Vehicle following : control design, simulation and experiments

Pham, Minh Tuan

2006

Pham, M. T. (2006). Vehicle following : control design, simulation and experiments. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/39113>

<https://doi.org/10.32657/10356/39113>

**Vehicle Following:
Control Design, Simulation and Experiments**

Pham Minh Tuan

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfilment of the requirement for the degree of
Doctor of Philosophy

2006

Vehicle Following: Control Design, Simulation and Experiments

Pham Minh Tuan

School of Electrical and Electronic Engineering

A thesis submitted to Nanyang Technological University
in fulfillment of the requirements for the degree of
Doctor of Philosophy

2005

Dedicated to
My dear wife Trang and little son Thi

Acknowledgements

I would like to express my deeply gratitude to my supervisor, Associate Professor Wang Dan Wei, for his invaluable and patient guidance throughout the study. His discussions on every issue were always helpful and put forward the way to the solutions.

I am also greatly indebted to School of Electrical and Electronic Engineering, Division of Control and Instrumentations for the provision of budgets and equipments to the research project and of the scholarship for the author.

Many thanks are given to the colleagues: Dr. He Bo, Mr. Yu Tienniu, and Mr. Low Chang Boon as well as all the technicians in M&E Workshop: Mr. Tan Chai Soon, Mr. Chia Chiu Kiat, and Mrs. Lim-Tan Geok Lan, Janet. They were always being around for constructive discussions on theoretic issues and actual hands-on work on repairing and maintaining the vehicles and carrying out experiments.

Last but not least, I would like to thank the FYP students who worked with me along the years, especially Mr. Tan Chee Hau and Mr. Ho Weng Ming. Their contributions to the project have always gained my appreciation and thankfulness.

Thank you all for being with me, always.

Summary

The thesis addresses the development and implementation of vehicle-following controllers for a fleet of two vehicles, a leader vehicle and a follower vehicle. Critical issues are studied including theoretical development, simulation study and experimental implementation for the proposed controllers.

The thesis starts off with a comprehensive review of previous work on the vehicle-following control systems which is normally divided into two independent control designs: longitudinal control and lateral control; as well as the issue of fault detection and identification and fault-tolerance for such systems, with applications mainly to car-like and/or land wheeled vehicles. Subsequently, this thesis presents its two main parts: the development and implementation of vehicle-following controllers for car-like vehicles and the development and implementation of a fault-tolerant vehicle-following controller for four-wheel-steering vehicles.

- **Vehicle-following control for car-like vehicles:** The kinematics-based and dynamics-based vehicle-following controllers are developed based on the definition of a focus point in front or behind the follower vehicle. The focus point is a function of the vehicle position as well as the steering angle. By controlling the follower vehicle in such a way that this focus point is able to track a reference point on the leader vehicle, the vehicle-following is proven to be successful for both look-ahead forward tracking and look-behind backward tracking. The feedback control laws integrate both longitudinal control and lateral control as one to make it possible for the vehicle to achieve any maneuvers. The effects of system parameter selections are also investigated especially for two basic tracking maneuvers: straight and circular trajectories. The results reveal that the necessary conditions to achieve tracking convergence are intuitive and similar to the human driving practices, for instance the driver looks ahead when following a car in front or he will not focus far away when engaging in a tight turn.

The proposed controllers are then simulated on a pseudo-reality co-simulation platform.

On one hand, instead of modeling the vehicles by using differential equations, the vehicles and the working environment are modeled in ADAMS in such a way similar to their normal construction process in practice of assembling designed and selected components and materials. On the other hand, the control algorithms are programmed comfortably in SIMULINK, with the vehicle system model imported and represented by a functional block. The co-simulation platform provides a high-fidelity simulation environment that can replicate the actual vehicle system and produce almost precisely what the behaviours could be when the control algorithm is implemented. Based on this simulation platform, the vehicle-following controllers are extensively studied with different sets of control parameters for different tracking situations. The results validate the effectiveness of and offer interesting insights from the proposed controllers.

The controller is also implemented on the experimental vehicle-following system, Cycab vehicles. A sensing system is designed to detect and measure the relative distance and orientation between two vehicles. The sensing system utilizing a laser scanner and reflective tapes provides a fast updating rate of $30Hz$, with the accuracy of $2cm$ for distance of up to $80m$, and 0.5° for the angle. Besides, by using three reflective tapes to define the tracked target, the reliability against noise and disturbances of the sensing system is significantly improved. The experiments are then carried out to illustrate the efficiency of the proposed controller.

- **Fault-tolerant vehicle-following control for four-wheel-steering vehicles:**

A fault-tolerant vehicle-following controller for four-wheel-steering vehicles is developed based on the one for car-like vehicles. The controller is able to carry on its vehicle-following task even if one of its driving systems or steering systems is at fault. The controller utilizes the additional steering actuator to switch among three steering modes, front-wheel-steering, rear-wheel-steering and four-wheel-steering, depending on the working condition of the two steering systems. The working principles of driving motors are exploited to gain the fault tolerance capacity for the driving control. The necessary conditions are constructed from the combination of conditions for each faulty situations.

In order to implement the fault-tolerant controller, the low-level control system of the four-wheel-driving four-wheel-steering vehicle is redesigned using a decentralized control architecture. Both hardware and software are developed in such a way that when a

fault occurs at any part of the system, it will be detected and isolated while the remaining parts of the vehicle continue to operate. Based on the physical structure of the vehicle, two separate but cooperative low-level controllers are built up to manage the front and rear sets of driving and steering systems. Each controller by itself can manipulate the vehicle, if the other is not working. When both controllers are working, they communicate with each other to cooperate in controlling the vehicle. The vehicle-following maneuver is supervised by the proposed controller which integrates all the fault diagnostic information from the vehicle to adjust its parameters and control commands. Simulation and experimental results show the efficiency of the fault-tolerant vehicle-following system.

Before the end of the thesis, recommendations are given for future work on the vehicle-following control and fault tolerant control.

Contents

Acknowledgements	i
Summary	ii
Contents	v
List of Figures	ix
List of Tables	xv
List of Symbols	xvii
List of Theorems/Lemmas	xix
1 Introduction	1
1.1 Background	1
1.2 Previous work	4
1.2.1 Vehicle following controllers	4
1.2.2 Fault-tolerant control of a vehicle following system	9
1.3 Contributions of This Thesis	12
1.4 Outline of This Thesis	14
I Vehicle-following system of car-like vehicles	16
2 Unified Vehicle-Following Controller or Car-Like Vehicles	17
2.1 Introduction	17
2.2 Vehicle model	18
2.2.1 Vehicle structure and kinematic constraints	18

Contents	vi
2.2.2 Kinematic model of a car-like vehicle	18
2.2.3 Dynamic model of a car-like vehicle	31
2.3 Development of vehicle-following controllers	34
2.3.1 Vehicle-following situations	34
2.3.2 Controller development	36
2.3.3 Following a circular trajectory	56
2.3.4 Requirement of practical measurements	66
2.4 Conclusion	67
3 A High-Fidelity Simulation Platform for Vehicle Control Algorithms	68
3.1 Introduction	68
3.2 ADAMS®: vehicle modeling and environment design	69
3.2.1 Vehicle modeling	69
3.3 SIMULINK®: high-level controller design	75
3.4 Closed-loop system modeling and co-simulation	76
3.4.1 Closed-loop system modeling	76
3.4.2 Co-simulation	79
3.5 Simulation Study	81
3.5.1 Kinematics-based	82
3.5.2 Dynamics-based	91
3.6 Conclusions	93
4 Implementation of a Vehicle-Following Controller	106
4.1 The low-level Control System of the Blue Cycab	106
4.1.1 Hardware configuration	107
4.1.2 Software configuration	109
4.1.3 Communication and control of the Cycab	110
4.2 The sensing system of relative measurements	112
4.2.1 Selection of sensor	112
4.2.2 Using a laser scanner and reflective tapes	114
4.3 Experimental results	127
4.3.1 Setting up	127
4.3.2 Experimental results and discussions	129

Contents	vii
4.4 Conclusions	132
II Fault-tolerant control system for a four-wheel-steering vehicle	133
5 Fault-Tolerant Control Design for a Four-Wheel-Steering Vehicle	134
5.1 Introduction	134
5.2 Vehicle operation and vehicle model	135
5.2.1 Operation principles of Cycab's driving and steering systems	135
5.2.2 Vehicle model	136
5.3 Fault-tolerant vehicle-following controller	144
5.4 Simulation studies	151
5.4.1 Simulation faults on the co-simulation platform	152
5.4.2 Setting-up	152
5.4.3 Results	154
5.5 Conclusions	156
6 Experiments on a Distributed Four-Wheel-Driving Four-Wheel-Steering Vehicle	157
6.1 Introduction	157
6.2 Design of fault-diagnostic low-level control system	157
6.2.1 Driving control	159
6.2.2 Steering control	161
6.2.3 Emergency stop subsystem	164
6.2.4 Communication among controllers	167
6.2.5 Software configuration for Red CyCab	167
6.2.6 Design of a fault detection and isolation system for the vehicle	168
6.3 Experiments	175
6.3.1 Fault simulation	175
6.3.2 Experimental results	176
6.4 Conclusion	179
7 Conclusions and Recommendations	181
7.1 Conclusions	181
7.2 Recommendations for further research	183

Contents	viii
Author's Publications	185
Bibliography	187

List of Figures

2.1	Car-like vehicle model	19
2.2	The kinematic constraints of the car-like vehicle	20
2.3	Instantaneous motion of the vehicle	24
2.4	Mapping set from $ \theta + \beta(\gamma_f) < \frac{\pi}{4}$ and $ v < 1$	28
2.5	Look-ahead vehicle-following configuration	34
2.6	Look-behind vehicle-following configuration	34
2.7	Look-ahead focus point	36
2.8	Look-behind focus point	36
2.9	Necessary conditions for circular trajectory look-behind tracking	61
2.10	Necessary conditions for circular trajectory look-ahead tracking	65
3.1	Body of a vehicle	70
3.2	Driving wheel model	71
3.3	Coulomb model of friction coefficient	72
3.4	Main steering joint	74
3.5	Links to a wheel	74
3.6	Steering system model	75
3.7	ADAMS model of vehicle platooning	75
3.8	Controller in SIMULINK	75
3.9	Closed-loop system in SIMULINK	76
3.10	ADAMS blocks	77
3.11	Command Inputs to ADAMS	78
3.12	Feedbacks from ADAMS	79
3.13	SIMULINK model	81
3.14	Necessary conditions for circular trajectory look-ahead tracking when $\gamma_d = 0.2$	83

List of Figures	x
3.15 Necessary conditions for circular trajectory look-behind tracking when $\gamma_d = 0.2$	86
3.16 Tracking error for $l = 4$ and $p = 2.8462$	89
3.17 Steering angle for $l = 4$ and $p = 2.8462$	89
3.18 Tracking error for $l = -4$ and $p = -1.5625$	89
3.19 Steering angle for $l = -4$ and $p = -1.5625$	89
3.20 Trajectories for $p = 0.5$	94
3.21 Tracking error for $p = 0.5$	94
3.22 Velocities for $p = 0.5$	94
3.23 Steering angles for $p = 0.5$	94
3.24 Trajectories for $p = 1.5$	94
3.25 Tracking error for $p = 1.5$	94
3.26 Velocities for $p = 1.5$	95
3.27 Steering angles for $p = 1.5$	95
3.28 Trajectories for $p = 2.5$	95
3.29 Tracking error for $p = 2.5$	95
3.30 Velocities for $p = 2.5$	95
3.31 Steering angles for $p = 2.5$	95
3.32 Trajectories for $l = 2$	96
3.33 Tracking error for $l = 2$	96
3.34 Velocities for $l = 2$	96
3.35 Steering angles for $l = 2$	96
3.36 Trajectories for $l = 4$	96
3.37 Tracking error for $l = 4$	96
3.38 Velocities for $l = 4$	97
3.39 Steering angles for $l = 4$	97
3.40 Trajectories for $l = 6$	97
3.41 Tracking error for $l = 6$	97
3.42 Velocities for $l = 6$	97
3.43 Steering angles for $l = 6$	97
3.44 Trajectories for $l = 7$	98
3.45 Tracking error for $l = 7$	98
3.46 Velocities for $l = 7$	98

List of Figures	xi
3.47 Steering angles for $l = 7$	98
3.48 Trajectories for $p = -1$	98
3.49 Tracking error for $p = -1$	98
3.50 Velocities for $p = -1$	99
3.51 Steering angles for $p = -1$	99
3.52 Trajectories for $p = -2$	99
3.53 Tracking error for $p = -2$	99
3.54 Velocities for $p = -2$	99
3.55 Steering angles for $p = -2$	99
3.56 Trajectories for $l = -1$	100
3.57 Tracking error for $l = -1$	100
3.58 Velocities for $l = -1$	100
3.59 Steering angles for $l = -1$	100
3.60 Trajectories for $l = -4$	100
3.61 Tracking error for $l = -4$	100
3.62 Velocities for $l = -4$	101
3.63 Steering angles for $l = -4$	101
3.64 Trajectories for $l = -7$	101
3.65 Tracking error for $l = -7$	101
3.66 Velocities for $l = -7$	101
3.67 Steering angles for $l = -7$	101
3.68 Trajectories for $p = 2$	102
3.69 Tracking error for $p = 2$	102
3.70 Velocities for $p = 2$	102
3.71 Accelerations for $p = 2$	102
3.72 Steering angles for $p = 2$	102
3.73 Heading angles for $p = 2$	102
3.74 Trajectories for $l = 2.5m$	103
3.75 Tracking error for $l = 2.5m$	103
3.76 Velocities for $l = 2.5$	103
3.77 Accelerations for $l = 2.5$	103
3.78 Steering angles for $l = 2.5$	103

List of Figures

xii

3.79	Heading angles for $l = 2.5$	103
3.80	Trajectories for $p = -1.5$	104
3.81	Tracking error for $p = -1.5$	104
3.82	Velocities for $p = -1.5$	104
3.83	Accelerations for $p = -1.5$	104
3.84	Steering angles for $p = -1.5$	104
3.85	Heading angles for $p = -1.5$	104
3.86	Trajectories for $l = -2.5m$	105
3.87	Tracking error for $l = -2.5m$	105
3.88	Velocities for $l = -2.5$	105
3.89	Accelerations for $l = -2.5$	105
3.90	Steering angles for $l = -2.5$	105
3.91	Heading angles for $l = -2.5$	105
4.1	The CyCab vehicles	107
4.2	Original hardware configuration of CyCab	108
4.3	The mounting of a driving motor	108
4.4	The mechanism of a steering system	109
4.5	Software configuration of CyCab	110
4.6	The SICK laser scanner	115
4.7	Scanning direction of the laser scanner	115
4.8	A scan from laser scanner	116
4.9	The detection and measurement of the reflective tapes	119
4.10	The distance measurements in a stationary situation	120
4.11	The angle measurements in a stationary situation	121
4.12	The heading-difference measurements in a stationary situation	121
4.13	Sampling time of the method	122
4.14	Trajectories for $p = 1.5$	130
4.15	Heading angles for $p = 1.5$	130
4.16	Velocities for $p = 1.5$	130
4.17	Steering angles for $p = 1.5$	130
4.18	Vehicular spacing for $p = 1.5$	130
4.19	Relative angle ϕ for $p = 1.5$	130

List of Figures	xiii
4.20 Trajectories for $p = 2$	131
4.21 Heading angles for $p = 2$	131
4.22 Velocities for $p = 2$	131
4.23 Steering angles for $p = 2$	131
4.24 Vehicular spacing for $p = 2$	131
4.25 Relative angle ϕ for $p = 2$	131
5.1 The vehicle model	137
5.2 Values of p_f, p_r to guarantee tracking stability	153
5.3 Faults occurred against time	154
5.4 Trajectory ζ of the vehicles	155
5.5 Absolute output tracking error $ \tilde{z} $	155
5.6 Velocity v of the vehicles	155
5.7 Steering angles γ_f, γ_r of the vehicles	155
6.1 Red Cycab low-level controllers: The two towers	159
6.2 Block diagram of the driving system	159
6.3 Schematic design of the mapping circuit	160
6.4 Block diagram of the steering system	162
6.5 Schematic diagram of the encoder output converter	163
6.6 The Emergency Stop subsystem	166
6.7 Transmission Path	166
6.8 Red Cycab's low-level control system	168
6.9 Schematic diagram of fault diagnostic circuit for Curtis amplifier	171
6.10 Timing of simulated faults	177
6.11 Trajectory ζ of the vehicles	178
6.12 Relative spacing d	178
6.13 Velocity v of the vehicles	178
6.14 Steering angles γ_f, γ_r of the vehicles	178
6.15 Heading angle θ of the vehicles	178
6.16 Relative angle ϕ	178
6.17 Trajectory ζ of the vehicles	180
6.18 Relative spacing d	180

List of Figures	xiv
6.19 Velocity v of the vehicles	180
6.20 Steering angles γ_f, γ_r of the vehicles	180
6.21 Heading angle θ of the vehicles	180
6.22 Relative angle ϕ	180

List of Tables

2.1	Parameters of wheel configuration	21
3.1	Torus’s parameter settings to model a tyre	71
3.2	Dataflow from SIMULINK to ADAMS	78
3.3	Dataflow from ADAMS to SIMULINK	80
3.4	Variables for display	80
3.5	Tracking results for selected sets of (l, p) for look-ahead tracking with p priori chosen	83
3.6	Tracking results for selected sets of (l, p) for look-ahead tracking with l priori chosen	85
3.7	Tracking results for selected sets of (l, p) for look-behind tracking with p priori chosen	87
3.8	Tracking results for selected sets of (l, p) for look-behind tracking with l priori chosen	88
4.1	Specifications of CyCab	108
4.2	Frequently used commands of Blue Cycab	111
4.3	Specifications of the SICK LMS221 Type-6 laser scanner	115
4.4	Reflectivity of some materials	117
5.1	Parameters of wheel configuration for 4WS vehicle	137
6.1	The resistors used in the mapping circuit	160
6.2	Voltage Mapping of the driving system	161
6.3	PID gains used for the driving system	161
6.4	PID gains used for the steering systems	164
6.5	Truth table of the logic brake control circuit	166

List of Tables	xvi
6.6 Frequently called functions of the POSYS card	169
6.7 Status fault codes of the amplifier	172
6.8 Commands in C++ to read the diagnostic results of driving system	173
6.9 Commands in C++ to read the diagnostic signals of steering system	174

List of Symbols

Symbol	Unit	Definition
β	rad	Side-slip angle
$\gamma_{fl}, \gamma_{fr}, \gamma_{rl}, \gamma_{rr}$	rad	Steering angle of the front left, front right, rear left and rear right wheels, respectively
γ_f, γ_r	rad	Steering angle of the virtual front and rear wheels respectively
γ_d	rad	steering angle of the leader vehicle
γ_{max}	rad	limit of all steering angles
θ	rad	heading angle
κ	m^{-1}	instantaneous curvature of the vehicle i.e. inverse of the instantaneous turning radius r
λ, ξ		coefficients of the first-order (λ only) and second-order (with ξ) closed loop systems
$\omega_f, \omega_r, \omega$	rad/s	front and rear and common steering rates
ϕ	rad	relative angle between the leader and follower vehicles
$\dot{\phi}$	rad/s	rotating speed of a wheel
τ	N	generalized forces
$\mu = [\ v \ \omega \]^T$	$[\ m/s \ rad/s \]^T$	velocity control inputs
$\eta = [\ \theta \ \gamma_f \]^T$	rad	
$\zeta = [\ x \ y \]^T$	m	global position of the vehicle
$\psi = [\ x \ y \ \theta \]^T$	$[\ m \ m \ rad \]^T$	the global posture of the vehicle

List of Symbols

xviii

Symbol	Unit	Definition
$2a$	m	Distance between front and rear pairs of wheels
a_f, a_r	m	Distances from the vehicular reference point to the front and rear axes, respectively, $a_f + a_r = 2a$
$2b$	m	Distance between left and right wheels
d	m	relative distance between the two vehicles
$d_{\max}$	m	maximum range of the laser scanner
f		tracking mode: 1 for look-ahead and -1 for look-behind
l	m	desired spacing between two vehicles
p		desired angle ratio, (l, p) defines the focus point based on steering angle for car-like vehicle
p_f, p_r		desired angle ratio, (l, p_f, p_r) defines the focus point based on steering angles for 4WS vehicle
r	m	instantaneous turning radius
v	m/s	velocity of the follower vehicle
v_d	m/s	velocity of the leader vehicle
q		state representation of the follower vehicle, $q = [x \ y \ \theta \ \gamma_f \ \gamma_r]^T$
q_d		state representation of the leader vehicle
u_m, u_s		linear and steering accelerations
$u = [u_m \ u_s]^T$	$[m/s^2 \ rad/s^2]^T$	acceleration control inputs
z	m	position of the focus point
z_d	m	position of the leader vehicle
$R(\theta)$		2-by-2 standard rotation matrix
$R_3(\theta)$		3-by-3 standard rotation matrix
k_{sf}, k_{sr}		boolean value of working status of the front and rear steering systems

List of Theorems/Lemmas

Theorem/Lemma	Brief Description	Page
Lemma 2.2.1	Conditions for a car-like vehicle to satisfy kinematic constraints	21
Lemma 2.3.2	Conditions to obtain the equivalence of tracking error convergence and desired spacing achievement for car-like following	38
Lemma 2.3.3	Conditions to make matrix $\bar{E}$ invertible and the controller for car-like vehicle existent	42
Lemma 2.3.4	Additional conditions to obtain state-tracking convergence.	45
Lemma 2.3.5	Proof of the state-tracking convergence for the straight-line trajectory under given conditions	48
Lemma 2.3.6	Proof of the convergence of the turning center for tracking on a circular trajectory under given conditions	49
Theorem 2.3.8	Description of the kinematic-based controller for car-like vehicle with necessary conditions of design parameters	52
Theorem 2.3.9	Description of the dynamic-based controller for car-like vehicle with necessary conditions of design parameters	56
Lemma 2.3.10	Additional conditions to obtain state-tracking convergence for look-behind tracking	58
Theorem 2.3.11	Conditions to achieve exact following of a circular trajectory for look-behind tracking	59
Lemma 2.3.12	Tracking behaviour for look-behind situation with $-2a \leq l < 0$	59

List of Theorems/Lemmas		xx
Theorem/Lemma	Brief Description	Page
Lemma 2.3.13	Additional conditions to obtain state-tracking convergence for look-ahead tracking	61
Theorem 2.3.14	Conditions to achieve exact following of a circular trajectory for look-ahead tracking	63
Lemma 2.3.16	Tracking behaviour for look-ahead situation with $0 < p < 2$	63
Lemma 5.3.1	Conditions to obtain the equivalence of tracking error convergence and desired spacing achievement for 4WS following	145
Lemma 5.3.2	Conditions to make matrix $\bar{E}$ invertible and the controller for 4WS vehicle existent	147

Chapter 1

Introduction

1.1 Background

Ever since the first automobile was invented by a French man named Nicolas Cugnot in 1769, people have been perpetually trying to improve cars, today's most popular land transportation means, with new designs, new power sources and new features. The various vehicles have been not only faster, safer and more eye-catching but also more reliable and efficient, less fuel consuming, cleaner to the environment and especially they are getting "smarter" in the sense that they are able to assist human drivers in decision making or even taking over the wheel to drive the vehicle with or without human supervision. Passengers just sit back and enjoy the ride. Together with the quick evolution of computers and computer-based controllers, this paradigm has been becoming more and more realistic. Many research efforts have been put on this interesting area of autonomous vehicles over the past decades. The very first autonomous vehicles were addressed to the Stanford Cart built in 1979 (derived from the Stanford's Shakey mobile robot which was built in 1970) and the CMU Rover built in 1983 [1]. Though being primitive, they put a golden landmark in the history of autonomous vehicles.

