inverted pendulum, ball-and-beam system, and vehicle lateral control. The linear controllers are then compared with the neural network (NN) controller under the variation of system parameters. In this section, the simulation results will be further discussed.

Inverted Pendulum

From the robustness analysis on both LC and NN controller for the inverted pendulum, both controllers are robust to variations of mass. From the model equation of inverted pendulum shown in (4.7), the m_c and m_r affect the motions of both control variables (i.e. both cart position x and rod angle θ) in matrix A_I . In the input matrix B_I , the m_c reduces the effective force delivered to both control variables.

The effect on the change of mass deduced from the model explains the robustness of both controller to the mass variation. Since the effects on the change of mass act on both control variables simultaneously with a similar scale, the controllers can compensate those effects by increasing forces.

For the change in the length of rod, the performances of controllers are deteriorated. From the equation (4.7), the change in l_r affects the motion on the rod angle only but not the cart position. In this case, controller cannot compensate the effect by applying strong force but generate control action with good coordination with the control variables. The results show that the NN controller is also robust to the change in the l_r . However, the LC controller cannot stabilize the system when the l_r is enlarged.

Ball-and-Beam balance system

The results show that both LC and NN controller are robust to the parameter variation. As the system models shown in equations (4.8) and (4.9) are linearized model with only one parameter that can be changed (α). Therefore the variation of the parameter only contributes to scaling up or scaling down of the system. Since there is no limitation on the controller output, the variation with linear scaling can be compensated by large control actions.

Vehicle lateral control

For the Monte-Carol analysis conducted in the vehicle lateral control, the NN controller shows much improved performance over the LC controller. Consider the bicycle model equation shown in (4.10), the change of a state variable (say \ddot{y}_r) is related to the sum of the all state variables, thus there are strong coupling between the state variables. Therefore the change in one state variable would inference the whole vehicle dynamics.

The NN controller shows better performance than LC which is due to the task

decomposition structure. Referring to Figure 4-1, individual task is handled by independent subnetwork. Each subnetworks outputs are towards the goal of their task. Thus the vehicle is stabilized by two independent controllers which enhance the robustness of the controller under parameters variations.

4.6 Conclusions

A novel and yet simple neural network controller structure utilizing task decomposition principle is proposed in this chapter. The controller can be employed to control systems that require two input parameters (an angle and a displacement) as their control variables. The neural network is optimized by genetic algorithms which enhance the flexibility for multi-objective optimization. The connection weight settings for inverted pendulum, ball-and-beam system, and vehicle lateral control are presented and simulated.

The proposed controller structure reduces the number of connection weight by 40% in comparison to fully connected network. The reduction in network connection weight enhances the effectiveness of the network optimization process. The proposed network is basically designed for vehicle lateral control system; however, it can be applied to many other systems. Simulation studies for lateral control systems and two benchmark problems have also been studied.

The robustness of proposed controller is compared with state feedback linear controller and the results show that the controller robustness is superior to the linear controller in vehicle lateral control problem through Monte-Carol evaluations. Comprehensive discussions have been conducted on individual systems and show the advantages of decomposition structure design offer better performance in compare to

linear controller. This controller will be tested on a scaled prototype semi-autonomous vehicle and the experimental result will be presented in Chapter 6.

In this chapter, widespread studies on the decomposition structure including various simulations on different systems and robustness analysis have been conducted. The proposed technique offers a possible method to reduce redundant elements involved in the GA optimization process and gives reasonable performance. However, it is only applicable to systems with a particular dynamics. In the next chapter, a general purpose hybrid controller based on fuzzy theory and GA search will be devised that can be applied to lateral control of vehicles as well as other systems.

Chapter 5. GA/FS Convergence Enhancement by Fired Rules Chromosome

5.1. Introduction

An algorithm inspired by the task decomposition principle was introduced in the last chapter. The rationale was to simplify the conventional neural network structure by effectively reducing the number of connection weights. In this chapter, a chromosome encoding scheme is proposed as an alternative design to improve the optimization speed of hybrid fuzzy-genetic algorithms. In contrast to the algorithm discussed in the last chapter, the suggested scheme is not problem specific and imposes no requirements on the controlled system. Therefore the scheme can be applied in any fuzzy controller that is optimized by the genetic algorithm.

A fuzzy controller design process starts with representing the system knowledge in terms of fuzzy linguistic variables. Next, the corresponding membership functions of linguistic variables are assigned. The generation of the rule-table is normally resolved by exploiting the knowledge of an expert and forming the final rule-base by a trial-and-error approach. However, this procedure is time consuming and problem specific. Hence, there is a strong motivation to automate this process. To achieve automatic fuzzy system determination, the integration of Genetic Algorithms (GA) and Fuzzy System (FS) has been proposed by many researchers as reviewed in Chapter 2. A specific literature citation related to the theme of this Chapter is included in this section.

Giordano *et al.* (2006) proposed the combination of GA and Lyapunov theory for adaptive fuzzy controller design. The fuzzy controller structure and parameter information was encoded into chromosome for GA optimization in an offline mode. The adaptation was then achieved by modifying the output singleton of optimized fuzzy controller online based on Lyapunov theory.

Chou (2006) introduced an index function for mapping input fuzzy set to fuzzy rules. The index function contained six and nine parameter for the cases of linear and nonlinear mapping respectively. Then GA was applied on the index function for fuzzy rule optimization by encoding the function parameters into chromosome instead of entire fuzzy rules. This proposed method reduced the length of chromosome by representing the fuzzy rules table with a linear/nonlinear function.

When it comes to real-time control, the rather tedious process of GA could hinder its effectiveness in practical applications. Uddin *et al.* (2005) implemented genetic based fuzzy controller in real time for motor speed control. The scaling factors of the implemented fuzzy motor controller are tuned by GA in offline.

GA is a probabilistic search and as such requires several generations before convergence to an optimal solution. Therefore, there is a demand to increase the convergence rate of the integrated FS/GA systems. The integrated FS/GA approaches presented in Chapter 2 have focused on improved robustness rather than speed of convergence. To address this requirement, fired rules chromosome encoding scheme is suggested for the integrated FS/GA to increase the speed of optimization. This method reduces the search space by identifying the fired rules at each generation and only optimizes them in the next generation. Modeling errors and parameter

uncertainty are inevitable in any control system application. The robustness to parameter variation of the resulting FS/GA controller is critical to overall reliability of the control system. Thus, Monte-Carlo analysis has conducted to evaluate the controller performance under parameters variation.

This chapter is organized as follow: Section 5.2 describes the main idea and the operation procedure of the proposed encoding in genetic algorithm. Section 5.3 shows a demonstration example of the proposed algorithm to explain the improvement in convergence rate and the reduction in search space. Section 5.4 applies the proposed scheme to optimize the fuzzy vehicle lateral controller and compares with fuzzy lateral controller found by traditional scheme. The comparison is based on the simulation results and Monte-Carol analysis for robustness evaluation. Section 5.5 discusses the result on vehicle lateral control. Finally Section 5.6 gives the conclusion.

5.2. Fired Rules Chromosome for Genetic Algorithms

From the traditional GA architecture and procedure for fuzzy rule table generation stated in Chapter 2, the GA/FS system requires encoding optimization variables (i.e. fuzzy rules) into chromosome for the GA optimization process.

The suggested scheme improves optimization speed by modifying the traditional formation method of chromosome. Traditional encoding method encodes all the fuzzy rules into chromosome therefore chromosome length is equal to the number of fuzzy rules. Therefore the chromosome length is fixed through out the optimization process. For the proposed scheme, only fired rules are encoded into chromosome. Since the number of fired rules varies from generation to generation, the chromosome length is

not fixed and it is changing dynamically from generation to generation. The length of chromosome at generation N is equal to the number of rules fired in generation N-1. Details of encoding scheme and implementation procedure will be shown in the following sections.

5.2.1. Fired Rules Chromosome scheme

Fired Rules Chromosome (FRC) is a chromosome encoding scheme that may reduce both the search space and the time for optimization. FRC only modifies the “coding” module of the Genetic Algorithms. The scheme reduces the search space by encoding the fired rules in each generation into chromosome in contrast to conventional approaches that aim to encode all the rules in the rule table.

To elaborate the idea of FRC in counting the number of fired rules, consider a fuzzy system is optimized by GA and assuming the population size of the GA is 20. Therefore the performances of 20 sets fuzzy rules are evaluated in each generation. The rules fired in each evaluation are recorded, thus 20 sets of fired rules are recorded after 20 evaluations. The rules fired in that generation is found by union of the 20 sets recorded rules. Therefore only the rules that are not fired in all of the 20 evaluations will not be counted as fired rules.

In a fuzzy system, only fired rules contribute to the system output which implies that only those fired rules are related to the performance of the fuzzy rule table. Thus only fired rules are imported to the genetic pool in the FRC scheme without lose of useful genetic materials.

This paragraph presents the idea of search space reduction achieved by FRC

scheme. The search space is defined as the entire range covers all the possible solutions of the given problem. Thus the search space for a fuzzy rule table is all the possible combination of the fuzzy output variables in the rule table. For a Mammdani type fuzzy rule table with 7 linguistic output variables and 49 rules, the corresponding search space is 7^{49} (i.e. 2.57×10^{41}). Since the fuzzy rules are encoded into chromosome, the reduction in chromosome length implies reduction in search space.

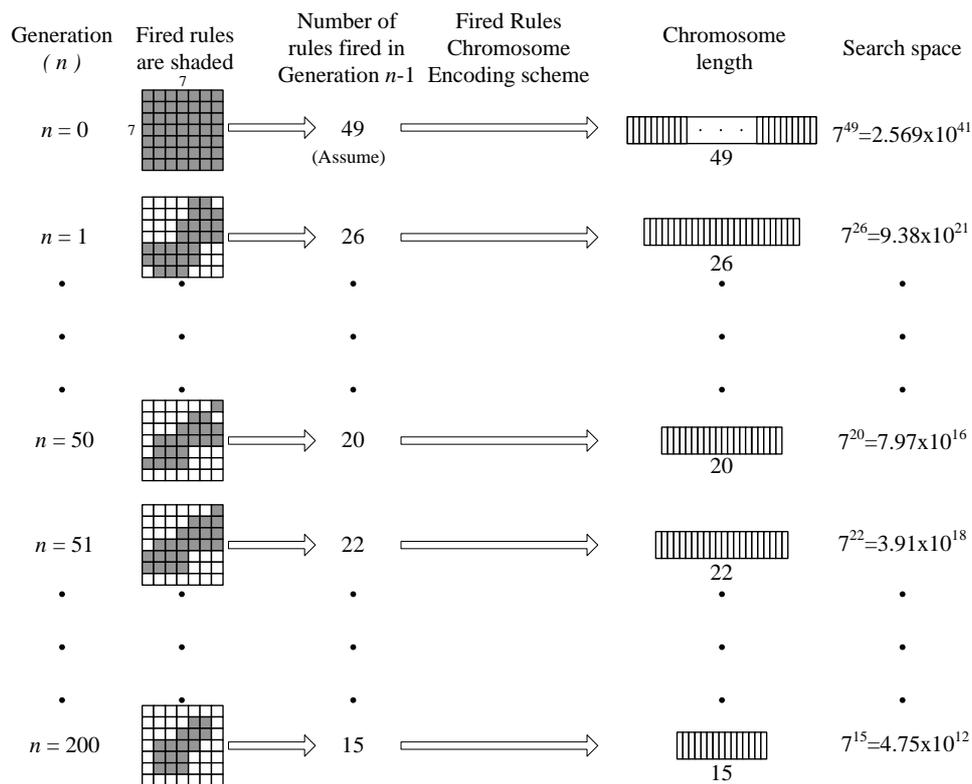


Figure 5-1 : Change of chromosome length in different generations.

Figure 5-1 shows an illustrative example on the search space reduction in FRC. At initialization, generation 0 ($n = 0$), FRC assumes the fired rules chromosome length to be 49. After evaluating the performance of chromosome in the generation 0, assuming 26 rules are fired in the generation 0. Then the FRC scheme will only encode the fired rule zone showed in the shaded part in the rule table. In the first generation ($n = 1$), the chromosome length is 26. As the generation change, Figure 5-1

shows chromosome length varies as the change in the number of fired rules. The length of the chromosome will not only reduce but increases as showed in Generation 50 to Generation 51 which the length of the chromosome is changed from 20 to 22. Since GA updates the fuzzy rules in each generation with result in altering the fuzzy system output characteristic, the number of fired rules may go up or down. Finally, the number of fired rules in Generation 199 is 15 and thus the chromosome length of Generation 200 is 15. As the chromosome length varies from generations to generations, the search space varies as well.

For the above described example, the search space remains at 2.569×10^{41} for traditional (TRD) FS/GA optimization (e.g. Steady State Without Duplicate, SSWOD (Davis (1991))). However, for FRC, the search space will be at maximum 2.56×10^{41} and decrease from 4.75×10^{12} to 9.38×10^{21} . Hence FRC gives a reduction in the search space optimization and substantially improve the convergence rate.

5.2.2. Optimization procedure with FRC

The optimization flow of GA/FS with FRC encoding scheme is shown in Figure 5-2. The optimization flow with FRC is the same as traditional GA/FS system discussed in Chapter 2. The FRC encoding scheme is inserted in between the evaluation and the selection module so that only fired rules will be evolved in the selection and reproduction modules.

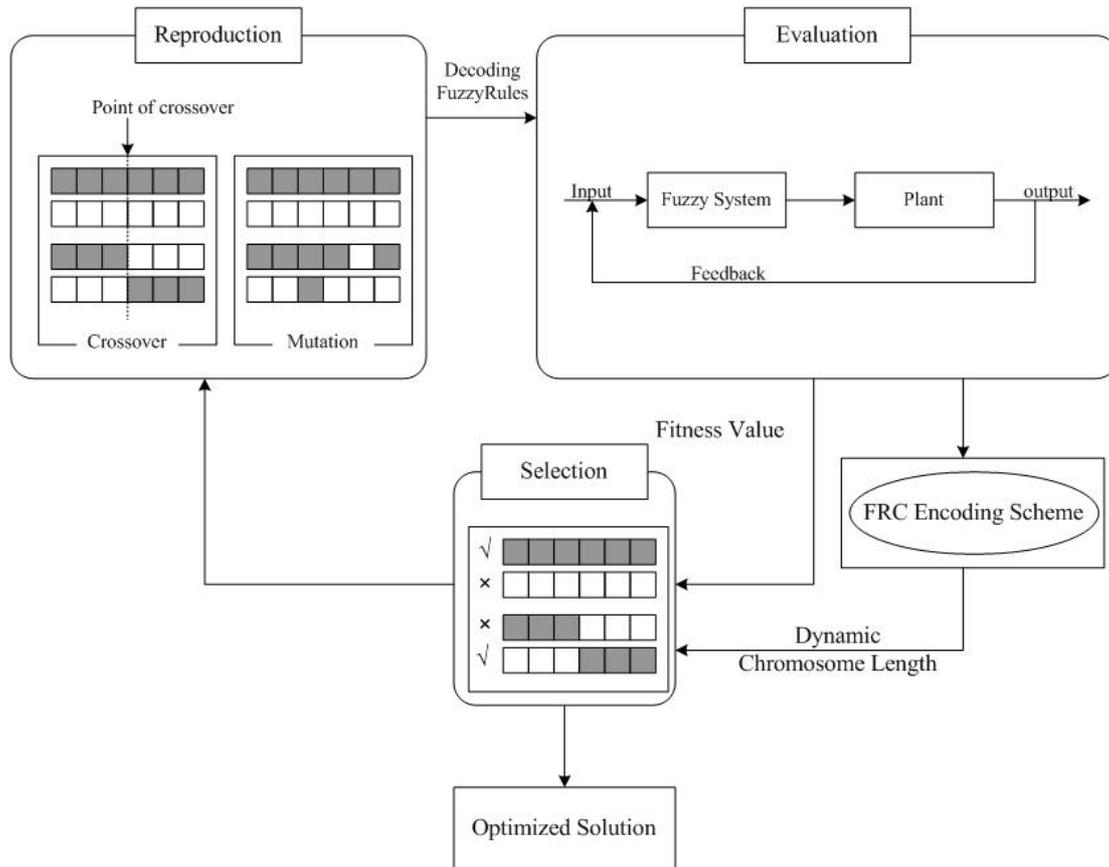


Figure 5-2: GA optimization flow with FRC encoding scheme.

Figure 5-3 shows the different between the traditional encoding and the FRC encoding scheme. At the N^{th} generation there are totally 37 rules fired (non-shaded) in the population. In conventional coding method all the fuzzy rules are encoded into chromosome therefore there are 49 genes in the chromosome. For the FRC encoding scheme, only fired rules are encoded into chromosome therefore the length of chromosome is reduced to 37. If some of the rules are not covered by any individual in the generation N , but they are fired in the evaluation module of generation N , they will be inserted in to the chromosome in generation $N+1$. Moreover if all the rules are fired, the length of the chromosome will include all the rules, then the search space is still the maximal one (49rules in Figure 5-3), then all of the rules are reachable at any point of the GA run. The main point is that the GA using FRC can ignore certain unfired rules at early stages of the run. A premature convergence of GA is prevented

by including those rules into the chromosomes in the subsequent generations whenever they are fired in the Evaluation Module in that generation.

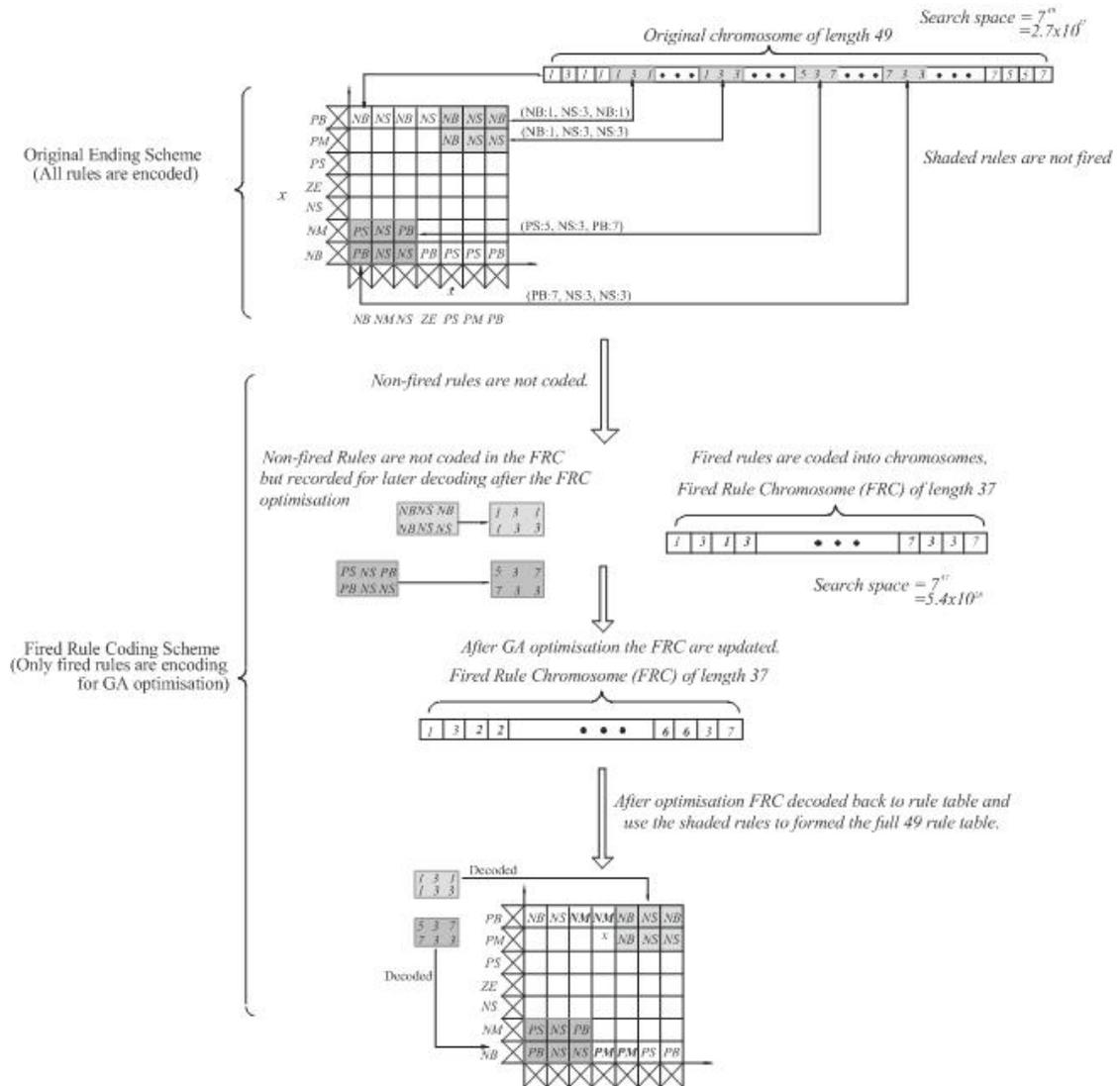


Figure 5-3 : Coding of the Fired Rules Chromosomes.

5.3. FRC demonstration example

To elaborate the presented idea, consider the optimization of a PI-fuzzy system (FS) to control the system (5.3) using GA. The fuzzy controller is optimization to follow a reference input with a unit step and followed by a falling step.

$$\frac{Y(s)}{U(s)} = \frac{(-0.5s+1)e^{-s}}{(s+1)^3} \quad (5.3)$$

where U(s) and Y(s) are the input and output in s-domain respectively.

The fuzzy controller is a 7×7-7 (i.e. the controller have 7 fuzzy sets for two inputs and 7 fuzzy sets for the output). The fuzzy inputs are the error (*e*) and rate of change of error (*ce*). Both *e* and *ce* have 7 fuzzy sets and the output of the fuzzy system is the rate of change of integral control action also has 7 fuzzy subsets. The scaling factors of the FS for *e* for *ce* are 1 and 0.1628, respectively. The scaling factors the control output is are 0.0347.

5.3.1. Search space reduction:

		<i>ce</i>						
		NB	NM	NS	ZE	PS	PM	PB
<i>e</i>	NB	PB	NB	NB	NB			
	NM	NM	NB	NB	NM			
	NS	NB	ZE	PS	NS	PB	NB	PS
	ZE	NB	NM	NM	ZE	PM	PS	PB
	PS	PB	PM	PB	PS	NS	ZE	PM
	PM				NM	PB	PM	ZE
	PB				PB	PM	PB	PB

Figure 5-4 : Fired rules in the 30th generation.

The fuzzy controller is first optimized with FRC scheme and the fired rules in each generation are recorded to show the reduction in search space. It turns out that not all the rules are fired. Figure 5-4 shows the fired rules in the 30th generation and there are 37 rules fired out of 49 rules. Note that some of the rules are not fired (i.e. not involved in the generation 30) in the optimization process which implies that the changes of the un-fired rules do not affect the performance of the FS in this generation. Then the length of the FRC in 31st generation is 37 instead of 49. The reduction of search space from the traditional chromosome to FRC is from 7^{49} to

7^{37} . i.e. a reduction of 2.57×10^{41} . However if the un-fired rules are fired during the 31st generation optimization, they will be again included in the 32nd generation optimization. A more compact representation of the fired rules in chromosomes can lead to better quality solutions for a given amount of function evaluations.

5.3.2. Controller performance and GA convergence.

In order to compare the performance of the proposed FRC scheme with traditional (TRD) scheme, the fuzzy controller is optimized by GA with both FRC and TRD scheme. As the genetic algorithm is a stochastic method, the optimization process with the FRC algorithms and the TRD GA/FS is repeated for 50 sets. Table 5-1 gives a summary of the parameters of the testing for both algorithms. The population size, crossover probability, mutation probability, maximum number of generation, and fitness function is 50, 0.7, 0.3, 50, and inverse of Integral Time Absolute Error (ITAE) respectively. The chromosome length is 49 for TRD, but not defined in FRC. 50 sets of initial population are randomly generated for both processes (GA with FRC and TRD), therefore both processes start with the same initial population in all the 50 simulations.

After the 50 sets optimization processes, the fuzzy controller fitness, performance, and time spent are follow. Figure 5-5 shows the average fitness of 50 sets verses number of generations. It shows that at the beginning generations (1st to 3rd) the fitness of fuzzy controller optimized by FRC and TRD encoding scheme are more or less the same, this is due to the initial populations are randomly generated. However, as number of generations increase, the FRC converged faster that the TRD encoding. It is due to the FRC scheme narrows the evolution zone to the rules that have effect on controller fitness, but the evolution in TRD scheme covers the

complete rule table which includes the rules have no contribution to the controller fitness. Figure 5-6 shows performance of FRC (solid line) and the TRD Encoding (dotted line) in the first testing. The reference signal is a unit step function and followed by a falling step after 100s. In the figure both solid and dotted lines are able to follow the reference signal but the solid line shows better tracking performance in the falling edge. This result shows that the newly proposed method improves the convergence rate and retains the performance as the traditional method. Finally Table 5-2 summarized the performance of the testing. The optimization time spent for FRC and TRD FS/GA are 110.6158s and 110.428s, respectively. This shows that the additional overhead of their Fired Rules Chromosome bookkeeping, is insignificant compared to the control loop testing spent in each generation. It's because in GA optimization, the most time process involved is the Elevation Module.

Population size	50
Chromosome length	49 (dynamic for FRC)
Maximum number of generation	50
Crossover probability	0.7
Mutation probability	0.3
The fitness function	$1/(1+ITAE)$

Table 5-1: GA parameters.

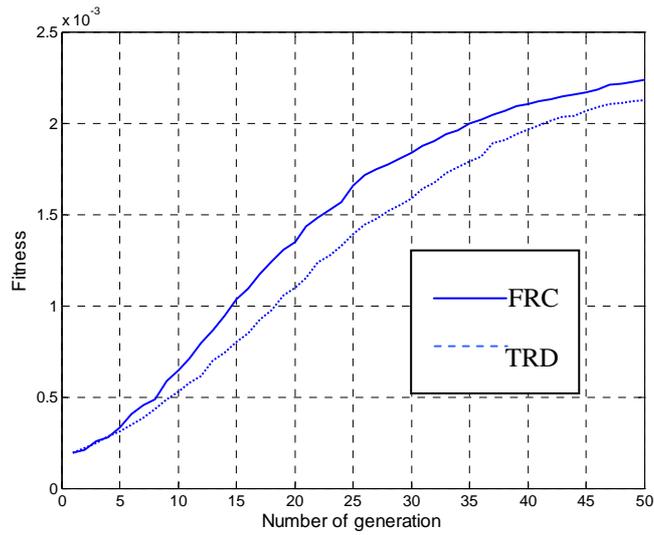


Figure 5-5: The convergence of FRC (solid line) and the TRD (dotted line).

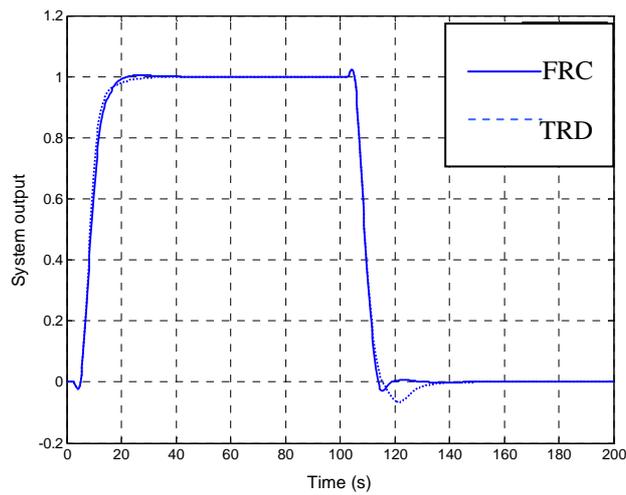


Figure 5-6: Performance of FRC (solid line) and the TRD encoding (dotted line) in the first testing.

	FRC	TRD
ITAE	445.8869	469.1287
Average Fitness (50 generations)	0.001454	0.001290
Time spent	110.6158s	110.428s

Table 5-2: Result of average of 50 test sets of FRC and TRD encoding.

5.4. Application to vehicle lateral control

To study the performance of the proposed method in real life example, a Mammdani type fuzzy controller (FC) is designed for the vehicle lateral control. The FC has 2 inputs with 7 triangular membership functions and there are 49 rules in the rule table. The inputs are lateral displacement error (e) and the vehicle angle (θ) with respect to the reference lane. Figure 5-7 shows the membership functions of the 2 inputs. The membership functions are normalized and the scaling factors for e and θ are 0.5 and 1.43 respectively. There are 9 output linguistic variables which namely NVB, NB, NM, NS, Z, PS, PM, PB, and PVB with singleton output function. The vehicle lateral model used for simulation and optimization is bicycle model. The bicycle model equation shows in (5.4) and the detail description on the model parameters refers to Chapter 3. Since the vehicle lateral displacement and vehicle angle are not explicitly shown in the bicycle model, equation (4.11) is used to obtain the lateral displacement and orientation.

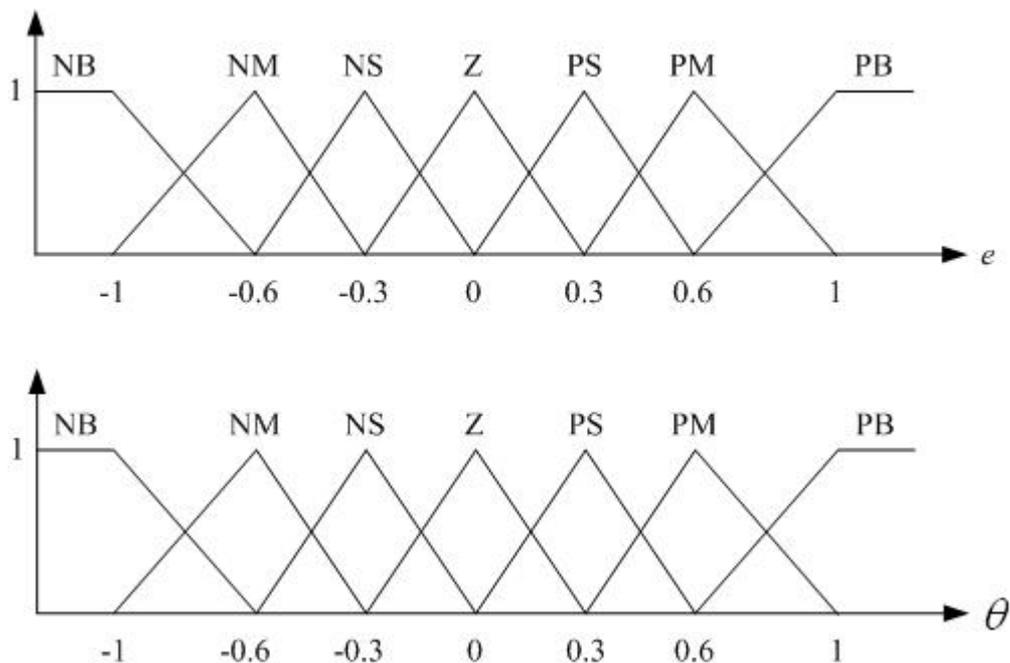


Figure 5-7. Input membership functions of the fuzzy lateral controller.

$$\frac{d}{dt} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a_{21} & a_{22} & -a_{21} & a_{24} \\ 0 & 0 & 0 & 1 \\ a_{41} & a_{42} & -a_{41} & a_{44} \end{bmatrix} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ 0 & b_{32} \\ b_{41} & b_{42} \end{bmatrix} \begin{bmatrix} \delta_f \\ \rho_{ref} \end{bmatrix} \quad (5.4)$$

The rule table of the above mentioned fuzzy controller is optimized by GA with both FRC and TRD encoding schemes. The model parameters (a_{21} to a_{44} and b_{21} to b_{41}) in (5.4) are defined in Chapter 3. Both optimization processes with FRC and TRD encoding schemes have included SSWOD technique to avoid duplicated chromosome in the population. To start optimization, an initial stable fuzzy system (symmetric) is designed. The control signal of the fuzzy system depends on the lateral displacement and the angular displacement.

The setting of evaluation process in the GA evaluation module is described as follow. The vehicle initial lateral position and orientation are $10cm$ and 0 degree with respect to the reference lane. The reference signal (lateral set-point) is initially at $20cm$ and step back to $10cm$ after $10s$. This reference signal ensures the fuzzy rule table is capable of both positive and negative steering. The total evaluation time is $20s$ therefore 2000 steps for step size with $0.01s$.

The GA parameters used in the optimization process are listed in Table 5-3. The population size, maximum number of generation, crossover probability, and mutation probability are 30 , 200 , 0.7 , and 0.3 respectively. For the fitness function, the ITAE shown in Table 5-3 is the integrated time absolute error. After optimization 2 different sets of rule table are yield. The GA processes with both encoding schemes take 200 generations to complete the optimization. These 2 sets of fuzzy rule table are

simulated and the simulation result will be shown in Section 5.4.1. In addition to the simulation results, the robustness of both fuzzy rule tables subject to vehicle parameter uncertainties is evaluated by Monte-Carlo analysis and it will be presented in Section 5.4.2.

Population size	30
Chromosome length	49 (dynamic for FRC)
Maximum number of generation	200
Crossover probability	0.7
Mutation probability	0.3
The fitness function	$1/(1+ITAE)$

Table 5-3 : GA parameter.

5.4.1. Simulation result on vehicle lateral control

In the simulation, the vehicle is assumed running at a straight road and thus the road curvature is zero ($\rho_{ref} = 0$). The vehicle initial position is located at 10cm from the reference lane and the lateral position set-point is 20cm. The vehicle is assumed running at $0.6ms^{-1}$.

Figure 5-8 shows simulated performance of fuzzy controller optimized by FRC (solid line) and TRD (dotted line) scheme. The simulation results show that the fuzzy controller optimized by both FRC and TRD encoding scheme are able to maneuver the vehicle to the lateral set-point. Figure 5-8 shows that the settling time of both controllers are 2s and the maximum overshoot of FRC and TRD is 5% and 10% respectively. The performance index (ITAE) of FRC and TRD are 192.031 and 212.072 respectively. The above performance indexes are listed in Table 5-4 for comparison.

	FRC	TRD
Settling Time	2s	2s
Maximum percentage of overshoot	5%	10%
ITAE	192.031	212.072

Table 5-4. Performance indexes of both fuzzy controllers.

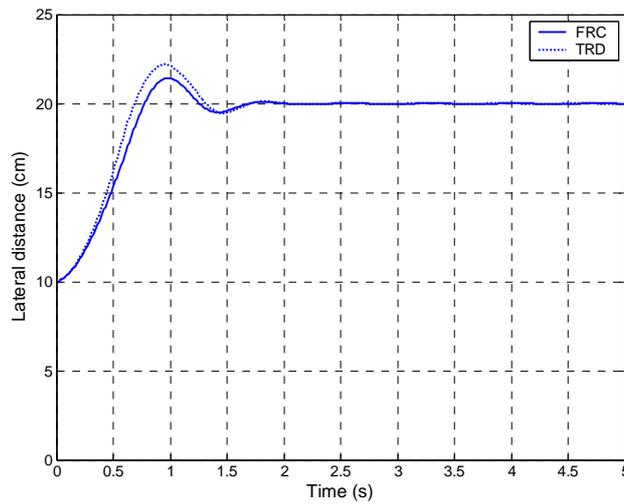


Figure 5-8: Lateral control simulation result.

5.4.2. Robustness evaluation by Monte-Carlo analysis

In this section, the controller robustness is examined by the Monte-Carlo evaluation method (Ray and Stengel (1993)). The fuzzy controller optimization process bases on mathematical model which inevitably includes incomplete system dynamics. Taking the above simulation as example, the parameters change of the bicycle model due to the change in vehicle speed, mass, road friction and tire cornering stiffness are not reflected in the model. Therefore it is important to show that the controller found by the proposed scheme is as robust as controller optimized by conventional scheme.

To illustrate the effect of parameter variation on the step response of the fuzzy

controller, 10,000 Monte-Carlo evaluations are performed. Two parameters are randomly selected and are varied $\pm 30\%$ of uniform uncertainty in each evaluation. Figure 5-9 and Figure 5-10 show the 10,000 step responses with $\pm 30\%$ randomly selected parameters uncertainty of the bicycle model. Envelopes shown in figures are constructed based on the nominal step responses of the vehicle with fuzzy controller. The step response is multiplied by factors 1.2 and 0.8 to define $\pm 20\%$ deviation limit respectively. The purpose of defining the envelope is to estimate the probability of step response violating the envelope. That is the probability of a step response to fall out of the envelope if system parameters have $\pm 30\%$ variation.

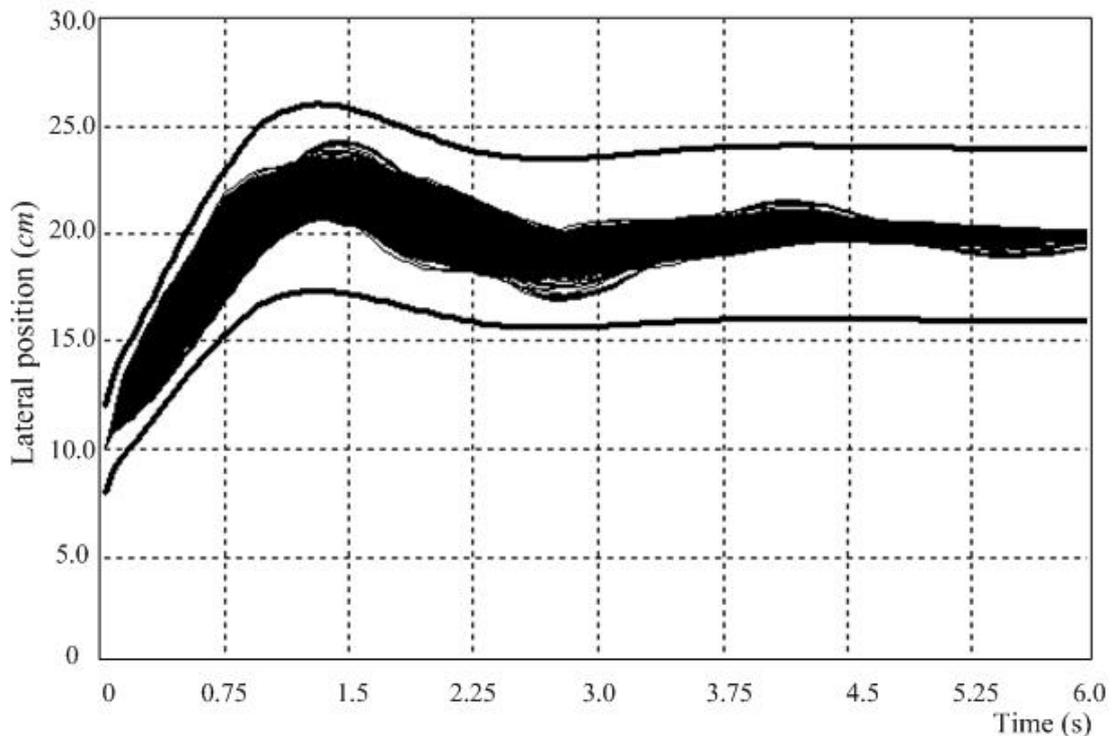


Figure 5-9 : FRC Monte-Carlo analysis result.

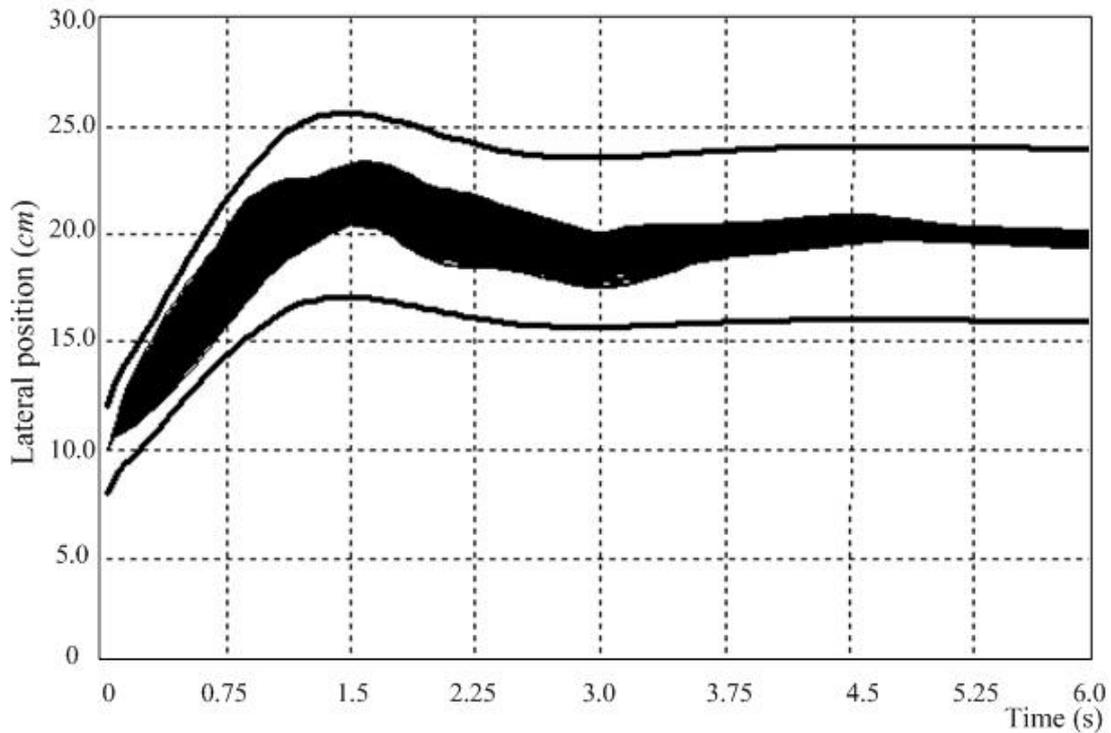


Figure 5-10 : TRD Monte-Carlo analysis result.

Figure 5-9 shows 10,000 Monte-Carlo evaluations for $\pm 30\%$ parameter uncertainties of the fuzzy controller optimized with FRC encoding scheme. The result shows that all the evaluations do not violate the envelope. In Figure 5-10 shows the result of Monte-Carlo evaluation on the fuzzy controller optimized with TRD encoding scheme. The result also shows that all the 10,000 evaluations are within the envelope.

The result presented here has two folds significance. First, it shows that the fuzzy controller designed and optimized with FRC encoding and TRD encoding schemes is robust to $\pm 30\%$ parameter uncertainties. Second, in comparison with the robustness of fuzzy controller optimized with TRD scheme, FRC encoding scheme does not affect the robustness of the controller yet it improves the convergence rate (result shown in Section 5.3.2).

5.5. Discussion

In this section, the simulation and Monte-Carol analysis results on the vehicle lateral control are discussed. The two fuzzy controllers optimized by FRC and TRD methods are tested in simulations as described in the previous sections. The fuzzy controller structure is a two inputs single output fuzzy system. The controller inputs are lateral distance and vehicle orientation.

From the simulation results, there is 10% overshoot observed on the controller optimized by TRD method. It is due to the changes of the control variables are not involved in the controller design. Thus the controller performance can be further enhanced by including those parameters. For the performance index, although there is large deviation observed at the initial stage between the two results, there is only insignificant difference in the value of performance index. It is due to the ITAE index gives large penalties at the steady state but not at the initial stage.

5.6. Conclusions

The suggested Fired rules chromosome (FRC) encoding scheme offers a novel encoding module for the integrated system of fuzzy system and genetic algorithm. The advantage of the proposed FRC module is the enhancement of convergence rate. The enhancement is achieved by narrowing the search space to fired rules only. The special feature of the presented encoding method is that the chromosome length is changed dynamically which depends on the number of fired rules.

The comparison results between FRC scheme and conventional (TRD) scheme

are presented. Controller performance evaluation bases on performance index (such as setting time, overshoot and ITAE) and the results show that both fuzzy controllers have similar performance. For the controller robustness, the Monte-Carol analysis results show that both controllers are robust to $\pm 30\%$ parameters variation. Thus the newly proposed encoding scheme offers higher convergence rate but not affects the controller robustness.

In the next chapter, both fuzzy controllers optimized by FRC and TRD scheme will be implemented on a scaled prototype vehicle for performance evaluation in experimental study.

Chapter 6. Experimental Results on Vehicle Lateral Control

6.1. Introduction

Lateral control of autonomous vehicles is a fundamental task within the context of an automated highway system (AHS). It is also crucial for successful operation of an autonomous vehicle driving in mixed traffic highways where it shares the road with other human driven vehicles. Simply stated, the lateral control refers to automatic vehicle steering in order to maintain a prescribed direction and follow a reference track. The automatic steering control can be grouped into either look-ahead or look-down reference systems (Guldner *et al.* (1997)). This classification is based on the lateral displacement of the vehicle from a reference lane. The look-ahead system mimics human driving behavior by measuring the lateral displacement ahead of the vehicle. Look-ahead systems using machine vision are extensively studied such as *VaMoRs-P* by Maurer *et al.* (1994) and *VITA I* and *II* by Ulmer (1992 and 1994). The automatic steering can be considerably improved by a preview of up to a approaching road curvature so that allowing the controller to anticipate the change of curvature for smooth transition (Hernandez and Kuo (2003)).

Hessburg and Tomizuka (1994) proposed a modularized fuzzy lateral controller for preview type lateral orientation. The fuzzy structure consisted of feedback, preview, and gain scheduling modules. The feedback modules minimized lane tracking error where as the preview module modified the steering angle based on the perceived road curvature from a database. Finally the steering angle was scaled by the

gain scheduling module according to the vehicle speed. Hessburg and Tomizuka (1995) further enhanced the fuzzy controller to a model reference adaptive fuzzy controller. Stability was ensured by a supervisory controller based on Lyapunov theory.

Hernandez and Kuo (2003) simulated the use of GPS and geometry database to preview road curvature for vehicle lateral control. The proposed control structure used a single feedback state controller which was already suggested by Guldner *et al.* (1999) for the steering angle generation. The preview action was achieved by replacing the state controller input from measured state to an estimated state which was based on the previewed road curvature.

Lateral controller based on look-down reference is also an active area due to availability of robust lateral displacement measurement sensors. Guldner *et al.* (1999) suggested the use of two sensors installed at the front and tail bumper of the vehicle in a look-down reference system design in order to solve the automatic steering problem at high speed. A robust state feedback controller based on parameter space approach was proposed to tackle the lateral control problem. The state feedback controller consisted of two controllers for front and tail lateral measurements respectively. The final controller output was the sum of the two controllers outputs. Choi (2000) designed an adaptive lateral controller for look down feedback reference system. The proposed algorithm uses single magnetic sensor to measurement vehicle lateral displacement. Adaptive PD type controller based on Lyapunov criterion is then designed as the vehicle lateral controller.

In this chapter, the prototype semi-autonomous scaled vehicle developed in

Chapter 3 is configured for a look-down reference system. The lateral measurement is obtained by two infrared sensors installed at each side of the vehicle. The lateral control system generally requires the information of vehicle lateral displacement and its orientation. The controlled vehicle should maintain a desired lateral distance from the reference lane. In addition, it should be parallel to the reference lane. Therefore, both displacement and orientation should be considered in the control system.

In the previous two chapters, two novel controllers were reported and their performance was tested by simulations studies. In this Chapter, the two controllers (Fused Neural Network Controller in Chapter 4, and FRC and TRD Fuzzy controller in Chapter 5) will be implemented as lateral controllers on the scaled vehicle in order to verify their performances in experimental studies and provide alternative solution to vehicle lateral problem.

The organization of this chapter is follow: Section 6.2 describes the experimental setup for the vehicle lateral control experiment. Section 6.3 presents the experimental result of the fused neural network controller. Section 6.4 reports the result of the fuzzy lateral controllers optimized by GA with FRC and TRD encoding schemes. Section 6.5 discusses the experimental results of the three controllers mentioned above. Finally Section 6.6 concludes this chapter.

6.2. Lateral control Experiment setup

As noted in Chapter 3, four infrared sensors and industrial computer are installed on the prototype vehicle for lateral displacement measurement and control algorithms implementation. The controllers developed in Chapters 4 and 5 were implemented on

the onboard industrial computer for lateral control experiments. The lane keeping problem is to maintain the vehicle at a desired lateral displacement from a reference lane and to minimize the angle between the vehicle and the reference lane. In the following experiments, the reference wall is used as reference lane instead of lane makers so that the infrared sensors can measurement the lateral displacement form the wall.

6.2.1. Experimental setup

The experiments were conducted in a long and narrow corridor inside the electrical engineering building. The wall of the corridor was taken as the reference lane. The experiments in this chapter emulate straight road scenario on a highway, therefore the road curvature is zero. In order to test the controller capability under different speeds, three sets of experiments at speed of 42cms^{-1} , 60cms^{-1} , and 78cms^{-1} were conducted. The speed setting was based on $\pm 30\%$ of the nominal speed (60cms^{-1}) used during the controller design process.

The vehicle was required to maintain a lateral displacement of 20cm from the reference lane ($y_{ref} = 20\text{cm}$) throughout the experiments. The vehicle initially was located at 10cm from the reference lane with a zero angle orientation (i.e. parallel to the lane). During the experiments, the infrared sensor readings and speed were recorded for performance evaluations.

The control system for both lateral and speed control is shown in Figure 6-1. The upper loop describes the lateral feedback control loop. The details of the steering actuator and speed drive are already described in Chapter 3. Three vehicle lateral

controllers, namely Fused Neural Network (NN), Fuzzy (FRC), and Fuzzy (TRD) lateral controller are implemented by replacing the “Lateral Controller module” shown in Figure 6-1. A conventional PID speed controller was designed and manually tuned to achieve a reasonable performance under the desired speed range.

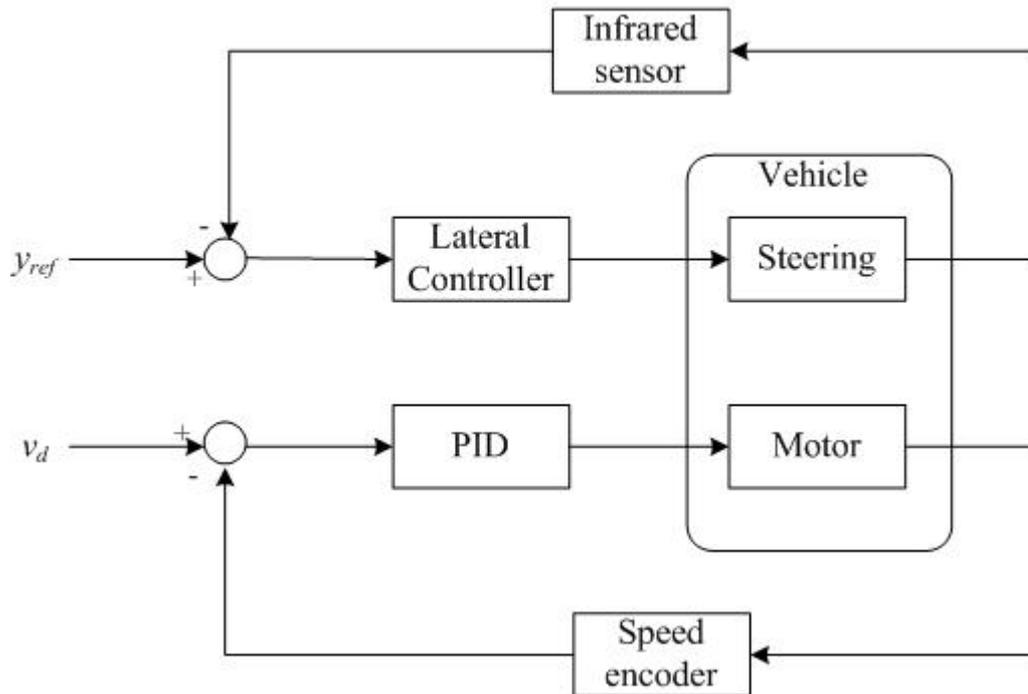


Figure 6-1: Lateral control experiment control loop.

6.2.2. Real time implementation of the lateral controller

Selection of sampling time is critical to real time applications. In Chapter 3 presented the onboard computer generates fixed hardware timer interrupts every 50ms and the corresponding interrupt frequency is 20 Hz for initiate data sampling. Brennan (1999) presented most of full-size vehicle frequency range was from 0 to 2 Hz or 3 Hz. The onboard timer offers around 10 times of the bandwidth of the system which lies in acceptable range for data sampling. Therefore 50ms is selected as the sampling time (t_s).

As stated in Chapters 4 and 5 vehicle state variables are manipulated to obtain

vehicle lateral displacement (d) and orientation (θ) based on (4.11). In real time implementation, the measured lateral displacements are also manipulated. However, the rate of lateral displacement is not directly measurable by the infrared sensors. Thus the rate of lateral displacement is calculated mathematically by equation (6.1). Then equation (4.11) is converted to equation (6.2) for real time implementation.

$$\begin{aligned}\dot{\bar{y}}_f(k) &= \frac{\bar{y}_f(k) - \bar{y}_f(k-1)}{t_s} \\ \dot{\bar{y}}_r(k) &= \frac{\bar{y}_r(k) - \bar{y}_r(k-1)}{t_s}\end{aligned}\tag{6.1}$$

where \bar{y}_f and \bar{y}_r are lateral distance measured by infrared sensors as mentioned in Chapter 3.

$$\begin{bmatrix} \theta \\ \dot{\theta} \\ d \\ \dot{d} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0.5 & 0 & 0.5 & 0 \\ 0 & 0.5 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} \bar{y}_f \\ \dot{\bar{y}}_f \\ \bar{y}_r \\ \dot{\bar{y}}_r \end{bmatrix}\tag{6.2}$$

To summarize the real time implementation conditions for vehicle lateral control, the sampling interval of the system is $50ms$ for data collection and control algorithm computation. For the sensor measurement, the measured lateral displacement information is manipulated by equation (6.1) and (6.2) so that the information can be processed by the Fused Neural Network controller and Fuzzy controller proposed in Chapter 4 and 5 respectively. The experimental results of the three controllers under different speed of operation are presented next.

6.3. Fused Neural Network Controller experimental result

The neural network controller proposed in Chapter 4 is examined experimentally in this section. The controller has four inputs and a single output (u i.e. steering angle for vehicle lateral control). The inputs are θ , $\dot{\theta}$, d , and \dot{d} as shown in Figure 6-2

where θ , $\dot{\theta}$, d , and \dot{d} are angle, rate of angle, displacement, and rate of displacement respectively. The controller connection weights for lane keeping problem were shown in Table 4-3. The values of the connection weights W_1 , W_2 , W_3 , W_4 , W_5 , and W_6 , are 0.9846, 0.1342, 0.1303, 0.2148, -16.1873, and 12.4779 respectively.