Since then, many works have been carried out over the globe in theory development as well as in practical implementation and commercial production that involve in different key issues of autonomous vehicles such as navigation and cartography, sensors and sensing strategies, motion planning and vehicle guidance, and fault management, a recent emerging topic. These issues are closely coupled since they all work on the same object, the vehicle, and serve for the same goal, to complete the vehicle's designated tasks. Amongst the tasks of an autonomous vehicle, tracking control is one of the most challenging tasks. Typically, the tracking problem can be

classified into 4 categories as follows

1. **point tracking:** the vehicle must travel to and stop at a desired configuration in its state coordinate system, e.g parking problem, from an initial configuration. This is, in fact, a stabilization problem. Unfortunately, this problem does not satisfy Brockett's necessary condition [2] for stabilization via smooth time-invariant feedback. It means unsmooth and/or time-varying feedback controllers should be included to tackle the problem [3–6]. A widely used method to deal with this point-tracking problem is to define an artificial potential fields whose gradient is attractive to the goal point and repellent from the obstacles posed along the way [7–9].
2. **path following:** the vehicle is driven to reach and follow a purely geometric path, that is independent of timing law, from its initial configuration on or off the given path. Since only the lateral dynamics is concerned, the path following problem can be solved by applying a stabilization controller [10–14]. Further improvement on the performance has been done with the path being modeled or estimated using clothoids, which are the curves whose curvature varies linearly with the arc length. Some remarkable results were presented in [14–16].
3. **trajectory tracking:** the vehicle needs to follow a geometric path with an associated time law, from its initial configuration. In [17], it was proven that in order to guarantee its exact reproducibility, the desired cartesian trajectory should be three time differentiable almost everywhere. With assumption that the desired velocity is not zero, different methods have been developed such as approximate linearization-based methods [18], exact feedback linearization [19–21] and full-state tracking [22,23].
4. **vehicle following:** the vehicle needs to follow another moving vehicle, called *the leader vehicle*, while maintaining a safe distance away from that vehicle. Since the leader vehicle moves around with respect to time, this vehicle following can be considered as a trajectory following problem for the follower vehicle with an output function. In general, the trajectory of the leader vehicle is not priori known or predictable to the follower vehicle, and to retrieve the desired trajectory, the follower vehicle must use some kinds of sensors to detect the leader vehicle and to carry out some measurements of the relative distance and orientation between the two vehicles. Then controllers taking into account this information as inputs are developed to drive the vehicle accordingly. When a few

vehicles follow one another, it is said that a platoon (or a convoy) of vehicles is formed. A platoon is composed of a leader vehicle and several follower vehicles each of which needs to keep a certain distance from its immediately preceding vehicle. The advantage of platooning maneuver in traffic is apparent especially when cars run closely on a highway. In [24] Varaiya pointed out that as the number of vehicles joining the platoon increased the capacity of flow, or the traffic throughput, would be greatly increased. The benefit of platoon driving for congestion dispersion was also investigated in [25]. That is why a lot of attention has been paid to IVHS (Intelligent Vehicle / Highway System) or AHS (Automated Highway Systems) as discussed later. However, when the number of vehicles in a platoon increases, the stability issue of the platoon becomes more important. The vehicle-following controller used in the platoon must make sure that the peak error is not amplified as it passes through the platoon from the very first controller. This problem is normally referred as *string stability* of the platoon. As being examined in [26], the string instability of the platoon is increased when the number of vehicles increases if the vehicle-following controllers use only relative spacing information to its preceding vehicle. When the leader vehicle's information is added to the controller, the error propagation can be controlled so that it will not grow through the platoon. Unfortunately, when disturbance is present, it will have an effect on the string stability no matter what information is in use. The effect of disturbance is only minimized and not propagated when only the leader vehicle's information is used for all the follower vehicles' controller. Thus, there is always a tradeoff between minimizing the disturbance propagation and error propagation when adding one more vehicle to a large platoon.

In this thesis, the focus is on the vehicle following issue of autonomous vehicles for both forward and backward motions in an urban or industrial environments such as those handling materials in warehouses or manufacturing floors, or shuttle bus systems at institutions. In the highway applications, the velocity/acceleration control is more important whereas the steering control is based on a linearized model of the vehicle as it is assumed that the highway is virtually straight. The two controls are normally separated and more attention is paid on the driving control. In contrast, in urban or industrial vehicle-following applications, the designated working path is generally located in a restricted area and with many turns, sometimes even tight turns. Following closely the leader vehicle's trajectory becomes more challenging and requires a more advanced steering controller. Meanwhile, the velocity/acceleration control still plays an equally

important role in the entire system. The idea of combining both controls into one controller is always a good direction to improve the following performance and has received efforts from many researchers. However, because of the increased complexity, there have reportedly been no such integrated vehicle-following controllers that are suitable for this kind of tracking situations, i.e. in an industrial environment with many turns. Moreover, the reliability of the system is much more concerned, especially like those in a fully automated factory. The system should be able to carry on its tasks even if some part of it is at fault. The fault-tolerant functionality can therefore reduce the downtime and hence increase the productivity of the entire factory.

The following section conducts a literature review with focus on the vehicle following problem. Related issues such as longitudinal and lateral controllers as well as fault-tolerant control for vehicle-following, which are of the interests of this thesis, will be discussed.

1.2 Previous work

1.2.1 Vehicle following controllers

As the leader vehicle moves on the planar ground, it creates a 2-dimensional trajectory for the follower vehicle to track and follow. Therefore, the vehicle following controller should be composed of two controls to deal with the 2D motion of the leader vehicle. Very commonly, due to the design of the car or car-like vehicle, the two chosen controls are the longitudinal control and lateral control. The longitudinal control deals with how to accelerate or decelerate the vehicle by controlling the engine throttle, the brake or the motors as for electric vehicles, while the lateral control is in charge of steering the front and/or rear pairs of wheels to adjust the heading orientation of the vehicle on the ground. In vehicle following control, the two controls are equally important. However, the longitudinal control has received more attention from researchers with the assumption that the platoon runs on a straight path, for instance on a highway lane. In this case, the lateral control is to keep the vehicle on the lane by tracking the lane center. The lateral control is more concerned when the platoon is involved with turning and lane changing maneuvers.

1.2.1.1 Longitudinal control

Research on longitudinal control of a string of vehicles begun late in 1960s. The work in [27] presented an optimal control based method to regulate the position and velocity error of each

vehicle to its desired ones. The results were primitive and purely mathematic.

The emergent trend of development on longitudinal control for vehicle following only started early in 1990s with the lead of the PATH (Partners for Advanced Transit and Highways) program in USA [28, 29]. Basically, a longitudinal control defines a desired spacing between two consecutive vehicles and tries to keep the actual distance between the two vehicles at that spacing. At first, the vehicle was modeled using Newton's second law and the control input was the acceleration of the vehicle. The desired spacing was kept constant all the time, thus the error needed to be regulated was the actual distance between two vehicles minus the desired constant spacing. Different controllers were proposed to regulate this error. The most straightforward way was to use a PID-type feedback controller for the velocity tracking [28, 30]. To achieve a better description of the vehicle dynamics, especially for those running on highway, the engine was modeled model by a first-order model and integrated in the controller development [31, 32]. This integration was to cope with the delay of the actuator (engine) of the vehicle. Other methods to deal with the uncertainty of the vehicle model were based on Fuzzy Logic [33] or adaptive control [34–37].

To guarantee the string stability of the platoon, all the mentioned methods needed information from the leader vehicle. Hence, a communication channel was required which is not always available. Moreover, the performance and reliability of the communication system had to be taken under consideration due to their important role in the platoon system [38, 39]. Partial or complete loss or long delay of communication may result in instability of the platoon. Thus, another trend was proposed to avoid the requirement of the leader vehicle's information. The methods that belong to this class utilized *varying spacing policies*. That is it. The desired spacing changed with respect to time. In most cases it was simply proportional to the velocity of the follower vehicle with a coefficient namely *time headway* and an added minimum safety distance. The desired spacing was larger when the vehicles traveled at higher speeds [40–44]. For heavy vehicles like trucks, due to the physical limits of the engine, their feasible acceleration and deceleration are extremely limited. Hence, time headway might not be constant anymore, it was also made dependent on the velocity. The result was the desired spacing became a quadratic function of velocity [45–51]. And with no doubts, if the information of the leader vehicle were also available and integrated, the performance and stability of the platoon would be much improved [52, 53].

1.2. Previous work

6

1.2.1.2 Lateral control

For the highway system applications, the platoon of vehicles stay in a dedicated lane most of the time. Therefore, the longitudinal control has received more attention than the lateral control. The task of lateral controllers are simply to keep the vehicle in the lane. Since all the vehicle in the platoon are supposed to stay in the same lane, for lateral control, the follower vehicles do not require information of the leading or preceding vehicles but the location of the lane center. These controllers are call *Lane-Keeping* controllers. In other words, they are just path-following controllers that take the deviation between the vehicle and the lane center as input to issue steering commands. Since these controllers are independent of the driving (or longitudinal) ones, the methods presented earlier in the review of path-following controllers are applicable. In terms of implementation, to obtain the deviation, there are two ways. In the first way, the path in Cartesian coordinate system was prerecorded and then compared with the current position of the vehicle computed by the onboard inertial navigation system and/or a GPS (Global Positioning System) system [54, 55]. The second way measured directly the derivation. In order to do so, markers were installed and detected by a device onboard the vehicle. The most common device set to detect the lane center was a single camera [56, 57] or a pair cameras [58]. The lane boundaries were detected and then the center calculated with respect to the longitudinal axis of the vehicle. Alternatively, the usage of magnetometers and magnetic patches buried along the lane center was first proposed by the PATH team at University of California, Berkeley [29, 59, 60] and also adopted by the group in Postech University, Korea [33]. This method surpasses the vision-based in terms of processing time and reliability against noise as it is not affected by environmental conditions such as lighting or shadow like how the vision is. However, it introduces more costs and only works on the roads with magnetic markers.

When the road infrastructure is unavailable to support a lane-keeping controller, the follower vehicle has no choice but to steer itself based on its immediately preceding vehicle's movements. This approach was named the *electronic tow-bar* maneuver as the follower vehicle is like being towed by its preceding vehicle with a virtual bar. Basically, the follower vehicle is expected to follow the track of the preceding vehicle. Therefore, if the position coordinates of the preceding vehicle can be made available to the follower vehicle, a simple trajectory generation algorithm can be applied to interpolate an instantaneous constant-curvature trajectory and then to generate the steering command based on that trajectory [61]. However, this assumption is, in practice,

1.2. Previous work

7

not easily satisfied because a good positioning system and a fast communication system are required. Thus, normally the lateral position of the preceding vehicle relative to the follower vehicle is taken as input. The lateral controllers developed based on this input are to regulate this so-called *lateral deviation* to zero. In [41], Daviet and Parent proposed two methods. The first one was a very simple controller that always oriented the wheels directly toward the preceding vehicle. This controller was prove to be stable but also created corner-cutting. The second one was trying to minimize the vehicle's distance to the path of the leader vehicle. The heading derivation relative to the path and the path curvature were taken into account to compute the steering rate command. The stability was guaranteed by a Hurwitz polynomial of the lateral deviation. In the European CHAUFFER project [42], the lateral control was proposed as a PID-like controller with the *Tow-Bar angle* as input and distance- and velocity-dependent PID gains. The controller worked well for truck platooning with distance between two vehicles as small as 6 meters. By using a linearized model of the vehicle, three lateral controllers were proposed by White and Tomizuka [62] for truck following. The first controller was to minimize the relative lateral distance by linearly interpolating the trajectory between the following and preceding vehicles. The second one regulated the projected lateral error by interpolating the trajectory with a circle. And the last one further improved the tracking performance by integrating the relative yaw angle. Though the practical implementation of the third method was almost impossible due to the difficulty in measurements, simulations showed very good results since the follower vehicle was tracking the center of gravity of the preceding vehicle instead of its rear end like how the previous two methods did.

1.2.1.3 Combination of two controls

It is understandable that the longitudinal and lateral controls have been developed separately due their difference in nature. A unified controller that integrates both of the controls are not available at the moment. Moreover, most of the platooning projects focused on AHS applications where the longitudinal control plays a more important role. After the velocity and/or acceleration commands have been generated by the longitudinal control, the lateral control would take it as input in order to calculate the steering command with or without the road infrastructure information. This process worked fine with AHS systems [42, 63, 64]. Yet, it is well known that the longitudinal and lateral dynamics are coupled especially when the vehicle proceeds turns. The degree of coupling is dependent on the severity of the maneuver. Thus, an integrated

controller becomes more desirable. An attempt to integrate longitudinal and lateral controls for highway vehicles was presented in [65]. This was a combination of a longitudinal control and a lane-keeping controller with concerns to bounded disturbances. The work showed the necessity and the increased complexity of such an integrated controller. For applications where the road infrastructure is not available such as those in urban or industrial environments, both longitudinal and lateral controls must rely on the on the relative distance and orientation between two vehicles. An integrated controller that generates driving and steering commands simultaneously becomes challenging. The platoon controller developed at the HONDA company in Japan [66] was a good example of such integrations for urban environment. However, the method required a very good sensing system to calculate accurately vehicle positions and a communication system to transmit among all the vehicles . The approach based on the global position and orientation has been therefore undesirable.

1.2.1.4 Backward vehicle-following

So far, the vehicle-following controllers that have been reviewed were meant for forward driving, i.e. the velocity was positive. Though backward driving have been received much less attention from researchers, it is also a necessity in some situations [67]. For example, on a highway, if there is an accident that blocks the traffic, all the vehicles trapped on the highway must go back to the nearest exit. There may be too many vehicles that the U-turn, especially for a long platoon, could be virtually impossible. That is when the backward following maneuver is needed. Even if the crashed vehicles do not block the entire way, the platoon of vehicles sometimes needs to go backward to have enough space before changing to an open lane. For those vehicles operating in a warehouse, backward driving is even as useful as forward driving due to the restrained working space. It is well known that for a car-like vehicle, backward driving is a much more challenging maneuver. The difficulties of this situation were analyzed carefully in [68] by using root locus method based on a linearized model of the vehicle. The work showed that for a *look-down* path following method, the transfer function of the steering angle to the lateral deviation for forward driving had one double pole at the origin and one stable (negative) zero whereas that of backward driving had also 1 double pole at the origin but with one positive (unstable) zero that made the controller design challenging since it had to create a closed-loop control with all the poles appearing on the left half plane. Thus, to overcome these difficulties, Patwardhan *et. al.* proposed a method that simulated the human practice in han-

dling a boat with a rudder at the rear end to develop a lane-following controller for backward driving. A look-back point coupled with the steering angle was defined and made follow the path. The simulations and experiments illustrated the effectiveness of the proposed controller. Path following of tractor-trailer like vehicles in backward motion were discussed in [69–72].

1.2.2 Fault-tolerant control of a vehicle following system

In the previous section, all the vehicle-following controllers were proposed with an assumption that the sensors and actuators of the vehicles were operational as usual. When a fault occurred, most of these controllers would simply fail to work. For instance, a fault at the steering encoder would make the encoder send out abnormally wrong readings. A closed-loop controller that takes these readings as feedbacks would generate strange steering command to lead the vehicle off the road. Therefore, a fault-tolerant ability is necessary and has recently become an essential part of a control system.

Fault-tolerant control concerns the situation that the plant is subject to some fault, which prevents the overall system to satisfy its goal in the future. A fault-tolerant controller has the ability to react on the existence of the fault by adjusting its activities to the faulty behaviour of the plant. – [73]

The system, as a whole, can not be made fault-tolerant against its own failures because once the failure has already occurred, nothing more can be done. However, it can be made fault-tolerant against the failure of its components. Therefore, a fault-tolerant system is the one that is able to avoid its overall failure when some of its subsystems fail. In general, a fault-tolerant controller is composed of two parts

1. **Fault diagnosis:** to detect whether there is a fault, then locate the fault source and/or measure the size of the fault. Normally, the fault detection and identification are done based on the observation of the input and the output of the components in accordance with their working principles (models).
2. **Control redesign:** to adapt the faulty situation to maintain the the overall goal. The redesign process may start with an activity called *damage confinement* where mechanisms are used to limit the spreading of the identified fault to certain boundaries. The faulty components are separated from the rest of the system to ensure their faults are not prop-

agated to other components. Finally, the controller is changed in such a way to deal with the degraded system where those faulty components are not used.

It is not really surprising that there have been very few works done on the fault-tolerance for a vehicle-following controller of a highway system. It is because, basically, the vehicles in use have only one driving system, e.g. engine, and one steering system. With just one actuator being faulty, the vehicle would not be able to maneuver properly. Thus, most of the reported fault-management systems for land vehicles focused on the first part, fault diagnosis, of the fault tolerance issue. Basically, for such an engineering system like a vehicle, a fault diagnostic method could fall in one of the two categories: model-based and model-free. The model-based methods require a mathematic model of the monitored component or subsystem whereas the model-free ones do not. To deal with the absence of the model, the missing model could be rebuilt from the inputs and outputs using trained neural networks and/or fuzzy logic system [74]. Another type of model-free methods is called *Case-Based Reasoning* in what a huge database of known faulty cases was built up with the detailed description of key features, the fault causes and probably the solution. When a faulty situation is detected, all the required features are acquired and compared with each of the stored cases to find out the best matches. A well-done work in this direction has been addressed to Crossman, Murphey *et. al.* at University of Michigan [75, 76]. The disadvantage of all the model-free methods is that a huge number of data sets, not just of the normal cases but much more preferably of the faulty cases, must be collected beforehand. This can only be done with the tight collaborations from a car company. Once the correct model is made available, it describes closely the kinematic/dynamic behaviours of the vehicle or a vehicle component. The fault, if any, can be detected by checking whether the behaviour is in order or not. That is why the model-based methods have long received more attention from researchers. Again, the model-based methods can break into two types. The first type, namely *parameter change detection*, monitors the inputs and outputs to estimate parameters of the given model. Significant changes of these parameters induce that faults have occurred. The typical applications of this type were those that monitored the tires and/or suspensions of the vehicle [77–79]. The second type, which is more popular, is based on *redundancy*. Redundancy means a few sensors are used to monitor the same subject, e.g signal, supposed that at most only one sensor can be faulty at a time. A simple voting scheme can be applied to detect and locate the faulty sensor. If a term is not directly measurable, it can be estimated based on other measurements provided the relations are explicitly known. Similarly, to detect a faulty actuator,

a model based on the nominal operation of the actuator is first developed. Then *an observer* can be built taking the same input as that of the actuator. The output from the observer and that of the actuator are compared to determine the working condition of the actuator. Many good results for vehicle diagnosis have been reported along this direction over the last decade, [80–86] to name a few. For a vehicle platoon, besides the vehicles themselves, the fault may occur in the communication channel, if there is one in use. Though this kind of faults is not as serious as an actuator fault, it, in fact, affects the platooning maneuvers and stability as discussed earlier. Some work has been reported for this particular problem [84, 87, 88]. However, with today's technologies, this threat can be solved easily and completely.

The fault diagnosis of a vehicle and its components has been studied extensively. For a vehicle-following system, the most complete system should be referenced to good work done at the PATH team [85]. The system integrated lots of works done on fault detection and identification to make sure most of the faults that could happen are detectable and, if possible, the necessary information and measurements are not corrupted due to the faults. On the contrary to the success of fault diagnosis, there have been very few papers on the fault-tolerant issue for a platoon system. In [89], Lygeros *et al.* presented a multiple-layered fault-tolerant control architecture for AHS. For the sake of vehicle safety, faults were classified into 6 groups depending on their severity: *Vehicle Stopped/Must Stop*, *Vehicle Needs Assistance to Exit*, *Vehicle Needs no Assistance to Exit*, *Vehicle Does Not Need to Exit*, *Infrastructure Failures*, and *Driver-Vehicle Interaction Failures*. Based on these classifications, outlined strategies for each control layer were proposed. The strategies revealed that the vehicle would try to use all its remaining resources to continue the operation or to exit from the highway as soon as possible. The fault-tolerant control of a vehicle platoon in [90] presented a task-optimized position/velocity control algorithm where the entire platoon of vehicles was treated as one and the control inputs were generated in order to optimize suitable task functions. The proposal architecture required a centralized controller to collect the position and velocity of the vehicle members and calculate their next position/velocity commands. If a vehicle were found faulty, it would be eliminated from the computations of the controller so that the primary-task error would converge to zero. A global positioning system and a communication system were needed, anyway. Regarding the fault-tolerance of longitudinal and lateral controls for AHS, [91, 92] considered the additive and multiplicative faults that only led to performance degradation not those severe enough to stop the vehicle from being driven. A sliding-mode controller with state estimations were developed

and stability proven even with the presence of degradation. One successful attempt at dealing with faults at sensors while keeping the vehicle following on a lane was done recently at the PATH team [93]. By applying the *Simultaneously Stabilizing Controllers* design method, the controller managed to deal with the loss of one of the two magnetometers, installed onboard to measure the lateral deviation from the dedicated lane's center. The experimental results showed satisfactory performances when the rear magnetometer was at fault. With the front magnetometer faulty, the performance became worse, especially at low speed. It is because the vehicle was in 'look-back' mode (rear sensor) while running forward, which has been proven difficult to be stable.

1.3 Contributions of This Thesis

In this thesis, the development a vehicle following system is worked out regarding different related issues including theoretic development, pseudo-reality simulation, and practical implementation as well as fault-tolerant control.

- **Vehicle-following problem formulation:**

The problem is formulated in such a way that it integrates both key control issues of any vehicle-following systems: the longitudinal control that is responsible for driving commands to follow and keep a certain distance away from the leader vehicle; and the lateral control that is in charge of steering the vehicle to align it with the leader vehicle's motion course. The leader vehicle is referred to as a moving desired point in a polar coordinate system of the follower vehicle. By defining a focus point in the same coordinate system that is at the desired distance from the vehicle and moving in accordance with the steering angle, the vehicle-following problem is converted into the tracking problem of the focus point to the desired point. Moreover, the focus point is also defined in such a way that it unifies two tracking situations: following a vehicle running in front and following a vehicle running behind. The unification facilitates the development of the control laws and the implementation as well by just a simple parameter selection of the desired tracking mode.