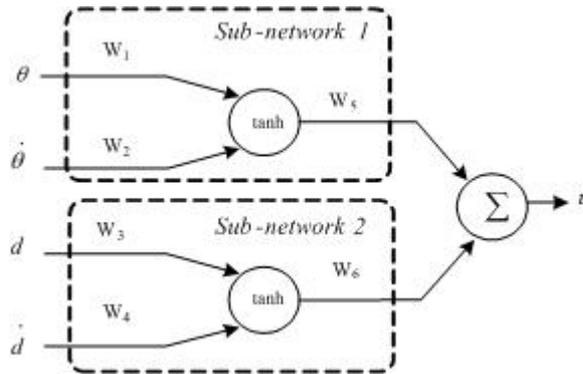


Figure 6-2: Fused neural network controller.

The experimental results of the Fused NN controller with speeds of 42cms^{-1} , 60cms^{-1} , and 78cms^{-1} are shown in Figure 6-3 to 6-5 respectively. The figures contain vehicle lateral position, vehicle orientation, and speed during the experiment. The vehicle orientation is evaluated by the different between the lateral displacement sensor at front and tail.

In Figure 6-3a, Figure 6-4a, and Figure 6-5a, vehicle reaches the lateral set-point at time 3.5s , 2.5s , and 2.2s respectively. The mentioned time is the time taken to adjust vehicle lateral position from the initial position to the lateral set-point. The results show that the NN controller is able to stabilize the vehicle at the steady state under different speeds. In Figure 6-5a, $\pm 10\%$ deviation from the set-point is observed.

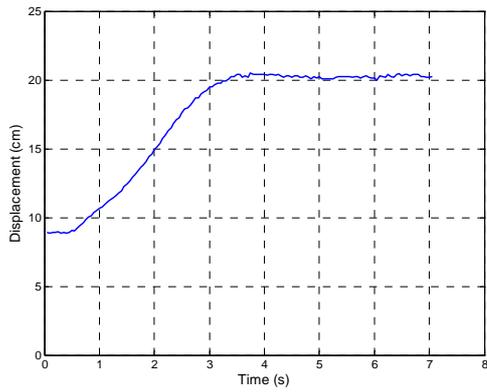


Figure 6-3a

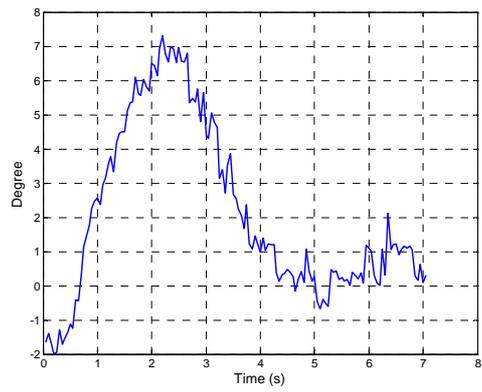


Figure 6-3b

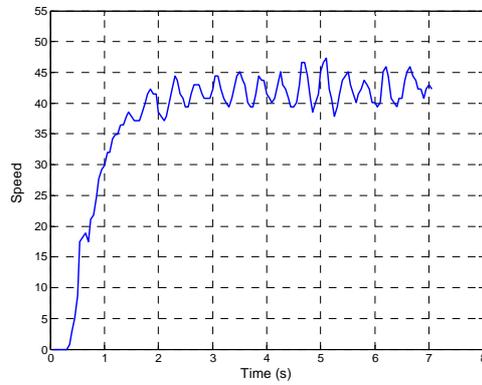


Figure 6-3c

Figure 6-3: Fused NN controller experimental result with speed= 42cm s^{-1} .

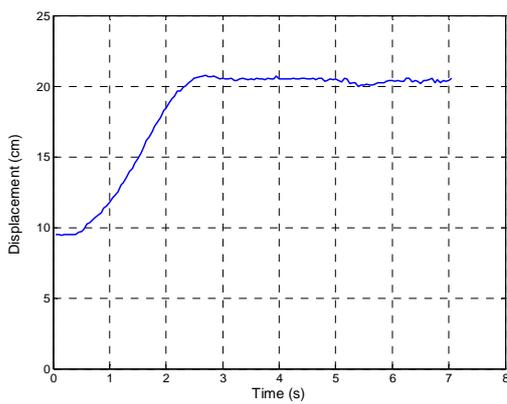


Figure 6-4a

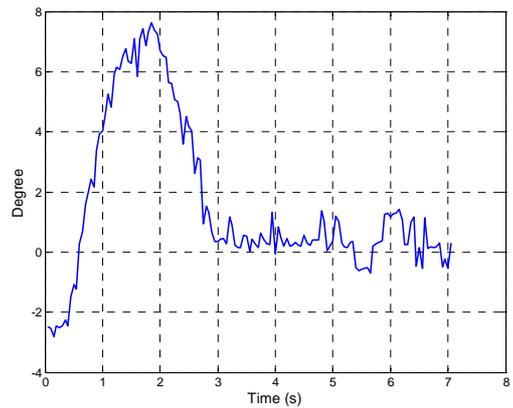


Figure 6-4b

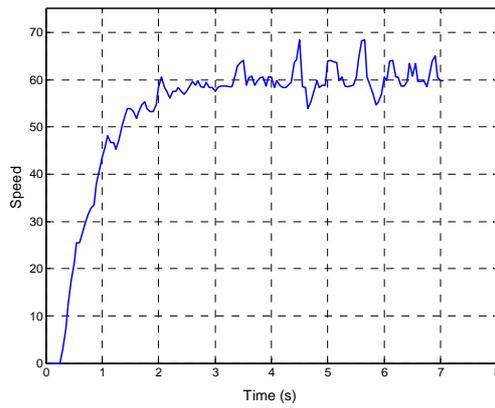


Figure 6-4c

Figure 6-4: Fused NN controller experimental result with speed= 60cm/s .

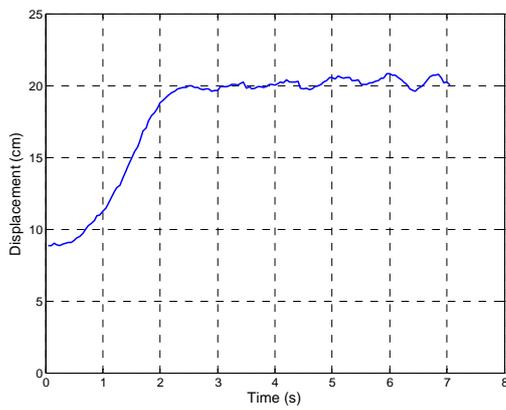


Figure 6-5a

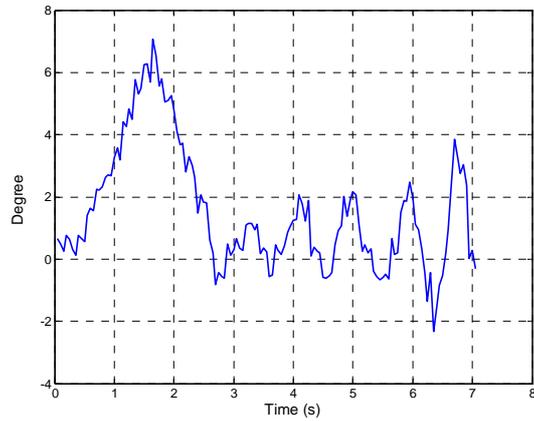


Figure 6-5b

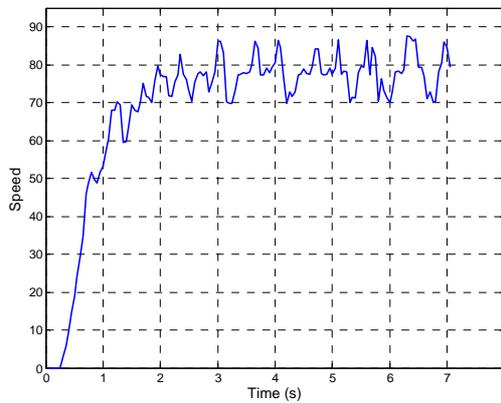


Figure 6-5c

Figure 6-5: Fused NN controller experimental result with speed= 78cm/s .

From Figure 6-3b, Figure 6-4b, and Figure 6-5b, the maximum degree of vehicle orientation (around 7°) appears at time 2.1s, 1.9s, and 1.5s respectively. Referring to the time spent recorded at the above paragraph, the maximum degree of orientation occurs during the time period of lateral position adjustment from the initial position to the set-point. At the steady state operation, the controller stabilizes the vehicle within $\pm 2^\circ$ which is well within the acceptable level. Finally Figure 6-3c, Figure 6-4c, and Figure 6-5c show the speed profile of the vehicle during the lateral control experiment

The experimental results show that the proposed NN controller designed by task decomposition can be applied to tackle vehicle lateral control problem. The controller shows acceptable performance in different speed of operation. The experimental results with different speed show that the controller maintains vehicle lateral position at the desired lateral displacement from the reference lane and adjusts vehicle orientation parallel to the reference lane.

6.4. Fuzzy lateral controllers experimental results

To study the performance of the FRC encoding scheme in a practical setting, a Mammdani type fuzzy controller (FC) was designed for the vehicle lateral control. The FLC has 2 inputs with 7 triangular membership functions and there are 49 rules in the rule table. The inputs are displacement error (e) and the vehicle angle (θ) with respect to the reference lane. In order to show the FRC scheme does not degrade the performance, a Mammdani type FLC optimized by traditional method (TRD) is constructed.

The two controllers were implemented on a scaled prototype autonomous vehicle for experimental testing. The experimental results of the fuzzy controller

optimized by FRC encoding scheme (FRC controller) with speeds 42cms^{-1} , 60cms^{-1} , and 78cms^{-1} are shown in Figure 6-6, Figure 6-7, and Figure 6-8 respectively. For the fuzzy controller optimized by TRD encoding scheme (TRD controller), the experimental results with speed 42cms^{-1} , 60cms^{-1} , and 78cms^{-1} are shown in Figure 6-9, Figure 6-10, and Figure 6-11 respectively.

6.4.1. Fuzzy (FRC) controller experimental results

In Figure 6-6a, Figure 6-7a, and Figure 6-8a, the FRC controller maneuver vehicle to reach lateral set-point at time 3s , 1.9s , and 1.5s respectively. it is the time taken to adjust vehicle lateral position from the initial position to the lateral set-point. The FRC controller shows better performance at steady state with speed 42cms^{-1} and 60cms^{-1} . Bounded ($\pm 10\%$) lateral oscillation is observed in the experiment with speed 78cms^{-1} .

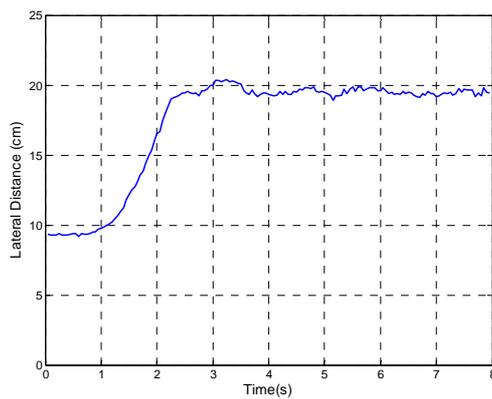


Figure 6-6a

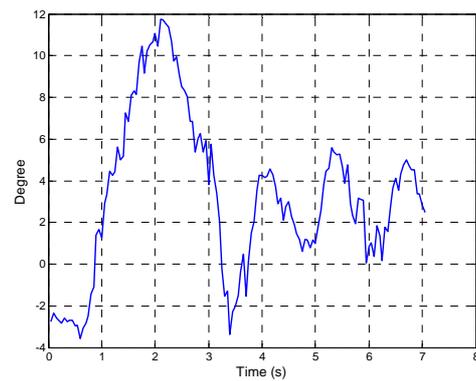


Figure 6-6b

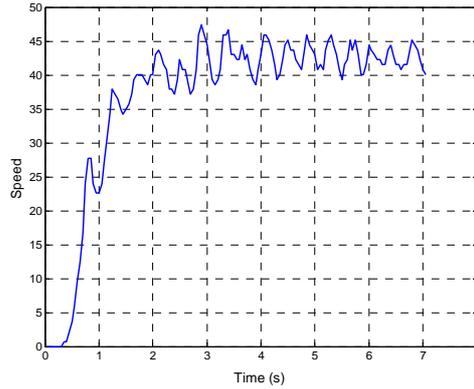


Figure 6-6c

Figure 6-6: FRC fuzzy controller experimental result with speed= 42cm/s .

From Figure 6-6b, Figure 6-7b, and Figure 6-8b, the maximum degree of vehicle orientation is recorded as 11.7° , 12.4° , and 12.4° at time 2.1s, 1.7s, and 1.25s respectively. The recorded maximum degree of orientation occurs during the time period of lateral position adjustment from the initial position to the set-point. In general the FRC controller stabilizes the vehicle at the steady state within $\pm 6^\circ$ in different speed of operation. Finally Figure 6-6c, Figure 6-7c, and Figure 6-8c show the speed profile of the vehicle during the lateral control experiment.

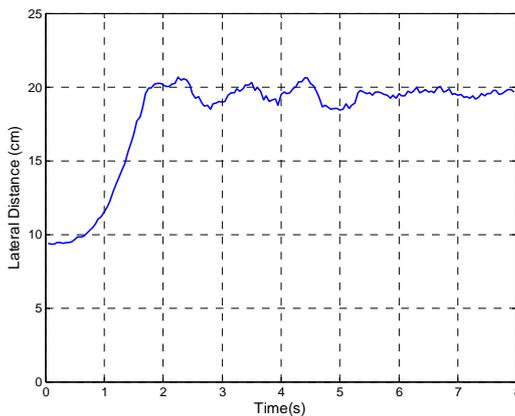


Figure 6-7a

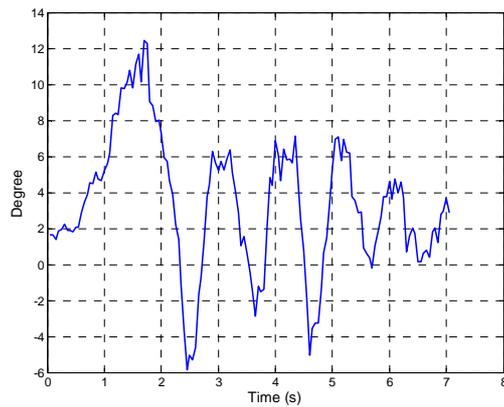


Figure 6-7b

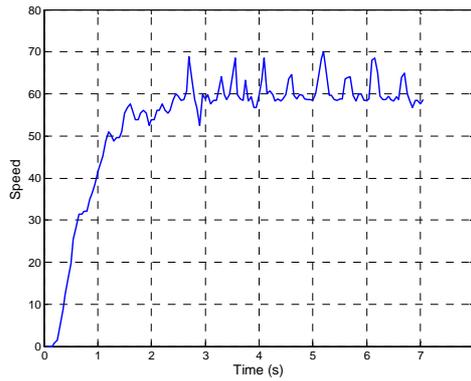


Figure 6-7c

Figure 6-7: FRC fuzzy controller experimental result with speed= 60cm/s^{-1} .

The experimental results show that the FRC controller performs well under the speeds of 42cm/s^{-1} and 60cm/s^{-1} ; although steady-state oscillation is recorded at a speed of 78cm/s^{-1} , it is bounded and thus the system is stable in operation.

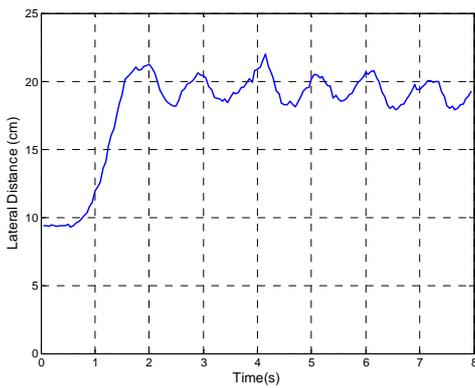


Figure 6-8a

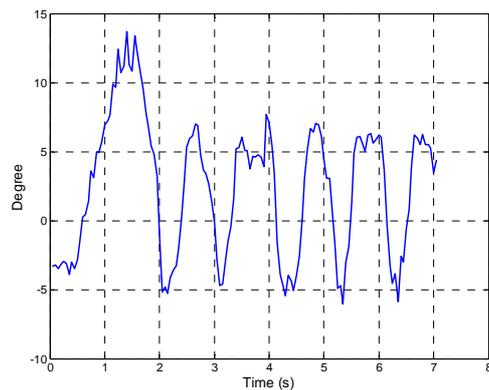


Figure 6-8b

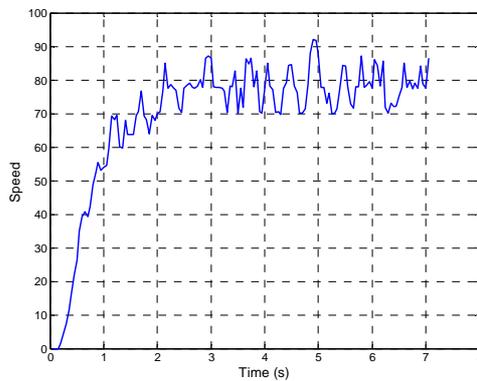


Figure 6-8c

Figure 6-8: FRC fuzzy controller experimental result with speed= 78cm/s^{-1} .

6.4.2. Fuzzy (TRD) Controller experimental results

In Figure 6-9a, Figure 6-10a, and Figure 6-11a, the TRD controller adjusts vehicle lateral position to reach lateral set-point at time 3s, 2s, and 1.9s respectively. The TRD controller also shows better performance at steady state with speed 42cms^{-1} and 60cms^{-1} . Bounded ($\pm 10\%$) oscillation is observed in the experiment with speed 78cms^{-1} .

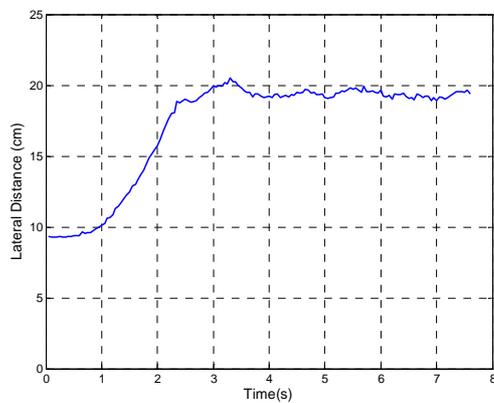


Figure 6-9a

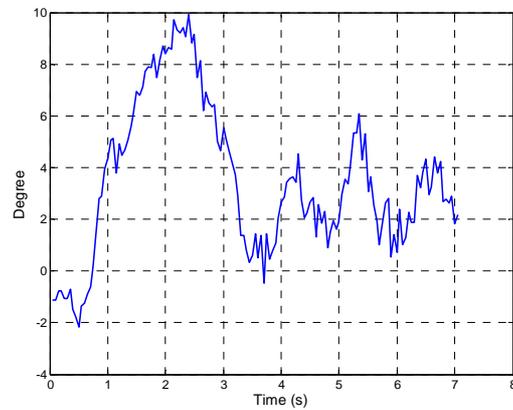


Figure 6-9b

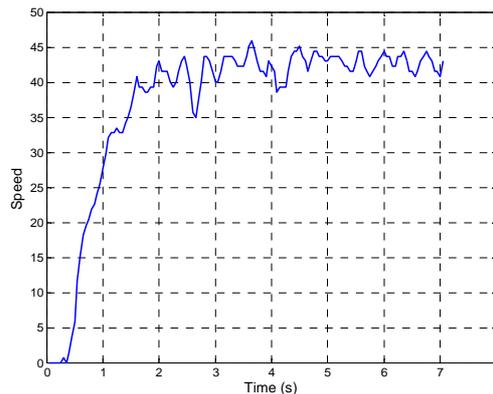


Figure 6-9c

Figure 6-9: TRD fuzzy controller experimental result with speed= 42cms^{-1} .

From Figure 6-9b, Figure 6-10b, and Figure 6-11b, the maximum degree of

vehicle orientation is recorded as 9.9° , 9.9° , and 10.7° at time $2.4s$, $1.9s$, and $1.6s$ respectively. The recorded maximum degree of orientation occurs during the time period of lateral position adjustment from the initial position to the setpoint. In general the TRD controller stabilizes the vehicle at the steady state within $\pm 6^\circ$ in different speed of operation. Finally Figure 6-6c, Figure 6-7c, and Figure 6-8c show the speed profile of the vehicle during the lateral control experiment.

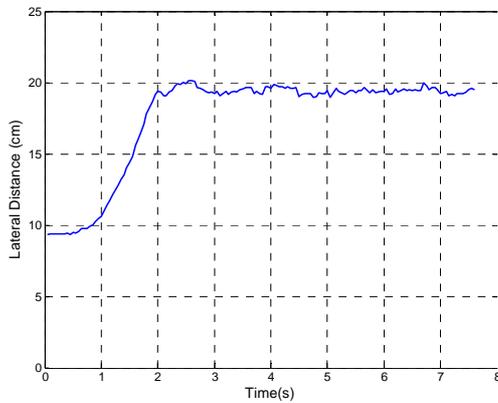


Figure 6-10a

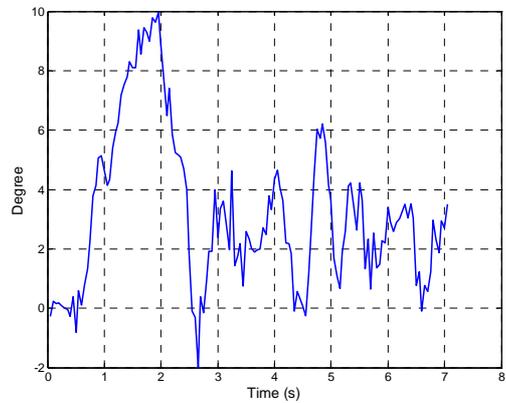


Figure 6-10b

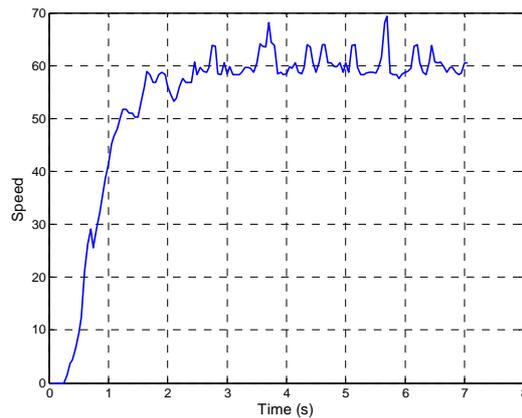


Figure 6-10c

Figure 6-10: TRD fuzzy controller experimental result with speed= 60cm s^{-1} .

The experimental results show that the FRC controller performs better under

speed 42cms^{-1} and 60cms^{-1} . Although steady stated oscillation is recorded in the speed of 78cms^{-1} , it is bounded and thus the system is stable in operation.

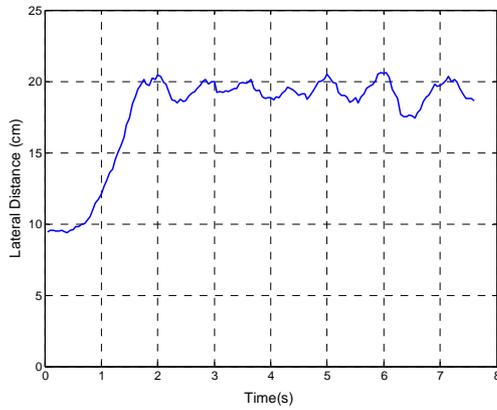


Figure 6-11a

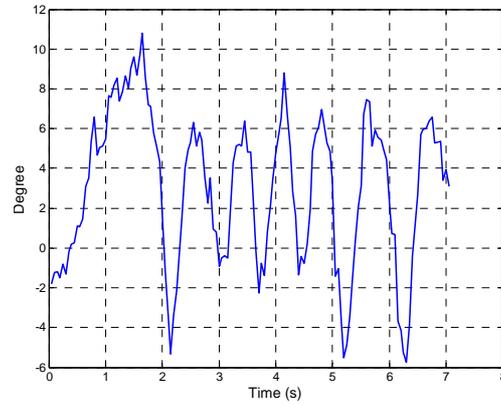


Figure 6-11b

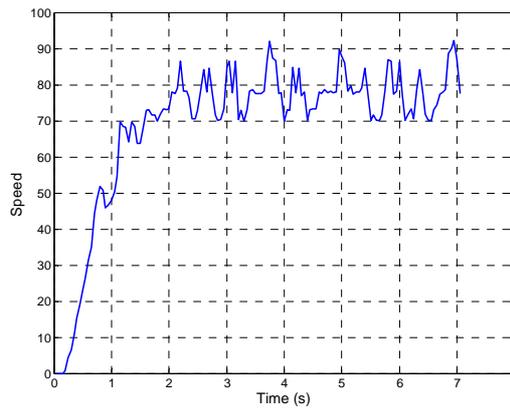


Figure 6-11c

Figure 6-11: TRD fuzzy controller experimental result with speed= 78cms^{-1} .

From the experimental results presented in Sections 6.4.1 and 6.4.2, the fuzzy controller optimized either by the FRC or the TRD encoding scheme demonstrate similar performance. The results show that both fuzzy controllers have better performance at speed 42cms^{-1} and 60cms^{-1} . Both FRC and TRD fuzzy controllers show bounded angle and lateral displacement oscillation result at the steady state with

speed 78cms^{-1} . Based on the 6 results shown in Figure 6-6 to Figure 6-11, the fuzzy controller optimized by GA with the proposed FRC does not affect controller stability in comparison to the TRD fuzzy controller optimized by conventional GA method.

6.5. Discussion

This section discusses the experimental controller performance described in sections 6-2 to 6-4 based on a performance index. Table 6-1 shows the Integral Time of Absolute Error (ITAE) and Integral Square Error (ISE) of the three controllers. The calculation of ITAE and ISE are based on equations (6.3) and (6.4) respectively.

$$ITAE = \sum_{i=m}^n \left(\left(|y_{ref} - y_i| + |\theta_i| \right) \cdot t_i \right) \quad (6.3)$$

$$ISE = \sum_{i=m}^n \left((y_{ref} - y_i)^2 + \theta_i^2 \right) \quad (6.4)$$

where y_{ref} , y_i , θ_i and t_i are lateral set-point, lateral displacement measured at i^{th} time step, vehicle angle evaluated at i^{th} time step and the time at i^{th} time step respectively.

Performance Index	Speed	FRC	TRD	NN
ITAE	42cms^{-1}	599.37	672.81	651.47
	60cms^{-1}	475.47	521.66	254.40
	78cms^{-1}	626.77	601.72	355.30
ISE (From Time 3s to 7s)	42cms^{-1}	30.29	32.98	33.29
	60cms^{-1}	40.67	35.56	20.24
	78cms^{-1}	95.66	89.981	29.32

Table 6-1: Controller performance index.

Consider the ITAE performance index; the error term is defined as the sum of absolute displacement error and absolute vehicle orientation. Therefore the system error is zero only if the vehicle located at lateral setpoint with zero vehicle orientation

with respect to the reference lane. Furthermore, the ITAE index associates the controller performance with time therefore the index penalizes slow response and large steady state error system.

For the ISE, the system error is evaluated in the same way of ITAE but using square error instead of absolute error. In addition the penalty factor of time is not used therefore the ISE only reflects the system error with respect to the set-point.