- **Vehicle-following control design for car-like vehicles:**

Based on the formulation, unified kinematics- and dynamics-based vehicle-following

controllers for car-like vehicles are developed. The tracking error which is the difference between the focus point and the desired point is regulated through a target closed-loop feedback system. We prove that, under some conditions of design parameters, the tracking convergence is guaranteed. Furthermore, the influence of these parameters on the tracking performance is studied, especially for the two basic situations when the vehicles move on a straight or circular path. It is shown that the selection of parameters may result in the saturation of the state variables or even the instability of the tracking; or it can manages to follow the leader vehicle with different effects such as corner-cutting, overshooting or exact repeating of the leader vehicle's trajectory.

- **Fault-tolerant vehicle-following controller for four-wheel-steering vehicles:**

A unified kinematics-based vehicle-controller for four-wheel-steering vehicles. The focus point is now defined depending on both front and rear steering wheels, which results in the vehicle's ability to follow the leader vehicle even in the case one of the two steering systems malfunction. The proposed velocity and steering coordination mechanisms guarantee that the vehicle is able to utilize all of its operational components and offer the best performance that it could in the presence of faults in some of its components.

- **Pseudo-reality cooperative simulation platform:**

A cooperative simulation platform is designed and developed. The platform integrates the advanced functionalities of two simulation platforms: ADAMS for vehicle and environment modeling and SIMULINK for controller design. The vehicle system and the controller are built separately and then run simultaneously and cooperatively with simulated measurement data and control inputs being exchanged. The physical system is modeled in detail with real parameters and attributes while the SIMULINK model runs as a high-level controller that receives necessary measurements and issues control commands to the low-level simulated vehicle system. Utilizing this platform, an extensive study of the vehicle-following controllers is carried out to verify the analysis and to find out more insights and behaviours of he controllers that are not foreseen by the analysis. The simulation platform is also open and extendable for other controllers.

- **Experimental platform setup and implementation:**

An experimental platform have been setup consisting of two vehicles, namely CyCab vehicles, and a reliable, fast-updating and accurate sensing system using a laser scanner and

strongly reflective tapes to detect and measure relative distance and orientation between two vehicles. One of the vehicles is a car-like vehicle that can be controlled autonomously by a computer via an RS232 communication channel. The other is a four-wheel-steering vehicle with a similar original low-level control system. After the low-level control system is spoiled, a distributed fault-tolerant low-level control system is designed and deployed. Two separate but incorporative controllers are built up to oversee the front and rear parts of the vehicle, respectively. When one fails to work, the vehicle is still able to maneuver by the other. Moreover, the working condition of key components are gathered and made available to the high-level controller in order to make right decisions and carry out necessary changes in control strategy. The vehicle is easy to control by any high-level controller via Ethernet and/or RS232 communication channels.

Based on these platform, the kinematics-based vehicle-following controllers have been implemented in a nearby car-park. The results again verify the effectiveness of the controllers as well as of the experimental platform, especially the sensing system.

1.4 Outline of This Thesis

The rest of this thesis is organized as follows

- **Chapter 2:** A unified vehicle-following controller is developed for car-like vehicles. The controller combines both longitudinal and lateral controls and generates both driving and steering commands at one stroke. The controller is also able to work with both situations: following a vehicle moving forward ahead and following a vehicle moving backward behind. Kinematics-based and dynamics-based controllers are proposed to meet the different requirements of control inputs for different vehicles. Analysis on the stability and tracking results for two fundamental maneuvers, tracking on a straight line and tracking on a circular line, are investigated. Conditions for tracking stability and for the desired of maintaining platoon velocity and stability are investigated.
- **Chapter 3:** A high-fidelity simulation platform which integrates a 3D mechanical design environment, ADAMS, with a powerful simulation software, MATLAB is presented. The platform serves as a pseudo-realistic testing environment to verify and judge the proposed controllers. The controllers developed in Chapter 2 are simulated using this platform. The

results are to verify the mathematic analysis and to gain detailed insights of the tracking performance under different conditions.

- **Chapter 4:** The investigation on the architecture of a car-like four-wheel-driving electric vehicle named Blue CyCab is carried out. The vehicle and another similar one, namely Red Cycab, are used as the test beds for the vehicle-following controllers developed in Chapter 2. The setting-up and results of the experiments are shown in this chapter.
- **Chapter 5:** A unified fault-tolerant vehicle-following controller is derived for the four-wheel-driving four-wheel-steering vehicle. The controller is able to continue its job even when one of the steering subsystems or one of the driving subsystems fails to work. The development of a simulation platform, based on the one in Chapter 3, with some modifications to simulate and implement the faults and fault tolerance, is illustrated.
- **Chapter 6:** The design and development of a distributed fault-tolerant low-level control system for Red Cycab are presented. Subsequently, the experimental setup and implementation are described and the results and discussions shown.
- **Chapter 7:** Conclusions and recommendations for future work are given.

Part I

Vehicle-following system of car-like vehicles

Chapter 2

Unified Vehicle-Following Controller or Car-Like Vehicles

2.1 Introduction

This chapter presents the development of both kinematic-based and dynamic-based vehicle-following controllers for car-like vehicles. Car-like vehicles are mostly used in practice. A car-like vehicle is the one whose two back wheels are used for driving and always kept parallel to the vehicle's longitudinal axle and whose two front wheels are steerable. Basically, the main objective of a vehicle-following controller is to drive a vehicle to follow another while keeping a predetermined distance between the two vehicles. Moreover, if the vehicles involve in turns, the controller must be able to steer the follower vehicle accordingly. Therefore, there are two types of control objectives: one being longitudinal control responsible for driving and the other being lateral control in charge of steering the vehicle. In the previous chapter, many different vehicle-following controllers have been reviewed. They were generally developed independently for either of the two control objectives. In this chapter, a vehicle-following controller that meets both control objectives is presented. The control design mimics human driving practices. It is also able to work when the vehicle attempts to follow another running forward in front of it or running backward behind it.

The chapter presents firstly the development of the vehicle kinematic and dynamic models. Secondly, the vehicle following problem is formulated and a controller is developed. Thirdly, the stability conditions are developed to ensure the stable platoon velocity.

2.2 Vehicle model

2.2.1 Vehicle structure and kinematic constraints

Physically, a vehicle operating on a horizontal plane, the ground, is composed of two main components: its rigid heavy body and its four wheels. In general, the wheels are joined with the body at their centers. During the motion of the vehicle, the body is assumed to be parallel to the ground, while the wheels to be **always in contact** with the ground and rotating around their own axis which is parallel to the ground. As a result, the wheels are perpendicular to the ground. This 2-dimensional scenario is obtained with the assumption that all the 4 tyres are always non-deformable and always in contact with ground.

The wheels of the vehicle are further assumed to belong to the class of *conventional* wheels as described in [94]. A conventional wheel is such a wheel that the contact between it and the plane is reduced to a single point, called *contacting point*, and this contact satisfies that the velocity at the contacting point is zero. The zero-velocity contacting point assumption induces that the two components of this velocity are both zero

- *along the wheel plane*: the physical meaning of the zero velocity along the wheel plane at the contacting point is that the wheel, in fact, rolls purely around the contacting point without any slippage. This is called “**Pure Rolling**” assumption.
- *orthogonal to the wheel plane*: this assumption implies the wheel does not skid along its rotating axis. This is therefore called “**No Skidding**” assumption.

Besides being able to roll on the plane, a conventional wheel may or may not be able to steer around a vertical axis. If the wheel is steerable and the vertical axis across the wheel center, the wheel is said to be of *centered steerable wheels*. If the axis does not contain the wheel point, then the wheel belongs to *off-centered steerable wheels*. If the conventional wheel can not steer, it is called a *fixed wheel*. For our practical vehicles, the wheels always belong to either of the two groups: fixed wheels and center steerable wheels. Thus, from now on, the term *steering wheel* implies that the wheel is centered steerable.

2.2.2 Kinematic model of a car-like vehicle

A car-like vehicle has two back wheels fixed and parallel to the vehicular center axle. Its two front wheels are steerable. The back and front wheels are linked by a center axle namely the

Figure 2.1: Car-like vehicle model

longitudinal axle.

An orthogonal basis coordinate system XOY is fixed on the ground as in Figure 2.1, called the global coordinate system. A point is chosen on the vehicle as *the reference point*. Its coordinates $\zeta = (x, y)$, or in the vector form: $\zeta = [x \ y]^T$, in the global coordinate system is considered as the position of the vehicle. Another coordinate system is fixed on the vehicle with the origin at the reference point and the first principle axis aligned with the longitudinal axle. This coordinate system is called the *vehicular coordinate system*.

The position $\zeta = (x, y)$ and the heading angle θ , formed by the axis OX and the longitudinal axle of the vehicle, describe the vehicle posture, ψ , in the global coordinate system, $\psi = [x \ y \ \theta]^T$.

Very often, the reference point is chosen on the longitudinal axle of the vehicle due to the assumption that the vehicle is symmetric about its longitudinal axis. Suppose its distance to the front and the rear wheels are $a_f \geq 0$ and $a_r \geq 0$, respectively, the length of the vehicle is $a_f + a_r = 2a > 0$. The distance between two front wheels is equal to the distance between two rear wheels and equal to $2b > 0$, the width of the vehicle. All the four wheels are supposed to have the same radius r_φ . If the steering angle of a wheel with respect to the vehicle longitudinal axis is γ , $\gamma = 0$ if the wheel is a fixed wheel, α is a constant angle formed by the longitudinal axis of the vehicle and the line linking the reference point to the wheel center, and φ is the rotating position of the wheel, then the kinematic constraints at the wheel can be expressed in terms of mathematic equations as follows

- along the wheel plane:

$$\begin{bmatrix} \cos \gamma & \sin \gamma & L \sin(\gamma - \alpha) \end{bmatrix} R_3(\theta) \dot{\psi} - r_\varphi \dot{\varphi} = 0 \quad (2.1)$$

2.2. Vehicle model

20

Figure 2.2: The kinematic constraints of the car-like vehicle

- *orthogonal to the wheel plane:*

$$\begin{bmatrix} -\sin \gamma & \cos \gamma & L \cos(\gamma - \alpha) \end{bmatrix} R_3(\theta) \dot{\psi} = 0 \quad (2.2)$$

where L is the distance from reference point to the wheel center, and the $R_3(\theta)$ is an orthogonal rotation matrix

$$R_3(\theta) = \begin{bmatrix} R(\theta) & 0 \\ 0 & 1 \end{bmatrix} \quad (2.3)$$

$$R(\theta) = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \quad (2.4)$$

Note that $R_3(\theta) \dot{\psi}$ represents the longitudinal velocity, the lateral velocity and the yaw rate of the vehicle.

Figure 2.2 illustrates the vehicle kinematics and the parameters for each of the 4 wheels are listed in Table 2.1 with γ_{fl} , γ_{fr} , γ_{rl} and γ_{rr} being the steering angles of the front left, front right, rear left and rear right wheels, respectively, and formed by the vehicular longitudinal direction and the respective wheel's plane.

By applying these parameters to (2.1) and (2.2) and putting these resultant equations into matrix forms, we obtain

$$J_1(\gamma_{fl}, \gamma_{fr}) R_3(\theta) \dot{\psi} - J_2 \dot{\varphi} = 0 \quad (2.5)$$

$$C_1(\gamma_{fl}, \gamma_{fr}) R_3(\theta) \dot{\psi} = 0 \quad (2.6)$$

Table 2.1: Parameters of wheel configuration

Wheel	L	α	γ
Front left	$\sqrt{a_f^2 + b^2}$	$\alpha_{fl} = \arctan \frac{b}{a_f}$	γ_{fl}
Front right	$\sqrt{a_f^2 + b^2}$	$\alpha_{fr} = -\arctan \frac{b}{a_f}$	γ_{fr}
Rear left	$\sqrt{a_r^2 + b^2}$	$\alpha_{rl} = \pi - \arctan \frac{b}{a_r}$	$\gamma_{rl} = 0$
Rear right	$\sqrt{a_r^2 + b^2}$	$\alpha_{rr} = \arctan \frac{b}{a_r} - \pi$	$\gamma_{rr} = 0$

$$J_1 = \begin{bmatrix} \cos \gamma_{fl} & \sin \gamma_{fl} & a_f \sin \gamma_{fl} - b \cos \gamma_{fl} \\ \cos \gamma_{fr} & \sin \gamma_{fr} & a_f \sin \gamma_{fr} + b \cos \gamma_{fr} \\ 1 & 0 & -b \\ 1 & 0 & b \end{bmatrix} \quad (2.7)$$

$$J_2 = r_\varphi I_4 \quad (2.8)$$

where I_4 is a 4-by-4 identity matrix.

$$C_1 = \begin{bmatrix} -\sin \gamma_{fl} & \cos \gamma_{fl} & a_f \cos \gamma_{fl} + b \sin \gamma_{fl} \\ -\sin \gamma_{fr} & \cos \gamma_{fr} & a_f \cos \gamma_{fr} - b \sin \gamma_{fr} \\ 0 & 1 & -a_r \\ 0 & 1 & -a_r \end{bmatrix} \quad (2.9)$$

1. Steering coordination

(2.6) implies that $R_3(\theta)\dot{\psi}$ lies in the null space of $C_1(\gamma_{fl}, \gamma_{fr})$, i.e.,

$$R_3(\theta)\dot{\psi} \in \ker[C_1(\gamma_{fl}, \gamma_{fr})] \quad (2.10)$$

Since C_1 is a 4-by-3 matrix, its rank is less than or equal to 3, i.e. $\text{rank}(C_1) \leq 3$. Moreover, if $\text{rank}(C_1) = 3$, we deduce that $\dot{\psi} = [\dot{x} \ \dot{y} \ \dot{\theta}]^T = 0$, which implies the vehicle is in fact not moving at all, thus we do not consider in this thesis cause we only consider the moving feasible trajectories of the vehicle.

Lemma 2.2.1 *For a car-like vehicle to maneuver with all wheels satisfying the kinematic constraint (2.2), i.e. no skidding at all wheels, its two front steerable wheels must be coordinated in such a way that*

$$\tan \gamma_{fl} - \tan \gamma_{fr} = \frac{b}{a} \tan \gamma_{fl} \tan \gamma_{fr}$$

2.2. Vehicle model

22

Proof: Since row 3 and row 4 of matrix C_1 are identical, $\text{rank}(C_1) < 3$ implies that the determinant of the first three rows of C_1 is zero, i.e.

$$\begin{vmatrix} -\sin \gamma_{fl} & \cos \gamma_{fl} & a_f \cos \gamma_{fl} + b \sin \gamma_{fl} \\ -\sin \gamma_{fr} & \cos \gamma_{fr} & a_f \cos \gamma_{fr} - b \sin \gamma_{fr} \\ 0 & 1 & -a_r \end{vmatrix} = 0$$

$$\Leftrightarrow 2a(\sin \gamma_{fl} \cos \gamma_{fr} - \sin \gamma_{fr} \cos \gamma_{fl}) = 2b \sin \gamma_{fl} \sin \gamma_{fr}$$

$$\Leftrightarrow \tan \gamma_{fl} - \tan \gamma_{fr} = \frac{b}{a} \tan \gamma_{fl} \tan \gamma_{fr}$$

■

Lemma 2.2.1 implies that the two steering angles, γ_{fl} and γ_{fr} , are not independent of each other. When being steered, the two steering angles must be coordinated. The steering maneuver guarantees that there is no skidding. Extensive studies on designs of feedback steering control that prevent car skidding were conducted by Ackermann [95, 96]. The steering angle is compensated with an angle that depends on the velocity and the yaw rate of the vehicle. Thus, the steering maneuver without skidding since then has been referred as *Ackermann steering* maneuver as opposed to skid steering control.

For car-like vehicles with restricted steering angles such that $|\gamma_{fl}, \gamma_{fr}| \ll 1$, the condition of Lemma 2.2.1 can be approximated as follows

$$\gamma_{fl} - \gamma_{fr} \approx \tan \gamma_{fl} - \tan \gamma_{fr} = \frac{b}{a} \tan \gamma_{fl} \tan \gamma_{fr} \approx 0$$

or $\gamma_{fl} \approx \gamma_{fr}$, which implies that the two steering angles can be equally steered and the skidding is so small that it can be negligible. This practice is in fact implemented in most of the vehicles to save one steering actuator. The two wheels are linked by a passive mechanism composed of links and joints and then steered by a sole actuator. In some vehicles like the ones we have, the mechanism could be slightly complex to better approximate the Ackermann steering maneuver, especially for low-speed applications.

With the assumption of Ackermann steering, the modeling development is carried on.

Property 2.2.2 For a car-like vehicle with restricted steering angles, i.e. $|\gamma_{fl}, \gamma_{fr}| \leq \gamma_{\max} \ll \frac{\pi}{2}$, the matrix C_1 in (2.9) is always of rank 2 when Ackermann steering condition is satisfied.

Proof: Since $\text{rank}(C_1) < 3$ when the steering wheels are Ackermann steering wheels, we will prove that $\text{rank}(C_1)$ is also greater than 1.

2.2. Vehicle model

23

By definition, rank of matrix C_1 is the number of independent nonzero rows or columns in the matrix. It is easy to note that there always exists a nonzero independent column in matrix C_1 , i.e. column 2, which implies $\text{rank}(C_1) \geq 1$.

Assuming $\text{rank}(C_1) = 1$, which implies that column 1 and column 3 are linearly dependent on column 2, i.e.

$$c_1 = k_1 c_2 \text{ and } c_3 = k_2 c_2$$

where c_1, c_2 and c_3 are the 1st, 2nd and 3rd columns of matrix C_1 , respectively, and k_1 and k_2 are two scalars. Based on (2.9) we find out that, $k_1 = 0$ and $k_2 = -a_r$.

Then $c_1 = k_1 c_2 = 0$ deduces $\sin \gamma_{fl} = \sin \gamma_{fr} = 0$, thus

$$c_3 = \begin{bmatrix} a_f \cos \gamma_{fl} + b \sin \gamma_{fl} \\ a_f \cos \gamma_{fr} - b \sin \gamma_{fr} \\ -a_r \\ -a_r \end{bmatrix} = \begin{bmatrix} a_f \cos \gamma_{fl} \\ a_f \cos \gamma_{fr} \\ -a_r \\ -a_r \end{bmatrix} = k_2 c_2 = -a_r c_2 = \begin{bmatrix} -a_r \cos \gamma_{fl} \\ -a_r \cos \gamma_{fr} \\ -a_r \\ -a_r \end{bmatrix}$$

or it can be induced that $a_f + a_r = 2a = 0$ which is not true since the physical length of the vehicle is strictly positive, i.e. $2a > 0$.

Consequently, we obtain $1 < \text{rank}(C_1) < 3$ which leads to $\text{rank}(C_1) = 2$. ■

Since the vehicle can be considered as a rigid object moving on a plane, its motion at any instant is a rotation about a point called *Instantaneous Center of Rotation (ICR)*. The velocity at each point on the vehicle is then orthogonal to the line linking it to the *ICR*. With the no skidding assumption (2.2), the velocity at a wheel is actually aligned with the wheel plane. As a result, all wheels are orthogonal to the line linking its center to the *ICR* (Figure 2.3). Obviously, since the two rear wheels are in parallel, the *ICR* must always lie on the line across the centers of these two wheels. Likewise, the velocity at the reference point is also orthogonal to the line linking it to the *ICR*. The angle β formed by the velocity vector and the longitudinal axis of the vehicle is called the *side-slip angle*.

Since the two steering angles are coordinated as shown in Lemma 2.2.1, the two steering wheels can be imaginarily replaced by a virtual steering wheel mount at the mid-point of the link between the two physical steering wheels, resulting in a tricycle configuration. Let assume the steering angle of this virtual steering wheel is γ_f . This additional wheel must also satisfy

Figure 2.3: Instantaneous motion of the vehicle

the pure-rolling and no-skidding assumptions. Thus, matrix C_1 becomes C'_1

$$C'_1 = \begin{bmatrix} -\sin \gamma_f & \cos \gamma_f & a_f \cos \gamma_f \\ -\sin \gamma_{fl} & \cos \gamma_{fl} & a_f \cos \gamma_{fl} + b \sin \gamma_{fl} \\ -\sin \gamma_{fr} & \cos \gamma_{fr} & a_f \cos \gamma_{fr} - b \sin \gamma_{fr} \\ 0 & 1 & -a_r \\ 0 & 1 & -a_r \end{bmatrix} \quad (2.11)$$

with

$$C'_1 R_3(\theta) \dot{\psi} = 0 \quad (2.12)$$

C'_1 still retains the ranking property of C_1 , i.e. $\text{rank}(C'_1) = 2$, with what we can deduce

$$2a(\sin \gamma_f \cos \gamma_{fl} - \cos \gamma_f \sin \gamma_{fl}) + b \sin \gamma_f \sin \gamma_{fl} = 0 \quad (2.13)$$

$$2a(\sin \gamma_f \cos \gamma_{fr} - \cos \gamma_f \sin \gamma_{fr}) - b \sin \gamma_f \sin \gamma_{fr} = 0 \quad (2.14)$$

And they lead to

$$\tan \gamma_f = \frac{\tan \gamma_{fl}}{1 + \frac{b}{2a} \tan \gamma_{fl}} = \frac{\tan \gamma_{fr}}{1 - \frac{b}{2a} \tan \gamma_{fr}} \quad (2.15)$$

γ_f can be calculated from γ_{fl} or γ_{fr} using (2.15).

The reverse computations can be retrieved as well from (2.13) and (2.14)

$$\tan \gamma_{fl} = \frac{\tan \gamma_f}{1 - \frac{b}{2a} \tan \gamma_f} \quad (2.16)$$

$$\tan \gamma_{fr} = \frac{\tan \gamma_f}{1 + \frac{b}{2a} \tan \gamma_f} \quad (2.17)$$

2.2. Vehicle model

25

It is desirable to have all the front steering angles, γ_f, γ_{fl} and γ_{fr} , steered in the same direction, i.e. γ_f, γ_{fl} and γ_{fr} always have the same sign (positive or negative). And (2.16) and (2.17) imply that

$$\begin{aligned} 1 - \frac{b}{2a} \tan \gamma_f &= \frac{\tan \gamma_f}{\tan \gamma_{fl}} > 0 \\ 1 + \frac{b}{2a} \tan \gamma_f &= \frac{\tan \gamma_f}{\tan \gamma_{fr}} > 0 \end{aligned}$$

or

$$|\tan \gamma_f| \leq \tan \gamma_{\max} < \frac{2a}{b} \quad (2.18)$$

Condition (2.18) is satisfied for most of the long vehicles, $2a \gg b$, or those with restricted steering angles, $\gamma_{\max} \ll \frac{\pi}{2}$.

The *ICR* lies at the intersection of the extended rear wheels' axis and the line across the reference point and orthogonal to the velocity v at the reference point. Thus, its global coordinates are

$$ICR = \zeta + R^T(\theta) \begin{bmatrix} -a_r \\ r' \end{bmatrix} \quad (2.19)$$

and the distance from the rear midpoint to the *ICR*

$$r' = r \cos \beta = \frac{2a}{\tan \gamma_f} \quad (2.20)$$

with $r' < 0$ meaning that the *ICR* is on the right side of the rear midpoint, and $r' > 0$ for the cases where *ICR* is on the left side on the rear wheels' axis. Subsequently, condition (2.18) can be rewritten

$$b < |r'|$$

which indicates that the *ICR* lies outside the two back wheels.

2. Posture kinematic model

Since matrix C_1 always has a rank of 2, the null space of C_1 is therefore of one dimension. A nonzero one-dimensional vector can be defined as the basis for the null space. All other vectors in the null space are linearly dependent on the basis with a multiplying scalar. Different selections of the basis will result in different value of the scalar for the same vector in the null space. From (2.10), we note that $R_3(\theta)\dot{\psi}$ belongs to the null space of C_1 . In addition, $R_3(\theta)\dot{\psi}$ consists of three vehicular velocity terms: the longitudinal velocity, the lateral velocity and the yaw rate. Thus, it is reasonable to choose the basis in such a way that the scalar coefficient has a meaning in terms of velocity.

2.2. Vehicle model

26

Since at any instant the vehicle rotate about the *ICR*, the rotation velocity, or the yaw rate $\dot{\theta}$, can be computed

$$\dot{\theta} = \frac{v}{r} = v\kappa \quad (2.21)$$

where v is the velocity at the reference point, r is the distance from the reference point to the *ICR*, and κ is the inverse of r . r and κ are dependent on the steering angle γ_f and can be calculated by geometry

$$\kappa(\gamma_f) = \frac{1}{r} = \frac{\tan \gamma_f}{2a} \cos \beta \quad (2.22)$$

$$\tan \beta = \frac{a_r}{a_f + a_r} \tan \gamma_f = \frac{a_r}{2a} \tan \gamma_f \quad (2.23)$$

In (2.21), the yaw rate relies on a velocity term v , thus, we choose v for the coefficient of the representation of $R_3(\theta)\dot{\psi}$ in the null space

$$R_3(\theta)\dot{\psi} = \begin{bmatrix} R(\theta)\dot{\zeta} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} Q(\gamma_f) \\ \kappa(\gamma_f) \end{bmatrix} v(t) \quad (2.24)$$

By projecting the velocity $v(t)$ on the longitudinal and lateral axes of the vehicle, respectively, the longitudinal and lateral velocities can be obtained

$$R(\theta)\dot{\zeta} = \begin{bmatrix} v \cos \beta \\ v \sin \beta \end{bmatrix} = \begin{bmatrix} \cos \beta \\ \sin \beta \end{bmatrix} v(t) \quad (2.25)$$

Subsequently, $Q(\gamma_f)$ is identified

$$Q(\gamma_f) = \begin{bmatrix} \cos \beta \\ \sin \beta \end{bmatrix}$$

Multiplying both sides of (2.24) with the orthogonal matrix $R_3^T(\theta)$ produces

$$\dot{\psi} = R_3^T(\theta) \begin{bmatrix} Q(\gamma_f) \\ \kappa(\gamma_f) \end{bmatrix} v(t) = \begin{bmatrix} R^T(\theta)Q(\gamma_f) \\ \kappa(\gamma_f) \end{bmatrix} v(t) \quad (2.26)$$

Then, a state space representation of the vehicle can be achieved

$$\dot{q} = \begin{bmatrix} \dot{\psi} \\ \dot{\gamma}_f \end{bmatrix} = G(\theta, \gamma_f)\mu(t) \quad (2.27)$$

$$G(\theta, \gamma_f) = \begin{bmatrix} R^T(\theta)Q(\gamma_f) & 0 \\ \kappa(\gamma_f) & 0 \\ 0 & 1 \end{bmatrix} \quad (2.28)$$

2.2. Vehicle model

27

where

$$q = \begin{bmatrix} \psi \\ \gamma_f \end{bmatrix} = \begin{bmatrix} \zeta \\ \theta \\ \gamma_f \end{bmatrix} \quad ; \quad \mu(t) = \begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix}$$

with ω being the steering rate of the virtual steering wheel.

The vehicle representation (2.27) is called the *posture kinematic model* of the vehicle. [94] revealed the following important properties of this model

Property 2.2.3 (Frobenius) *The posture kinematic model (2.27) is irreducible, i.e. the dimension of its involutive closure is smaller than the dimension of the state vector q .*

$$\dim \bar{\Delta}(q) < \dim q$$

where

$$\begin{aligned} \bar{\Delta}(q) &= \text{span}\{g_1(q), g_2(q), g_3(q), g_4(q)\} \\ g_1(q), g_2(q) &\text{ are two columns of } G(\theta, \gamma_f) \\ g_3(q) &= [g_1(q), g_2(q)] \\ g_4(q) &= [g_1(q), g_3(q)] \end{aligned}$$

Some other properties of this model were investigated in [94] and can be easily verified

Property 2.2.4 *The controllability rank of the linear approximation of the posture kinematic model (2.27) around an equilibrium is 2.*

This implies the linear approximation of the posture kinematic model is not controllable.

Property 2.2.5 *The posture kinematic model (2.27) is controllable.*

This is due to the fact that its involutive distribution has constant maximum rank (i.e. 4).

Property 2.2.6 [97–99] *The posture kinematic model (2.27) is a “differentially flat system”.*

Basically, a system is said to be differentially flat if there exist linearizing outputs $y = (y_1, \dots, y_m)$ differentially independent such that

- the linearizing outputs can be expressed as a function of the system variables and their successive derivatives
- any system variable (state, controls, etc.) can be expressed only from the linearizing outputs and their successive derivatives.

2.2. Vehicle model

28

Property 2.2.7 [2] *The posture kinematic model (2.27) is (a) not stabilizable by a continuous static time-invariant state feedback; but (b) stabilizable by a continuous time-varying static state feedback.*

The Brockett Theorem states that a necessary condition for the system (2.27) to be asymptotically stabilizable by a continuous static time-invariant state feedback is that it satisfies the Brockett condition, which is “the image of every open neighbourhood of the origin contains an open neighbourhood of the origin.”

From the model (2.27), it suffices to show that the mapping

$$\begin{bmatrix} \theta \\ v \\ \gamma_f \end{bmatrix} \mapsto \begin{bmatrix} v \cos(\theta + \beta(\gamma_f)) \\ v \sin(\theta + \beta(\gamma_f)) \end{bmatrix} : \mathbb{R}^3 \longrightarrow \mathbb{R}^2$$

does not satisfy Brockett condition. It is clear that this function maps the open set

$$|\theta + \beta(\gamma_f)| < \frac{\pi}{4}, |v| < 1$$

onto the set as in Figure 2.4 which obviously does not contain an open neighbourhood of the origin. ■

From (2.16) and (2.17), γ_{fl} and γ_{fr} in (2.9) can be replaced with

$$\gamma_{fl} = \arctan \frac{\tan \gamma_f}{1 - \frac{b}{2a} \tan \gamma_f} \quad (2.29)$$

$$\gamma_{fr} = \arctan \frac{\tan \gamma_f}{1 + \frac{b}{2a} \tan \gamma_f} \quad (2.30)$$

Figure 2.4: Mapping set from $|\theta + \beta(\gamma_f)| < \frac{\pi}{4}$ and $|v| < 1$

2.2. Vehicle model

29

Then matrix C_1 becomes matrix $\bar{C}_1(\gamma_f)$ dependent only on the steering angle γ_f . Note that matrix $\bar{C}_1$ also has a rank of 2. And from (2.6), we have

$$\bar{C}_1(\gamma_f)R_3(\theta)\dot{\psi} = 0 \quad (2.31)$$

which leads to

$$\begin{bmatrix} \bar{C}_1(\gamma_f)R_3(\theta) & 0 \end{bmatrix} \dot{q} = \bar{C}_s(\theta, \gamma_f)\dot{q} = 0 \quad (2.32)$$

Replacing $\dot{q}$ by that from (2.27) we have

$$\bar{C}_s(\theta, \gamma_f)\dot{q} = \bar{C}_s(\theta, \gamma_f)G(\theta, \gamma_f)\mu(t) = 0 \quad (2.33)$$

Time-varying control input $\mu(t) = [v(t) \ \omega(t)]^T$ is independent of other state variables and hence we deduce that

$$\bar{C}_s(\theta, \gamma_f)G(\theta, \gamma_f) = 0 \quad (2.34)$$

Equation (2.34) shows that both of the columns of matrix $G(\theta, \gamma_f)$ are in the null space of $\bar{C}_s(\theta, \gamma_f)$. Since $\text{rank}(\bar{C}_s(\theta, \gamma_f)) = \text{rank}(\bar{C}_1(\gamma_f)) = 2$, the null space of $\bar{C}_s(\theta, \gamma_f)$ is actually a two-dimensional subspace. With the fact that two columns of $G(\theta, \gamma_f)$ are linearly independent of each other, they can be chosen as the basis of the null space of $\bar{C}_s(\theta, \gamma_f)$.

The kinematic constraint (2.34) will be used later when deriving the dynamic model.

3. Velocity coordination

Substituting (2.26) into (2.5) yields

$$\begin{aligned} \dot{\varphi} &= J_2^{-1}J_1(\gamma_{fl}, \gamma_{fr})R_3(\theta)\dot{\psi} = J_2^{-1}J_1(\gamma_{fl}, \gamma_{fr}) \begin{bmatrix} Q(\gamma_f) \\ \kappa(\gamma_f) \end{bmatrix} v(t) \\ &= \frac{1}{r_\varphi} \begin{bmatrix} \cos \gamma_{fl} & \sin \gamma_{fl} & a_f \sin \gamma_{fl} - b \cos \gamma_{fl} \\ \cos \gamma_{fr} & \sin \gamma_{fr} & a_f \sin \gamma_{fr} + b \cos \gamma_{fr} \\ 1 & 0 & -b \\ 1 & 0 & b \end{bmatrix} \begin{bmatrix} \cos \beta \\ \sin \beta \\ \frac{1}{2a} \tan \gamma_f \cos \beta \end{bmatrix} v(t) \\ &= \frac{\cos \beta}{r_\varphi} \begin{bmatrix} \sqrt{(1 - \frac{b}{2a} \tan \gamma_f)^2 + \tan^2 \gamma_f} \\ \sqrt{(1 + \frac{b}{2a} \tan \gamma_f)^2 + \tan^2 \gamma_f} \\ 1 - \frac{b}{2a} \tan \gamma_f \\ 1 + \frac{b}{2a} \tan \gamma_f \end{bmatrix} v(t) = \Phi(\gamma_f)v(t) \end{aligned} \quad (2.35)$$

This equation (2.35) shows the relationships of the four wheel velocities to the velocity $v(t)$ at the reference point. This relationship again implies all the velocities must be coordinated to one independent velocity.

2.2. Vehicle model

30

For small steering angle, γ_f , such that $0 \approx \frac{b}{2a} \tan |\gamma_f| \ll 1$, i.e. $b \ll 2a$ and/or $|\gamma_f| \leq \gamma_{\max} \ll 1 < \frac{\pi}{2}$, then the velocity coordination (2.35) can be approximated as follows

$$\dot{\varphi} \approx \frac{\cos \beta}{r_\varphi} \begin{bmatrix} \frac{1}{\cos \gamma_f} \\ \frac{1}{\cos \gamma_f} \\ 1 \\ 1 \end{bmatrix} v(t) \quad (2.36)$$

The two front wheels rotate at the same speed and so do the two rear wheels.

4. Configuration kinematic model

The combination of (2.27) and (2.35) describes all the kinematic constraints of the vehicle and therefore is called the *configuration kinematic model*. The model is subject to a velocity vector $\mu(t)$ consisting of 2 independent velocities: the velocity $v(t)$ and the steering rate $\omega(t)$.

$$\begin{bmatrix} \dot{q} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} G(\theta, \gamma_f) \\ \Phi(\gamma_f) \quad 0 \end{bmatrix} \mu(t) = \begin{bmatrix} R^T(\theta)Q(\gamma_f) & 0 \\ \kappa(\gamma_f) & 0 \\ 0 & 1 \\ \Phi(\gamma_f) & 0 \end{bmatrix} \mu(t) \quad (2.37)$$

Very often, for the car-like vehicle, the reference point is chosen at the midpoint of the rear axle, $a_r = 0$ and $a_f = 2a$. This selection sets the side-slip angle to zero, $\beta = 0$, and, hence, simplifies the model

$$\begin{bmatrix} \dot{q} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ \frac{1}{2a} \tan \gamma_f & 0 \\ 0 & 1 \\ \frac{1}{r_\varphi} \sqrt{(1 - \frac{b}{2a} \tan \gamma_f)^2 + \tan^2 \gamma_f} & 0 \\ \frac{1}{r_\varphi} \sqrt{(1 + \frac{b}{2a} \tan \gamma_f)^2 + \tan^2 \gamma_f} & 0 \\ \frac{1}{r_\varphi} (1 - \frac{b}{2a} \tan \gamma_f) & 0 \\ \frac{1}{r_\varphi} (1 + \frac{b}{2a} \tan \gamma_f) & 0 \end{bmatrix} \mu(t) \quad (2.38)$$

If the reference point is chosen at the midpoint of front axle, i.e. $a_r = 2a$ and $a_f = 0$, the side-slip angle is equal to the steering angle, $\beta = \gamma_f$. And we have the following model

$$\begin{bmatrix} \dot{q} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos(\theta + \gamma_f) & 0 \\ \sin(\theta + \gamma_f) & 0 \\ \frac{1}{2a} \sin \gamma_f & 0 \\ 0 & 1 \\ \frac{1}{r_\varphi} \sqrt{(\cos \gamma_f - \frac{b}{2a} \sin \gamma_f)^2 + \sin^2 \gamma_f} & 0 \\ \frac{1}{r_\varphi} \sqrt{(\cos \gamma_f + \frac{b}{2a} \sin \gamma_f)^2 + \sin^2 \gamma_f} & 0 \\ \frac{1}{r_\varphi} (\cos \gamma_f - \frac{b}{2a} \sin \gamma_f) & 0 \\ \frac{1}{r_\varphi} (\cos \gamma_f + \frac{b}{2a} \sin \gamma_f) & 0 \end{bmatrix} \mu(t) \quad (2.39)$$

2.2.3 Dynamic model of a car-like vehicle

A proper method to derive the dynamic model for the vehicle is to use the Lagrange's method. For a vehicle operating on a horizontal plane, its Lagrangian is dependent solely on its kinetic energy assuming its potential energy is always constant. Moreover, because the two steering angles, γ_{fl} and γ_{fr} , are coordinated subject to γ_f as in (2.16)(2.17), and all the four rotating velocities of the wheels can be calculated from $\dot{\psi}$ as in (2.5), the Lagrangian of the vehicle is in fact in the following form

$$\mathcal{L} = \frac{1}{2} \dot{q}^T M_1(q) \dot{q} \quad (2.40)$$

where $M_1(q)$ is a positive definite symmetric matrix depending on the mass distribution and inertia moments of the rigid body and wheels of the vehicle. Therefore, the dynamics of the vehicle is described as follows

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}(q, \dot{q})}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}(q, \dot{q})}{\partial q} = f_a + B_r(q) \tau_r + B(q) \tau \quad (2.41)$$

where τ is the generalized forces of the control input; τ_r is the resistant forces such as frictions and air resistance; $B(q)$ and $B_r(q)$ are transformation matrices of the forces, respectively; f_a denotes the constraint forces. When the nonholonomic constraints (2.34) are imposed, the constraints forces are given

$$f_a = \bar{C}_s^T(\theta, \gamma_f) \Lambda$$

where $\Lambda \in R^4$ is the associated Lagrangian multipliers.

Equation (2.41) leads to

$$M_1(q) \ddot{q} + W_1(q, \dot{q}) \dot{q} = \bar{C}_s^T(\theta, \gamma_f) \Lambda + B_r(q) \tau_r + B(q) \tau \quad (2.42)$$

2.2. Vehicle model

32

where $W_1(q, \dot{q})$ is the Coriolis matrix and

$$W_1(q, \dot{q}) = \dot{M}_1(q) - \frac{1}{2} \frac{\partial}{\partial q} (\dot{q}^T M_1(q))$$

The Lagrangian multipliers Λ can be eliminated by multiplying both sides of (2.42) with $G^T(\theta, \gamma_f)$ and using equation (2.34)

$$G^T(\theta, \gamma_f) M_1(q) \ddot{q} + G^T(\theta, \gamma_f) W_1(q, \dot{q}) \dot{q} = \bar{B}_r(q) \tau_r + \bar{B}(q) \tau \quad (2.43)$$

where

$$\bar{B}_r(q) = G^T(\theta, \gamma_f) B_r(q) \quad (2.44)$$

$$\bar{B}(q) = G^T(\theta, \gamma_f) B(q) \quad (2.45)$$

Differentiating (2.27) yields

$$\ddot{q} = \dot{G}(\theta, \gamma_f) \mu + G(\theta, \gamma_f) \dot{\mu} \quad (2.46)$$

Substituting (2.27) and (2.46) into (2.43), we obtain a reduced state-space description of the vehicle

$$\begin{cases} \dot{q} = G(\theta, \gamma_f) \mu \\ M(q) \dot{\mu} + W(q, \dot{q}) \mu = \bar{B}_r(q) \tau_r + \bar{B}(q) \tau \end{cases} \quad (2.47)$$

where

$$M(q) = G^T(\theta, \gamma_f) M_1(q) G(\theta, \gamma_f)$$

$$W(q, \dot{q}) = G^T(\theta, \gamma_f) M_1(q) \dot{G}(\theta, \gamma_f) + G^T(\theta, \gamma_f) W_1(q, \dot{q}) G(\theta, \gamma_f)$$

The following properties of matrix $M(q)$ and $W(q, \dot{q})$ were pointed out in [100]

Property 2.2.8 *Matrix $M(q)$ is positive definite and symmetric. Moreover, it satisfies*

$$m_1 I \leq M(q) \leq m_2 I$$

where m_1 and m_2 are known.

Property 2.2.9 *The matrix $\dot{M}(q) - 2W(q, \dot{q})$ is skew-symmetric.*

From (2.45), it is noted that $\bar{B}(q)$ is a 2-row matrix. The number of columns of $\bar{B}(q)$ is equal to the number of generalized forces, or in other words, the number of actuators on the

2.2. Vehicle model

33

vehicle. The only required condition that the vehicle must satisfy is that $\bar{B}(q)$ must have rank of 2 in order to make the vehicle moveable and steerable. In practice, the number of actuators onboard a car-like vehicle may be equal to or greater than 2. The redundancy may be exploited to enhance the performance such as larger physical driving forces or tighter turning angle. A selection of actuators in use must always give the vehicle the abilities to move forward/backward and to steer its front wheels. Let us define the two required forces for driving and steering as τ_m and τ_s , respectively. Mathematically, the actuator torques τ can relate to these two forces as follows

$$\tau = P \begin{bmatrix} \tau_m \\ \tau_s \end{bmatrix} \quad (2.48)$$

where P is a two-column full-rank matrix.

Substituting (2.48) into (2.47) produces

$$\begin{cases} \dot{q} = G(\theta, \gamma_f)\mu \\ M(q)\dot{\mu} + W(q, \dot{q})\mu = \bar{B}_r(q)\tau_r + \bar{B}(q)P \begin{bmatrix} \tau_m \\ \tau_s \end{bmatrix} \end{cases} \quad (2.49)$$

It is obvious that the selection of actuators in use, represented by matrix P , must satisfy the condition that square matrix $\bar{B}(q)P$ is full-rank, i.e. $\text{rank}(\bar{B}(q)P) = 2$.

The description (2.49) can be transformed by a smooth static time-invariant state feedback into a simpler form

$$\begin{cases} \dot{q} = G(\theta, \gamma_f)\mu \\ \dot{\mu} = u = \begin{bmatrix} u_m & u_s \end{bmatrix}^T \end{cases} \quad (2.50)$$

where u represents auxiliary control inputs and consists of the driving acceleration u_m and the steering acceleration u_s . The following smooth time-invariant state feedback is well defined everywhere in the state space

$$\begin{bmatrix} \tau_m \\ \tau_s \end{bmatrix} = [\bar{B}(q)P]^{-1} (M(q)u + W(q, \dot{q})\mu - \bar{B}_r(q)\tau_r) \quad (2.51)$$

And from (2.48) and (2.51) we obtain

$$\tau = P [\bar{B}(q)P]^{-1} (M(q)u + W(q, \dot{q})\mu - \bar{B}_r(q)\tau_r) \quad (2.52)$$

The dynamic model (2.50) describes the dynamics between the posture ψ of the vehicle and the control input u . All the coordinates φ disappear and in fact are hidden in the feedback (2.52). The model (2.50) is actually a dynamic extension of the kinematic model (2.27). But

now, the control input μ is also part of the state vector. Thus, the model (2.50) inherits of the properties of the kinematic model (2.27).

We would like to use this rather simple dynamic model (2.50) in our development of a unified vehicle tracking controller and focus on how to generate the control input u .

2.3 Development of vehicle-following controllers

2.3.1 Vehicle-following situations

Figure 2.5: Look-ahead vehicle-following configuration

Figure 2.6: Look-behind vehicle-following configuration

In the low-speed applications, such as those in warehouses or seaports, the number of vehicles in a platoon is usually limited. The stability of individual vehicle-following controllers

2.3. Development of vehicle-following controllers

35

on each vehicle plays a much more important role than the platoon's string stability where the errors are small. Therefore, we focus on development of a vehicle-following controller for a platoon of two vehicles. In general, the vehicle that leads the platoon, normally manually driven, is called the *leader vehicle*. The other vehicle which is supposed to follow the leader vehicle autonomously is called the *follower vehicle*.

This vehicle-following system can be executed in one of two modes: *look-ahead tracking* and *look-behind tracking*. We define a variable f representing the desired tracking mode

- $f = 1$: *Look-Ahead Tracking*: The leader vehicle moves forward in front of the follower vehicle as in Figure 2.5 and both vehicles move with positive velocities. In this case, the tracked point is the center point P_d at the rear axle of the leader vehicle. The relative distance between two vehicles is measured by the length $d > 0$ of $P_f P_d$ and the relative orientation angle ϕ is formed by the longitudinal axis $P_b P_f$ and $P_f P_d$ ($-\frac{\pi}{2} \leq \phi \leq \frac{\pi}{2}$).
- $f = -1$: *Look-Behind Tracking*: The leader vehicle moves backward behind the follower vehicle as in Figure 2.6 and both vehicles move with negative velocities. In this case, the tracked point is the center point P_d at the front axle of the leader vehicle. The relative distance between two vehicles is measured by the length $d > 0$ of $P_b P_d$ and the relative orientation angle ϕ formed by the longitudinal axis $P_f P_b$ and $P_b P_d$ ($-\frac{\pi}{2} \leq \phi \leq \frac{\pi}{2}$).