From Table 6-1, the ITAE values of the 3 controllers at speed 42cms^{-1} tend to be higher even though the experimental results (Section 6.3 and 6.4) show that the controller performs better at speed 42cms^{-1} which is obviously due to slow response under low speed operation. Comparing the ITAE index among the 3 controllers at the same speed of operation, NN controller in general is the most outstanding. The FRC controller performs the best at speed 42cms^{-1} . Referring to the fuzzy controller experimental result shown in Section 6.4.1 and Section 6.4.2, the bounded oscillation at the steady state gains large penalty from ITAE therefore the ITAE indexes are high for both fuzzy controllers.

In order to examine the controller performance at steady-state and regardless of the time factor, the ISE performance index is applied to evaluate each controller performance during the time period from time 3s to 7s. Referring to Table 6-1, NN controller also shows outstanding performance out of the 3 controllers especially at high speed operation. For the 2 fuzzy controllers, their ISE indexes are similar to each other.

The performance index evaluated on the two fuzzy controllers has a twofold

meaning. First as implied it shows both fuzzy controller performances on vehicle lateral control. Second the similarity in performance index of the FRC and TRD fuzzy controller shows that the newly proposed FRC encoding scheme in Chapter 5 does not affect the fuzzy system stability.

Considering the inputs of controller, the fuzzy controllers minimize tracking error based on lateral displacement (y) and vehicle orientation (θ). For the NN controller, the inputs are y , rate of y , θ , and rate of θ . The rate of change of the control variables help the NN controller to stabilize the vehicle and thus its performance is superior to the 2 fuzzy controllers.

6.6. Conclusions

NN controller proposed in Chapter 4 and Fuzzy controllers suggested in Chapter 5 are evaluated experimentally with different speed of operation in this Chapter. The experimental results show that both Fused Neural Network controller and fuzzy controller are able to control the under the variation of speed.

The performance index ITAE and ISE of the experimental results have been evaluated for comparison. Among the three controllers studied in this Chapter, the Fused Neural Network controller has outstanding performance in all speed of operation. In the rest of the thesis the Fused Neural Network controller is selected as the vehicle lateral controller for experimental studies in the following Chapters.

In this chapter, the focus was concentrated on steering (lane keeping) aspect of vehicle lateral control. However, in addition to lane keeping, lane changing is another

challenging topic on vehicle lateral control. Autonomous vehicles with capability of lane changing would enhance the flexibility of vehicle navigation and coordination. In the next chapter, a lane change algorithm will be presented to handle this challenging task.

Chapter 7. Lane Change Algorithm with Virtual Curvature and Lane Change Abortion Analysis

7.1. Introduction

Vehicle lateral control, including lane keeping as well as lane change maneuver, is a prime issue in the design of autonomous vehicles. Lane change maneuver consists of lane selection, smooth and safe travel to the desired lane while avoiding obstacles. To perform lateral control, detection of vehicle lateral information with respect to road is required such as vehicle orientation and lateral displacement. In general, the detection of vehicle lateral information is referred to the detection of reference lane. The method of lane detection engaged may introduce difficulties in lateral control application which due to the limitation of the detection method.

The lane detection method can be classified into two categories: infrastructure independent or infrastructure dependent. Infrastructure independent lane detection methodology is based on vision system mounted on the vehicle to capture image of the road for painted lane markers detection. Researchers Li *et al.* (2004) and Yim and Oh (2003) have proposed infrastructure independent lane detection algorithms using vision systems to detect public road painted lane markers for vehicle guidance. The advantage of this detection method is that it does not require modification of public road. However, the detection system robustness to weather condition is a challenging task.

However, the infrastructure dependent type lane detection method requires installation of lane markers that can be recognized by non-vision sensor, frequency selective strips (FSS)

by radar detection or magnetic markers, on the road as reference lane. Farkas *et al.* (1997) and Hatipoglu *et al.* (2003) introduced forward looking radar to measure vehicle lateral position based on the sensing energy from FSS. Zhang and Parsons (1990) designed magnetic marker system to measure the lateral position based on the magnetic field strength. This detection method is robust to weather condition, but this reference system requires heavy infrastructure modification.

During lane changing, vehicle travels from one lane to another which also implies that the reference of the lateral control system is also changed from one lane to another. For vision based system, the reference signal is extracted from captured image and the neighborhood lane is visible in the image, thus the problem of changing reference can be alleviated by image processing algorithm. However, for infrastructure dependent type reference systems, the sensors are not able to detect the markers on neighborhood lane thus this problem should be tackled by some ad hoc scheme or method.

To tackle the problem of reference lane transition in the systems with infrastructure dependent type during lane change, Tan *et al.* (1998) reported results on infrastructure supported lane changes schemes and dead reckoning schemes. The infrastructure supported lane changes method installed extra markers to mark a path between adjacent lanes to provide continuous lateral position information. However this scheme imposes limitations on the location of lane transition. The dead reckoning scheme did not require the installation of extra markers. The method assists lane change by using internal (soft) sensors to estimate vehicle current lateral position. However, there is a dead-zone period during lane transition, since no extra markers provide lateral information for feedback control. Therefore the lane change process is actually an open loop problem with respect to continuously measurable lateral position in the infrastructure supported lane changes schemes. Since there is no

limitation in lane transition location, it offers higher flexibility. Various algorithms have been proposed to tackle the open loop problem.

Hatipoglu *et al.* (2003) proposed a reference lateral jerk signal calculated based on the desired lateral displacement and maximum lateral jerk. The reference yaw rate signal guides the vehicle to adjacent lane in a closed-loop fashion by switching from lane keeping controller to sliding mode yaw rate follower. PATH researchers Chee and Tomizuka (1997) propose a trapezoidal lateral acceleration profile for lane change. They first implemented a tracking controller in experiment by converting the acceleration profile to a virtual reference trajectory. Then, they estimated the vehicle position by a state estimator. Next, they designed a unified yaw rate lateral tracking controller to handle both lane keeping and lane changing tasks to alleviate the controller switching problem. Shamir (2004) proposed an optimal trajectory planning which based on minimizing total kinetic energy and “minimum-jerk trajectory” for overtaking slower moving vehicle. Papadimitriou and Tomizuka (2003) proposed a fast algebraic computation algorithm to design an obstacle avoidance trajectory by using polynomials to represent lateral and longitudinal position.

The aforementioned existing works focus on the determination of lane change trajectory and follow the trajectory by tracking controllers. However, the main objective of the proposed algorithm is introducing a concept of virtual curvature for lane changing. From the autonomous vehicle point of view, the virtual curvature transforms the physical lane and guides the vehicle to adjacent lane.

The suggested algorithm incorporates virtual road curvature with bicycle model for vehicle lane change guidance. The virtual road curvature, does not physically exist, is a user assigned radius of a curved lane changing path which connects the current lane to the

adjacent lane. The lane changing path guidance is achieved by assigning a virtual road curvature to the bicycle model to transform existing physical reference path curvature to the desired lane change path curvature. This transforming effect is accomplished by the inherent property of the bicycle model. The method is inspired by the observation that any change in the road curvature affects the vehicle lateral dynamics (this can be verified by equation (7.1) in Section 7.3.2). The virtual road curvature is parameterized by road curvature and time.

The lane change capability of autonomous vehicle offers flexibility in vehicle navigation, coordination and obstacle avoidance. However, during lane changing the merging vehicle should cross lanes which imply that the vehicle should consider obstacles or vehicles on both lanes that may inference the vehicle safety. To encounter danger situation during lane changing, the possible actions to avoid collision are steering, braking or both. In the literatures on the safety of lane changing, the determination of longitudinal safety spacing and safety strategies have been studied.

Jula *et al.* (2000) suggested Minimum Longitudinal Safety Spacing (MLSS) as the minimum spacing to initiate lane change and established lane change safety region based on the MLSS. The study also proposed a specific acceleration strategy during lane change to enlarge the safety region. Kanaris *et al.* (2001) presented an algorithm for calculation Minimum Safe Spacing for Lane Changing (MSSLC). The calculation of MSSLC considers the vehicle braking limit during lane change for collision avoidance. The MSSLC ensures the merging vehicle have enough spacing to perform deceleration when vehicle enters emergency situation during lane change.

Jeich and Lin (2005) proposed a cascade fuzzy controller for car following and lane changing collision prevention. The proposed system handles danger situation during lane

change by using both emergency braking and abortion of lane change. However, the details of the abortion process have not shown.

In the literature, studies on abortion trajectory and the associated safety region are not explored. The abortion trajectory analysis may provide valuable information for vehicle coordination during lane change. In this chapter, the lane change abortion trajectory analysis is conducted and it evaluates the total abortion distance and collision free abortion point (P_{cf}). The P_{cf} is defined as the point of abortion on the lane change trajectory that guarantees no collision occurs. The general P_{cf} under different speed of operation and lateral acceleration limit will be presented in this chapter.

The chapter is organized as below: Section 7.2 defines the problem of lane changing in automated highway. Section 7.3 gives essential background information and the concept of the proposed lane change algorithm. Section 7.4 illustrates the algorithm design and the scheme for implementing the proposed algorithm. Section 7.5 shows the determination of abortion trajectory. Section 7.6 analyzes the abortion trajectory to formulate the safety abortion region and collision free abortion point. Section 7.7 shows both simulation and experimental results. At last, Section 7.8 concludes this chapter.

7.2. Lane change problem

This section states the assumption on the vehicle system and defines the lane change problem. Assuming the Lane detection method of the vehicle is infrastructure dependent type therefore specific reference system is required on the road. Furthermore, dead-reckoning scheme is used during lane changing therefore no infrastructure support exists. The main problem in infrastructure non-supported lane change within an infrastructure dependent

reference system is that physical reference path does not exist and no measurable lateral feedback signal available during lane transition.

The lane change problem is characterized as initiating, on demand, a path following routine along a specified lateral path from the current lane to the adjacent lane while keeping a desired longitudinal speed and aligning the vehicle at the end of the maneuver such that the lane keeping task can be resumed in a smooth and safe manner. Obstacle detection and avoidance during lane change is critical to vehicle safety, however it is not in the scope of this thesis therefore the obstacle detection scheme is not mentioned.

To solve the lane change problem with the above assumed vehicle reference system, one has to overcome the difficulty on measuring lateral position during lane changing. Since lateral information is not available (actually the information is not correct) during lane transition, the problem is handled in open-loop with respect to the lateral position information. The concept of Virtual curvature method is introduced to tackle this problem and the details will be presented in the later sections.

7.3. Road Profile and Bicycle Model

This section introduces the background information of the proposed lane change algorithm on road profile and bicycle model. The proposed algorithm is inspired by the relationship between the road curvature (road profile) and the vehicle lateral dynamics (bicycle model). In the coming 2 sections, the formation of road profile and its relationship with bicycle model will be discussed.

7.3.1. Road profile

The road profile describes the shape of highways by using the curvature of the road. Highway consists of both straight and curved sections and the path can be represented as an integration of circular arcs and straight lines. Assuming the radius of the circular arcs is R_{ref} . For the straight road segment, the radius is infinity ($R_{ref}=\infty$). For the road curvature (ρ_{ref}), it is defined by inverse of the radius ($\rho_{ref}=1/R_{ref}$). Furthermore, the curvature is defined to be positive for left movements and negative for right movements. Therefore, different road profiles can be established with different values of ρ_{ref} .

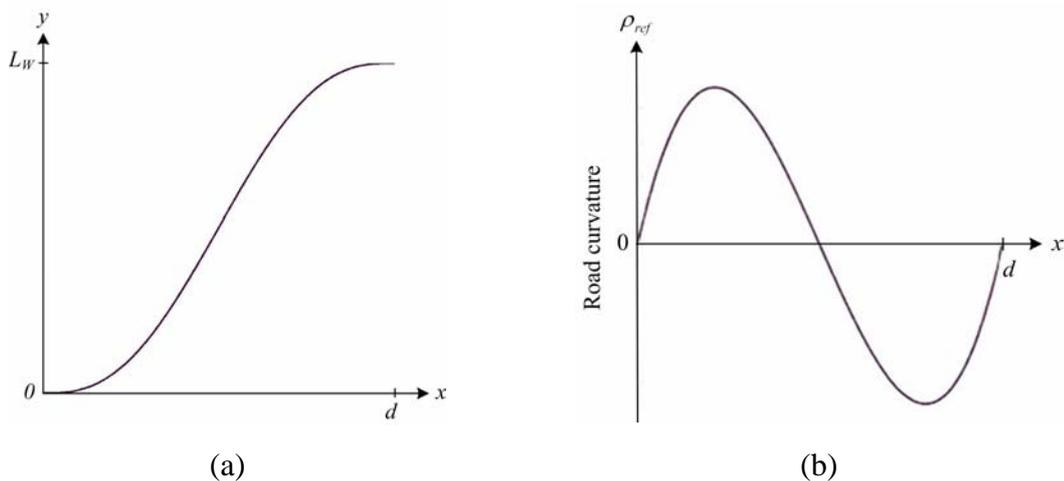


Figure 7-1 Typical road and road curvature profile

To illustrate the idea of road profile, Figure 7-1 show a typical road and road curvature profile of the so-called “lazy-S” formation for lane change. This curve produces a transition path between parallel lanes with L_w as the lane width. The lateral position at original lane and the lane width are assumed as zero and L_w respectively. In Figure 7-1(a), y and x is the vehicle lateral position and longitudinal position respectively. The road starts at $y=0$ and terminate at $y=L_w$. The longitudinal displacement of the road from $y=0$ to $y=L_w$ is d . The road

trajectory shows that it starts with left cornering and gradually changes to right cornering.

Figure 7-1(b) shows that the curvature of the “lazy-S” road. The curvature contains both positive value and negative value which implies the change of cornering direction of the road. The curvature is continuous and it starts and finishes with zero curvature so that the curve connects the two parallel lanes smoothly.

7.3.2. Bicycle model

To study the automatic steering of an autonomous car, a vehicle steering model including velocity, vehicle heading and lateral distance from the sensors to a reference path is required. The bicycle model is well known and is used in most applications of lateral control design including the works presented by Guldner *et al.* (1997), Ackermann (1993) and Tan *et al.* (1998).

The bicycle model (7.1) describes the lateral dynamics with respect to steering angle and the reference road curvature. Where σ_f is steering angle, ρ_{ref} is reference road curvature, a_{ij} and b_{ij} are vehicle parameters, y_f and y_r are lateral displacement of vehicle from front and rear sensor to reference lane respectively. The sign notation of both σ_f and ρ_{ref} are the same, therefore defined as positive and negative for left and right cornering respectively.

$$\frac{d}{dt} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a_{21} & a_{22} & -a_{21} & a_{24} \\ 0 & 0 & 0 & 1 \\ a_{41} & a_{42} & -a_{41} & a_{44} \end{bmatrix} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ 0 & b_{32} \\ b_{41} & b_{42} \end{bmatrix} \begin{bmatrix} \sigma_f \\ \rho_{ref} \end{bmatrix} \quad (7.1)$$

In vehicle lateral control simulation, lateral controller output is σ_f which used to control

the steering angle of the bicycle model. However, the ρ_{ref} is used to describe the curvature of the reference lane (i.e. straight or curved). The bicycle model gives vehicle lateral position with respect to the reference lane based on these 2 input parameters

The proposed algorithm in this chapter will assign a virtual reference curvature to the model so that transforms the shape of the physical reference system to a virtual reference system. The shape of the virtual reference system is a user defined trajectory for lane changing.

7.3.3. Concept of virtual road curvature

The above sections of road profile and bicycle model introduce the formation of road curvature profile and the relation between vehicle lateral dynamics and road curvature respectively. The concept of virtual road curvature is inspired by the simulation study of lane keeping controller performance on curved highway. To simulate a vehicle riding on a desired curved highway, road profile of the highway will be constructed and assigned to the bicycle model in the simulation. The bicycle model calculates the vehicle lateral position with respect to the curved highway. Then the lane keeping controller adjusts the vehicle steering angle according to the bicycle model output so that the vehicle follows the curved highway.

The property of the above mentioned simulation is used for lane changing by scale down the highway curvature to lane transition curvature. The profile of the transition lane connects current lane to adjacent lane. Since this transition lane does not physically exist, we name it as virtual lane, and its corresponding curvature (ρ_{ref}) as virtual road curvature. During lane change, the virtual road curvature profile is used as the bicycle model input instead of the physical lane curvature profile so that the lane keeping controller maneuver the vehicle towards the adjacent lane. The detail of methodology on curvature determination and

controller coordination will be described in the Section 7.4.2.

7.4. Lane Change Algorithm

7.4.1. Illustration of the lane change algorithm

The lane change maneuver was divided into 3 Stages and Figure 7-2 illustrates the concept of the proposed algorithm on a straight double lane highway. Since the highway is straight, the physical road curvature is zero in all Stages. In Stage I and III, The merging vehicle (M) performs lane keeping at Lane A (original lane) and lane B (Destination lane) by measuring lateral distance from physical reference lane.

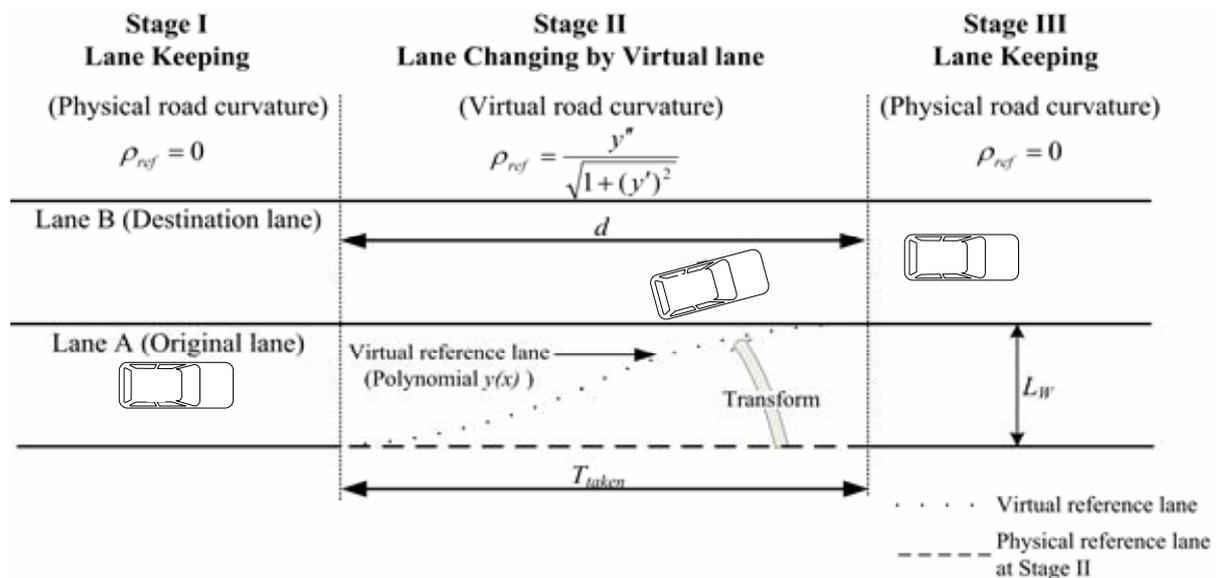


Figure 7-2: Illustration of lane change algorithm

In Stage II, a virtual reference lane is constructed (dotted line) for lane changing. The virtual reference lane is formed according to desired lane change longitudinal distance (d)

and the lane width (L_w) by a polynomial equation (Section 7.4.2 (7.2)). The curvature of the polynomial formulated is then calculated by equation (7.3) shown in Section 7.4.2. The sign of the curvature is designed according to the definition made in Section 7.3.1, therefore positive for left movements and negative for right movements respectively. The aforementioned virtual reference lane and its curvature guide the vehicle to Lane B (Destination lane) by using the property of bicycle model. The calculation and evaluation of polynomial and its curvature will be described in the next Section.

During Stage II, bicycle model estimates the vehicle lateral position based on the virtual reference lane curvature instead of the curvature of physical lane (dashed line). Since the estimated vehicle lateral position is with respect to virtual reference lane instead of the physical lane, the physical lane is transformed to a virtual reference lane from the vehicle point of view. Lane keeping controller follows the virtual reference lane by using the estimated lateral position as input and guides the vehicle to Lane B. The time duration of using bicycle model for lateral position estimation is T_{taken} . The vehicle speed (V) and d determine the length of T_{taken} and the calculation will be shown in the coming Section.

7.4.2. Virtual road trajectory and curvature determination

To achieve the lane change algorithm presented in at the above section, a virtual reference lane should be constructed and a lane change scheme should be designed. This Section presents the determination of the virtual reference lane and its curvature. Then the lane change scheme with virtual lane method will be described.

There are several types of lane change trajectories proposed in the literatures. Different types of trajectories have different characteristic in the curvature profile. The design of lane change trajectory should consider passenger's ride comfort. Caywood *et. al.* (1977) proposed

the transient lateral acceleration and the lateral jerk should not exceed 0.12 g and 0.24 g/s, respectively for ride comfort.

There are 3 candidates for the virtual trajectories design (1) Circular trajectory, (2) Cosine trajectory, and (3) 5th order polynomial trajectory. Chee and Tomizuka (1994) have presented the characteristic of these 3 trajectories and are summarized as follow. For those reader interested in the proof of trajectories characteristic, please refer to the report by Chee and Tomizuka (1994).

In the following description L_w , V , and a_{\max} are referred to the lane width, longitudinal speed, and maximum allowable lateral acceleration respectively.

(1) Circular trajectory:

The trajectory consists of 2 circular arcs with a curvature equal in magnitude but opposite in direction, and a straight segment between the two arcs. The lengths of the straight segment have proven to be zero to minimize the transition time. Therefore, the circular trajectory is effectively consists of 2 arcs only. The radius of curvature (R) of the 2 circular arcs can be expressed by equation (7.2). The time spent (t_c) for completing the circular arc depends on the vehicle speed, lane width, and the radius of curvature. Equation (7.3) calculates the time spent (t_c) and the total transition time (T) is $2t_c$. Equation (7.4) shows the circular trajectory equations under different time period.

$$R = \frac{V^2}{a_{\max}} \quad (7.2)$$

$$t_c = \frac{R}{V} \cos^{-1} \left(1 - \frac{L_w}{2R} \right) \quad (7.3)$$

$$y(t) = R \left(1 - \cos \frac{V}{R} t \right) \quad , 0 \leq t \leq t_c \quad (7.4)$$

$$R \left\{ 1 + \cos \frac{V}{R} (T - t) - 2 \cos \frac{V}{R} t_c \right\} \quad , t_c \leq t \leq T$$

The radius of curvature calculated by equation (7.2) limits the lateral acceleration to maximum allowable level. Therefore this trajectory satisfies the requirement on lateral acceleration. However, the lateral jerk at the time of $t = 0$, t_c , and T is infinity. It is due the discontinuous curvature and thus this trajectory is not desirable for lane changing.

(2) Cosine trajectory

The cosine trajectory is an approximation of the circular trajectory. Equations (7.5) and (7.6) show the cosine trajectory equation and its second derivative respectively. The α is selected by equation (7.7) to satisfy the lateral acceleration constraint.

$$y(t) = \frac{L_w}{2} (1 - \cos(\alpha t)) \quad , 0 \leq \alpha t \leq \pi \quad (7.5)$$

$$\ddot{y}(t) = \frac{L_w \alpha^2}{2} (\cos(\alpha t)) \quad (7.6)$$

$$\alpha = \sqrt{\frac{2a_{\max}}{L_w}} \quad (7.6)$$

$$T_{\cos} = \pi \sqrt{\frac{L_w}{2a_{\max}}} \quad (7.7)$$

The total transition time for cosine trajectory (T_{\cos}) is defined in equation (7.7). The cosine approximation alleviates the problem of infinite lateral jerk at the middle of the trajectory that occur at $t = t_c$. However, the second derivative of cosine trajectory still

discontinuous at $t = 0$ and T_{cos} .

(3) *5th order polynomial trajectory*

The 5th order polynomial shown in equation (7.8) proposed by Nelson for lane change maneuvers. The variables in equation (7.8), y , x , L_w and d are the lateral position, longitudinal distance with respect to the starting point of the trajectory, lane width and the terminal point of the trajectory respectively.

$$y(x) = L_w \left\{ 10 \left(\frac{x}{d} \right)^3 - 15 \left(\frac{x}{d} \right)^4 + 6 \left(\frac{x}{d} \right)^5 \right\} \quad (7.8)$$

$$\kappa = \frac{\frac{d^2 y}{dx^2}}{\sqrt{1 + \left(\frac{dy}{dx} \right)^2}} \quad (7.9)$$

In order to find the polynomial trajectory that satisfies the lateral acceleration limit, the point of maximum curvature (x_m) on the trajectory is found. The ratio of x_m to the terminal point (d) is also evaluated as 0.2113 (i.e. $x_m/d = 0.2113$). The equation (7.10) relates the lane change distance d (i.e. the point of trajectory termination) to maximum allowable lateral acceleration (a_{1max}).

$$d = V \sqrt{\frac{L_w}{a_{1max}} \left\{ 60 \left(\frac{x_m}{d} \right) - 180 \left(\frac{x_m}{d} \right)^2 + 120 \left(\frac{x_m}{d} \right)^3 \right\}} \quad (7.10)$$

$$\frac{d}{x_m=0.2113} = V \sqrt{\frac{L_w}{a_{max}}} (5.774)$$

The 5th order polynomial offers a closed form of trajectory presentation and continuous

curvature which satisfy both constraints on lateral acceleration and jerk. Thus consider the 3 candidates described above, the 5th order polynomial is selected as the reference trajectory for virtual curvature determination.

To implement the polynomial equation (7.8) and the curvature equation (7.9), the equation (7.8) is modified to equation (7.11) as a function of time by substituting the vehicle longitudinal distance (x) with velocity times time (i.e. Vt). The equation (7.9) is simplified to the second derivative of the polynomial trajectory. Equation (7.12) shows the express of the simplified version of (7.9) that expressed as a function of Vt . The modifications of both equations are based on the works by Chee and Tomizuka (1994).

$$y(t) = L_w \left\{ 10 \left(\frac{Vt}{d} \right)^3 - 15 \left(\frac{Vt}{d} \right)^4 + 6 \left(\frac{Vt}{d} \right)^5 \right\} \quad (7.11)$$

$$\kappa = y''(t) = \frac{L_w}{d^2} \left\{ 60 \left(\frac{Vt}{d} \right) - 180 \left(\frac{Vt}{d} \right)^2 + 120 \left(\frac{Vt}{d} \right)^3 \right\} \quad (7.12)$$

The total transition time (T_{taken}) of the lane change trajectory is obtained by

$$T_{taken} = \frac{d}{V} \quad (7.13)$$

The lane width (L_w) and maximum lateral acceleration (a_{1max}) are substituted into (7.10) to find d . Then d is substituted to equation (7.12) for setting up the function of polynomial curvature (κ). When the function of curvature profile of the virtual reference lane is ready, the lane change maneuver can be started through the lane change scheme described below.

7.4.3. Lane change scheme

The proposed lane change scheme is a unified controller approach, using the same

lateral controller in both lane keeping and lane changing. Owing to the difficulties described in Section 7.2, hardware sensors are not feasible to measure lateral position during lane change. The lane change scheme is based on the property of bicycle model to establish a soft sensor to estimate the vehicle lateral position during lane change according to the assigned virtual road curvature and steering angle. The flow of the lane change scheme is described as follows.