The system dynamics is therefore defined as the collective motions of both vehicles as well as the relative distance and orientation angle between two vehicles. A good performance of platoon maneuvers is ensured only if the follower vehicle can follow the leader vehicle with a specified spacing and a tracking error bounded or going to zero.

Many controllers have been developed for vehicle following as discussed in Chapter 1. In this chapter we will develop a unified vehicle-following controller that works for both tracking modes and integrates both longitudinal and lateral controls.

The controller will also utilize a laser scanner system, as discussed in Chapter 4, to measure relative information. When implementing the look-ahead tracking case, the laser scanner is mounted at the midpoint of the front axle to capture the leader vehicle in front, whereas for the look-behind tracking mode, the laser scanner is mounted at the midpoint of the rear axle to capture the back view of the vehicle. Thus, the physical limit of the sensing system is also taken under considerations.

2.3. Development of vehicle-following controllers

36

2.3.2 Controller development

Figure 2.7: Look-ahead focus point

Figure 2.8: Look-behind focus point

The idea of the controller is to imitate human driving practices. Human drivers always focus on a point ahead or behind of the car to keep track of the road. His focus is not fixed but rather moving when he tries to make turns. When following another car, he also keeps his eyes on the car and maneuvers in accordance with the leading car movement.

Therefore, a point P_r , called *the focus point*, is defined as an output function that relies on the steering angle of the vehicle

$$P_r = z = \zeta + R^T(\theta)d_\gamma(\gamma_f) \quad (2.53)$$

$$d_\gamma = \begin{bmatrix} \bar{a} + l \cos p\gamma_f \\ l \sin p\gamma_f \end{bmatrix} \quad (2.54)$$

2.3. Development of vehicle-following controllers

37

$$\bar{a} = \frac{1+f}{2}a_f - \frac{1-f}{2}a_r \quad (2.55)$$

where $\zeta = (x, y)$ is the position of the vehicle in the global coordinate system; and l, p are design parameters. $(l, p\gamma_f)$ defines the focus point P_r in the vehicular polar coordinate system, whose polar axis is aligned with the vehicular center axle and whose pole is at where the laser scanner is mounted, i.e. the front point P_f in the look-ahead tracking mode (Figure 2.7) and at the back point P_b in the look-behind tracking mode (Figure 2.8). $(l, p\gamma_f)$ with $l < 0$ defines a point symmetric to point $(|l|, p\gamma_f)$ about the pole, or equivalent to point $(|l|, p\gamma_f + \pi)$, in the polar coordinate system. As illustrated later, the physical meaning of $|l|$ is the desired spacing between two vehicles. p has no explicit physical meaning but the desired ratio of focus angle to the steering angle.

As discussed in the literature view, the string stability of the platoon can be guaranteed using a time-varying desired spacing which depends on the velocity of the vehicle with a headway time parameter. However, for the low-speed vehicle-following applications with limited number of vehicles, the desired spacing can be simply constant to simplify the controller.

During the look-ahead tracking mode, the leader vehicle moves forward in front of the follower vehicle. The controller will take the midpoint of the rear axle of the leader vehicle, P_d , as the desired point to track. In contrast, when in the look-behind tracking mode, the desired point P_d is the front point of the vehicle. By using the sensing system, the desired point P_d is measured with respect to the follower vehicle

$$P_d = z_d = \zeta + R^T(\theta)d_\phi(d, \phi) \quad (2.56)$$

$$d_\phi = \begin{bmatrix} \bar{a} + fd \cos \phi \\ fd \sin \phi \end{bmatrix} \quad (2.57)$$

(d, ϕ) defines the desired point P_d in the same polar coordinate system as that for the focus point P_r . d is the relative distance between the desired point and the point where the laser scanner is mounted ϕ is the relative angle formed by the longitudinal axis of the follower vehicle and the line linking the polar origin to the desired point. Due to the physical limits of the laser scanner, d and ϕ are bounded: $0 < d \leq d_{\max}$ and $|\phi| \leq 90^\circ$.

The output tracking error becomes

$$\tilde{z} = z - z_d = R^T(\theta)(d_\gamma(\gamma_f) - d_\phi(d, \phi)) = R^T(\theta) \begin{bmatrix} l \cos p\gamma_f - fd \cos \phi \\ l \sin p\gamma_f - fd \sin \phi \end{bmatrix} \quad (2.58)$$

We have the following definition

2.3. Development of vehicle-following controllers

38

Definition 2.3.1 The tracking convergence of a vector $z(t)$ to a given vector $z_d(t)$, means that, for any $\varepsilon > 0$, there exists a $t_0 \geq 0$ such that, for all $t_0 \leq t < \infty$, then

$$\|\tilde{z}(t)\| = \|z(t) - z_d(t)\| < \varepsilon$$

Since ε can be arbitrarily small, the above definition is actually equivalent to

$$\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = \lim_{t \rightarrow \infty} \|z(t) - z_d(t)\| = 0$$

The tracking convergence of $z(t)$ to $z_d(t)$ is equivalent to the zero regulation of the tracking error $\tilde{z}(t)$.

Lemma 2.3.2 Consider a car-like vehicle with restricted steering angle, $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, and a vehicle tracking problem formulated as the forward tracking or the backward tracking. If parameter p is chosen such that $|p| < \frac{\pi}{2\gamma_{\max}}$ and l is chosen as a finite constant, then the following pairs of statements are equivalent:

1. First pair of statements

a. The vehicle tracking error $\tilde{z}(t)$ is convergent to zero, i.e.,

$$\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = 0$$

b. The relative orientation angle ϕ converges to $p\gamma$, i.e.,

$$\lim_{t \rightarrow \infty} (\phi - p\gamma_f) = 0$$

and the relative intervehicular spacing d converges to fl , i.e.,

$$\lim_{t \rightarrow \infty} (d - fl) = 0$$

2. Second pair of statements

a. The vehicle tracking error $\tilde{z}(t)$ and its first derivative $\dot{\tilde{z}}(t)$ are convergent to zero, i.e.,

$$\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = \lim_{t \rightarrow \infty} \|\dot{\tilde{z}}(t)\| = 0$$

b. The convergence as in statement 1b and the relative orientation rate $\dot{\phi}$ converges to $p\omega$, i.e.,

$$\lim_{t \rightarrow \infty} (\dot{\phi} - p\omega) = 0$$

and the relative intervehicular speed $\dot{d}$ converges to 0, i.e.,

$$\lim_{t \rightarrow \infty} \dot{d} = 0$$

2.3. Development of vehicle-following controllers

39

3. Third pair of statements

The vehicle tracking error $\tilde{z}(t)$ and its first derivative $\dot{\tilde{z}}(t)$ and second derivative $\ddot{\tilde{z}}(t)$ are convergent to zero, i.e.,

$$\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = \lim_{t \rightarrow \infty} \|\dot{\tilde{z}}(t)\| = \lim_{t \rightarrow \infty} \|\ddot{\tilde{z}}(t)\| = 0$$

and 3b. The convergence as in statement 2b and the relative orientation acceleration $\ddot{\phi}$ converges to $p\dot{\omega}$, i.e.,

$$\lim_{t \rightarrow \infty} (\ddot{\phi} - p\dot{\omega}) = 0$$

and the relative intervehicular acceleration $\ddot{d}$ converges to 0, i.e.,

$$\lim_{t \rightarrow \infty} \ddot{d} = 0$$

Proof: From the definition of the tracking error (2.58), we have

$$\|\tilde{z}\|^2 = l^2 + d^2 - 2lfd \cos(p\gamma_f - \phi) = [d - fl \cos(p\gamma_f - \phi)]^2 + [fl \sin(p\gamma_f - \phi)]^2 \quad (2.59)$$

Note that l is a finite constant, statement 1a is equivalent to

$$\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = 0 \Leftrightarrow \begin{cases} \lim_{t \rightarrow \infty} \sin(\phi - p\gamma_f) = 0 & \text{(i)} \\ \lim_{t \rightarrow \infty} [d - fl \cos(\phi - p\gamma_f)] = 0 & \text{(ii)} \end{cases} \quad (2.60)$$

1. If statement 1b is true, we will prove that statement 1a is true.

It is easy to check that statement 2b ensures both (2.60-i) and (2.60-ii) satisfied. Hence, $\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = 0$, i.e., statement 1a is true.

2. If statement 1a is true, we now prove that statement 1b is true. Since $|\gamma_f| \leq \gamma_{\max}$ and $|p| < \frac{\pi}{2\gamma_{\max}}$, we have $|p\gamma_f| < \frac{\pi}{2}$. Furthermore, the relative orientation angle ϕ is also bounded, $|\phi| \leq \frac{\pi}{2}$. Thus, we have

$$|p\gamma_f - \phi| \leq |p\gamma_f| + |\phi| < \pi$$

As a result, (2.60-i) leads to

$$\lim_{t \rightarrow \infty} (\phi - p\gamma_f) = 0 \quad (2.61)$$

Combining (2.61) with (2.60-ii) produces

$$\lim_{t \rightarrow \infty} (d - fl) = 0 \quad (2.62)$$

3. If statement 2b is true, we will prove that statement 2a is true.

2.3. Development of vehicle-following controllers

40

We calculate the first derivative of the tracking error $\tilde{z}(t)$ by taking time derivative of (2.58)

$$\dot{\tilde{z}} = R^T(\theta) \begin{bmatrix} -l(\dot{\theta} + p\omega) \sin p\gamma_f - f\dot{d} \cos \phi + fd(\dot{\theta} + \dot{\phi}) \sin \phi \\ l(\dot{\theta} + p\omega) \cos p\gamma_f - f\dot{d} \sin \phi - fd(\dot{\theta} + \dot{\phi}) \cos \phi \end{bmatrix} \quad (2.63)$$

thus,

$$\begin{aligned} \|\dot{\tilde{z}}\|^2 = & l^2(\dot{\theta} + p\omega)^2 + \dot{d}^2 + d^2(\dot{\theta} + \dot{\phi})^2 - 2lf\dot{d}(\dot{\theta} + p\omega) \sin(p\gamma_f - \phi) \\ & - 2lfd(\dot{\theta} + p\omega)(\dot{\theta} + \dot{\phi}) \cos(p\gamma_f - \phi) \end{aligned} \quad (2.64)$$

If statement 2b is true, i.e.,

$$\lim_{t \rightarrow \infty} (\phi - p\gamma_f) = \lim_{t \rightarrow \infty} (d - fl) = \lim_{t \rightarrow \infty} (\dot{\phi} - p\omega) = \lim_{t \rightarrow \infty} \dot{d} = 0$$

then from (2.64), it is deduced that

$$\lim_{t \rightarrow \infty} \|\dot{\tilde{z}}\|^2 = 0 \Rightarrow \lim_{t \rightarrow \infty} \|\dot{\tilde{z}}\| = 0$$

Together with statement 1a deduced from statement 1b, we obtain statement 2a.

4. If statement 2a is true, we will prove that statement 2b is true.

Statement 2a, which contains statement 1a, implies statement 1b, which is

$$\lim_{t \rightarrow \infty} (\phi - p\gamma_f) = \lim_{t \rightarrow \infty} (d - fl) = 0$$

Thus, from (2.64) and $\lim_{t \rightarrow \infty} \|\dot{\tilde{z}}\| = 0$, we have

$$\begin{aligned} 0 &= \lim_{t \rightarrow \infty} \|\dot{\tilde{z}}\|^2 = \lim_{t \rightarrow \infty} \left\{ l^2(\dot{\theta} + p\omega)^2 + \dot{d}^2 + l^2(\dot{\theta} + \dot{\phi})^2 - 2l^2(\dot{\theta} + p\omega)(\dot{\theta} + \dot{\phi}) \right\} \\ &= \lim_{t \rightarrow \infty} \left\{ \dot{d}^2 + l^2(\dot{\phi} - p\omega)^2 \right\} \end{aligned}$$

Subsequently, we retrieve

$$\lim_{t \rightarrow \infty} \dot{d} = \lim_{t \rightarrow \infty} (\dot{\phi} - p\omega) = 0$$

5. If statement 3b is true, we will prove that statement 3a is true.

We calculate the second derivative of the tracking error $\tilde{z}(t)$ by taking time derivative of (2.63)

$$\ddot{\tilde{z}} = R^T(\theta) \begin{bmatrix} -l(\ddot{\theta} + p\dot{\omega}) \sin p\gamma_f - l(\dot{\theta} - p\omega)^2 \cos p\gamma_f - f[\ddot{d} - d(\dot{\theta} + \dot{\phi})^2] \cos \phi \\ \quad + f[2\dot{d}(\dot{\theta} + \dot{\phi}) + d(\ddot{\theta} + \ddot{\phi})] \sin \phi \\ l(\ddot{\theta} + p\dot{\omega}) \cos p\gamma_f - l(\dot{\theta} - p\omega)^2 \sin p\gamma_f - f[\ddot{d} - d(\dot{\theta} + \dot{\phi})^2] \sin \phi \\ \quad - f[2\dot{d}(\dot{\theta} + \dot{\phi}) + d(\ddot{\theta} + \ddot{\phi})] \cos \phi \end{bmatrix} \quad (2.65)$$

2.3. Development of vehicle-following controllers

41

then

$$\begin{aligned} \|\ddot{z}\|^2 = & l^2(\ddot{\theta} + p\dot{\omega})^2 + l^2(\dot{\theta} + p\omega)^4 + [\ddot{d} - d(\dot{\theta} + \dot{\phi})^2]^2 + [2\dot{d}(\dot{\theta} + \dot{\phi}) + d(\ddot{\theta} + \ddot{\phi})]^2 \\ & + 2lf \left\{ (\ddot{\theta} + p\dot{\omega})[\ddot{d} - d(\dot{\theta} + \dot{\phi})^2] + (\dot{\theta} + p\omega)^2[2\dot{d}(\dot{\theta} + \dot{\phi}) + d(\ddot{\theta} + \ddot{\phi})] \right\} \sin(p\gamma_f - \phi) \\ & + 2lf \left\{ (\dot{\theta} + p\omega)^2[\ddot{d} - d(\dot{\theta} + \dot{\phi})^2] - (\ddot{\theta} + p\dot{\omega})[2\dot{d}(\dot{\theta} + \dot{\phi}) + d(\ddot{\theta} + \ddot{\phi})] \right\} \cos(p\gamma_f - \phi) \end{aligned} \quad (2.66)$$

If statement 3b is true, i.e.,

$$\lim_{t \rightarrow \infty} (\phi - p\gamma_f) = \lim_{t \rightarrow \infty} (d - fl) = \lim_{t \rightarrow \infty} (\dot{\phi} - p\omega) = \lim_{t \rightarrow \infty} \dot{d} = \lim_{t \rightarrow \infty} (\ddot{\phi} - p\dot{\omega}) = \lim_{t \rightarrow \infty} \ddot{d} = 0$$

then from (2.66), it is deduced that

$$\lim_{t \rightarrow \infty} \|\ddot{z}\|^2 = 0 \Rightarrow \lim_{t \rightarrow \infty} \|\ddot{z}\| = 0$$

Together with statement 2a deduced from statement 3b, we obtain statement 3a.

6. If statement 3a is true, we will prove that statement 3b is true.

Statement 3a, which contains statement 2a, implies statement 2b, which is

$$\lim_{t \rightarrow \infty} (\phi - p\gamma_f) = \lim_{t \rightarrow \infty} (d - fl) = \lim_{t \rightarrow \infty} (\dot{\phi} - p\omega) = \lim_{t \rightarrow \infty} \dot{d} = 0$$

Thus, from (2.66) and $\lim_{t \rightarrow \infty} \|\ddot{z}\| = 0$, we have

$$0 = \lim_{t \rightarrow \infty} \|\ddot{z}\|^2 = \lim_{t \rightarrow \infty} \left\{ \dot{d}^2 + l^2(\ddot{\phi} - p\dot{\omega})^2 \right\}$$

Subsequently, we retrieve

$$\lim_{t \rightarrow \infty} \dot{d} = \lim_{t \rightarrow \infty} (\ddot{\phi} - p\dot{\omega}) = 0$$

■

Lemma 2.3.2 implies that a control law that ensures the convergence of the tracking error $\tilde{z}(t)$ can guarantee that the intervehicular spacing ultimately converges to the desired distance $|l|$. In practice, sensing range is limited, $0 < d < d_{\max}$ and parameter l must be chosen such that fl lies in the valid range of the sensor

$$0 < fl < d_{\max} \quad (2.67)$$

Condition (2.67) shows that l must be positive in the look-ahead tracking, i.e. $f = 1$, and negative in the look-behind tracking, i.e. $f = -1$. To ensure robust and reliable performance,

2.3. Development of vehicle-following controllers

42

fl should be chosen well away from the boundaries, $0 \ll fl \ll d_{max}$, so that d can be effectively kept within the valid range of the sensor. Equation (2.61) gives an interpretation of the parameter p . At steady state, $\phi = p\gamma_f$, and p is a multiplier relating the steering angle, γ_f , of the follower vehicle and the relative orientation angle, ϕ .

To obtain the dynamic relationship between the output function $z(t)$ and the control input μ , take time derivative of (2.53)

$$\begin{aligned} \dot{z} &= \frac{\partial z}{\partial q} \dot{q} = \frac{\partial z}{\partial q} G(\theta, \gamma_f) \mu \\ &= \begin{bmatrix} I_2 & R^T(\theta) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} d_\gamma & R^T(\theta) \frac{\partial d_\gamma}{\partial \gamma_f} \end{bmatrix} \begin{bmatrix} R^T(\theta) Q(\gamma_f) & 0 \\ \kappa(\gamma_f) & 0 \\ 0 & 1 \end{bmatrix} \mu \\ &= R^T(\theta) \left[Q(\gamma_f) + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} d_\gamma \kappa(\gamma_f) \frac{\partial d_\gamma}{\partial \gamma_f} \right] \mu = E(\theta, \gamma_f) \mu \end{aligned} \quad (2.68)$$

$$E(\theta, \gamma_f) = R^T(\theta) \bar{E}(\gamma_f) \quad (2.69)$$

$$\begin{aligned} \bar{E}(\gamma_f) &= Q(\gamma_f) + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} d_\gamma \kappa(\gamma_f) \frac{\partial d_\gamma}{\partial \gamma_f} \\ &= \begin{bmatrix} \cos \beta \left(1 - \frac{l}{2a} \tan \gamma_f \sin p\gamma_f\right) & -lp \sin p\gamma_f \\ \cos \beta \tan \gamma_f \left(\frac{1+f}{2} + \frac{l}{2a} \cos p\gamma_f\right) & lp \cos p\gamma_f \end{bmatrix} \end{aligned} \quad (2.70)$$

To ensure the existence of a feedback control, the matrix $E(\theta, \gamma_f)$ has to be nonsingular and the following lemma presents such a set of sufficient conditions.

Lemma 2.3.3 Consider a car-like vehicle with restricted steering angle, $|\gamma_f| \leq \gamma_{max} < \frac{\pi}{2}$, and a vehicle tracking problem formulated as the forward tracking or the backward tracking. A control input μ exists for (2.68) if the design parameters l and p are chosen so that the following two conditions are satisfied

1. $lp \neq 0$
2. $\left| p - \frac{1+f}{2} \right| < \frac{\pi}{2\gamma_{max}}$

Proof: The existence of the input μ is guaranteed if and only if matrix $E(\theta, \gamma_f)$ or, equivalently, matrix $\bar{E}(\gamma_f)$ is nonsingular. This is equivalent to the determinant of matrix $\bar{E}(\gamma_f)$ is nonzero

$$\det(\bar{E}) = lp \cos \beta \cos p\gamma_f + \frac{1+f}{2} lp \cos \beta \tan \gamma_f \sin p\gamma_f \neq 0 \quad (2.71)$$

2.3. Development of vehicle-following controllers

43

Since f only takes two values -1 or 1 , we have the following equality

$$\frac{1+f}{2} \tan \gamma_f = \tan \left(\frac{1+f}{2} \gamma_f \right) \quad (2.72)$$

and (2.71) becomes

$$\det(\bar{E}) = lp \cos \beta \cos p \gamma_f + lp \cos \beta \tan \left(\frac{1+f}{2} \gamma_f \right) \sin p \gamma_f = lp \cos \beta \frac{\cos \left[\left(p - \frac{1+f}{2} \right) \gamma_f \right]}{\cos \left(\frac{1+f}{2} \gamma_f \right)} \neq 0 \quad (2.73)$$

Thus, for restricted steering angle, $|\gamma_f| \leq \gamma_{\max}$, the side-slip angle is also constrained

$$|\tan \beta| = \left| \frac{a_r}{2a} \tan \gamma_f \right| \leq |\tan \gamma_f| \Rightarrow |\beta| \leq |\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$$

and condition (2.73) leads to

$$\begin{cases} 1. & lp \neq 0 \\ 2. & \left| p - \frac{1+f}{2} \right| |\gamma_f| \leq \left| p - \frac{1+f}{2} \right| \gamma_{\max} < \frac{\pi}{2} \Rightarrow \left| p - \frac{1+f}{2} \right| < \frac{\pi}{2\gamma_{\max}} \end{cases} \quad (2.74)$$

■

Condition 1 in Lemma 2.3.3 requires (1) ($l \neq 0$), i.e. the focus point P_r cannot be fixed at the front center point P_f of the follower vehicle in forward tracking or at the back point P_b in backward tracking; and (2) ($p \neq 0$), i.e., P_r can not be fixed on the longitudinal center axis. Condition 2 in Lemma 2.3.3 indicates that the selectable range of parameter p is bounded.

Lemmas 2.3.2 and 2.3.3 provide some sufficient conditions in choosing the design parameters l and p . It can be expected that vehicle tracking stability require more conditions on l and p . By examining the basic maneuvers, we can gain some insights and necessary conditions on l and p for tracking stability. Vehicle tracking along a straight or circular path is a basic maneuver and its requirement on stability will offer some insights and a set of necessary conditions.

Without loss of generality, let us assume that both leader and follower vehicles are identical. Then, $\zeta_d = (x_d, y_d)$ represents the reference point of the leader vehicle in the global coordinate, θ_d is the heading angle, γ_d the steering angle, v_d the velocity, and $\omega_d = 0$ the steering rate of the leader vehicle. The reference point ζ_d of the leader vehicle is chosen similarly to that of the follower vehicle, i.e. its distances to front and rear mid-points are a_f and a_r , respectively.

The turning center of the leader vehicle's trajectory, namely ICR_d , is identified

$$ICR_d = \zeta_d + R^T(\theta_d) \begin{bmatrix} -a_r \\ r_d \end{bmatrix} \quad (2.75)$$

2.3. Development of vehicle-following controllers

44

where r_d is the constant distance from ICR_d to the rear midpoint of the leader vehicle, and

$$r_d = \frac{2a}{\tan \gamma_d} \quad (2.76)$$

When $\gamma_d = 0$, the ICR_d is at infinity. When γ_d is a nonzero constant, the ICR_d is a fixed point.

The posture kinematic model of the leader vehicle is

$$\dot{q}_d = G(\theta_d, \gamma_d) \mu_d = \begin{bmatrix} R^T(\theta_d)Q(\gamma_d) & 0 \\ \kappa(\gamma_d) & 0 \\ 0 & 1 \end{bmatrix} \mu_d \quad (2.77)$$

where $q_d = [x_d \ y_d \ \theta_d \ \gamma_d]^T$, $\mu_d = [v_d \ 0]^T$. And the constant side-slip angle of the leader vehicle β_d is computed

$$\tan \beta_d = \frac{a_r}{2a} \tan \gamma_d \quad (2.78)$$

During the look-ahead tracking mode, the desired point P_d is the rear midpoint of the leader vehicle. However, when in the look-behind mode, it is the front midpoint of the leader vehicle that is taken as the desired point. Combining the two situations together in one equation we have

$$P_d = z_d = \zeta_d + R^T(\theta_d) \begin{bmatrix} \underline{a} \\ 0 \end{bmatrix} \quad (2.79)$$

$$\underline{a} = \frac{1-f}{2}a_f - \frac{1+f}{2}a_r \quad (2.80)$$

Taking time derivative of (2.79) produces

$$\dot{z}_d = \dot{\zeta}_d + R^T(\theta_d) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta}_d \begin{bmatrix} \underline{a} \\ 0 \end{bmatrix} = R^T(\theta_d) \bar{z}_d \quad (2.81)$$

$$\bar{z}_d = Q(\gamma_d)v_d + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta}_d \begin{bmatrix} \underline{a} \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ \frac{1-f}{2} \tan \gamma_d \end{bmatrix} v_d \cos \beta_d \neq \mathbf{0} \quad (2.82)$$

It is easy to see that when $\gamma_d = 0$, then $\dot{\theta}_d = 0$ and the leader vehicle moves on a straight path. When γ_d is a nonzero constant, r_d is a finite constant while ICR_d is a fixed point, i.e. its path is a circle.

We also define the following tracking errors

$$\tilde{\eta} = \eta - \eta_d = \begin{bmatrix} \tilde{\theta} \\ \tilde{\gamma} \end{bmatrix} = \begin{bmatrix} \theta - \theta_d \\ \gamma_f - \gamma_d \end{bmatrix} \quad (2.83)$$

2.3. Development of vehicle-following controllers

45

where $\eta = \begin{bmatrix} \theta & \gamma_f \end{bmatrix}^T$ and $\eta_d = \begin{bmatrix} \theta_d & \gamma_d \end{bmatrix}^T$.

We have the following lemma.

Lemma 2.3.4 Consider a basic maneuver of vehicle following a leader vehicle moving on a straight or circular path (γ_d is constant) at a constant speed ($v_d \neq 0$). Suppose there exists a feedback vehicle-following controller that guarantees the convergence of the tracking error $\bar{z}(t)$ and its derivative $\dot{\bar{z}}(t)$ to zero, i.e.

$$\lim_{t \rightarrow \infty} \|\bar{z}(t)\| = \lim_{t \rightarrow \infty} \|\dot{\bar{z}}(t)\| = 0$$

In addition to the conditions in Lemma 2.3.2 and Lemma 2.3.3, parameters l , p and f are necessarily to satisfy $lp > 0$ and $fv_d > 0$ in order to obtain the convergence of $\dot{\eta}$ to zero.

Proof: Taking time derivation of (2.83) yields

$$\dot{\eta} = \begin{bmatrix} \dot{\theta} \\ \omega \end{bmatrix} = \dot{\eta} - \dot{\eta}_d = \Pi(\gamma_f)\mu - \dot{\eta}_d \quad (2.84)$$

where

$$\Pi(\gamma_f) = \begin{bmatrix} \kappa(\gamma_f) & 0 \\ 0 & 1 \end{bmatrix} \quad (2.85)$$

When the convergence of $\dot{\bar{z}}$ to zero is achieved, we have

$$\mathbf{0} \equiv \dot{\bar{z}} = \dot{z} - \dot{z}_d = E(\theta, \gamma_f)\mu - \dot{z}_d$$

Since the matrix E is nonsingular (Lemma 2.3.3), we obtain

$$\mu = E^{-1}(\theta, \gamma_f)\dot{z}_d = \bar{E}^{-1}(\gamma_f)R(\theta)R^T(\theta_d)\bar{z}_d = \bar{E}^{-1}(\gamma_f)R(\tilde{\theta})\bar{z}_d \quad (2.86)$$

Substituting (2.86) into (2.84) produces

$$\dot{\eta} = \Pi(\gamma_f)\bar{E}^{-1}(\gamma_f)R(\tilde{\theta})\bar{z}_d - \dot{\eta}_d \quad (2.87)$$

With the assumption that v_d and γ_d are constants, $\bar{z}_d$ in (2.82) and $\dot{\eta}_d$ are also constants. Then, differentiating (2.87) produces

$$\begin{aligned} \ddot{\eta} &= \dot{\eta} = \left(\Pi\bar{E}^{-1}(\gamma_f) \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \dot{\tilde{\theta}} + \frac{\partial(\Pi\bar{E}^{-1})}{\partial\gamma_f}\omega \right) R(\tilde{\theta})\bar{z}_d \\ &= \left[\Pi\bar{E}^{-1}(\gamma_f) \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} R(\tilde{\theta})\bar{z}_d \quad \frac{\partial(\Pi\bar{E}^{-1})}{\partial\gamma_f}R(\tilde{\theta})\bar{z}_d \right] \dot{\eta} = A(\tilde{\eta}, v_d, \gamma_d)\dot{\eta} \quad (2.88) \end{aligned}$$

2.3. Development of vehicle-following controllers

46

For any frozen $\tilde{\eta} = (\tilde{\theta}, \tilde{\gamma})$, (2.88) is a linear system of $\dot{\tilde{\eta}}$, with the equilibrium $\dot{\tilde{\eta}} = 0$. It is well known that the condition for such a linear system as (2.88) to be exponentially stable is that all of its eigenvalues have negative real part. Thus, we will find the eigenvalues of $A(\tilde{\eta}, v_d, \gamma_d)$ and analyze them to obtain necessary conditions for exponential stability of $\dot{\tilde{\eta}}$.

For simplification, we focus on the case where all the angles γ_f, γ_d and $\tilde{\theta}$ are sufficiently small, i.e. $|\gamma_f| \ll 1$, $|\gamma_d| \ll 1$ and $|\tilde{\theta}| \ll 1$, and find the conditions for this case. These conditions will play the role of necessary conditions for the system (2.88) to be stable.

We have

$$\Pi(\gamma_f) \approx \begin{bmatrix} \frac{1}{2a}\gamma_f & 0 \\ 0 & 1 \end{bmatrix}$$

$$\bar{E}^{-1}(\gamma_f) \approx \frac{1}{lp} \begin{bmatrix} lp & lp^2\gamma_f \\ -(\frac{1+f}{2} + \frac{l}{2a})\gamma_f & 1 \end{bmatrix}$$

thus,

$$\Pi\bar{E}^{-1}(\gamma_f) \approx \frac{1}{lp} \begin{bmatrix} \frac{lp}{2a}\gamma_f & 0 \\ -(\frac{1+f}{2} + \frac{l}{2a})\gamma_f & 1 \end{bmatrix}$$

and

$$\frac{\partial(\Pi\bar{E}^{-1})}{\partial\gamma_f} \approx \frac{1}{lp} \begin{bmatrix} \frac{lp}{2a} & 0 \\ -(\frac{1+f}{2} + \frac{l}{2a}) & 0 \end{bmatrix}$$

Then, from (2.88) we obtain

$$A(\tilde{\eta}, v_d, \gamma_d) = \frac{1}{lp} \left[\begin{bmatrix} 0 & \frac{lp\gamma_f}{2a} \\ -1 & -(\frac{1+f}{2} + \frac{l}{2a})\gamma_f \end{bmatrix} R(\tilde{\theta})\bar{z}_d \begin{bmatrix} \frac{lp}{2a} & 0 \\ -(\frac{1+f}{2} + \frac{l}{2a}) & 0 \end{bmatrix} R(\tilde{\theta})\bar{z}_d \right]$$

Let us set

$$\mathbf{u}_1 = \frac{1}{lp} \begin{bmatrix} 0 & \frac{lp}{2a}\gamma_f \\ -1 & -(\frac{1+f}{2} + \frac{l}{2a})\gamma_f \end{bmatrix} R(\tilde{\theta})\bar{z}_d$$

$$\mathbf{u}_2 = \frac{1}{lp} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} \frac{lp}{2a} & 0 \\ -(\frac{1+f}{2} + \frac{l}{2a}) & 0 \end{bmatrix} R(\tilde{\theta})\bar{z}_d = \frac{1}{lp} \begin{bmatrix} -(\frac{1+f}{2} + \frac{l}{2a}) & 0 \\ -\frac{lp}{2a} & 0 \end{bmatrix} R(\tilde{\theta})\bar{z}_d$$

Two eigenvalues of matrix A are the roots of the following characteristic equation

$$|A(\tilde{\eta}, v_d, \gamma_d) - \lambda I| = 0$$

or

$$\lambda^2 - \lambda \begin{bmatrix} 1 & 0 \end{bmatrix} (\mathbf{u}_1 + \mathbf{u}_2) + \mathbf{u}_1^T \mathbf{u}_2 = 0$$

2.3. Development of vehicle-following controllers

47

The conditions to obtain both negative real eigenvalues are

$$\begin{cases} \Delta = \left\{ \begin{bmatrix} 1 & 0 \end{bmatrix} (\mathbf{u}_1 + \mathbf{u}_2) \right\}^2 - 4\mathbf{u}_1^T \mathbf{u}_2 \geq 0 \\ \mathbf{u}_1^T \mathbf{u}_2 > 0 \\ \begin{bmatrix} 1 & 0 \end{bmatrix} (\mathbf{u}_1 + \mathbf{u}_2) < 0 \end{cases}$$

while the condition to obtain both complex eigenvalues with negative real parts are

$$\begin{cases} \Delta = \left\{ \begin{bmatrix} 1 & 0 \end{bmatrix} (\mathbf{u}_1 + \mathbf{u}_2) \right\}^2 - 4\mathbf{u}_1^T \mathbf{u}_2 < 0 \\ \begin{bmatrix} 1 & 0 \end{bmatrix} (\mathbf{u}_1 + \mathbf{u}_2) < 0 \end{cases}$$

Having combined both cases we obtain the conditions for the two eigenvalues to have negative real part

$$\mathbf{u}_1^T \mathbf{u}_2 > 0 \quad (2.89)$$

$$\begin{bmatrix} 1 & 0 \end{bmatrix} (\mathbf{u}_1 + \mathbf{u}_2) < 0 \quad (2.90)$$

Calculations show that

$$\begin{aligned} \mathbf{u}_1^T \mathbf{u}_2 &= \frac{1}{lp} (R(\tilde{\theta}) \tilde{z}_d)^T \begin{bmatrix} 0 & -1 \\ \frac{lp}{2a} \gamma_f & -\left(\frac{1+f}{2} + \frac{l}{2a}\right) \gamma_f \end{bmatrix} \frac{1}{lp} \begin{bmatrix} -\left(\frac{1+f}{2} + \frac{l}{2a}\right) & 0 \\ -\frac{lp}{2a} & 0 \end{bmatrix} (R(\tilde{\theta}) \tilde{z}_d) \\ &= \frac{1}{l^2 p^2} (R(\tilde{\theta}) \tilde{z}_d)^T \begin{bmatrix} \frac{lp}{2a} & 0 \\ 0 & 0 \end{bmatrix} (R(\tilde{\theta}) \tilde{z}_d) = \frac{1}{2alp} \left((R(\tilde{\theta}) \tilde{z}_d)^T \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} (R(\tilde{\theta}) \tilde{z}_d) \right) \end{aligned}$$

With the term inside the brackets being always positive, (2.89) leads to $lp > 0$.

$$\begin{aligned} \begin{bmatrix} 1 & 0 \end{bmatrix} (\mathbf{u}_1 + \mathbf{u}_2) &= \frac{1}{lp} \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} -\left(\frac{1+f}{2} + \frac{l}{2a}\right) & \frac{lp}{2a} \gamma_f \\ -\frac{lp}{2a} - 1 & -\left(\frac{1+f}{2} + \frac{l}{2a}\right) \gamma_f \end{bmatrix} (R(\tilde{\theta}) \tilde{z}_d) \\ &= \frac{1}{lp} \begin{bmatrix} -\left(\frac{1+f}{2} + \frac{l}{2a}\right) & \frac{lp}{2a} \gamma_f \end{bmatrix} (R(\tilde{\theta}) \tilde{z}_d) \\ &= -\frac{v_d \cos \beta_d}{lp \cos \frac{1-f}{2} \gamma_d} \left\{ \left(\frac{1+f}{2} + \frac{l}{2a} \right) \cos \left(\tilde{\theta} - \frac{1-f}{2} \gamma_d \right) + \frac{lp}{2a} \gamma_f \sin \left(\tilde{\theta} - \frac{1-f}{2} \gamma_d \right) \right\} \end{aligned}$$

With assumption that γ_f, γ_d and $\tilde{\theta}$ are sufficiently small, we then have

$$\begin{bmatrix} 1 & 0 \end{bmatrix} (\mathbf{u}_1 + \mathbf{u}_2) \approx -\frac{v_d}{lp} \left(\frac{1+f}{2} + \frac{l}{2a} \right) = -\frac{fv_d}{lp} \left(\frac{1+f}{2} + \frac{lf}{2a} \right)$$

Lemma 2.3.2 implies that $lf > 0$, thus, $\left(\frac{1+f}{2} + \frac{lf}{2a} \right) > 0$. As a result, conditions (2.89) and (2.90) lead to

$$lp > 0 \text{ and } fv_d > 0$$

2.3. Development of vehicle-following controllers

48

These conditions are the necessary conditions for matrix $A(\tilde{\eta}, v_d, \gamma_d)$ to have all eigenvalues with negative real part. In other words, the system (2.88) is exponentially stable at its equilibrium $\dot{\tilde{\eta}} = 0$. ■

The condition $f v_d > 0$ in Lemma 2.3.4 implies that vehicle-following maneuver is feasible and successful only if the leader vehicle moves forward ($v_d > 0$) in the look-ahead tracking mode ($f = 1$) and moves backward ($v_d < 0$) in the look-behind tracking mode ($f = -1$). This condition is satisfied automatically based on the formulations of the forward tracking and backward tracking defined earlier in Section (2.3.1). Condition $l p > 0$ implies that the vehicle tracking must be in the formations defined in Figure 2.5 for forward tracking and Figure 2.6 for backward tracking. In another word, look-ahead control can only be used for forward tracking formation and look-behind control can only be used for backward tracking control.

Lemma 2.3.5 *Consider a basic maneuver of vehicle following a leader vehicle moving along a straight path ($\gamma_d = 0$) at a constant speed ($v_d \neq 0$). Suppose there exists a feedback vehicle-following controller that guarantees the convergence of the tracking error $\tilde{z}(t)$ and its derivative $\dot{\tilde{z}}(t)$ to zero, i.e.*

$$\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = \lim_{t \rightarrow \infty} \|\dot{\tilde{z}}(t)\| = 0$$

Parameters l and p are chosen to satisfy the conditions in Lemma 2.3.2 and Lemma 2.3.3. Then, the conditions in Lemma 2.3.4 are necessary conditions to achieve the convergence of $\tilde{\eta}$ to zero and v to v_d .

Proof: Lemma 2.3.4 has shown that $\dot{\tilde{\eta}}$ converges to zero. With $\gamma_d = 0$, we have $\dot{\theta}_d = 0$. Thus $\dot{\tilde{\eta}} = \dot{\eta} = \begin{bmatrix} \dot{\theta} & \omega \end{bmatrix}^T \rightarrow 0$.

Since $\dot{\theta} = \frac{v}{2a} \tan \gamma_f \cos \beta$ and $v \cos \beta$ is nonzero, the convergence of $\dot{\theta}$ to zero implies the convergence of γ_f to zero, i.e. $\lim_{t \rightarrow \infty} \gamma_f = 0$.

Consequently, we obtain

$$\dot{z} = R^T(\theta) \bar{E}^{-1}(\gamma_f) \mu = R^T(\theta) \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{l p} \end{bmatrix} \begin{bmatrix} v \\ \omega = 0 \end{bmatrix} = R^T(\theta) \begin{bmatrix} v \\ 0 \end{bmatrix}$$

Additionally, with $\gamma_d = 0$, from (2.81) we have

$$\dot{z}_d = R^T(\theta_d) \begin{bmatrix} v_d \\ 0 \end{bmatrix}$$

2.3. Development of vehicle-following controllers

49

Since $\dot{\tilde{z}} = \dot{z} - \dot{z}_d$ converges to zero, we obtain

$$\dot{\tilde{z}} = R^T(\theta) \begin{bmatrix} v \\ 0 \end{bmatrix} - R^T(\theta_d) \begin{bmatrix} v_d \\ 0 \end{bmatrix} = R^T(\theta) \begin{bmatrix} v - v_d \cos \tilde{\theta} \\ v_d \sin \tilde{\theta} \end{bmatrix} = 0$$

which leads to $\tilde{\theta} = 0$ and $v = v_d$.

In conclusion, $\tilde{\eta} = (\tilde{\theta}, \tilde{\gamma}) = (\tilde{\theta}, \gamma_f)$ converges to zero and v converges to v_d . The follower vehicle eventually follows the leader vehicle on the same straight path and at the same speed.

■

For the circular path situation, we have the lemma.

Lemma 2.3.6 Consider a basic maneuver of vehicle following a leader vehicle moving along a circular path (γ_d is a nonzero constant) at a constant speed $v_d \neq 0$. Suppose there exists a feedback vehicle-following controller that guarantees the convergence of the tracking error $\tilde{z}(t)$ and its derivative $\dot{\tilde{z}}(t)$ to zero, i.e.

$$\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = \lim_{t \rightarrow \infty} \|\dot{\tilde{z}}(t)\| = 0.$$

Parameters l and p are chosen to satisfy the conditions in Lemma 2.3.2 and Lemma 2.3.3. Then, the conditions in Lemma 2.3.4 are necessary conditions to achieve the convergence of the ICR of the follower vehicle to the ICR_d of the leader vehicle, i.e.

$$\lim_{t \rightarrow \infty} \|ICR - ICR_d\| = 0$$

Proof: From (2.19), with ω converging to 0, $r' = r \cos \beta$ and $\kappa = \frac{1}{r}$, then we have

$$\begin{aligned} I\dot{C}R &= R^T(\theta)Q(\gamma_f)v + R^T(\theta) \left(\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta} \begin{bmatrix} -a_r \\ r' \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{\partial r'}{\partial \gamma_f} \omega \end{bmatrix} \right) \\ &= R^T(\theta) \left\{ Q(\gamma_f)v + \begin{bmatrix} -r' \\ -a_r \end{bmatrix} \kappa(\gamma_f)v + \begin{bmatrix} 0 \\ \frac{\partial r'}{\partial \gamma_f} \omega \end{bmatrix} \omega \right\} \\ &= R^T(\theta) \left\{ \left(\begin{bmatrix} 1 \\ \tan \beta \end{bmatrix} + \begin{bmatrix} -r' \frac{\kappa}{\cos \beta} \\ -a_r \frac{\kappa}{\cos \beta} \end{bmatrix} \right) v \cos \beta + \begin{bmatrix} 0 \\ \frac{\partial r'}{\partial \gamma_f} \omega \end{bmatrix} \omega \right\} \rightarrow 0 \end{aligned}$$

The ICR of the follower vehicle converges to a fixed point. From (2.19) and (2.53) we have

$$ICR = z - R^T(\theta) \left(d_\gamma - \begin{bmatrix} -a_r \\ r' \end{bmatrix} \right)$$

2.3. Development of vehicle-following controllers

50

Similarly, from (2.75) and (2.81) we have

$$ICR_d = z_d - R^T(\theta_d) \begin{bmatrix} \underline{a} + a_r \\ -r_d \end{bmatrix} = z_d - R^T(\theta_d) \begin{bmatrix} \frac{1-f}{2} 2a \\ -r_d \end{bmatrix}$$

The tracking error $\widetilde{ICR} = ICR - ICR_d$ is

$$\widetilde{ICR} = \tilde{z} - R^T(\theta) \left(d_\gamma - \begin{bmatrix} -a_r \\ r' \end{bmatrix} \right) + R^T(\theta_d) \begin{bmatrix} \frac{1-f}{2} 2a \\ -r_d \end{bmatrix} \quad (2.91)$$

Since $\dot{ICR}_d = 0$, and $\dot{\tilde{\theta}}$, ω and $\dot{ICR}$ converge to zero, when $\tilde{z}$ and $\dot{\tilde{z}}$ converges to zero, γ_f converges to a constant angle and we obtain

$$\begin{aligned} \dot{\widetilde{ICR}} &= \dot{\tilde{z}} - \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} R^T(\theta) \dot{\theta} \left(d_\gamma - \begin{bmatrix} -a_r \\ r' \end{bmatrix} \right) + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} R^T(\theta_d) \dot{\theta}_d \begin{bmatrix} \frac{1-f}{2} 2a \\ -r_d \end{bmatrix} \\ &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta}_d \left\{ -R^T(\theta) \left(d_\gamma - \begin{bmatrix} -a_r \\ r' \end{bmatrix} \right) + R^T(\theta_d) \begin{bmatrix} \frac{1-f}{2} 2a \\ -r_d \end{bmatrix} \right\} \\ &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta}_d \{ \widetilde{ICR} - \tilde{z} \} \rightarrow 0 \end{aligned}$$

With the convergence of $\tilde{z}$ to zero and with $\dot{\theta}_d$ being nonzero, the above equation leads to

$$\widetilde{ICR} \rightarrow 0 \Leftrightarrow \lim_{t \rightarrow \infty} \|\widetilde{ICR}\| = 0$$

■

2.3.2.1 Kinematic-based tracking controller

The target performance of the vehicle tracking maneuvers can be specified by a first-order system for the closed-loop output tracking error

$$\dot{\tilde{z}} + \lambda \tilde{z} = 0 \quad (2.92)$$

where the convergence rate $\lambda > 0$ can be specified for a desired target performance. By applying (2.92), the tracking error $\tilde{z}$ and its time derivative $\dot{\tilde{z}}$ are exponentially convergent to zero.