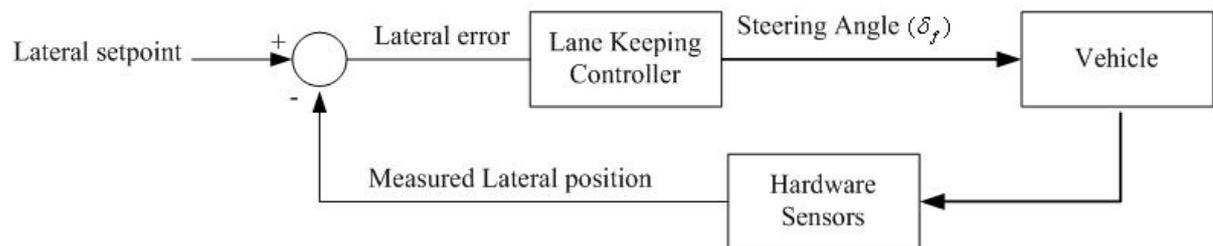


Figure 7-3 : Lane keeping scheme at Stage I and III.

Referring to Figure 7-2, the proposed lane change algorithm is divided into three Stages. In Stage I, vehicle performs lane keeping at the original lane with the control block diagram shown in Figure 7-3. In the figure, hardware lateral sensor measures vehicle lateral displacement from the reference lane. Then the lane keeping controller minimizes the lateral error based on the measured lateral information. The full architecture of the lateral controller is described in Chapter 4.

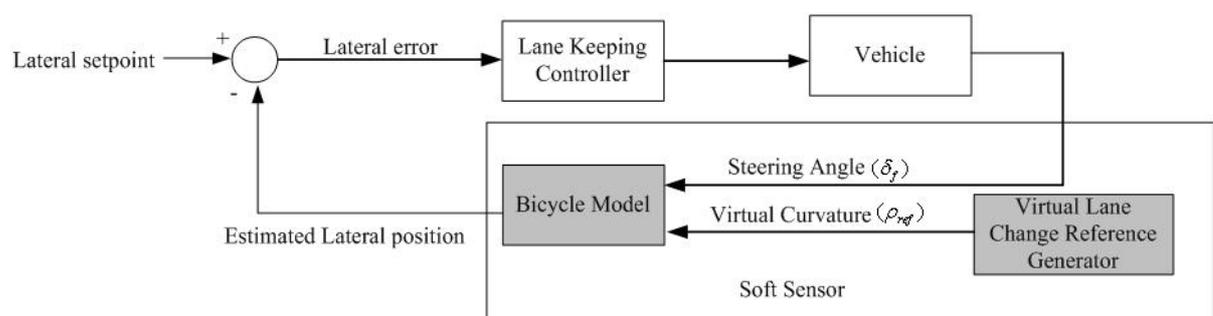


Figure 7-4: Lane change scheme at Stage II.

In Stage II, vehicle starts lane changing with the proposed lane change scheme as shown in Figure 7-4. In Figure 7-4, the hardware sensor is replaced by soft sensor and the

lateral controller remains the same as lane keeping therefore there is no switching of controller. The soft sensor calculates virtual curvature (ρ_{ref}) and measures actual steering (σ_f) from the steering wheel. The virtual curvature is generated from the virtual lane reference generator using equations (7.11) and (7.12). The ρ_{ref} and σ_f are then feed into bicycle model for lateral position estimation. Then the lane keeping controller maintains the vehicle lateral position with the estimated lateral displacement from the bicycle model.

In Stage III, after the lane changing from Lane A to B shown in Figure 7-2, vehicle resumes lane keeping in the destination lane with the control block diagram shown in Figure 7-3. Then the lateral controller maintains the vehicle lateral position according to sensor measurement in Lane B.

The lane change algorithm and scheme is summarized by the following steps:

1. Obtain the lane width (L_w) and desired lane change longitudinal distance (d).
Formulate the virtual reference lane by substituting L_w and d into equation (7.11) of the lane change polynomial ($y(x)$).
2. Differentiate $y(x)$ and substitute it into equation (7.12) to find the curvature of the polynomial (κ). Establish the virtual road curvature profile by putting $\rho_{ref} = \kappa$ for implementation.
3. Perform lane keeping at original lane that the lateral controller inputs are physical sensors measurement.
4. Assign the calculated virtual road curvature and the actual steering angle to bicycle model for lateral position estimation during lane change at Stage II.
5. Use the lateral position estimated by bicycle model with respect to virtual reference lane as the lateral controller inputs in Stage II (Figure 7-3). The lateral

controller maneuvers the vehicle to adjacent lane by using the same controller as lane keeping in Stage I.

6. Resume lane keeping at the Stage III and uses physical measurement as the lateral controller inputs.

The implementation of the proposed lane change algorithm and scheme will be presented later in Section 7.7. In the coming section, the lane change abortion trajectory is discussed.

7.5. Lane Change Abortion - Collision Free Abortion Point Analysis

The lane change maneuver offers flexibility in vehicle navigation, coordination and obstacle avoidance. However, during lane changing the merging vehicle should cross lanes which imply that the vehicle should consider obstacles or vehicles on both lanes that may inference the vehicle safety. To encounter danger situation during lane changing, the typical actions are abortion of lane change or performing emergency braking. In general, emergency braking is the last alternative if avoidance maneuvers can be executed. For the abortion of lane change, the vehicle should maneuvers back to the original lane to avoid danger and resume lane keeping at original lane.

In this section, the determination of collision free abortion point will be discussed. The collision fee abortion point defined here refers to the point that vehicle initiates abortion would not colloid with obstacles at the adjacent lane. To my best knowledge, this is the first analysis to be documented for the evaluation of collision free abortion point. The details of the analysis are shown below.

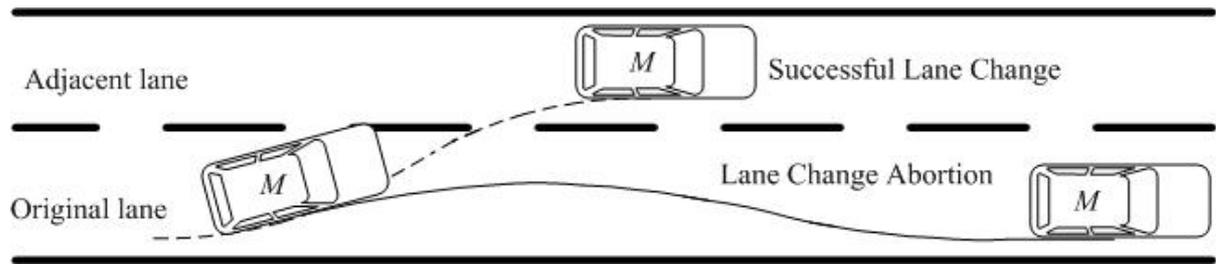


Figure 7-5: Lane change and lane change abortion.

Figure 7-5 illustrates the trajectory of the merging vehicle (M) performing lane change or lane change abortion. In the figure, vehicle M initiates lane change maneuver at its original lane and terminate at the adjacent lane which follows the dotted trajectory to complete lane change successfully. However if risky situation is encountered during lane change, lane change abortion will be activated which follows the solid trajectory to escape from danger. To analyze the abortion process, the abortion trajectory should be obtained explicitly and it will be discussed as follow.

The longitudinal distance between the abortion point of lane change and the position resume lane keeping is defined as the abortion distance (d_a). The length of d_a depends on varies of parameters such as lateral position of the abortion point, vehicle speed, lane width, and the allowable lateral acceleration. In general, the abortion trajectory can be regarded as the mirror image of the lane change trajectory. On the lane change trajectory, the vehicle orientation at both initial and termination points are zero so that the vehicle maneuvers to the adjacent lane smoothly. However during lane changing, the vehicle orientation at the abortion point is not zero which implies that the initial condition of the abortion trajectory is not equivalent to the lane change trajectory. Owing to the difference in the initial condition, mirror image of the lane change trajectory cannot be directly applied to determine the abortion trajectory.

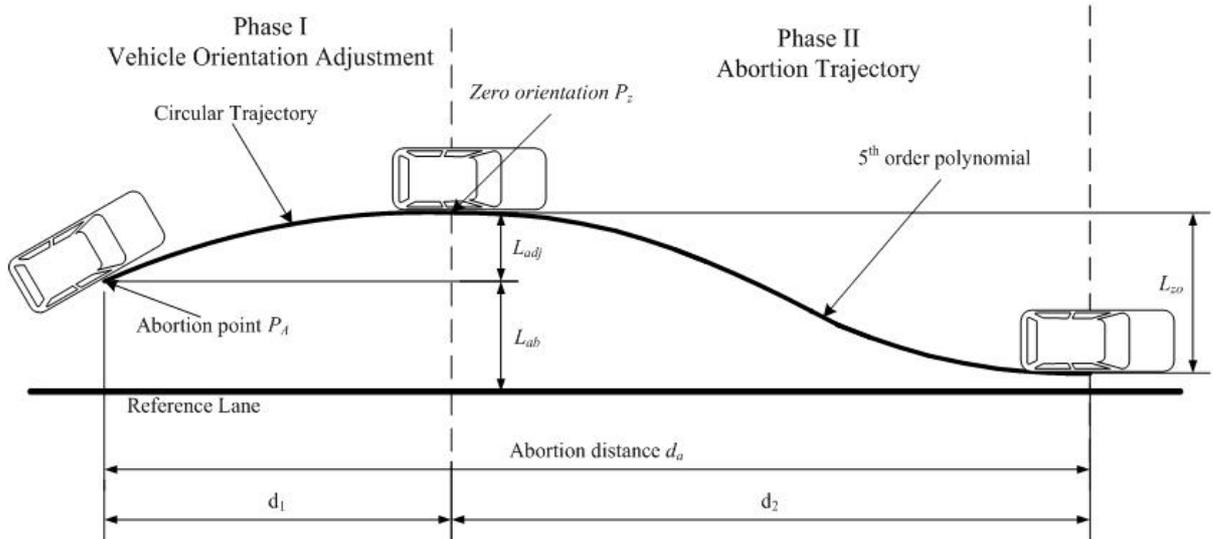


Figure 7-6: Lane change abortion process.

To obtain the abortion trajectory and the associated longitudinal abortion distance, the process of lane change abortion is divided into two Phases as shown in Figure 7-6. In Phase I, the vehicle adjusts its orientation to zero from the abortion point (P_A) to zero orientation point (P_z). Then In the Phase II, the vehicle maneuvers back to the original lane from P_z with lateral displacement L_{zo} . The total longitudinal distance required of the 2 Phases mentioned above is the required abortion distance (d_a).

The collision free abortion point is determined from the maximum lateral position of the abortion trajectory. From the figure, it can be seen that the maximum lateral position of the abortion trajectory is located at P_z which depends on the lateral displacement required for orientation adjustment (L_{adj}). The followings show the trajectories determination of the two Phases described above.

7.5.1. Phase I – Vehicle Orientation adjustment

To determine the trajectory in Phase I, consider a circle with equation (C_1) as shown in

(7.14) with centre located at x_0 and y_0 with radius R . The variables x and y are horizontal and vertical position respectively. Further consider the tangent lines to the circle C_1 , the angle of the tangent lines is directly proportional to the first derivative of C_1 with respect to x . Equation (7.15) shows the first derivative of C_1 (y') which also shows that y' equals to zero when $x = x_0$. Therefore it shows that the angle of tangent line at $x = x_0$ is zero.

$$C_1 : (y - y_0)^2 + (x - x_0)^2 = R^2 \quad (7.14)$$

$$y = \sqrt{R^2 - (x - x_0)^2} + y_0$$

$$y' = \frac{dy}{dx} = \frac{2(x - x_0)}{\sqrt{R^2 - (x - x_0)^2}} \quad \begin{cases} y' > 0, & x > x_0 \\ y' < 0, & x < x_0 \\ y' = 0, & x = x_0 \end{cases} \quad (7.15)$$

Assuming the vehicle follows the circular trajectory and the vehicle orientation is equal to the angle of the tangent lines. Then it can be further assumed that the vehicle orientation is zero when its horizontal position equals to the centre of the circle. Therefore the vehicle orientation can be adjusted to zero by using a circular trajectory that connects the point of abortion and the point of zero orientation.

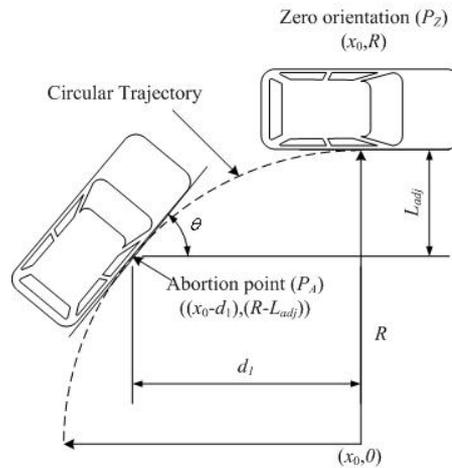


Figure 7-7: Abortion distance required in the first section of abortion.

Figure 7-7 illustrates abortion distance (d_1) required in the Phase I of the abortion process. The circular trajectory shows in dashed line. The vehicle orientation at abortion point is θ . The distance d_1 effectively is the horizontal distance between the abortion point (P_A) and the zero orientation point (P_Z). Let the vertical distance between P_A and P_Z be L_{adj} and the center of circular trajectory is located at $(x_0,0)$ with radius R . Then the position of the P_A and P_Z can be evaluated as $(x_0-d_1, R-L_{adj})$ and (x_0,R) respectively.

$$R = \frac{V^2}{a2_{\max}} \quad (7.16)$$

$$\tan \theta = \frac{2d_1}{\sqrt{R^2 - d_1^2}} \quad (7.17)$$

$$d_1 = \frac{R \tan \theta}{\sqrt{4 + (\tan \theta)^2}} = \frac{v^2 \tan \theta}{a2_{\max} \sqrt{4 + (\tan \theta)^2}}$$

$$L_{adj}^2 - 2RL_{adj} + d_1^2 = 0 \quad (7.18)$$

$$L_{adj} = R - \frac{\sqrt{4R^2 - 4d_1^2}}{2}$$

To formulate the circular trajectory for vehicle orientation adjustment, the variables R , d_1 and L_{adj} shown in the figure should be obtained. (7.16) shows the equation for determine the radius of curvature R which is subject to the vehicle speed (V) and the maximum allowable lateral acceleration ($a2_{\max}$). For the longitudinal abortion distance (d_1) calculation, (7.15) is modified to (7.17) by replacing the (x_1-x_0) and y' by d_1 and $\tan\theta$ respectively where θ is vehicle orientation at abortion point.

When the value of R and d_1 are obtained, the L_{adj} can be evaluated in terms of R and d_1 by substituting the positions of P_A and P_Z into the circle equation C_1 with center $(x_0, 0)$ and solve simultaneously to form equation (7.18). The value of L_{adj} associates with the maximum

lateral position (L_{max}) in the abortion process. L_{max} can be calculated by adding L_{adj} with the vehicle lateral position at the abortion point (L_{ab} refer to Figure 7-6).

7.5.2. Phase II – Abortion Trajectory

The abortion trajectory considered in the Phase II is the path that connects the point of zero orientation (P_z) to the original lane. As the aforementioned discussion, the abortion trajectory can be expressed as the mirror image of the lane change trajectory provided that the initial condition is the same. Since the initial vehicle orientation has been adjusted to zero in Phase I, the lane change trajectory equation can be employed to determine the abortion trajectory.

The abortion trajectory equation employed is the same as the 5th order polynomial discussed in Section 7.4.2. Equation (7.19) shows the abortion distance (d_2) required in the Phase II of the abortion process. In equation (7.19), L_{zo} (which shown in Figure 7-6) is the lateral distance between the point of P_Z and the target position at the original lane.

$$d_2 = V \sqrt{\frac{L_{zo}}{a2_{max}}} \quad (5.774) \quad (7.19)$$

To conclude the total abortion distance d_a required is equal to the sum of the distance determined in Phase I and Phase II of the abortion process which as shown in (7.20). The d_a is expressed in terms of the V , L_{zo} , and $a2_{max}$. Therefore the d_a can be calculated immediately when the abortion point is located.

$$d_a = d_1 + d_2$$

$$d_a = \frac{V^2 \tan \theta}{a2_{max} \sqrt{4 + (\tan \theta)^2}} + V \sqrt{\frac{L_{zo}}{a2_{max}}} \quad (5.774) \quad (7.207)$$

7.6. Abortion trajectory analysis

The lane change algorithm and the formation of lane change abortion trajectory have been presented in the previous sections. The abortion trajectory evaluated by circular and polynomial trajectories at the above sections provides two valuable information of the abortion process. First the abortion longitudinal distance is explicitly calculated. Second the maximum vehicle lateral position (L_{max}) in the abortion trajectory is calculated. The abortion trajectory analysis in this section introduces the collision free abortion point based on the abortion trajectory stated above.

Safety abortion region

The collision free abortion point (P_{cf}) is defined as the point of abortion on the lane change trajectory that guarantees no collision occurs with obstacles at adjacent lane. The P_{cf} is evaluated by investigating the abortion points which ensure the abortion trajectory does not violate the Safety abortion region (S_r). S_r is the lateral region that the merging vehicle will not collide with the vehicle at the adjacent lane. The determination of S_r is based on the assumption that the vehicle at the adjacent lane traveling at the lane center. Figure 7-8 illustrates the S_r with shaded region and it can be calculated by equation (7.21) where V_w and L_w are vehicle width and lane width respectively. Since the S_r is not occupied by other vehicles, collision free can be guaranteed if the merging vehicle M does not violate this region.

$$S_r = L_w + \left(\frac{L_w - V_w}{2} \right) \quad (7.21)$$

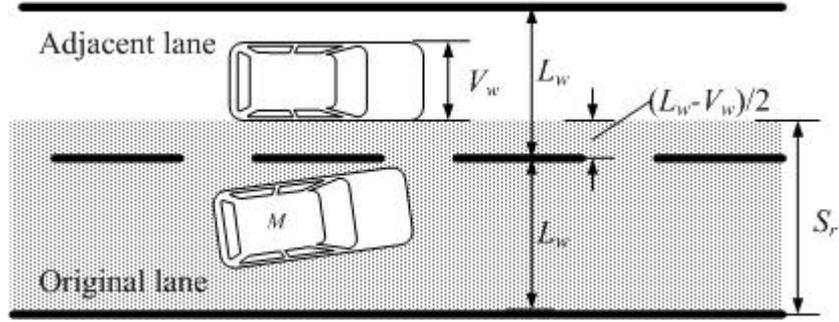


Figure 7-8: Safety region in lane change abortion.

Collision free abortion point

Once the S_r is defined as above, the P_{cf} can be established by calculating L_{max} of the abortion trajectory with respect to the abortion point. As stated in Section 7.5.1, the L_{max} is equal to sum of the abortion point lateral position (L_{ab}) and the lateral displacement (L_{adj}) required for orientation adjustment (i.e. $L_{max} = L_{ab} + L_{adj}$). The condition for collision free abortion can be established based on the S_r , L_{ab} , and L_{adj} as shown in (7.22). Therefore any abortion points and the associated L_{adj} satisfy (7.22) are P_{cf} .

$$S_r \geq L_{ab} + L_{adj} \quad (7.22)$$

Referring to Section 7.5.1, the calculation of L_{adj} is related to the vehicle speed (V), allowable lateral acceleration ($a_{2_{max}}$) and the vehicle orientation (θ) at the point of abortion as shown in (7.18). These 3 parameters are regarded as the initial condition of the abortion process. Since the abortion point initiates on the lane change trajectory, the initial condition can be generated from the lane change trajectory.

To generate initial condition of abortion point for P_{cf} evaluation from the lane change trajectory, the 5th order polynomial (7.8) and its first derivative (7.23) are used for L_{ab} and the vehicle orientation (θ) determination. Consider equation (7.8), $y(x)$ associate the vehicle

lateral position with respect to the longitudinal position x to $L_{ab}(x)$ by (7.23). The shape of $y(x)$ depends on V , L_w and d (d is associated with V , L_w , and $a1_{max}$ as shown in (7.10)). For (7.24), it is the slope ($\tan\theta$) of the lane change trajectory so that gives the vehicle orientation. Therefore by substituting x_{ab} as the longitudinal abortion point of the lane change process $L_{ab}(x_{ab})= y(x_{ab})$ and $\tan\theta = y'(x_{ab})$.

$$L_{ab}(x_{ab}) = y(x_{ab}) = L_w \left\{ 10 \left(\frac{x_{ab}}{d} \right)^3 - 15 \left(\frac{x_{ab}}{d} \right)^4 + 6 \left(\frac{x_{ab}}{d} \right)^5 \right\} \quad (7.23)$$

$$\tan \theta = y'(x_{ab}) = \frac{L_w}{d} \left(30 \left(\frac{x_{ab}}{d} \right)^2 - 60 \left(\frac{x_{ab}}{d} \right)^3 + 30 \left(\frac{x_{ab}}{d} \right)^4 \right) \quad (7.24)$$

For the determination of L_{adj} in the abortion process, equation (7.17) for abortion distance calculation is modified to (7.25) by substituting $\tan\theta = y'(x_{ab})$. Then equation (7.18) for L_{adj} determination is then modified to (7.26) as a function of abortion point position. Finally the condition for collision free abortion shown in (7.22) is changed to (7.27) as a function of longitudinal abortion point so that the P_{cf} can be found when the lane change trajectory is established.

$$d_1 = \frac{V^2 y'(x_{ab})}{a2_{max} \sqrt{4 + (y'(x_{ab}))^2}} \quad (7.25)$$

$$L_{adj}(x_{ab}) = \frac{V^2}{a2_{max}} - \frac{\sqrt{4 \left(\frac{V^2}{a2_{max}} \right)^2 - 4d_1^2}}{2} \quad (7.26)$$

$$S_r \geq L_{ab}(x_{ab}) + L_{adj}(x_{ab}) \quad (7.27)$$

7.6.1. Determination of collision free abortion point

The condition for collision free abortion point presented at the above section provides the foundation for analyzing the range of collision free abortion. In this section, the collision free abortion point under different speed and lateral acceleration requirements will be evaluated. From the evaluated P_{cf} , a generalized collision free abortion range will be presented.

In order to generalize the location of P_{cf} , the P_{cf} is represented by the percentage of longitudinal lane change distance completed. The procedure to evaluate the P_{cf} summarized as follows:

1. Calculates lane change distance d by (7.10) subject to V , L_w , and $a1_{max}$.
2. Calculates $L_{ab}(x_{ab})$ and $\tan\theta$ of all possible abortion points along the lane change trajectory by (7.23) and (7.24) respectively. (where $0 < x_{ab} < d$)
3. Evaluates abortion distance (d_1) corresponding to the abortion point x_{ab} by (7.25) subject to V and $a2_{max}$.
4. Obtains $L_{adj}(x_{ab})$ by (7.26) subject to V and $a2_{max}$.
5. Obtains the L_{max} with respect to the percentage of lane change completed $\left(\frac{x_{ab}}{d} \times 100\% \right)$ by $L_{max} = L_{ab}(x_{ab}) + L_{adj}(x_{ab})$.

The objective of the above procedure for P_{cf} calculation is evaluating all the possible abortion points on the lane change trajectory for collision free analysis. The P_{cf} calculation consists of 5 steps and it starts with establishing the lane change trajectory subject to V , L_w , and $a1_{max}$. When the lane change trajectory is established, step 2 to 5 is repeated with constant V and $a2_{max}$ until all the possible point is evaluated. In the rest of this section, the analysis on the range of P_{cf} under different speed and lateral acceleration limit will be shown.

The parameters setting for P_{cf} determination under different condition are summarized in Table 7-1. The Lane width (L_w), vehicle width (V_w), and the safety region (S_r) are 3.4m, 1.7m and 4.25m respectively. The $a1_{max}$ in lane change process is fixed at 0.2g. For the range of $a2_{max}$ in abortion process and vehicle speed are 0.1g to 0.5g and 20km/h to 80km/h respectively.

L_w	3.4m	$a1_{max}$	0.2g
V_w	1.7m	$a2_{max}$	0.1g to 0.5g
S_r	4.25m	V	20km/h to 80km/h

Table 7-1: Collision free abortion point analysis

Figure 7-9 shows the results on the maximum lateral position (L_{max}) of the abortion process against the percentage of lane change completed under different speed and allowable lateral acceleration ($a2_{max}$). The shaded region shown in the figures is the safety region S_r which assumed as 4.25m. To begin the discussion on the evaluations results, the relationship between the $a2_{max}$ and the abortion trajectories will be described. Since the trajectories in the abortion process are curved path, the $a2_{max}$ would limit the curvature of the path. The radius of curvature is proportional to speed but inversely proportional to $a2_{max}$ (as shown in equation (7.16)). Normally as speed increase, the radius increase and enlarge the L_{max} .

Figure 7-9a to Figure 7-9e show the results on L_{max} with $a2_{max} = 0.1g$ to 0.5g respectively. Figure 7-9a shows that the L_{max} are almost the same with different speed under $a2_{max} = 0.1g$. This result suggests that the effect of vehicle speed is suppressed by the large radius of curvature with $a2_{max} = 0.1g$. The range of collision free abortion is about 42% of the lane change.

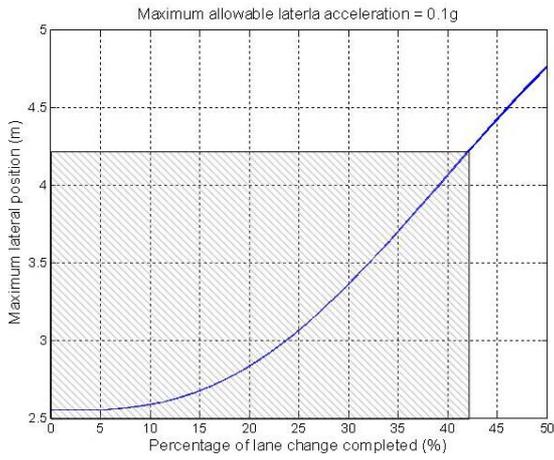


Figure 7-9a

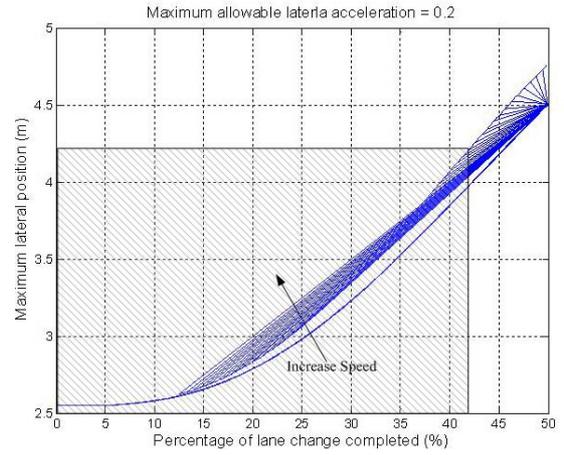


Figure 7-9b

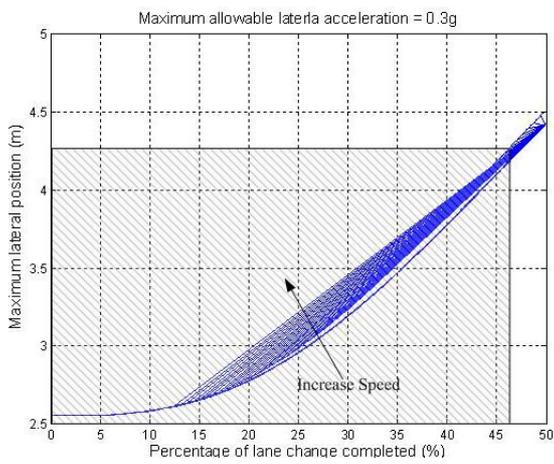


Figure 7-9c

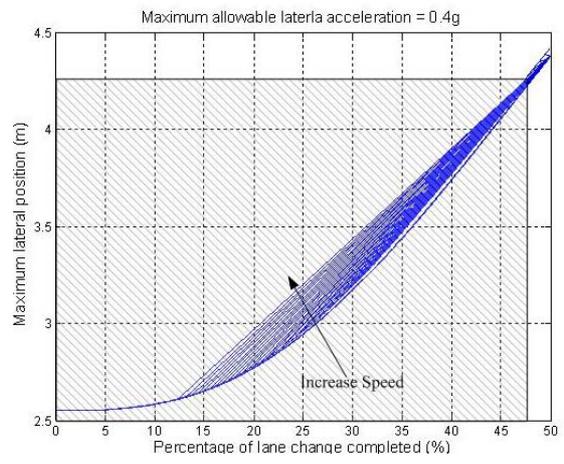


Figure 7-9d

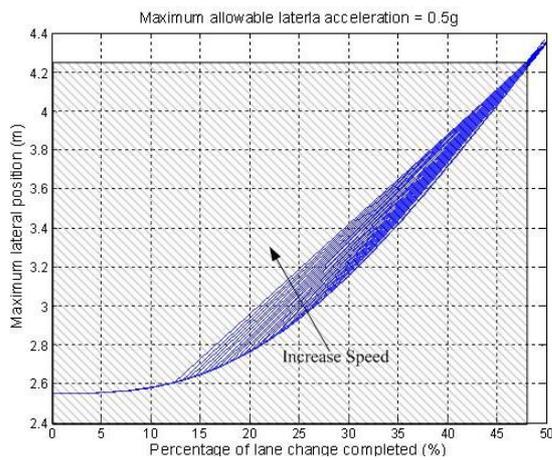


Figure 7-9e

Figure 7-9: Collision free abortion region.