Equation (2.92) can be rewritten equivalently as

$$\dot{z} = \dot{z}_d - \lambda \tilde{z} = F_{kin}(q, d, \dot{d}, \phi, \dot{\phi}) \quad (2.93)$$

2.3. Development of vehicle-following controllers

51

By differentiating (2.56) and utilizing the kinematic model (2.27) we obtain

$$\begin{aligned}\dot{z}_d &= \dot{\zeta} + R^T(\theta) \left\{ \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta} d_\phi + \dot{d}_\phi \right\} = R^T(\theta) \left\{ Q(\gamma_f)v + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta} d_\phi + \dot{d}_\phi \right\} \\ &= R^T(\theta) \left\{ \begin{bmatrix} v \cos \beta \\ \frac{1+f}{2} v \cos \beta \tan \gamma_f \end{bmatrix} + f R^T(\phi) \begin{bmatrix} \dot{d} \\ d(\dot{\theta} + \dot{\phi}) \end{bmatrix} \right\}\end{aligned}\quad (2.94)$$

Having substituted (2.68), (2.94) and (2.58) into (2.93), we have

$$R^T(\theta) \bar{E}(\gamma_f) \mu = \dot{z}_d - \lambda \bar{z} = F_{kin}(\theta, v, \gamma_f, d, \dot{d}, \phi, \dot{\phi}) \quad (2.95)$$

$$F_{kin}(\theta, v, \gamma_f, d, \dot{d}, \phi, \dot{\phi}) = R^T(\theta) \bar{F}_{kin}(v, \gamma_f, d, \dot{d}, \phi, \dot{\phi}) \quad (2.96)$$

$$\bar{F}_{kin} = \begin{bmatrix} 1 \\ \frac{1+f}{2} \tan \gamma_f \end{bmatrix} v \cos \beta - \lambda l \begin{bmatrix} \cos p \gamma_f \\ \sin p \gamma_f \end{bmatrix} + f R^T(\phi) \begin{bmatrix} \dot{d} + \lambda d \\ d(\dot{\theta} + \dot{\phi}) \end{bmatrix} \quad (2.97)$$

The orthogonal matrix $R^T(\theta)$ can be eliminated from both sides of (2.95)

$$\bar{E}(\gamma_f) \mu = \bar{F}_{kin}(v, \gamma_f, d, \dot{d}, \phi, \dot{\phi}) \quad (2.98)$$

With the parameters l and p satisfying Lemma 2.3.3 that guarantees the regularity of matrix $\bar{E}$ and hence E , the feedback control law can be achieved from (2.98)

$$\mu_{kin} = \begin{bmatrix} v_{kin} \\ \omega_{kin} \end{bmatrix} = \bar{E}^{-1}(\gamma_f) \bar{F}_{kin}(v, \gamma_f, d, \dot{d}, \phi, \dot{\phi}) \quad (2.99)$$

Property 2.3.7 The control law μ_{kin} in (2.99) is independent of the global posture $\zeta = (x, y)$ and the heading angle θ of the vehicle.

Both of the driving and steering commands are tightly coupled and generated at one stroke. Detailed computations reveal that

$$\begin{aligned}v_{kin} &= v + \frac{\cos \frac{1+f}{2} \gamma_f}{\cos \beta \cos \left(p - \frac{1+f}{2}\right) \gamma_f} \times \\ &\quad \left\{ -\lambda l + f(\dot{d} + \lambda d) \cos(p \gamma_f - \phi) + f d(\dot{\theta} + \dot{\phi}) \sin(p \gamma_f - \phi) \right\}\end{aligned}\quad (2.100)$$

$$\begin{aligned}\omega_{kin} &= -\frac{\dot{\theta}}{p} - \frac{\lambda}{p} \tan \left(p - \frac{1+f}{2}\right) \gamma_f + \frac{f}{lp \cos \left(p - \frac{1+f}{2}\right) \gamma_f} \times \\ &\quad \left\{ (\dot{d} + \lambda d) \sin \left(\phi - \frac{1+f}{2} \gamma_f\right) + d(\dot{\theta} + \dot{\phi}) \cos \left(\phi - \frac{1+f}{2} \gamma_f\right) \right\} \\ &\quad - \frac{\tan \gamma_f \cos \frac{1+f}{2} \gamma_f}{2ap \cos \left(p - \frac{1+f}{2}\right) \gamma_f} \times \\ &\quad \left\{ -\lambda l + f(\dot{d} + \lambda d) \cos(p \gamma_f - \phi) + f d(\dot{\theta} + \dot{\phi}) \sin(p \gamma_f - \phi) \right\}\end{aligned}\quad (2.101)$$

2.3. Development of vehicle-following controllers

52

With $(d, \dot{d}, \phi, \dot{\phi})$ measurable by the laser scanner and (v, γ_f) given by the onboard encoders, (2.100) and (2.101) are the control laws in an explicit form of measurable parameters. This makes the implementation in real-time a lot easier.

The above development for kinematics-based vehicle-following controller can be summarized in the following theorem.

Theorem 2.3.8 *Consider the car-like vehicle tracking maneuvers of forward tracking, shown in Figure 2.5, and backward tracking, shown in Figure 2.6. The kinematic motion of these tracking maneuvers is defined collectively as the kinematics (2.27) of both vehicles and the virtual intervehicular connection (2.56). Define the tracking error $\tilde{z}$ in (2.53) as the difference between the output of the follower vehicle (2.53) and the virtual intervehicular connection (2.56). The tracking target performance in $\tilde{z}$ is defined by the stable first order system (2.92) and can be ensured if the nonlinear control laws (2.100) for driving and (2.101) for steering are applied, and the following conditions are satisfied*

- Forward tracking: $f = 1, \lambda > 0$

$$\begin{cases} v_d > 0 \\ \lambda > 0 \\ 0 \ll l \ll d_{\max} \\ 0 < p < \frac{\pi}{2\gamma_{\max}} \end{cases} \quad (2.102)$$

- Backward tracking: $f = -1, \lambda > 0$

$$\begin{cases} v_d < 0 \\ \lambda > 0 \\ -d_{\max} \ll l \ll 0 \\ -\frac{\pi}{2\gamma_{\max}} < p < 0 \end{cases} \quad (2.103)$$

Proof: Combining the target performance specification, the conditions for tracking convergence equivalence in Lemma 2.3.2, the conditions for the existence of control input in

2.3. Development of vehicle-following controllers

53

Lemma 2.3.3, and the necessary conditions for the tracking stability in Lemma 2.3.4, we obtain

$$\left\{ \begin{array}{l} \text{Target dynamics} \Rightarrow \lambda > 0 \\ \text{Lemma 2.3.2} \Rightarrow \left\{ \begin{array}{l} |p| < \frac{\pi}{2\gamma_{\max}} \\ 0 \ll fl \ll d_{\max} \end{array} \right. \\ \text{Lemma 2.3.3} \Rightarrow \left\{ \begin{array}{l} lp \neq 0 \\ |p + \frac{1+f}{2}| < \frac{\pi}{2\gamma_{\max}} \end{array} \right. \\ \text{Lemma 2.3.4} \Rightarrow \left\{ \begin{array}{l} lp > 0 \\ fv_d > 0 \end{array} \right. \end{array} \right. \Rightarrow \left\{ \begin{array}{l} |p| < \frac{\pi}{2\gamma_{\max}} \\ 0 \ll fl \ll d_{\max} \\ |p + \frac{1+f}{2}| < \frac{\pi}{2\gamma_{\max}} \\ lp > 0 \\ fv_d > 0 \end{array} \right. \quad (2.104)$$

Thus, for $f = 1$, conditions (2.104) will lead to (2.102) and for $f = -1$, conditions (2.104) will lead to (2.103). ■

2.3.2.2 Dynamics-based controller

As in most of the low-speed applications, kinematics-based controllers are proven to be sufficient since the access to the low-level control such as driving torques or steering forces is covered by fast PID controllers. When the torque/force controls, or their corresponding acceleration controls, are accessible, e.g. the current to the motor is directly controllable, a dynamics-based controller is more desirable.

From the analysis of dynamic model presented earlier, with the control inputs are chosen as whenever an acceleration command is available, it can be implemented though a smooth time-invariant feedback control τ as in (2.52). Hence, the problem lies on how to generate the acceleration command. The dynamic model (2.50) is suitable for this purpose while it still describes sufficiently well the dynamics of the vehicle.

The focus point is still defined as in (2.53). The desired point P_d is measured relative to the follower vehicle as in (2.56). And the tracking error is (2.58). Now, in order to make u appear, instead of using the first-order system (2.92) to regulate the error, a second-order closed-loop system is applied on the tracking error

$$\ddot{z} + 2\xi\lambda\dot{z} + \lambda^2 z = 0 \quad (2.105)$$

where $\lambda > 0$ and $\xi > 0$ are the natural frequency and the damping ratio of the second-order system. They can be chosen beforehand for a target performance. The system (2.105) guarantees that the tracking error z and its derivatives $\dot{z}$ and $\ddot{z}$ converge exponentially to zero.

2.3. Development of vehicle-following controllers

54

The closed-loop system (2.105) can be rewritten equivalently as

$$\ddot{z} = \ddot{z}_d - 2\xi\lambda\dot{z} - \lambda^2 z \quad (2.106)$$

$\ddot{z}_d$ is achieved from the time differentiation of (2.94)

$$\begin{aligned} \ddot{z}_d &= R^T(\theta) \left\{ \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta} \left(Q(\gamma_f)v + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta}d_\phi + \dot{d}_\phi \right) \right. \\ &\quad \left. + Q(\gamma_f)\dot{v} + \frac{\partial Q}{\partial \gamma_f}v\omega + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} (\ddot{\theta}d_\phi + \dot{\theta}\dot{d}_\phi) + \ddot{d}_\phi \right\} \\ &= R^T(\theta) \left\{ \begin{bmatrix} \dot{v} \cos \beta - v\dot{\beta} \sin \beta - \frac{1+f}{2}2a\dot{\theta}^2 \\ v\dot{\theta} \cos \beta + \frac{1+f}{2}2a\ddot{\theta} \end{bmatrix} + fR^T(\phi) \begin{bmatrix} \ddot{d} - d(\dot{\theta} + \dot{\phi})^2 \\ d(\ddot{\theta} + \ddot{\phi}) + 2\dot{d}(\dot{\theta} + \dot{\phi}) \end{bmatrix} \right\} \end{aligned} \quad (2.107)$$

Likewise, from (2.68), we take the derivative with respect to q and time t

$$\dot{z} = \frac{d}{dt}\dot{z} = \frac{\partial(E(\theta, \gamma_f)\mu)}{\partial q}\dot{q} + E(\theta, \gamma_f)\dot{\mu} = E(\theta, \gamma_f)u + H(\theta, \gamma_f)\mu \quad (2.108)$$

where

$$\begin{aligned} H(\theta, \gamma_f) &= \frac{\partial(E(\theta, \gamma_f)\mu)}{\partial q}G(\theta, \gamma_f) = \frac{\partial(R^T(\theta)\bar{E}(\gamma_f)\mu)}{\partial q}G(\theta, \gamma_f) \\ &= R^T(\theta) \left[\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \bar{E}(\gamma_f)\mu\kappa(\gamma_f) \quad \frac{\partial\bar{E}(\gamma_f)}{\partial \gamma_f}\mu \right] = R^T(\theta)\bar{H}(\gamma_f) \end{aligned} \quad (2.109)$$

Then, we have

$$\begin{aligned} H(\theta, \gamma_f)\mu &= R^T(\theta)\bar{H}(\gamma_f)\mu = R^T(\theta) \left(\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \bar{E}(\gamma_f)\dot{\theta} + \frac{\partial\bar{E}(\gamma_f)}{\partial \gamma_f}\omega \right) \mu \\ &= R^T(\theta) \left(\begin{bmatrix} -v\dot{\beta} \sin \beta - \frac{1+f}{2}2a\dot{\theta}^2 \\ v\dot{\theta} \cos \beta + \frac{1+f}{2}2a\left(\ddot{\theta} - \frac{v\dot{\theta}}{v}\right) \end{bmatrix} - lR^T(p\gamma_f) \begin{bmatrix} (\dot{\theta} + p\omega)^2 \\ -\ddot{\theta} + \frac{v\dot{\theta}}{v} \end{bmatrix} \right) \end{aligned} \quad (2.110)$$

where

$$\dot{\beta} = \frac{a_r \cos^2 \beta}{2a \cos^2 \gamma_f} \omega \quad (2.111)$$

$$\dot{\theta} = \frac{1}{2a} v \tan \gamma_f \cos \beta \quad (2.112)$$

$$\ddot{\theta} = \frac{1}{2a} \left(\dot{v} \tan \gamma_f \cos \beta + \frac{v\omega \cos \beta}{\cos^2 \gamma_f} - v\dot{\beta} \tan \gamma_f \sin \beta \right) \quad (2.113)$$

2.3. Development of vehicle-following controllers

55

Having substituted (2.58), (2.63), (2.107) and (2.110) into (2.106), we obtain

$$E(\theta, \gamma_f)u = \ddot{z}_d - H(\theta, \gamma)\mu - 2\xi\lambda\dot{\ddot{z}} - \lambda^2\ddot{z} = F_{dyn}(\theta, v, \dot{v}, \gamma_f, \omega, d, \dot{d}, \ddot{d}, \phi, \dot{\phi}, \ddot{\phi}) \quad (2.114)$$

$$F_{dyn}(\theta, v, \dot{v}, \gamma_f, \omega, d, \dot{d}, \ddot{d}, \phi, \dot{\phi}, \ddot{\phi}) = R^T(\theta)\bar{F}_{dyn}(v, \dot{v}, \gamma_f, \omega, d, \dot{d}, \ddot{d}, \phi, \dot{\phi}, \ddot{\phi}) \quad (2.115)$$

$$\begin{aligned} \bar{F}_{dyn} = & \begin{bmatrix} \dot{v} \cos \beta \\ \frac{1+f}{2} \dot{v} \cos \beta \tan \gamma_f \end{bmatrix} + lR^T(p\gamma_f) \begin{bmatrix} (\dot{\theta} + p\omega)^2 - \lambda^2 \\ -\ddot{\theta} + \frac{1}{2a} \dot{v} \tan \gamma_f \cos \beta - 2\xi\lambda(\dot{\theta} + p\omega) \end{bmatrix} \\ & + fR^T(\phi) \begin{bmatrix} \ddot{d} + 2\xi\lambda\dot{d} + \lambda^2 d - d(\dot{\theta} + \dot{\phi})^2 \\ d(\ddot{\theta} + \ddot{\phi}) + 2(\dot{d} + \xi\lambda d)(\dot{\theta} + \dot{\phi}) \end{bmatrix} \end{aligned} \quad (2.116)$$

Multiplying the orthogonal matrix $R(\theta)$ to both sides of (2.114) produces

$$\bar{E}(\gamma)u = \bar{F}_{dyn}(v, \dot{v}, \gamma_f, \omega, d, \dot{d}, \ddot{d}, \phi, \dot{\phi}, \ddot{\phi}) \quad (2.117)$$

Again, with l and p chosen to satisfy Lemma 2.3.3 that guarantee the regularity of matrix $\bar{E}$, the dynamics-based controller is obtained

$$u_{dyn} = \bar{E}^{-1}(\gamma_f)\bar{F}_{dyn}(v, \dot{v}, \gamma_f, \omega, d, \dot{d}, \ddot{d}, \phi, \dot{\phi}, \ddot{\phi}) \quad (2.118)$$

This control input u_{dyn} is nonlinear and dependent solely on relative parameters such as $(v, \dot{v}, \gamma_f, \omega)$ and $(d, \dot{d}, \ddot{d}, \phi, \dot{\phi}, \ddot{\phi})$ measurable by the onboard inertial navigation system and a ranging sensor. The commanded control input $u_{dyn} = [u_{m_{dyn}} \quad u_{s_{dyn}}]^T$ is then computed

$$\begin{aligned} u_{m_{dyn}} = & \dot{v} + \frac{\cos \frac{1+f}{2} \gamma_f}{\cos \beta \cos (p - \frac{1+f}{2}) \gamma_f} \left\{ l[(\dot{\theta} + p\omega)^2 - \lambda^2] \right. \\ & + f[\ddot{d} + 2\xi\lambda\dot{d} + \lambda^2 d - d(\dot{\theta} + \dot{\phi})^2] \cos(p\gamma - \phi) \\ & \left. + f[d(\ddot{\theta} + \ddot{\phi}) + 2(\dot{d} + \xi\lambda d)(\dot{\theta} + \dot{\phi})] \sin(p\gamma - \phi) \right\} \end{aligned} \quad (2.119)$$

$$\begin{aligned} u_{s_{dyn}} = & -\frac{1}{p} \left(\ddot{\theta} + 2\xi\lambda(\dot{\theta} + p\omega) \right) + \frac{1}{p} \left((\dot{\theta} + p\omega)^2 - \lambda^2 \right) \tan \left(p - \frac{1+f}{2} \right) \gamma_f \\ & + \frac{\cos \frac{1+f}{2} \gamma_f}{p \cos (p - \frac{1+f}{2}) \gamma} \left\{ -\frac{l}{2a} \left((\dot{\theta} + p\omega)^2 - \lambda^2 \right) \tan \gamma_f \right. \\ & + f \left[\frac{\sin (\phi - \frac{1+f}{2} \gamma_f)}{l \cos \frac{1+f}{2} \gamma_f} - \frac{\tan \gamma_f}{2a} \cos(p\gamma_f - \phi) \right] [\ddot{d} + 2\xi\lambda\dot{d} + \lambda^2 d - d(\dot{\theta} + \dot{\phi})^2] \\ & \left. + f \left[\frac{\cos (\phi - \frac{1+f}{2} \gamma_f)}{l \cos \frac{1+f}{2} \gamma_f} - \frac{\tan \gamma_f}{2a} \sin(p\gamma_f - \phi) \right] [d(\ddot{\theta} + \ddot{\phi}) + 2(\dot{d} + \xi\lambda d)(\dot{\theta} + \dot{\phi})] \right\} \end{aligned} \quad (2.120)$$

2.3. Development of vehicle-following controllers

56

Theorem 2.3.9 Consider the car-like mobile robot performing forward tracking, shown in Figure 2.5, and backward tracking, shown in Figure 2.6. The dynamic motion of these tracking maneuvers is defined collectively as the dynamics (2.50) of both vehicles and the virtual intervehicular connection (2.56).

Define the tracking error $\tilde{z}$ in (2.58) as the difference between the output of the follower vehicle (2.53) and the virtual intervehicular connection (2.56). The tracking target performance in $\tilde{z}$ is defined by the stable second order system (2.105) and can be ensured if the nonlinear controls (2.119) for driving and (2.120) for steering are applied and the following necessary conditions are satisfied

- Forward tracking: $f = 1$, $\lambda > 0$, $\xi > 0$, l and p satisfying (2.102);
- Backward tracking: $f = -1$, $\lambda > 0$, $\xi > 0$, l and p satisfying (2.103).

Proof: By combining conditions from Lemmas 2.3.2, 2.3.3 and 2.3.4, we retrieve the same conditions as those for kinematics-based controller, i.e. (2.102) for the look-ahead tracking mode and (2.103) for the look-behind mode. In addition, the conditions for the target system (2.105) to be stable are $\lambda > 0$ and $\xi > 0$. Consequently, the theorem's claim holds. ■

2.3.3 Following a circular trajectory

It is privileged for the follower vehicle to follow as much closely as possible the leader vehicle's trajectory especially when turning to avoid potential obstacles near the trajectory. Moreover, since the trajectory traced by the leader vehicle is a really feasible one, successful tracking this trajectory can virtually ensure the control inputs lie in its physical limits.

Lemma 2.3.5 has shown that when the leader vehicle moves on a straight path, i.e. $\gamma_d = 0$, $lp > 0$ is the necessary condition to achieve the convergence of the follower vehicle's steering angle, heading angle and velocity to the corresponding ones of the leader vehicle.

For the circular-path tracking situation, Lemma 2.3.4 and Lemma 2.3.6 show that if the tracking convergence of $\tilde{z}$ and its derivatives $\dot{\tilde{z}}$ is guaranteed, both vehicles will eventually move on circular paths with the same turning center, i.e. $ICR = ICR_d$, and at the same heading rate, i.e.

$$\dot{\theta} = v\kappa(\gamma_f) = v \frac{\tan \gamma_f \cos \beta}{2a} = \dot{\theta}_d = v_d \frac{\tan \gamma_d \cos \beta_d}{2a}$$

Since the two velocities have the same sign, i.e. $vv_d > 0$, it is deduced that γ_f and γ_d have the same sign, i.e. $\gamma_f \gamma_d > 0$.

2.3. Development of vehicle-following controllers

57

It is also easy to see that if $\gamma_f = \gamma_d$, or $r' = r_d$, then we also achieve $v = v_d$ and vice versa, which is very much desirable. It implies that the convergence of the steering angle to the leader vehicle's steering angle, which also means the two vehicles moves on the same circular path, is equivalent to the convergence of the velocity to that of leader vehicle. When $\gamma_f < \gamma_d$, we have $v > v_d$ that may lead to the saturation of the velocity due to its physical limit. If $v < v_d$, it implies $\gamma_f > \gamma_d$. The saturation of the steering angle is very likely, especially with the restricted steering angle, $|\gamma_f| \leq \gamma_{\max} \ll \frac{\pi}{2}$. That is why having $v = v_d$ and $\gamma_f = \gamma_d$ is very important. As shown later, the values of v and γ_f , if existent, for a given set of (γ_d, v_d) rely strongly on the selection of parameters l and p . The following study will focus on this issue.

With z converging to z_d and ICR converging to ICR_d , (2.91) leads to

$$\begin{aligned} R^T(\theta) \left\{ d_\gamma - \begin{bmatrix} a_r \\ -r' \end{bmatrix} \right\} &= R^T(\theta_d) \begin{bmatrix} \frac{1-f}{2} 2a \\ -r_d \end{bmatrix} \\ \Leftrightarrow \begin{bmatrix} \frac{1+f}{2} 2a + l \cos p\gamma_f \\ -r' + l \sin p\gamma_f \end{bmatrix} &= R(\tilde{\theta}) \begin{bmatrix} \frac{1-f}{2} 2a \\ -r_d \end{bmatrix} \end{aligned}$$

The norm of both side should be equal as well

$$(r' - l \sin p\gamma_f)^2 + \left(\frac{1+f}{2} 2a + l \cos p\gamma_f \right)^2 = r_d^2 + \frac{1-f}{2} (2a)^2 \quad (2.121)$$

Unfortunately, it is not easy to retrieve γ_f from equation (2.121). Nonetheless, it is clear that the convergent steering angle γ_f is a function of r_d , l and p . Some necessary conditions on l and p for a given r_d may be obtained from (2.121) to ensure the validity of the steering angle γ_f .

Note that if γ_f satisfies (2.121) then so does $(-\gamma_f)$. And because γ_f and γ_d must have the same sign. Then, (2.121) is equivalent to

$$(|r'| - l \sin p|\gamma_f|)^2 + \left(\frac{1+f}{2} 2a + l \cos p\gamma_f \right)^2 = r_d^2 + \frac{1-f}{2} (2a)^2 \quad (2.122)$$

and $|\gamma_f| = |\gamma_d|$ also means $\gamma_f = \gamma_d$.