From Figure 7-9b to Figure 7-9e, the effect on vehicle speed variation is significant to

the L_{max} that the L_{max} increase with the increase in vehicle speed. In the figures, the deviation of L_{max} between different speeds is decreasing as the increase in percentage of lane change completed. It is due to the decrease in vehicle orientation as it close to the end of the lane change process, the vehicle orientation adjustment becomes insignificant.

For the effect of the allowable lateral acceleration on the collision free abortion region, the result shows that the increase in a_{2max} (from 0.1g to 0.5g) would enlarge the range of collision free abortion region (from 42% to 47.5%). Finally from the result in Figure 7-9, the collision free abortion region is concluded as 40% for different speed and allowable lateral acceleration operation.

To summarize the achievements in lane change abortion analysis, the lane change abortion trajectory is decomposed into two phases for evaluation. The total longitudinal abortion distance is explicitly calculated which can be employed in longitudinal control scheme. Collision free point is evaluated under different speed of operation and lateral acceleration constraints.

7.7. Simulation and Experimental Results on Lane Change

The simulation and experiment of the proposed lane change algorithm and lane change scheme are implemented on a scaled prototype autonomous vehicle that shows in Chapter 3. The speed controller is a conventional PID controller to maintain the vehicle speed constant at $0.7ms^{-1}$.

7.7.1. Simulation Result

The proposed virtual curvature algorithm was simulated under MATLAB SIMULINK

environment. The control scheme used in the simulation is the block diagram described in Section 7.4.2 (Figure 7-3).

The bicycle model of the prototype vehicle shown in Chapter 3 is used for simulation. The bicycle model parameters used in the simulation are shown in Table 7-2. In the simulation, we assumed the vehicle traveling on a double lane highway with lane width $0.6m$ ($L_w = 0.6m$). The maximum allowable lateral acceleration ($a_{1_{max}}$) is $0.05g$ ($0.49ms^{-2}$) which gives margin for controller action. The lateral controller had to keep the vehicle $0.2m$ from the reference lane. The minimum lane change distance d with above mentioned constraints was $1.86m$ which calculated by (7.10).

In the simulation, the desired longitudinal distance to complete lane was assumed as $3m$ (longer than the minimum distance) with constant speed at $0.7ms^{-1}$ ($V = 0.7ms^{-1}$). The desired L_w , d , and V were substituted into equations (7.11), (7.12), and (7.13) to calculate the curvature of the virtual transition lane (κ) and total transition time (T_{taken}). Figure 7-10 shows the curvature (κ) of the virtual transition lane and the total time of transition (T_{taken}) is $4.28s$.

a_{21}	251.64	a_{44}	-60.25
a_{22}	-130.13	b_{21}	63.77
a_{24}	61.7	b_{22}	-0.49
a_{41}	139.56	b_{32}	0.14
a_{42}	-4.9	b_{41}	-6.67

Table 7-2: Vehicle parameters.

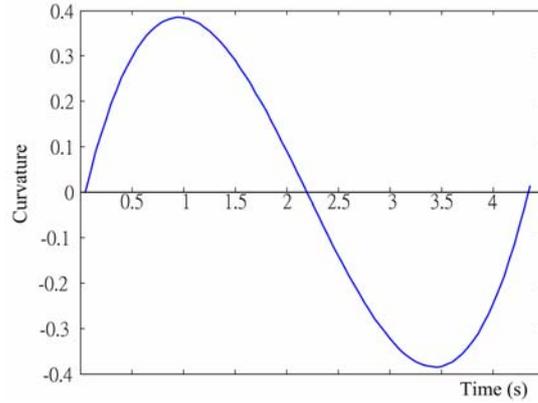


Figure 7-10 : Curvature of virtual transition lane

The vehicle performs right lane change maneuvers in the simulation. The lane change starting time (T_s) is 5s. The lane change finishing time (T_f) is $T_s + T_{taken} = 9.28s$. Figure 7-11 shows the simulation result of the vehicle trajectory with respect to the original lane. In Stage I ($t = 0$ to T_s), the vehicle performs lane keeping and maintains at $0.2m$ from the original lane. At Stage II ($t = T_s$ to T_f), vehicle performs lane changing and its position changes from $0.2m$ to $0.8m$ with respect to the original lane. Therefore the virtual transition lane guides the vehicle towards the destination lane. Finally in Stage III ($t > T_f$), the vehicle resumes lane keeping at destination lane.

The simulation result shows that the proposed control scheme and the lane change algorithm are able to solve the lane change problem.

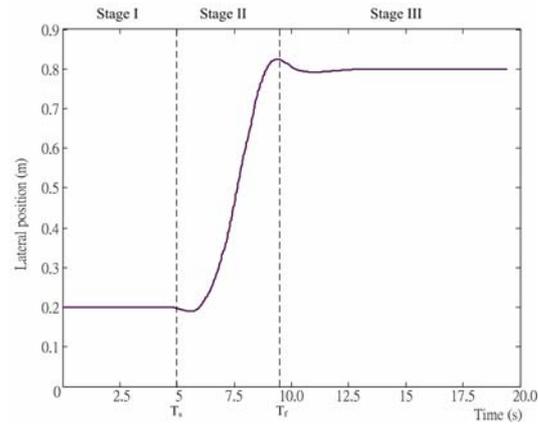


Figure 7-11 : Lane change simulation

7.7.2. Experiment setup

This section shows the physical setting of the test track for lane changing experiment. The proposed algorithm was implemented on the scaled prototype autonomous vehicle. A double lane test track was constructed with lane length (L_w) $0.6m$ for the experiment. Figure 7-12 shows the experimental setup for the lane change experiment with the proposed algorithm. The vehicle had to complete the lane change from Lane A to Lane B within a desired lane change longitudinal distance (d).

Two different lane change longitudinal distances $4m$ and $5m$ were tested. Assuming the vehicle starts lane change at $5s$ (i.e. $T_s = 5s$) with constant speed at $0.7ms^{-1}$. The lane transition time for $4m$ and $5m$ calculated by equation (7.13) are $5.71s$ and $7.14s$ respectively. Therefore the lane change finishing time (T_f) for $4m$ and $5m$ are $10.71s$ and $12.14s$ respectively.

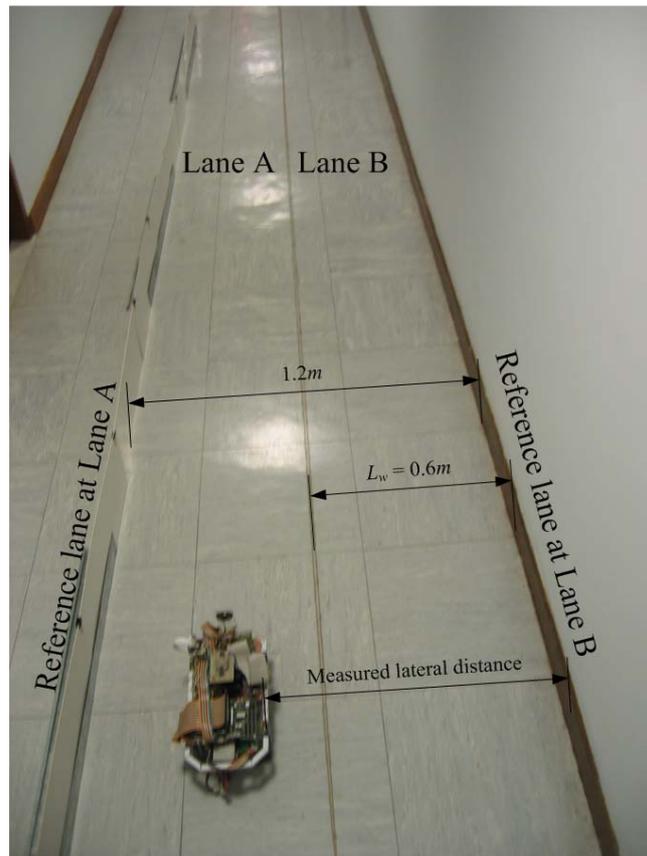


Figure 7-12: Lane change experiment setup with virtual transition lane.

The vehicle was programmed to perform the right lane change maneuver. The lane change maneuver started at time $5s$ (T_s) and finished at T_f for different cases of lane change distances. Infrared sensors installed on both sides of the vehicle to measure the lateral distance from the reference lane. Sensor measurements were used as feedback during the lane keeping (Stage I and III), however, sensor measurements were inaccurate during lane changing (Stage II). Therefore the sensor measurements in Stage II are used to illustrate vehicle lateral position only.

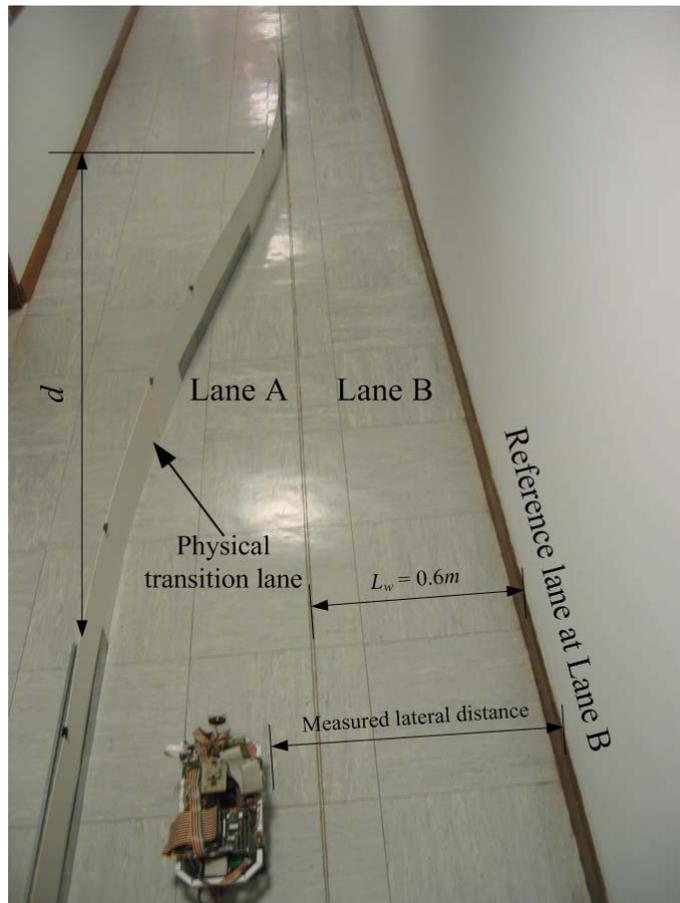


Figure 7-13: Lane change experiment setup with physical transition lane.

To show the performance of the virtual transition lane method is comparable with physical transition lane, a test track with physical transition lane was built. Figure 7-13 shows the experiment setup of the physical transition lane on a double lane scenario. The trajectory of the physical transition lane is based on the polynomial described in equation (7.8). The vehicle performs lane keeping at Lane A and follows the physical transition lane to Lane B.

Implementation requirement and hardware constrains

Some assumptions were made to implement this algorithm. The first assumption was that the bicycle model of vehicle had already been identified. The second assumption was that the lane keeping controller was able to follow the reference lane. The third assumption was that the vehicle speed was constant and the road was straight during lane change

maneuver. The last assumption was due to the fact that the infrared sensors installed on the scaled prototype vehicle are Sharp GP2D12, which measurement range is $0.1m$ to $0.8m$. These sensors are inaccurate for long distance measurement. So we further assume that the reliable measurement range was $0.1m$ to $0.4m$.

7.7.3. *Experimental results*

Figure 7-14 and Figure 7-15 show the experimental results of longitudinal lane change distances (d) equal to $4m$ and $5m$ respectively. In these Figures, different lane changing stages are labeled and separated by dashed line. The vehicle lateral distance with respect to Lane B is recorded. The 2 dotted lines located at $0.8m$ and $0.2m$ are the desired lane keeping distance at Lane A and Lane B respectively. In Stage I ($t = 0$ to T_s), vehicle perform lane keeping at Lane A. The lane changing process starts at Stage II ($t = T_s$ to T_f). In Stage II, the virtual transition lane curvature is feed into bicycle model to estimate the lateral distance from the virtual transition lane. After T_f the vehicle enters the final stage, Stage III the vehicle resume lane keeping at Lane B till the end of the experiment.

Figure 7-14(a) shows the lane change result with $4m$ virtual transition lane. At Stage II, the result shows that the vehicle lateral position with respect to the Lane B is decreasing therefore the vehicle is moving toward to Lane B. At the end of Stage II ($t = T_f$), the vehicle lateral position is $0.28m$ which is higher than the desired $0.2m$ from Lane B. At Stage III, the vehicle resume lane keeping at Lane B and the lane keeping controller adjust the vehicle lateral position from $0.28m$ to the desired lane keeping distance till the end of the experiment.

Figure 7-14 (b) shows the lane change result with $4m$ length physical transition lane, the dashed lines on $t = 5s$ and $t = 10.7s$ indicate the starting and termination of physical

transition lane respectively. To show the lane change performance with virtual transition lane is comparable with physical transition lane, the trajectories (time 5s to 10.7s) shown in Figure 7-14(a) are compared with Figure 7-14(b). The difference between these 2 trajectories is evaluated by Integral Square of Error (ISE). The ISE evaluated for 4m lane change is 0.2576.

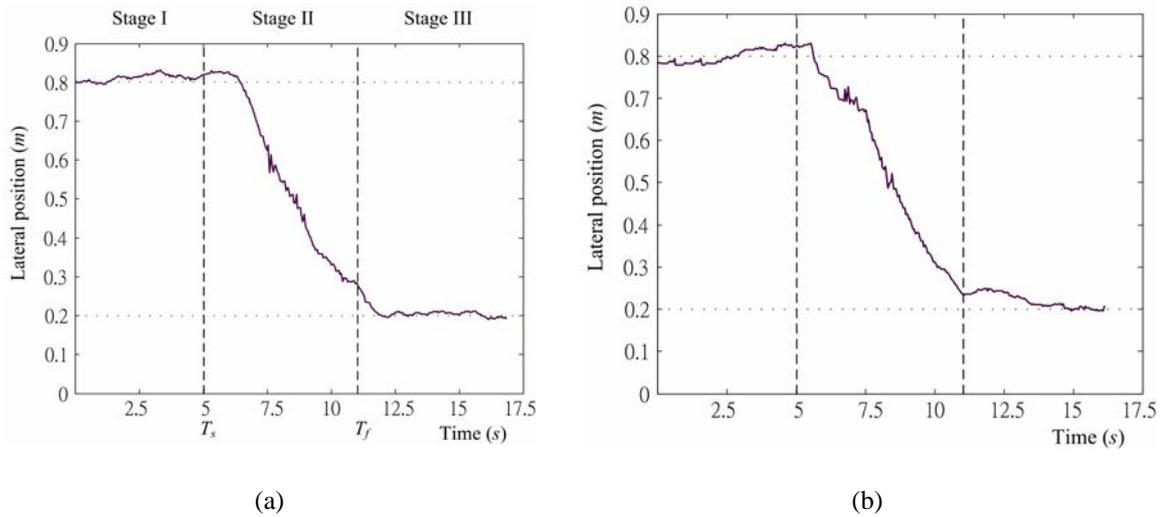


Figure 7-14: Experiment result with virtual transition lane (a) and physical transition lane (b) ($d = 4m$).

For the experiment with 5m lane change distance, the results are shown in Figure 7-15. Figure 7-15(a) shows the result with 5m length virtual transition lane. The result also shows that the vehicle lateral position is moving towards to Lane B. The vehicle lateral position at the end of Stage II ($t = T_f$) is 0.12m which is lower than the desired lane keeping distance. Finally, at Stage III lane keeping controller resumes lane keeping at Lane B and maintains the vehicle at the desired lateral position till the end of the experiment.

Figure 7-15(b) shows the result with 5m length physical transition lane. The vehicle starts lane change with physical transition lane at $t = 5s$ and terminated at 12.5s. The

difference between the 2 trajectories (time 5s to 12.5s) shown in Figure 7-15(a) and Figure 7-15 (b) is evaluated. The ISE for 5m length lane change trajectory is 1.578.

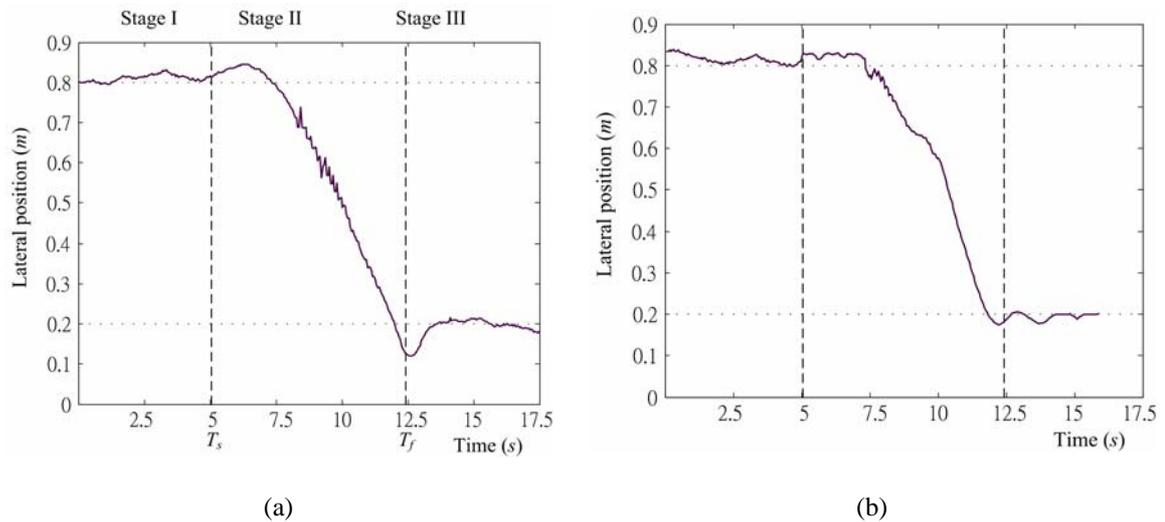


Figure 7-15: Experiment result with virtual transition lane (a) and physical transition lane (b) ($d = 5m$)

The lane change trajectories with virtual transition lane and physical transition lane under different longitudinal lane change distance ($d=4m$ and $d=5m$) are compared. The ISE in the case of $4m$ and $5m$ longitudinal lane change distance are 0.2576 and 1.578 respectively. The results show that the lane changing trajectory with the proposed virtual transition lane method is comparable with the lane change trajectory with physical transition lane.

However, the ISE of $d=5m$ is much larger than the ISE of $d=4m$. It may be due to 2 possible error sources of the experiment. The first possible error source is the accumulation of modeling error during the lane change process. Since the error in bicycle model estimation would be accumulated, longer transition time should have larger error. The second possible error source is the error in the physical transition lane trajectory. The physical transition lane trajectory with $5m$ long is difficult to be constructed as same as the 5th order polynomial.

7.8. Conclusions

The concept of virtual road curvature for assisting automatic lane change maneuvers is presented. This idea is inspired by the relationship between the vehicle lateral dynamics and reference lane curvature. A 5th order polynomial trajectory is chosen as the virtual lane change trajectory and its curvature is used in the proposed algorithm as the virtual curvature. In addition to the concept of virtual road curvature, a lane change control scheme is proposed. The proposed scheme tackles the lane change problem with unified lateral controller approach. Therefore, there is no switching of controller involved in the scheme.

In addition to the scheme of lane changing, safety consideration of lane change abortion trajectory is also investigated. In this chapter, the details on the abortion trajectory are presented by explicit calculation of the abortion distance and maximum vehicle lateral position in the abortion trajectory. Based on the evaluated abortion trajectory, collision free abortion region is proposed for collision free abortion point evaluation. In depth simulations on the relationship between vehicle speed, allowable lateral acceleration, and lane change process are conducted. The result concludes that the abortion process initiated at the first 40% (longitudinal) of lane change is within the defined collision free abortion region. The abortion point will be tested experimentally in Chapter 8.

The proposed lane change scheme is tested in both simulation and experimental test track on a scaled vehicle. Both simulation and experimental result shows that the proposed algorithm is able to solve the lane change problem. Furthermore, the experimental results show that the performance of the proposed virtual lane method is comparable with physical transition lane method.

Experimental case studies of the overall control system will be evaluated in the coming chapter by implementing the controllers suggested in the previous chapters on the semi-autonomous vehicle that presented in Chapter 3.

Chapter 8. Experimental Verification of Lateral Control Algorithms

8.1. Introduction

This chapter further examines the proposed lateral controllers by setting up realistic scenarios for experimental studies. The experimental vehicle used in this chapter is the scaled prototype semi-autonomous vehicle that was presented in Chapter 3. A lateral model of the prototype vehicle has been used for controller design based on the bicycle model as presented in the previous chapters.

The prototype vehicle is equipped with infrared sensors, a rotating encoder, a steering servo motor, and an industrial computer that is used as a data monitoring and control management center (lateral distance measurement, speed measurement, front wheel steering, and control algorithms implementation). The sensing, actuation and computing units installed form the main hardware platform for experimental studies. The detailed hardware configuration of the vehicle was presented in Chapter 3.

This chapter starts with experiments for speed and lateral control studies. These two tasks (speed and lateral control) are the building blocks of the autonomous vehicle control. To achieve further experimental development and analysis, both controllers are required. The speed control experiment shows the ability of the speed controller to maintain the vehicle speed at different desired levels.

The lateral control experiment verifies the ability of the lane keeping controller

working under different speeds on straight as well as curved road profiles. The lateral control experiment assumes that the preview of the road curvature is not available.

The lane change abortion analysis presented in Chapter 7 will also be verified experimentally in this chapter. Different abortion trajectories will be recorded under different speed settings. The analysis result suggests that collision free lane change abortion points are located within the first 40% of the lane change process. Finally, the lane change scheme, speed control, and lateral controller are integrated together to show the experimental verification of the overall control and management center.

The organization of the chapter is as follows: Section 8.2 presents the results on fundamental experiment which includes speed control, lateral control and the experimental abortion trajectories. Section 8.3 shows the experimental setup and result on the integrated experiment of the suggested control algorithms. Section 8.4 presents a discussion on the experiments conducted in this Chapter. Finally Section 8.5 concludes this Chapter.

8.2. Fundamental experiment analysis

The lateral controller and the lane change algorithm were experimentally tested in Chapters 6 and 7 respectively. In Chapter 6, the NN and Fuzzy controllers were studied on a straight road scenario and were compared against each other based on a step response test. The results demonstrated that the controllers were able to adjust the vehicle lateral position in the straight road scenario. In this chapter, the lateral controller performance is further evaluated on curved road profile.

In Chapter 7, the author proposed a lane change scheme with virtual curvature

algorithm and suggested the collision abortion points were located within the first 40% of the lane change process. In this section, lane changes with abortion are conducted with different speeds so that reaffirm experimentally the conclusions drawn in Chapter 7.

8.2.1. Speed control

The speed controller implemented on the vehicle is a PID controller. The PID parameters were manually tuned through experiments. The parameters of K_p , T_i , K_d were obtained as 15.0, 0.7, and 0.2 respectively. In the speed control experiment, the speed set-point was a staircase profile. The speed set-point was scheduled at 30, 60, and 80cms^{-1} (Figure 8.1). The speed set-point profile was changed every 8s and the sequence of the set-point changes were 30, 60, 80, 60, and 30cms^{-1} . Therefore there were four step changes -2 rising and 2 falling, in the profile in order to examine the vehicle speed response in both acceleration and deceleration modes. From the speed control experiment, the average acceleration and deceleration factor of the PID controller corresponding to speed error was determined.

Figure 8-1 shows the vehicle speed control experimental result. The solid and dashed lines are the measured vehicle speed and speed set-point respectively. As expected, the level of noise increases with increase in the speed. The noise is due to the microcontroller miss counting on the number of pulses generated by the rotating encoder. As the speed increases, the frequency of the pulses chain produced by the rotating encoder increase. Therefore, the chance of the microcontroller missing the number of pulses would increase as well.

The vehicle starts at standstill and accelerates to 30cm/s^{-1} with 2.5s . When the set-point is changed to 60cm/s^{-1} and 80cm/s^{-1} , it takes 2s and 1.7s to reach the set-point respectively. For the deceleration from 80cm/s^{-1} to 60cm/s^{-1} and from 60cm/s^{-1} to 30cm/s^{-1} , the deceleration times are 1.7s and 2s respectively.

From the experimental result, the acceleration and deceleration factors of the PID controller are similar and it is approximated as -0.5cm/s^{-1} . Therefore when the speed error is -20cm/s^{-1} , the corresponding vehicle acceleration rate is 10cm/s^{-2} .

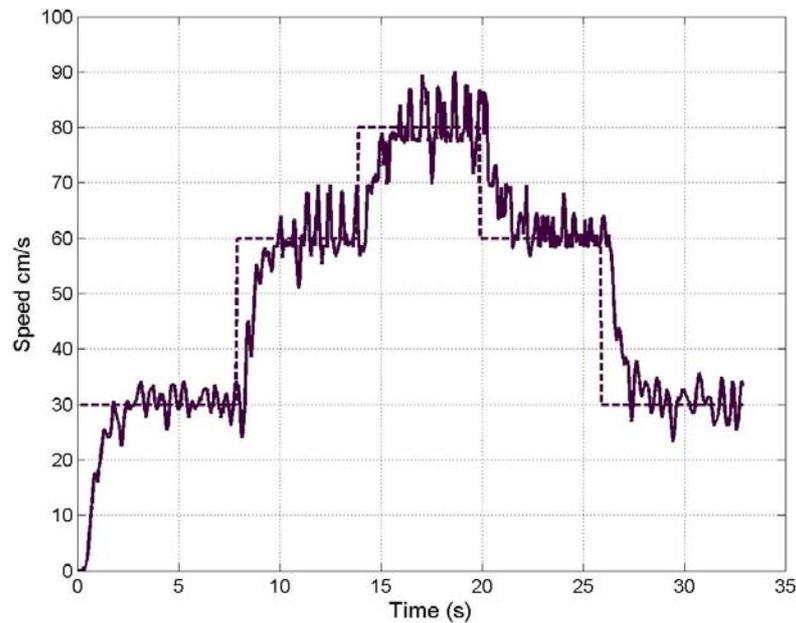


Figure 8-1: Vehicle speed control experimental result..

8.2.2. Lateral control

The NN and Fuzzy controllers proposed in Chapter 4 and 5 were verified experimentally in Chapter 6. The experiments conducted in Chapter 6 evaluated the controller performance by step response and performance index (such as ITAE and ISE). In the comparison of controller performance indices, NN controller

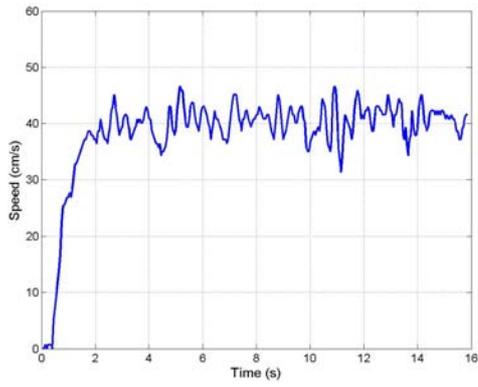
demonstrated a better performance than the two fuzzy controllers. Therefore the NN controller is selected as the vehicle lateral controller in this Chapter.

In this Chapter, the performance of the NN controller is further evaluated on a curved road scenario. Figure 8-2 shows the experimental setup with a curved reference lane. The curved reference lane was built by cardboard as depicted in the figure. As previously mentioned in Chapter 3, the vehicle is equipped with two infrared sensors at the sides to measure the vehicle lateral distance from the reference lane. There are two curved sections with opposite turning directions. The vehicle is required to turn the right and then to the left while keeping the desired lateral distance from the reference lane. The radii of curvature of the two curved sections are unknown and there is no feed-forward compensator to assist the steering. Therefore the NN controller minimizes the lateral error only based on the look-down measurement of infrared sensors.

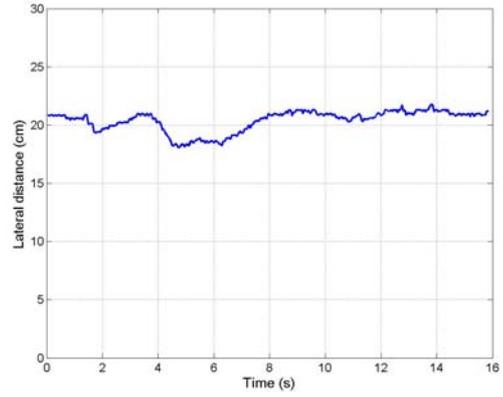


Figure 8-2: Lane keeping experiment setup.

Figure 8-3, Figure 8-4, and Figure 8-5 show the controller lane keeping performance with speed 40cms^{-1} , 70cms^{-1} , and 100cms^{-1} respectively. The figures show the vehicle speed as well as the lateral distance during the experiment. The results show that the maximum deviation from the set-point is 7cm (at 100cms^{-1}) during the right cornering. The vehicle is able to maintain at the desired lateral position (20cm) after the curved sections.

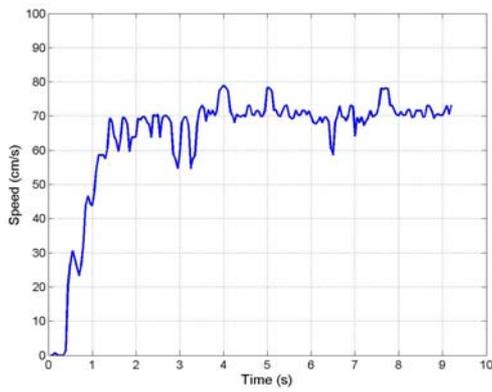


(a)

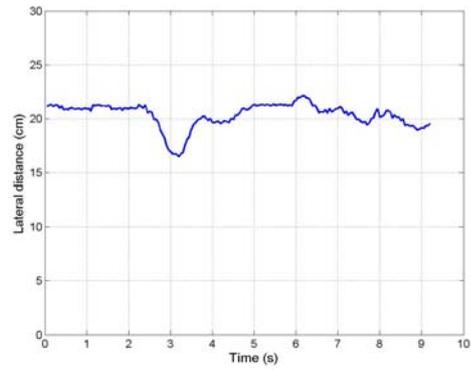


(b)

Figure 8-3: Lane keeping performance with speed $=40\text{cm/s}^{-1}$.

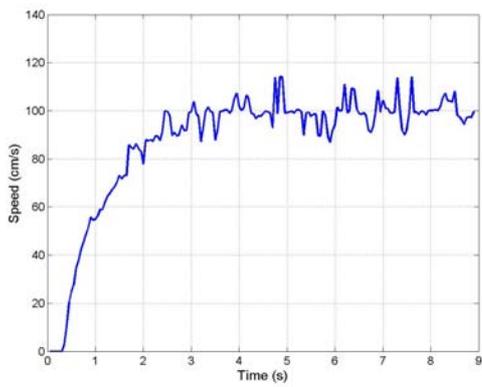


(a)

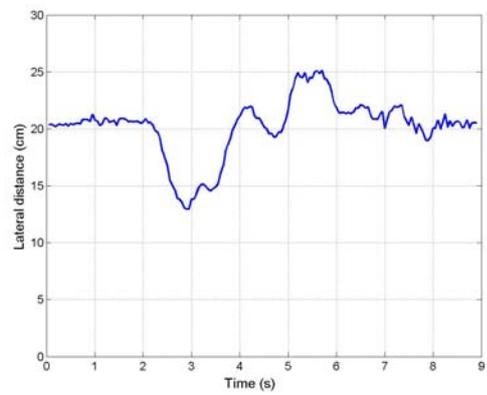


(b)

Figure 8-4: Lane keeping performance with speed $=70\text{cm/s}^{-1}$.



(a)



(b)

Figure 8-5: Lane keeping performance with speed $=100\text{cm/s}^{-1}$.

8.2.3. Lane change abortion

The analysis on lane change abortion trajectory was reported in Chapter 7. In the analysis, the collision free abortion point (P_{cf}) and the Safety abortion region (S_r) were introduced. S_r is the lateral region that the merging vehicle will not collide with the vehicle at the adjacent lane. From the analysis, the collision free abortion region is concluded as 40% of the lane change process for different speed and allowable lateral acceleration of operation.

In this section, the collision free abortion region will be tested experimentally. The scaled autonomous vehicle was programmed to perform lane changing with abortion at 40% of the lane change process. The infrared sensors on board will record the lateral distance from the reference lane throughout the experiment. The vehicle lateral position will be compared with the Safety abortion region (S_r) so that to verify the successfulness of the defined collision free abortion point.

The experiment was conducted in an indoor environment which the corridor wall served as the reference lane. Figure 8-6 illustrates the experimental setup of lane change abortion. The vehicle width (V_w) and lane width (L_w) are 20cm and 60cm respectively. The shaded region shown in the figure is the Safety abortion region (S_r) which defined in Chapter 7. Based on the assumption stated in Chapter 7 and equation (7.21), the S_r is 80cm from the reference lane. In order to show the vehicle lateral trajectory, the lateral distance y_l (as depicted in the figure) is recorded during the experiment.

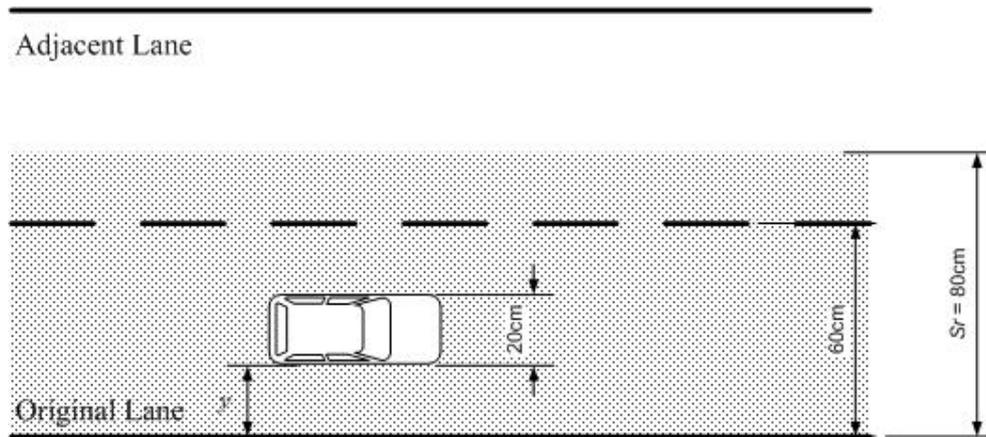


Figure 8-6: Lane change abortion experiment setup.

The determination of virtual lane change curvature profile was described in Chapter 7. The lane change longitudinal distance is $4m$ which is determined with the lateral acceleration limit at $0.01g$. In the experiment, the vehicle started with lane keeping which maintaining the vehicle $20cm$ from the reference lane and then the lane change maneuver was activated after time $5s$. Subsequently, the lane change process was aborted and the vehicle resumed lane keeping at the original lane, when 40% of the lane change process was completed.

Figure 8-7 shows the results of the abortion trajectories with dash line, dotted line and solid line for speed $50cms^{-1}$, $70cms^{-1}$, and $80cms^{-1}$ respectively. From time $0s$ to $5s$, vehicle was operated in lane keeping mode and the results show that the vehicle lateral distance was maintained at $20cm$ under different speeds.

After time $5s$, the vehicle started lane changing and it took $3s$, $2.5s$, and $2s$ to reach to the point of lane change abortion with speed $50cms^{-1}$, $70cms^{-1}$, and $80cms^{-1}$ respectively. Since the lane change distance was fixed at $4m$, the time to reach 40% was shortest at high speed. At the abortion point, the maximum lateral distances (y)

from the reference lane were 37cm , 35cm , and 38cm for speed 50cms^{-1} , 70cms^{-1} , and 80cms^{-1} respectively.

After the abortion point, the vehicle resumed the lane keeping at the original lane. Form the figure, a large overshoot on the resuming trajectories is observed. This is due to the angle between the vehicle and the reference lane is too large such that the infrared sensor can not provide accurate measurements. Therefore it affects the performance of the lateral controller.

As noted in Chapter 7, if the vehicle boundary does not violate the defined region S_r , the abortion process is collision free. Therefore in this experiment, if the vehicle lateral upper boundary y_u (Referring to Figure 8-6) is smaller than 80cm , the abortion process is safe.

However the recorded vehicle lateral position is the lower boundary y_l , thus the upper boundary y_u is reproduced from y_l by adding the vehicle width (V_w). Therefore $y_u = y_l + 20\text{cm}$. In the experimental result shown in Figure 8-7, the maximum lateral position is 38cm at speed 80cms^{-1} (i.e. $y_l = 38\text{cm}$). Then the maximum vehicle lateral upper boundary position (y_u) is 58cm ($y_u = y_l + 20\text{cm}$). Since $y_u = 58\text{cm}$ is smaller than $S_r = 80\text{cm}$, the experimental results conclude that the lane change abortion started on or before 40% of lane change process is collision free. The experiment results consolidate the findings in the abortion analysis stated in Chapter 7.

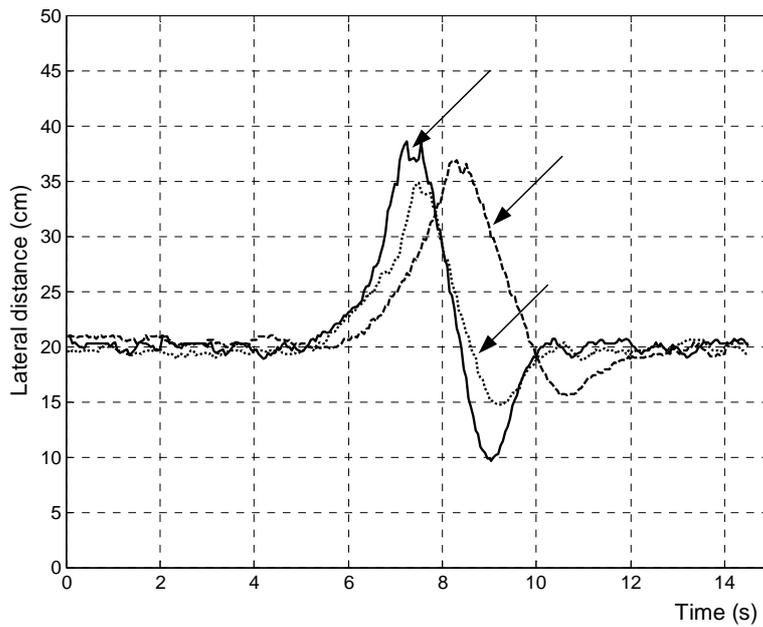


Figure 8-7: Experimental abortion trajectories under different speed.

8.3. *Integrated experiment*

In this section, the performance of the fused neural network lateral controller and the lane change algorithm are evaluated in a single trail. In this experiment, the autonomous vehicle is scheduled to perform lane keeping with the proposed fused neural network lateral controller on a curved reference path. The lane keeping controller is required to maintain the vehicle 20cm from the reference lane.

Then the vehicle is instructed to perform the lane changing with the proposed virtual curvature lane change algorithm to the adjacent lane on a straight road section. When the lane change process is completed, the vehicle resumes lane keeping at the adjacent lane till the end of the experiment.

Two cases with different speed profiles were tested as follows:

Case 1:

The vehicle initial speed was set at 40cms^{-1} for the first 12s. The vehicle speed was increased to 70cms^{-1} after time 12s. Since the length of the curved section was 6m (details on the test track setup will be shown on the coming section), the vehicle should take 15s with speed 40cms^{-1} to complete the riding on the curved road. Therefore after time 12s, the vehicle is close to the end of the curved section. Then the vehicle completes the lane changing and lane keeping on the adjacent lane with speed 70cms^{-1} to the end of the experiment.

Case 2:

The vehicle speed is kept constant at 70cms^{-1} throughout the experiment.

Figure 8-8 illustrates the test track setup in a corridor in the Department of Electrical Engineering. There are two lanes, namely Lane A (original lane) and Lane B (adjacent lane), in the test track with lane width 60cm . The reference lane of the Lane A is made by cardboard so that curved road sections can be implemented. The reference lane of the Lane B is the wall of the corridor.

The profile of Lane A starts with 2 curved sections and followed by straight road. The lengths of curved and straight sections are 6m and 8m respectively. Therefore the total length of Lane A is 14m . The radius of curvature of the right and left cornering as depicted in the figure are 2.2m and 2m respectively. Lane B is a straight road profile with length 8m .

As discussed in Chapter 3, infrared sensors are installed on both sides of the autonomous vehicle to measure lateral distance. To achieve lane changing and lane

keeping at both Lane A and B, sensors on both sides are required. In Figure 8-8, there is an illustration on the direction of sensor measurement and indicates the sensors used during lane keeping in Lane A and Lane B. The infrared sensors data at both side and vehicle speed are recorded during the experiments.

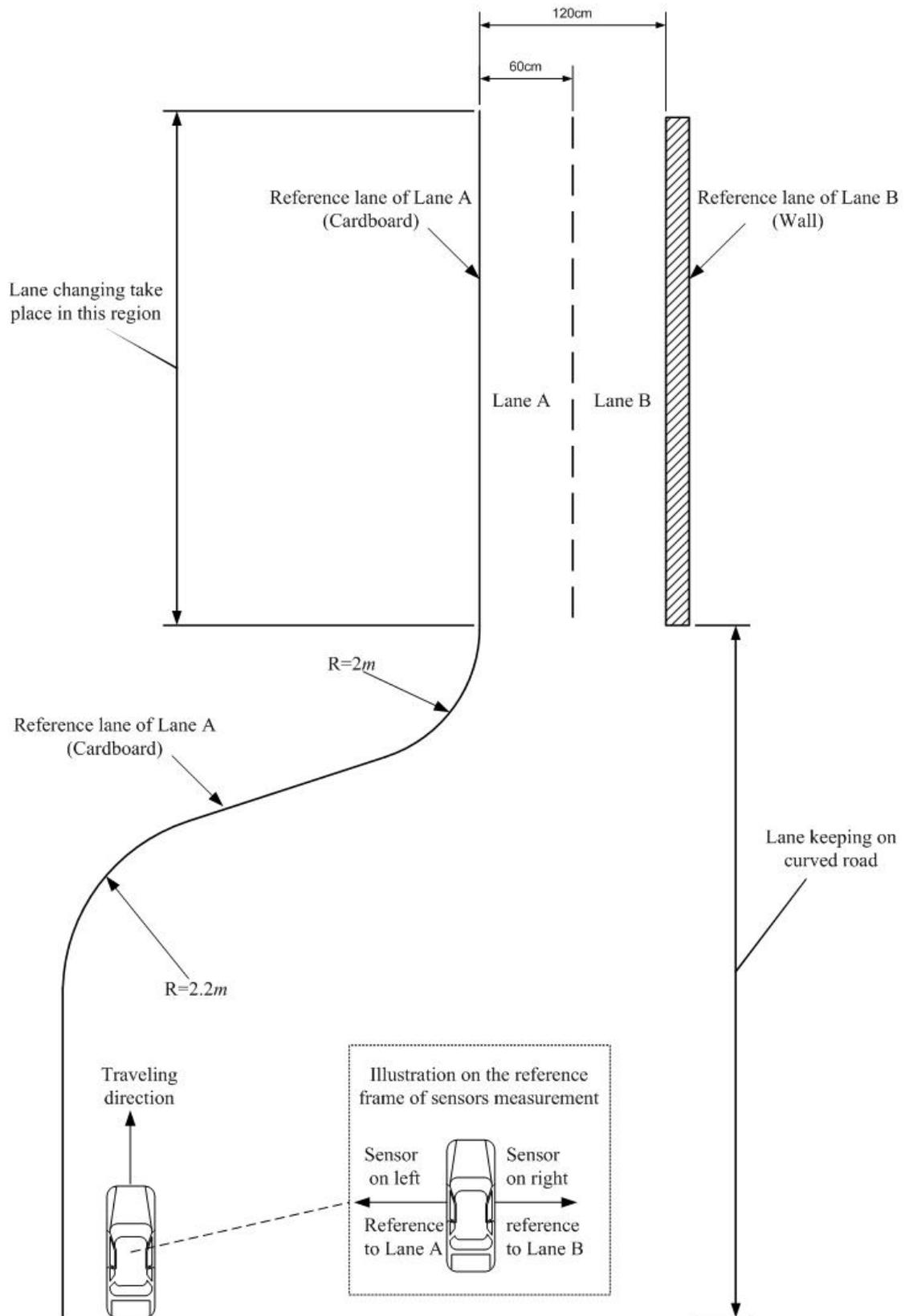


Figure 8-8: Integrated experiment setup.

8.3.1. Experimental Result

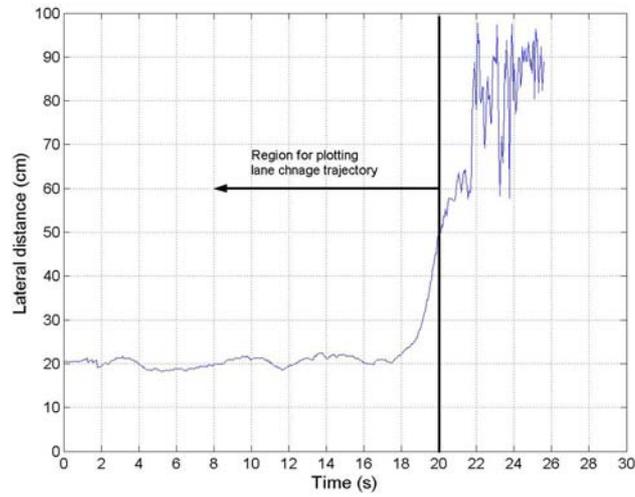
Data treatment for presenting vehicle trajectory

In this section the experimental result of the integrated experiment will be presented. The infrared sensors data is recorded throughout the experiment. However owing to limitation of infrared sensors on the detection range, the vehicle lateral trajectory cannot be reproduced from sensor data that measured with respect to a single reference frame (i.e. either reference to Lane A or Lane B).

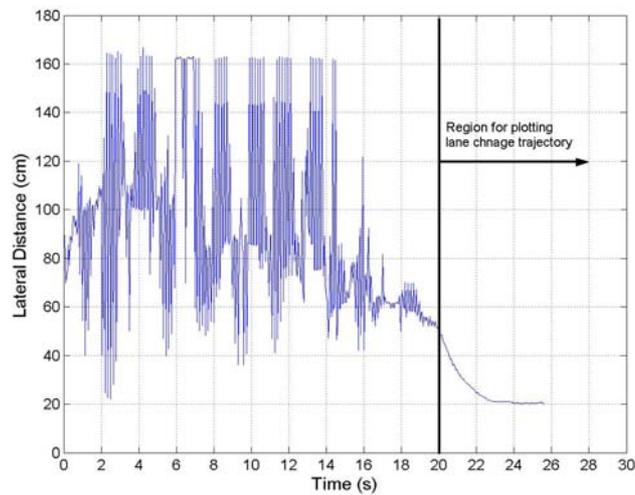
Figure 8-9(a) and (b) show the sensors measurements in case 1 with respect to the Lane A and Lane B respectively. The highly oscillatory measurement shown in Figure 8-9 is due to out of measurement range. From the figure, the complete vehicle trajectory cannot be reproduced either from Figure 8-9(a) or (b).

In order to reproduce the vehicle trajectory, the 2 sets of sensor data from different reference frame are truncated together. The fluctuated data at the end of Figure 8-9(a) is replaced by the normal data at the end of Figure 8-9(b). Since the data is taken from different frame, the sensor data on Figure 8-9(b) is mapped to reference Lane A by offsetting the data with the width of vehicle and lanes. Let the sensor data measured with respect to reference Lane B, lane width, and vehicle width be $y_b(t)$, L_w (i.e 60cm) and v_w (20cm from Chapter 3). Then the offset data of $y_b(t)$ with respect to reference lane A is calculated by $2L_w - v_w - y_b(t)$.

The truncated result of the complete vehicle lateral trajectory of case 1 is shown in Figure 8-10(a). The vehicle lateral trajectories presented in this section are all with respect to the Lane A.



(a)



(b)

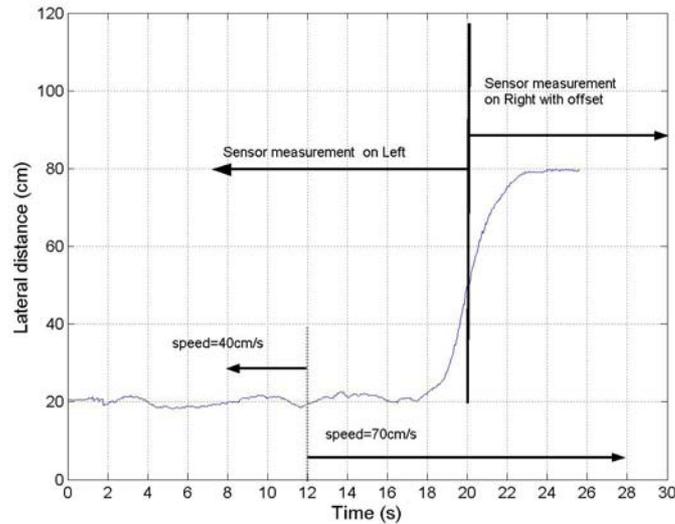
Figure 8-9: Sensors measurement in Case 1.

Result of Case 1:

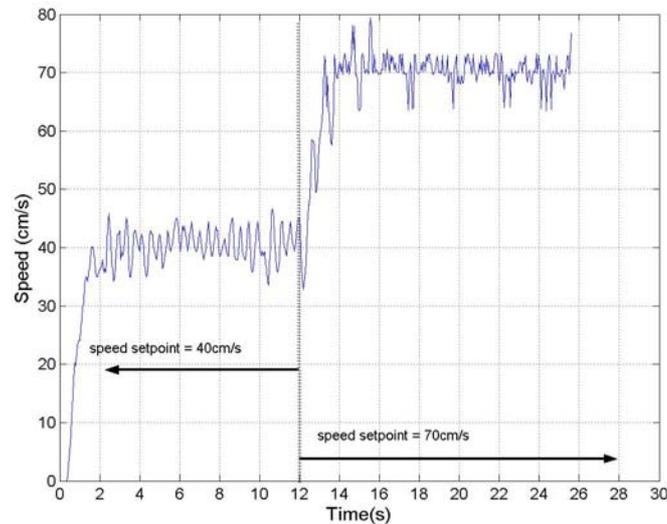
After the mapping and truncation of the sensor data, Figure 8-10(a) and (b) show the vehicle lateral trajectory and vehicle speed profile of Case 1 respectively. In Figure 8-10, from time 0s to 12s, the vehicle is moving on the curved road sections with speed 40cms^{-1} and the result shows good tracking on the lateral position which

maintains the vehicle 20cm from the reference Lane A.

After time 12s, the vehicle speed increases to 70cm^{-1} . At time 18s vehicle starts the lane changing maneuver, the vehicle takes 6s to complete lane change and resume lane keeping at Lane B at time 24s and maintains the vehicle 80cm from the reference Lane A (which equals to 20cm from the reference lane B).



(a)

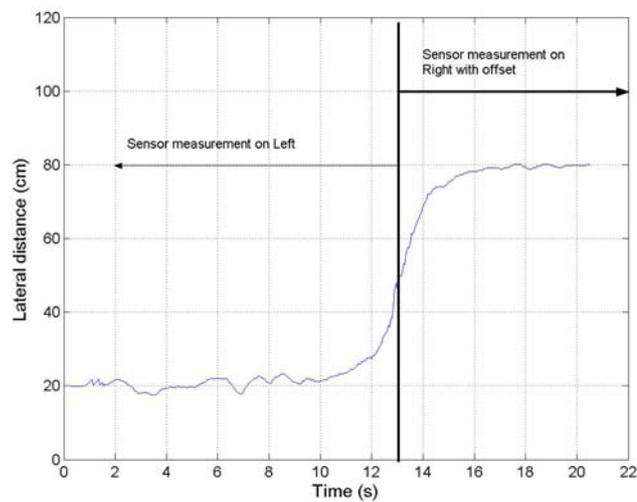


(b)

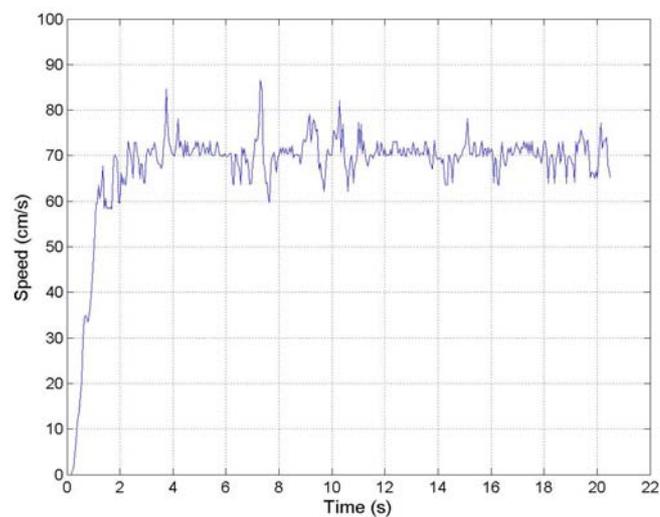
Figure 8-10: Case 1 experimental result.

Result of Case 2:

Figure 8-11 shows the vehicle lateral trajectory and vehicle speed profile of Case 2. In Case 2, the vehicle speed is maintain constant at 70cm/s^{-1} . From time 0s to 8s , the vehicle is riding on the curved road sections and it shows reasonable good tacking performance at 20cm from the reference lane A. At time 10s vehicle starts lane changing, the vehicle takes 6s to complete lane change and resume lane keeping at Lane B at time 16s . After time 16s , the vehicle is maintained 80cm from the reference lane A (i.e. 20cm from the reference lane B).



(a)



(b)

Figure 8-11: Case 2 experimental result.

8.4. Discussion

Lateral control experiment

The lateral control experiment conducted on curved road section with different speed settings from 40cms^{-1} to 100cms^{-1} . The results show that there is a strong coupling effect on the speed to vehicle lateral dynamic. The performance is deteriorated as the speed increase.

Improvement can be made by adding feed-forward compensator which generates steering angle based on the prior knowledge on the road curvature. However if the road curvature is unknown, feed-forward compensation cannot be achieved.

The other method to improve the controller performance is to modify the sensor location. In this thesis, the prototype vehicle equipped with 2 sensors for lateral distance measurement. The current location of the sensor at the front is installed in line with the front wheel. If the location of the sensor can be extended ahead to the front wheel, riding performance can be improved. Since the extension on the sensor location provides virtual look ahead lateral distance, the lateral controller adjusts the vehicle lateral position based on the virtual look-ahead distance.

Integrated experiment

The objective of the proposed lane changing algorithm with virtual curvature is to guide the vehicle from original lane to the adjacent lane without infrastructure support. The lateral sensing device used in the thesis is infrared sensor which provides

accurate short distance measurement. As mentioned in Chapter 7, the proposed virtual lane change algorithm and control scheme is activated to complete the overall lane change profile.

In the integrated experiment, the lane change profile may not be completely executed and the time of lane changing with estimation may be shortened. The lateral distance to the adjacent lane is measured continuously during lane change. When the distance falls into reliable measurement range (i.e $<40cm$), normal lane keeping at the adjacent lane is resumed. The early resume process offers smooth transition between the 2 schemes and enhance the safety by shortens the time of lane changing with estimation.

8.5. Conclusion

In this Chapter, comprehensive experimental studies on vehicle control have been conducted. The fundamental tests on vehicle speed control and lateral control shown in this chapter provide the basis for the further studies on vehicle lane change maneuver and integrated experiments.

The experimental lane change abortion trajectory accomplished in this chapter verifies the theoretical analysis on the collision free abortion point stated in Chapter 7. The suggested collision free abortion point is located at 40% of the lane change abortion process and it has been verified by the experiment under different speed.

The integrated experiment shows feasibility on the integration of the two proposed controllers and lane change algorithm. Since there is no controller switching

in the suggested lane change algorithm, the lateral controller implemented performs both lane keeping and lane changing in cope with the virtual lane change algorithm.

Chapter 9. Conclusions

We have come a long way since Cugnot steam car. The joy and excitement of driving is somehow embedded in each of us. On the other hand, the explosion of the number of cars has led to air pollution, road congestion, noise pollution, and most importantly, the loss of human life. Technology can be used to alleviate some of these problems. Car manufacturers have already implemented driver assistance systems, cruise controller, dynamic radar cruise controller, parking sensors, anti-lock braking systems, dynamic stability control, etc. However, at present, only the top end luxury cars provide such features. But the trend is towards a move to driver assistance systems and this consolidates the vision for full automation in the near future. To realize complete vehicle automation, the problems associated with longitudinal and lateral controls and their interactions should be completely resolved.

The studies on vehicle lateral control and automatic lane change maneuver are reported in thesis. The design of novel algorithms for lateral control and lane change are implemented in real-time control. Experiments conducted on the prototype vehicle demonstrated the effectiveness of the proposed control algorithms. The contributions of this thesis not only include studies on vehicle lateral control and lane changing, but also control engineering in a wider sense. In this chapter, the major contributions and some suggestions for future work are outlined.

9.1. Major Contributions

In this thesis, there are five major contributions toward the realization of autonomous vehicle and the applications of soft computing techniques in control

engineering. For the application of autonomous vehicles, a fused neural network controller is proposed as vehicle lateral controller and a virtual curvature lane change algorithm is established. Furthermore, Lane change abortion analysis is conducted to evaluate a collision free abortion point. For the application of soft computing method, a Fired Rules Chromosome encoding scheme is proposed to enhance convergence rate. Finally, experimental verification of the proposed controllers and algorithms are conducted. The major contributions are summarized as:

- A fused neural network controller is proposed which utilizing task decomposition principle to design a novel and yet simple controller structure. The network structure reduces the number of connection weight by 40% in comparison to a fully connected network. The contribution of this controller structure is not only restricted to vehicle control but for a class of systems with angle and displacement as control variables such as inverted pendulum and ball-and-beam systems. Genetic algorithms are used to optimize the network weights for different control systems. For the vehicle lateral controller, Monte-Carol analysis has been conducted to evaluate controller robustness subject to $\pm 20\%$ parameter variation. The result shows that the suggested controller is robust to $\pm 20\%$ parameter variation.
- Fire Rules Chromosome (FRC) encoding scheme is suggested to enhance the convergence rate of the fuzzy rules optimization process by GA. In FRC, the length of chromosome is changed dynamically from generation to generation. In each generation only fired rules will be encoded into chromosome for optimization in that generation. Since the length of chromosome governs the

search space, the reduction in search space would eventually enhance the convergence rate. Monte-Carlo robustness analysis are conducted on the fuzzy controller optimized by FRC scheme and fuzzy controller optimized by traditional method. The result shows that FRC scheme would enhance the convergence rate but retain the controller robustness.

- Virtual curvature lane change scheme is proposed to handle the automatic lane change problem. The lane change maneuver in the absence of infrastructure is a challenging task in autonomous vehicle research. The concept of virtual road curvature transforms the physical reference lane to the desired transition lane by the inherit property of vehicle lateral model. The proposed lane change scheme contains no switching of controller and thus offers smooth transition from lane change to lane keeping. Experimental studies have been conducted to show the feasibility of implementing the lane change scheme in realistic test track.
- Lane change abortion analysis is conducted to show the collision free abortion point of a lane change process. To best of the author knowledge, it is the first analysis to document the collision free abortion point. In the analysis, collision free region is defined in terms of vehicle width and lane width. The collision free abortion point is defined as the point which ensures the vehicle completes the abortion process within that collision free region. The analysis result suggests that the collision free abortion point is located at 40% of the lane change process.
- Experimental verification on the proposed controllers and lane change algorithms are conducted. Controllers and lane change scheme are integrated

and tested in a test track. The result shows the feasibility of implementing the suggested algorithms on a vehicle. In addition, the verification on the collision free abortion point under different speed consolidates the findings in the lane change abortion analysis.

9.2. Suggestion for future work

It is impossible to tackle all the problems of autonomous vehicle in thesis and this was never the intention. The following discussion gives a short summary of some topics, among others, that could be studied in future.

- ❖ Further studies on the proposed fused neural network controller can be taken. The controller performance can be improved by online adaptation of the connection weights. The controller structure is simple and the number of connection weight is not bulky which makes it feasible for online adaptation. The algorithm may be enhanced to adjust the connection weights according to the vehicle speed and the tracking error.

- ❖ The proposed lane change algorithm and control scheme could be further explored. In the proposed scheme, the period of lane changing with virtual curvature method is the full lane width. However, if the maximum measurement range of sensors can be considered in the lane change scheme, the period of lane change by estimation can be reduced. The reduction would enhance the safety and the accuracy of the whole lane change process.

- ❖ The development and improvement of the prototype vehicle is extensive for conducting precise experiment. The locations of the lateral sensor should be

extended to the front end of the vehicle so that to establish a virtual look-ahead distance to enhance the performance of the vehicle lateral control.

- ❖ In the thesis, the look-down reference system is assumed and no road curvature is available. However, on board navigation system with GPS and digital map already implemented in vehicle are further steps towards a full autonomous vehicle. The vehicle location and curvature information can be achieved from the GPS and digital map. Further studies on the uses of digital map to extract curvature information are suggested to improve the lateral control performance with feed-forward compensation.

- ❖ The implementation of the algorithms developed in this research on a real car is also the next logical step.

9.3 Final remarks

Crucial to a seamless transition from a semi-autonomous to an autonomous car is the integration of various instrumentation and electronic devices into a user-friendly and coordinated management system. This calls for a gradual change in the driver attitude, traffic regulations which in turn have numerous legal implications. Therefore, driver re-training is required to avoid loss of life and property as a result of misunderstanding and miscommunication between man and the machine. These issues are outside the scope of the investigation in this thesis. One would imagine that in a foreseeable future such autonomous cars are sharing the roads with human drivers. Still, there are many open ended questions: How can we cope from relinquishing our control of the wheels? What are the legal implications? What would happen to the joys and excitement of driving?

Bibliography

A. Aceves-Lopez, and J. Aguilar-Martin, (2006)

A simplified version of mamdani's fuzzy controller: the natural logic controller,
IEEE Trans. on Fuzzy Systems vol. 14 issue 1. Feb. 2006

C.W Anderson, (1989)

Learning to Control an Inverted Pendulum Using Neural Networks,
Control System Magazine, Vol. 9, Issue 3, April 1989.

J. Ackermann, J. Guldner, W. Sienel, R. Steinhauser, and V.I. Utkin, (1995)

Linear and Nonlinear Controller Design for Robust Automatic Steering,
IEEE Trans. on Control System Technology, Vol. 3, No. 1 March 1995.

L. Bouraoui, S. Petti, A. Laouiti, T. Fraichard, and M. Parent, (2006)

Cybercar cooperation for safe intersections,
Proc. of IEEE ITSC 2006, Canada, Sept. 17-20 2006

Sean N. Brunnan, (1999)

Modeling and Control Issues Associated with Scaled Vehicle,
Thesis (Msc.), The University of Illinois, Department of Mechanical Engineering

Lin Cai, A.B. Rad, W.L. Chan, and Kai-Yuan Cai, (2003)

A robust fuzzy PD controller for automatic steering control of autonomous vehicles,
Fuzzy Systems, 2003. FUZZ '03. The 12th IEEE International Conference on Volume
1, 25-28 May 2003 Page(s):549 - 554 vol.1

Lin Cai, (2003)

Novel algorithms for longitudinal and lateral control for application in autonomous
vehicles,

Thesis (Mphil.), The Hong Kong Polytechnic University, Department of Electrical
Engineering

J. Casillas, O. Cordon, M. J. del Jesus, and F. Herrera, (2005)

Genetic Tuning of Fuzzy Rule Deep Structures Preserving Interpretability and Its
Interaction With Fuzzy Rule Set Reduction,
IEEE Trans. on Fuzzy Systems, Vol. 13, No. 1, February 2005

P.T. Chan, W.F. Xie and A.B. Rad, (2000)
Tuning of Fuzzy Controller for an Open-loop Unstable System : A Genetic Approach,
Fuzzy Sets and Systems, Volume 111, Issue 2, 16 April 2000, 137-152.

P.T. Chan, K.M. Tsang and A.B. Rad, (2003)
Optimisation of Stable Rule Table for Fuzzy Controller using Genetic Algorithms,
Control and Intelligent System, Vol. 31 (3) pp.133-137, 2003

W. Chang, J. B. Park, and Y. H. Joo, (2003)
GA-Based Intelligent Digital Redesign of Fuzzy-Model-Based Controllers,
IEEE Trans. on Fuzzy Systems, Vol. 11, No. 1, February 2003

Chih-Hsun Chou, (2006)
Genetic algorithm-based optimal fuzzy controller design in the linguistic space,
IEEE Trans. on Fuzzy Systems vol. 14 issue 3. June 2006

Q. Chen, U. Oguner, and K. Redmill, (2004)
Ohio State University at the 2004 DARPA Grand Challenge: Developing a
Completely Autonomous Vehicle,
IEEE Intelligent System, Vol. 9, Issue 5, Sept.-Oct. 2004.

S.B. Choi, (2000)
The Design of a Look-Down Feedback Adaptive Controller for the Lateral Control of
Front-Wheel-Steering Autonomous Vehicles,
IEEE Trans. on Vehicular Technology, Vol. 49, No. 6, Nov. 2000.

W. Chen, and M. Saif, (2005)
A Novel Fuzzy System With Dynamic Rule Base,
IEEE Trans. on Fuzzy systems vol. 13 issue 5 Oct. 2005

Cybercar official website
<http://www.cybercars.org/>

E.P. Dadios and D.J. Williams, (1998)
Nonconventional Control of the Flexible Pole-Cart Balancing Problem: Experimental
Results,
IEEE Trans. on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 28, No. 6,
Dec. 1998.

L. Davis, ed.,(1991)

Handbook of Genetic Algorithms,
Wan Nestand, Reinhold

S. Dominic, R. Das, D. Whitley, C. Anderson, (1991)

Genetic Reinforcement Learning for Neural Networks,
Int. Joint Conference on Neural Network, Seattle, 1991.

P.H. Eaton, D.V. Prokhorov, and D.C. Wunsch, (2000)

Neurocontroller Alternatives for “Fuzzy” Ball-and-Beam Systems with Nonuniform
Nonlinear Friction,
IEEE Trans. on Neural Networks, Vol. 11, No. 2, March 2000.

A. Eidehall, J. Pohl, F. Gustafsson, and J. Ekmark (2007)

Toward Autonomous Collision Avoidance by Steering
IEEE Trans. on Intelligent Transportation Systems, Vol. 8, No. 1, March 2006 Pages
84 - 94

Chia-Feng Juang, (2005)

Combination of online clustering and Q-value based GA for reinforcement fuzzy
system design,
IEEE Trans. on Fuzzy systems vol. 13 issue 3 June. 2005

D. Farkas, J. Young, B. Baertlein, and U. Ozguner, (1997)

Forward-Looking Radar Navigation System for 1997 AHS Demonstration,
IEEE Conference on Intelligent Transportation System, Nov. 1997.

R.E. Fenton, (1988)

On the Optimal Design of automotive Lateral Control,
IEEE Trans. on Vehicular Technology, Vol. 37, No. 2, May 1988.

R.E. Fenton and R.J. Mayhan, (1991)

Automated Highway Studies at The Ohio State University-An Overview,
IEEE Trans. on Vehicular Technology, Vol. 40, No. 1, Feb. 1991.

R. L. French & Associates, E. R. Case & Associates, Y. Noguchi, C. Queree, and O.
Sviden, (1993)

A Comparison of IVHS Progress in the United State, Europe, and Japan,
31, Dec. 1993

M. Fu, J. Ruan, and H. Ding, (2004a)
Study on Virtual Path Track Control for Intelligent Vehicle,
IEEE Intelligent Transportation System Confererence, Washington, D.C. USA, Oct. 3-6,
2004.

M. Fu, J. Ruan, and Y. Li, (2004b)
A new kind of robust design method of intelligent vehicle lateral control,
Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on
Volume 3, 15-19 June 2004 Page(s):2438 - 2442 Vol.3

D.N. Godbole, R Sengupta, and V. Hagenmeyer, (1998)
Distributed Hybrid Control for Automated Vehicle Lane Changes,
Proc. Of the 37th IEEE Conference on Decision and Control, Tampa, Florida USA,
Dec. 1998. pages 2639-2644

J. Guldner, H.S. Tan, S. Patwardhan, (1997)
On Fundamental Issues of Vehicle Steering Control for Highway Automation,
PATH Working Paper, UCB-ITS-PWP-97-11.

J. Guldner, W. Sienel, H.S. Tan, J. Ackermann, S. Patwardhan, and T. Bunte, (1999)
Robust Automatic Steering Control for Look-down Reference Systems with Front and
Rear Sensor,
IEEE Trans. on Control Systems Technology, Vol. 7, No. 1, Jan. 1999.

V. Giordano, D. Naso, and B. Turchiano, (2006)
Combining Genetic Algorithms and Lyapunov-Based Adaptation for Online Design of
Fuzzy Controllers,
IEEE Trans. on system, Man, and Cybernetics – Part B: Cybernetics, Vol. 36, No. 5,
Oct. 2006.

C. Hatipoglu, K. Redmill, and U. Ozguner, (1998)
Steering and Lane Change: A Working System
IEEE of Proceedings of ITSC Conference, 1998. pp. 272-277

C. Hatipoglu, U. Ozguner, and K. A. Redmill, (2003)
Automatic Lane Change Controller Design
IEEE Trans. on Intelligent Transportation Systems

J. I. Hernandez and C.Y. Kuo, (2003)
Steering Control of Automated Vehicless Using Absolute Positioning GPS and
Magnetic Markers,
IEEE Trans. on Vehicular Technology, Vol. 52, No. 1, Jan. 2003.

F. Herrera, J.L. Verdegay, (1996)
Genetic Algorithms and Soft Computing,
Heidelberg, Physica-Verlag

T. Hessburg and M. Tomizuka, (1994)
Fuzzy Logic Control for Lateral Vehicle Guidance,
IEEE Control System Magazine. 1994

T. Hessburg, H. Peng, M. Tomizuka, and W.B. Zhang, (1991)
An Experimental Study on Lateral Control of a Vehicle,
PATH Research Report UCB-ITS-PRR-91-17.

R. Horowitz and P. Varaiya,
Control design of an automated highway system,
IEEE Proceedings of the IEEE, vol. 88, No. 7, July (2000)

J. Huang, and M. Tomizuka, (2005)
LTV Controller Design for Vehicle Lateral Control Under Fault in Rear Sensors,
IEEE Trans. on Mechatronics, Vol. 10, No. 1, Feb. 2005.

J. Jameson, (1990)
A Neurocontroller Based on Model Feedback and the Adaptive Heuristic Critic,
IEEE Int. Joint Neural Network Conference. San Diego, CA June 1990.

H. Jula, E. B. Kosmatopoulos, and P.A. Ioannou, (2000)
Collision Avoidance Analysis for Lane Changing and Merging,
IEEE Trans. on Vehicular Technology, Vol. 49, No. 6, Nov. 2000.

A. Kanaris, E. B. Kosmatopoulos, and P.A. Ioannou, (2001)
Strategies and Spacing Requirements for Lane Changing and Merging in Automated
Highway Systems,
IEEE Trans. on Vehicular Technology, Vol. 50, No. 6, Nov. 2001.

J. Kameko and A. Shimamura, (1998)
A Design of Lane Change Maneuver for Automated Vehicle,
Proc. Of IEEE 37th Conference on Decision and Control, Tampa, Florida USAm Dec.
1998.

C. Karr, (1991)
Genetic algorithms for fuzzy logic controllers,
AI Expert, vol. 6, no. 2, pp.26-33

K.R.S. Kodagoda, W.S. Wijesoma, and E. K. Teoh, (2002)
Fuzzy Speed and Steering Control of AGV,
IEEE Trans. on Control Systems Technology, Vol. 10, No. 1, Jan. 2002.

N. Kehtarnavaz, N. Griswold, K. Miller, and P. Lescoe, (1998)
A Transportable Neural-Network Approach to Autonomous Vehicle Following,
IEEE Trans. on Vehicular Technology, Vol. 47, No. 2, May 1998.

S. Kato, S. Tsugawa, K. Tokuda, T. Matsui, and H. Fujii, (2002)
Vehicle Control Algorithms for Cooperative Driving with Automated Vehicles and
Intervehicle Communication,
IEE Trans. on Intelligent Transportation System, Vol. 3, No. 3, Sept. 2002.

V.K. Koumousis, and C.P. Katsaras, (2006)
A saw-tooth Genetic Algorithm Combining the Effects of Variable Population Size
and Reinitialization to Enhance Performance,
IEEE Trans. on Evolution computation vol. 10 issue 1 Feb. 2006

S. Kumarawadu, and T. T. Lee (2006)
Neuroadaptive Combined Lateral and Longitudinal Control of Highway Vehicle
Using RBF Networks.
IEEE Trans. on Intelligent Transportation System Vol. 7, No. 4, Dec. 2006 Pages
500 – 512.

H.K. Lam and F.H.F. Leung, (2007)
Fuzzy controller with stability and performance rules for nonlinear systems,
Fuzzy sets and Systems. Vol. 158, Issue 2, 16 June 2007 Pages 147-163

G. Lu. And M. Tomizuka, (2005)
Vehicle Following as Backup Control Scheme for Magnet-Magnetometer-Based Lateral Guidance,
IEEE Trans. on Control System Technology, Vol. 13, No. 2, March 2005.

J. Manigel and W. Leonhard, (1992)
Vehicle Control by Computer Vision,
IEEE Trans. on Industrial Electronic, Vol.39, No. 3, June 1992.

N. Moehler, (2006)
Lane detection for a situation adaptive lane keeping support system,
10th International Forum on Advanced Microsystems for Automotive Applications
Berlin, Germany, 25 - 27 April 2006

J.E. Naranjo, C. Gonzalez, R. Garcia, T. de Pedro (2006),
ACC+Stop&go maneuvers with throttle and brake fuzzy control,
IEEE Trans. on Intelligent Transportation System Vol. 7 issue 2 June 2006

W. L. Nelson, (1989)
Continuous Steering-Function Control of Robot Car.,
IEEE Trans. on Industrial Electronics, Vol. 36, No. 3, Aug. 1989. Pages 330-337.

I. Papadimitriou and M. Tomizuka, (2003)
Fast Lane Change Computation using Polynomials,
Proc. Of the American Control Conference, Denver, Colorado June 4-6 2003.,
Pages48-53.

K. M. Passino and S. Yurkovich, (1998)
Fuzzy Control,
Addison Wesley Longman, Inc.

PReVENT official website
<http://www.prevent-ip.org/en/home.htm>

D. E. Rumelhart, G. R. Hinton, and R. J. Williams, (1986)
Learning Internal Representation by Error Propagation,
Parallel Digital Processing, Vol. 1, D. E. Rumelhart and J. L. McClelland, ed., MIT
Press, Cambridge, MA, 1986

- L. Saxton, (1993)
“Mobility 2000 and the roots of IVHS”.
IVHS Reviews, p.11-26, 1993
- S.E. Shladover, C.A. Desor, J.K. Hedrick, M. Tomizuka, J. Walrand, W.B. Zhang, D.H. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown, (1991)
Automatic Vehicle Control Development in the PATH Program,
IEEE Trans. on Vehicular Technology, Vol. 40, No. 1, Feb. 1991.
- S. E. Shladover, (1995)
Review of the State of Development of Advanced Vehicle Control Systems (AVCS),
Vehicle Systems Dynamics, 24 (1995), pp. 551-595
- T. Shamir, (2004)
How Should an Autonomous Vehicle Overtake a Slower Moving Vehicle: Design and
Analysis of an Optimal Trajectory,
IEEE Trans. on Automatic Control, Vol. 49, No. 4, April. 2004.
- S. K. Sharma, and G. W. Irwin, (2003)
Fuzzy Coding of Genetic Algorithms,
IEEE Trans. on Evolutionary Computation, Vol. 7, No. 4, August 2003
- S. Suryanarayanan, M. Tomizuka, and T. Suzuki, (2004)
Design of Simultaneously Stabilizing Controllers and Its Application to Fault-Tolerant
Lane-Keeping Controller Design dot Automated Vehicles,
IEEE Trans. on Control System Technology, Vol.12, No. 3, May 2004.
- A. Sonbol, and M.S. Fadali, (2006)
Stability Analysis of Discrete TSK Type II/III Systems,
IEEE Trans. on Fuzzy Systems vol. 14 issue 5. Oct. 2006
- H-S. Tan, J. Guldner, S. Patwardhan, C. Chen, and B. Bougler, (1999)
Development of an Automated Steering Vehicle Based on Roadway Magnets – A Case
Study of Mechatronic System Design,
IEEE Trans. on Mechatronic, Vol.4, No. 3, Sept. 1999

Y-W. Teng, and W-J. Wang, (2004)
Constructing a User-Friendly GA-Based Fuzzy System Directly From Numerical Data,
IEEE Trans. on Systems, Man, And Cybernetics—Part B. Cybernetics, Vol. 34, No. 5,
October 2004

A. Tahirovic, S. Konjicija, Z. Avdagic, G. Meier, and C. Wurmthaler, (2005)
Longitudinal Vehicle Guidance using Neural Networks,
IEEE Proceeding on Inter. Symposium on Computational intelligence in Robotics and
Automation, Espoo, Finland, June 27-30 2005.

The Annual Transport Digest 2006 from
http://www.td.gov.hk/mini_site/atd2006/main_en.htm

S. Tsugawa, (1994)
Vision Based Vehicles in Japan: Machine Vision Systems and Driving Control
Systems,
IEEE Trans. on Industrial Electronic, Vol. 41, No. 4, Aug. 1994.

J.T. Tsai, J.H. Chou, and T.K. Liu, (2006)
Tuning the Structure and Parameters of a Neural Network by Using Hybrid
Taguchi-Genetic Algorithm,
IEEE Trans. on Neural Network, Vol. 17, No. 1, Jan. 2006.

L.H. Tsoukalas and R. E. Uhrig, (1997)
Fuzzy and Neural Approaches in Engineering,
A Wiley-Interscience Publication

M. N. Uddin, M. A. Abido, and M. A. Rahman, (2005)
Real-Time Performance Evaluation of a Genetic-Algorithm-Based Fuzzy Logic
Controller for IPM Motor Drives,
IEEE Trans. on Industry Application, Vol. 41, No. 1, Jan. 2005.

A. Vahidi and A. Eskandarian, (2003)
Research Advances in Intelligent Collision Avoidance and Adaptive Cruise Control,
IEEE Trans. on Intelligent Transportation System, Vol. 4, No. 3, Sept. 2003

L.X. Wang, (1997)

A Course in Fuzzy Systems and Control,
Upper Saddle River, N.J. :Prentice Hall

Q. Wang, M. Mi, G. Ma, and P. Spronck, (2004)

Evolving a neural Controller for Ball-and-Beam System,
Proc. of third Int. Conference on Machine Learning and Cybernetics, Shanghai, Aug.
2004.

P. Werbos, (1974)

Beyond Regression: News Tools for Prediction and Analysis in the Behavioral
Sciences,
Ph.D. Dissertation, Harvard University, Boston

B.A. Whitehead and T. D. Choate, (1996)

Cooperative-Competitive Genetic Evolution of Radial Basis Function Centers and
Widths for Time Series Prediction,
IEEE Trans. on Neural Network, Vol. 7, No. 4, July 1996.

G. G. Yen and H. Lu., (2000)

Hierarchical Genetic Algorithm Based Neural Network Design,
IEEE Symposium on Combinations of Evolutionary Computation and Neural
Network, May 11-13 2000.

G. Zhenhai and Z. Bo, (2005)

Vehicle Lane Keeping of Adaptive PID Control with BP Neural Network Self-Tuning,
IEEE Proceeding of Intelligent Vehicle Symposium, 6-8 June 2005.