To facilitate the analysis, we will consider two tracking modes separately.

a. Look-behind tracking

Parameters l and p are then chosen satisfying condition (2.103), i.e.

$$\begin{cases} -d_{\max} \ll l \ll 0 \\ -\frac{\pi}{2\gamma_{\max}} < p < 0 \end{cases}$$

2.3. Development of vehicle-following controllers

58

Equation (2.122) becomes

$$(|r'| - l \sin p |\gamma_f|)^2 + (l \cos p \gamma_f)^2 = r_d^2 + (2a)^2 \quad (2.123)$$

$$\Leftrightarrow r'^2 + l^2 - 4al \frac{\sin p |\gamma_f|}{\tan |\gamma_f|} = \frac{(2a)^2}{\sin^2 \gamma_d} \quad (2.124)$$

Due to the physical limits of the steering angle, $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, we have the following lemma.

Lemma 2.3.10 *Consider the look-behind tracking situation where a car-like vehicle, with restricted steering angle $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, follows another vehicle that moves on a circular path at a constant speed $v_d < 0$ and with a constant steering angle $\gamma_d \neq 0$. The kinematics-based controller (2.99) or the dynamics-based controller (2.118) is applied with parameters l and p chosen satisfying condition (2.103). Then the necessary condition to achieve the convergence of the tracking error $\tilde{z}(t)$ to zero is*

$$l^2 - 4alp < (2a)^2 \left(\frac{1}{\sin^2 \gamma_d} - \frac{1}{\tan^2 \gamma_{\max}} \right) \quad (2.125)$$

Proof: Since the steering angle is restricted, $|\gamma_f| \leq \gamma_{\max} \ll \frac{\pi}{2}$, the radius $|r'|$ has a lower bound

$$|r'| \geq r_{\min} = \frac{2a}{\tan \gamma_{\max}} \quad (2.126)$$

Also, calculations show that

$$p < \frac{\sin p |\gamma_f|}{\tan |\gamma_f|} \leq \frac{\sin p \gamma_{\max}}{\tan \gamma_{\max}}$$

As a result the left side of (2.124) has a lower limit

$$\begin{aligned} \min_{|\gamma_f| \leq \gamma_{\max}} \left(r'^2 + l^2 - 4al \frac{\sin p |\gamma_f|}{\tan |\gamma_f|} \right) &> \min_{|\gamma_f| \leq \gamma_{\max}} (r'^2) + l^2 + 4a(-l) \min_{|\gamma_f| \leq \gamma_{\max}} \left(\frac{\sin p |\gamma_f|}{\tan |\gamma_f|} \right) \\ &> \frac{(2a)^2}{\tan^2 \gamma_{\max}} + l^2 - 4alp \end{aligned}$$

Therefore, the necessary condition to achieve tracking convergence of $\tilde{z}(t)$ is that the right side of equation (2.124) must be greater than this lower limit of the left side, i.e.

$$\frac{(2a)^2}{\sin^2 \gamma_d} = r'^2 + l^2 - 4al \frac{\sin p |\gamma_f|}{\tan |\gamma_f|} > \frac{(2a)^2}{\tan^2 \gamma_{\max}} + l^2 - 4alp$$

or it is equivalent to (2.125). ■

2.3. Development of vehicle-following controllers

59

Lemma 2.3.10 reveals a bound on parameters l and p for a particular steering angle γ_d of the leader vehicle.

Regarding the desire of having $\gamma_f = \gamma_r$, we have the following necessary condition

Theorem 2.3.11 *Consider the look-behind tracking situation where a car-like vehicle, with restricted steering angle $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, follows another vehicle that moves on a circular path at a constant speed $v_d < 0$ and with a constant steering angle $\gamma_d \neq 0$. The kinematics-based controller (2.99) or the dynamics-based controller (2.118) is applied with parameters l and p chosen satisfying condition (2.103) and Lemma 2.3.10. Then the necessary condition to achieve the convergence of the tracking error $\tilde{z}(t)$ and $\tilde{\gamma} = \gamma_f - \gamma_d$ to zero is*

$$l^2 - 4al \frac{\sin p|\gamma_d|}{\tan |\gamma_d|} - (2a)^2 = 0 \quad (2.127)$$

Proof: When the tracking convergence of $\tilde{z}(t)$ and $\tilde{\gamma}$ to zero is secured, $\gamma_f = \gamma_d$ and $r' = r_d$, and (2.124) becomes (2.127). ■

Condition (2.127) shows that the successful tracking of γ_f to γ_d , and hence v to v_d , is possible if l and p are chosen to satisfy not only (2.103) and (2.125) but also (2.127).

It is clear that for different values of γ_d , there are different sets of l and p satisfying (2.127). It implies that there does not exist a fixed set of (l, p) that satisfies (2.127) for any γ_d . In other words, if the leader vehicle changes its steering angle, with the same set of (l, p) , the follower cannot repeat the new circular path of the leader vehicle's.

Theorem 2.3.11 also reveals that, if l and p are not chosen such that satisfy (2.127), the convergent steering angle γ_f , if existent, may be greater or smaller than γ_d .

One immediate result we can obtain from equation (2.124) is the following lemma.

Lemma 2.3.12 *Consider the look-behind tracking situation where a car-like vehicle, with restricted steering angle $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, follows another vehicle that moves on a circular path at a constant speed $v_d \neq 0$ and with a constant steering angle $\gamma_d \neq 0$. The kinematics-based controller (2.99) or the dynamics-based controller (2.118) is applied with parameters l and p chosen satisfying condition (2.103) and Lemma 2.3.10. If l is chosen such that $-2a \leq l < 0$, then the convergence of the tracking error $\tilde{z}(t)$ to zero implies that the follower vehicle eventually moves on a circular path whose radius is greater than that of the leader vehicle's trajectory,*

2.3. Development of vehicle-following controllers

60

i.e.

$$|r'| > |r_d| \text{ or } |\gamma_f| < |\gamma_d|$$

Proof: From (2.124) we have

$$r'^2 - r_d^2 = (2a)^2 - l^2 + 4al \frac{\sin p |\gamma_f|}{\tan |\gamma_f|}$$

Since $\frac{\sin p |\gamma_f|}{\tan |\gamma_f|} < 0$ for all $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, with condition $-2a \leq l < 0$, we have

$$r'^2 - r_d^2 = [(2a)^2 - l^2] + 4al \frac{\sin p |\gamma_f|}{\tan |\gamma_f|} > 0$$

■

No matter what value is chosen for p , if $-2a \leq l < 0$, we always achieve $|r'| > |r_d|$ if the tracking error $\tilde{z}$ converges to zero. For other values of l and p that do not satisfy (2.127), it is more difficult to retrieve the convergent result of γ_f . But with the assumption that γ_f and γ_d are relatively small angles, i.e. $|\gamma_f| \leq \gamma_{\max} \ll \frac{\pi}{2}$, and γ_f and γ_d have the same sign, i.e. $\gamma_f \gamma_d > 0$, that the following approximation holds

$$\frac{\sin p |\gamma_f|}{\tan |\gamma_f|} \approx \frac{\sin p |\gamma_d|}{\tan |\gamma_d|} \approx p$$

then (2.124) becomes

$$r'^2 - r_d^2 \approx - \left[l^2 - 4al \frac{\sin p |\gamma_d|}{\tan |\gamma_d|} - (2a)^2 \right]$$

Then from (2.127), we state that with the sets of l and p that make the left side of (2.127) positive, the convergent radius $|r'|$, if existent, will tend to be smaller than $|r_d|$. And with the sets of l and p that make that left side (2.127), then the radius $|r'|$ tends to be greater than $|r_d|$, i.e.

$$l^2 - 4al \frac{\sin p |\gamma_d|}{\tan |\gamma_d|} - (2a)^2 > 0 \Rightarrow |r'| < |r_d|$$

$$l^2 - 4al \frac{\sin p |\gamma_d|}{\tan |\gamma_d|} - (2a)^2 < 0 \Rightarrow |r'| > |r_d|$$

All the above-developed necessary conditions for the look-behind tracking of a vehicle running on a circular path are shown in Figure 2.9 for $2a = 1.2m$, $\gamma_{\max} = \frac{\pi}{8}$ and $d_{\max} = 20m$. There are two surfaces in this figure: the lower is the bound (2.125) and the upper is the condition (2.127). Parameters l and p must be chosen such that the point (p, γ_d, l) is above the lower surface and below the top horizontal plane, i.e. $l = 0$. The points on the upper surface are those that can guarantee the tracking convergence of the steering angle and velocity of the follower

Figure 2.9: Necessary conditions for circular trajectory look-behind tracking

vehicle to those of the leader vehicle. The points above the upper surface may lead to $|r'| > |r_d|$ whereas those in between the two surfaces may lead to $|r'| < |r_d|$.

Figure 2.9 shows that $|l|$ can not be too large, especially when the vehicles are tightly turning. When $|p|$ decreases to zero, the bound of $|l|$ also decreases to $2a$.

b. Look-ahead tracking

In this case, l and p are chosen satisfying condition (2.102), i.e.

$$\begin{cases} 0 \ll l \ll d_{\max} \\ 0 < p < \frac{\pi}{2\gamma_{\max}} \end{cases}$$

With $f = 1$, equation (2.122) becomes

$$(|r'| - l \sin p |\gamma_f|)^2 + (2a + l \cos p \gamma_f)^2 = r_d^2 \quad (2.128)$$

$$\Leftrightarrow r'^2 + l^2 - 4al \frac{\sin(p-1)|\gamma_f|}{\sin |\gamma_f|} = \frac{(2a)^2}{\tan^2 \gamma_d} - (2a)^2 \quad (2.129)$$

Lemma 2.3.13 Consider the look-ahead tracking situation where a car-like vehicle, with restricted steering angle $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, follows another vehicle that moves on a circular path at a constant speed $v_d > 0$ and with a constant steering angle $\gamma_d \neq 0$. The kinematics-based controller (2.99) or the dynamics-based controller (2.118) is applied with parameters l and p chosen satisfying condition (2.102). Then the necessary condition to achieve the convergence of the tracking error $\tilde{z}(t)$ to zero is

2.3. Development of vehicle-following controllers

62

(a) for $1 < p < 2$

$$l^2 - 4al \frac{\sin(p-1)\gamma_{\max}}{\sin \gamma_{\max}} \leq (2a)^2 \left(\frac{1}{\tan^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}} \right) \quad (2.130)$$

(b) for $0 < p \leq 1$ or $2 \leq p < \frac{\pi}{2\gamma_{\max}}$

$$l^2 - 4al(p-1) < (2a)^2 \left(\frac{1}{\tan^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}} \right) \quad (2.131)$$

Proof: Since the steering angle is restricted, $|\gamma_f| \leq \gamma_{\max} \ll \frac{\pi}{2}$, from (2.126) we have

$$|r'| \geq r_{\min} = \frac{2a}{\tan \gamma_{\max}}$$

Examining on $\frac{\sin(p-1)|\gamma_f|}{\sin |\gamma_f|}$ reveals that in the range $|\gamma_f| \in (0, \gamma_{\max}]$ the function is monotonous. And

◇ For $0 < p \leq 1$

$$0 \geq p-1 > \frac{\sin(p-1)|\gamma_f|}{\sin |\gamma_f|} \geq \frac{\sin(p-1)\gamma_{\max}}{\sin \gamma_{\max}}$$

◇ For $1 < p < 2$

$$0 < p-1 < \frac{\sin(p-1)|\gamma_f|}{\sin |\gamma_f|} \leq \frac{\sin(p-1)\gamma_{\max}}{\sin \gamma_{\max}}$$

◇ For $2 \leq p < \frac{\pi}{2\gamma_{\max}}$

$$p-1 > \frac{\sin(p-1)|\gamma_f|}{\sin |\gamma_f|} \geq \frac{\sin(p-1)\gamma_{\max}}{\sin \gamma_{\max}} \geq 1$$

(a) With $1 < p < 2$, we have

$$\min_{|\gamma_f| \leq \gamma_{\max}} \left(r'^2 + l^2 - 4al \frac{\sin(p-1)|\gamma_f|}{\sin |\gamma_f|} \right) = \frac{(2a)^2}{\tan^2 \gamma_{\max}} + l^2 - 4al \frac{\sin(p-1)\gamma_{\max}}{\sin \gamma_{\max}}$$

Thus, the right side of equation (2.129) must not be smaller than the lower bound of the left side

$$\frac{(2a)^2}{\tan^2 \gamma_d} - (2a)^2 \geq \frac{(2a)^2}{\tan^2 \gamma_{\max}} + l^2 - 4al \frac{\sin(p-1)\gamma_{\max}}{\sin \gamma_{\max}}$$

or equivalently we have (2.130)

(b) Similarly, with $0 < p \leq 1$ or $2 \leq p < \frac{\pi}{2\gamma_{\max}}$, we have

$$\min_{|\gamma_f| \leq \gamma_{\max}} \left(r'^2 + l^2 - 4al \frac{\sin(p-1)|\gamma_f|}{\sin |\gamma_f|} \right) > r_{\min}^2 + l^2 - 4al(p-1)$$

With the same discussion, the necessary condition to obtain the tracking convergence is

$$\frac{(2a)^2}{\tan^2 \gamma_d} - (2a)^2 > r_{\min}^2 + l^2 - 4al(p-1) = \frac{(2a)^2}{\tan^2 \gamma_{\max}} + l^2 - 4al(p-1)$$

which lead to (2.131). ■

2.3. Development of vehicle-following controllers

63

Theorem 2.3.14 Consider the look-ahead tracking situation where a car-like vehicle, with restricted steering angle $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, follows another vehicle that moves on a circular path at a constant speed $v_d > 0$ and with a constant steering angle $\gamma_d \neq 0$. The kinematics-based controller (2.99) or the dynamics-based controller (2.118) is applied with parameters l and p chosen satisfying condition (2.103) and Lemma 2.3.13. Then the necessary condition to achieve the convergence of the tracking error $\tilde{z}(t)$ and $\tilde{\gamma} = \gamma_f - \gamma_d$ to zero is

$$l^2 - 4al \frac{\sin(p-1)|\gamma_d|}{\sin|\gamma_d|} + (2a)^2 = 0 \quad (2.132)$$

Proof: When the tracking convergence of $\tilde{z}(t)$ and $\tilde{\gamma}$ to zero is secured, $\gamma_f = \gamma_d$ and $r' = r_d$, and (2.129) becomes (2.132). ■

Corollary 2.3.15 The convergence of the tracking error $\tilde{\gamma} = \gamma_f - \gamma_d$ to zero is achieved with $l = 2a$ and $p = 2$ for any γ_d .

Proof: Conditions $l = 2a$ and $p = 2$ satisfy the condition (2.132). Furthermore, with $l = 2a$ and $p = 2$, (2.129) becomes $r'^2 = r_d^2$, or $\gamma_f = \gamma_d$. ■

Corollary 2.3.15 points out a set of parameters (l, p) independent of γ_d that can guarantee tracking convergence to zero of not only $\tilde{z}$ but $\tilde{\gamma}$ as well.

Lemma 2.3.16 Consider the look-ahead tracking situation where a car-like vehicle, with restricted steering angle $|\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$, follows another vehicle that moves on a circular path at a constant speed $v_d > 0$ and with a constant steering angle $\gamma_d \neq 0$. The kinematics-based controller (2.99) or the dynamics-based controller (2.118) is applied with parameters l and p chosen satisfying condition (2.102) and Lemma 2.3.13. If p is chosen such that $0 < p < 2$, then the convergence of the tracking error $\tilde{z}(t)$ to zero implies that the follower vehicle moves on a circular path whose radius is smaller than that of the leader vehicle's trajectory, i.e.

$$|r'| < |r_d| \text{ or } |\gamma_f| > |\gamma_d|$$

Proof: From (2.129) we have

$$\begin{aligned} r'^2 - r_d^2 &= -l^2 + 4al \frac{\sin(p-1)|\gamma_f|}{\sin|\gamma_f|} - (2a)^2 \\ &= -\left[l - 2a \frac{\sin(p-1)|\gamma_f|}{\sin|\gamma_f|}\right]^2 + (2a)^2 \left[\frac{\sin^2(p-1)\gamma_f}{\sin^2\gamma_f} - 1\right] \\ &= -\left[l - 2a \frac{\sin(p-1)|\gamma_f|}{\sin|\gamma_f|}\right]^2 + (2a)^2 \frac{\sin p|\gamma_f| \sin(p-2)|\gamma_f|}{\sin^2\gamma_f} \end{aligned}$$

2.3. Development of vehicle-following controllers

64

Therefore, condition $0 < p < 2$ leads to $\sin p |\gamma_f| \sin(p-2) |\gamma_f| < 0$ for any $0 < |\gamma_f| \leq \gamma_{\max} < \frac{\pi}{2}$. Hence, $r'^2 - r_d^2 < 0$. ■

No matter what value is chosen for l , if $0 < p < 2$, we always achieve $|r'| < |r_d|$ if the tracking error $\tilde{z}$ converges to zero. For other values of l and p that do not satisfy (2.132), we also try to estimate the convergent result. With the assumption that γ_f and γ_d are relatively small angles, i.e. $|\gamma_f| \leq \gamma_{\max} \ll \frac{\pi}{2}$, and γ_f and γ_d have the same sign, i.e. $\gamma_f \gamma_d > 0$, that the following approximation holds

$$\frac{\sin(p-1)|\gamma_f|}{\sin |\gamma_f|} \approx \frac{\sin(p-1)|\gamma_d|}{\sin |\gamma_d|} \approx p-1$$

then (2.129) becomes

$$r'^2 - r_d^2 \approx - \left[l^2 - 4al \frac{\sin(p-1)|\gamma_d|}{\sin |\gamma_d|} + (2a)^2 \right]$$

Then from (2.132), we state that with the sets of l and p that make the left side of (2.132) positive, the convergent radius $|r'|$, if existent, will tend to be smaller than $|r_d|$. And with the sets of l and p that make that left side (2.132), then the radius $|r'|$ tends to be greater than $|r_d|$, i.e.

$$\begin{aligned} l^2 - 4al \frac{\sin(p-1)|\gamma_d|}{\sin |\gamma_d|} + (2a)^2 > 0 &\Rightarrow |r'| < |r_d| \\ l^2 - 4al \frac{\sin(p-1)|\gamma_d|}{\sin |\gamma_d|} + (2a)^2 < 0 &\Rightarrow |r'| > |r_d| \end{aligned}$$

All the above-developed necessary conditions for the tracking of a vehicle running on a circular path are shown in Figure 2.10 for $2a = 1.2m$, $\gamma_{\max} = \frac{\pi}{8}$ and $d_{\max} = 20m$. There are two surfaces in this figure: the upper is the bound (2.130) and (2.131) and the lower, which is in a parabolic cylinder-like shape, is the condition (2.132). Parameters l and p must be chosen such that the point (p, γ_d, l) is below the upper surface and above the bottom horizontal surface, i.e. $l = 0$. The points on the lower surface are those that can guarantee the tracking convergence of the steering angle and velocity of the follower vehicle to those of the leader vehicle. The points inside the lower surface may lead to $|r'| > |r_d|$ whereas those in between the two surfaces may lead to $|r'| < |r_d|$.

Figure 2.10 also shows that when $|\gamma_d|$ is very big, $|\gamma_d| > \arctan(\sin \gamma_{\max})$ to be precise, there are some places where the upper surface touches the bottom horizontal plane, which means the selectable range of (l, p) is null and the tracking is impossible. This is not the case for look-behind tracking where there is always a valid range of (l, p) to choose from for any γ_d .

Figure 2.10: Necessary conditions for circular trajectory look-ahead tracking

For any γ_d , if $p \geq 2$ is chosen first, then there may be one or two values of l such that the point(s) (p, γ_d, l) belong to the lower surface. It means we can choose between two values of l and still be able to guarantee the two circular paths are identical, i.e. $\gamma_f = \gamma_d$. For look-behind tracking, for every pair (p, γ_d) , we have at most only one value of l that makes (p, γ_d, l) lie on the surface indicating $\gamma_f = \gamma_d$. However, the additional value of l that we obtain for look-ahead tracking is normally very small that make it impractical to choose with the assumption $l \gg 0$.

Leader vehicle moving on a complex trajectory

It is well known that a complex trajectory including curved paths can be approximated in terms of a series of instantaneous circles, with the instantaneous radius being the inverse of the curve's instantaneous curvature. However, as discussed in the previous section, for different turning radius r_d of the leader vehicle, different values of l and p may need to be chosen to guarantee the tracking desire of $r' = r_d$. There is only one set of (l, p) that can guarantee $r' = r_d$ for a varying r_d . It is $l = 2a$ and $p = 2$ for the look-ahead tracking mode. But sometimes the desired spacing is not $2a$, especially for the look-behind tracking mode where l is always negative. Subsequently, we will not be able to find another set of (l, p) that is independent of γ_d . That is the drawback of using constant desired spacing.

Nevertheless, by choosing parameters l and p to satisfy the conditions for the case where the leading radius reach to its minimum, i.e. when the leader vehicle turns the tightest angle,

with the preference of $|\gamma_f| \leq |\gamma_d|$, the steering angle can be guaranteed that it lies in its feasible range. As a result, $|v| \geq |v_d|$ but hopefully v still stays in the physical limit for low-speed applications and with powerful motors.

As we know, during the transient period before the two trajectories converge to each other, the initial condition and parameter λ will determine the magnitude of the peaks of the response. If the initial error and/or λ is too big, it will result in large overshoots in control command μ . Due to the physical limits of the vehicle, the saturation of velocity or steering angle may occur which leads to the tracking instability. This discussion will be illustrated in the simulation study.

2.3.4 Requirement of practical measurements

As stated in the development of the kinematics- and dynamics-based controllers, besides the vehicular state feedbacks such as velocity/acceleration and steering angle, few measurements are required including relative distance between two vehicles d , velocity $\dot{d}$, acceleration $\ddot{d}$ and relative angle ϕ as well as its derivatives $\dot{\phi}$ and $\ddot{\phi}$. These requirements are vital and also implemented in other vehicle-following systems. For example, the inclusion of relative distance, velocity and even acceleration in the controller have been well known and implemented in longitudinal controls in order to improve the stability of the tracking system [35, 42, 53]. Likewise, for steering control, the controllers developed based on kinematic model generally need the relative angle and/or its first derivative [41, 42] whereas those based on dynamics-model [62, 63, 101] may or may not require the second derivative.

In practice, the relative distance and angle can be measured by a ranging sensor. For the relative velocities and particularly relative accelerations, it is more difficult to obtain. In general, there are two ways of getting those measurements. The first one is to utilize a wireless communication channel to transit the vehicular measurements such as velocity, acceleration and yaw rate of the leader vehicle to the follower vehicle [42, 53]. The relative velocities and/or accelerations are computed based on the geometric and dynamic relationships of the two vehicles. The second way relies on the high accuracy of the ranging sensor to estimate the derivatives using numerical calculations or derivative filtering [41, 62]. This method is less accurate than the first one but more suitable for low-speed applications and it does not require a communication channel.

2.4 Conclusion

A unified controller for vehicle-following has been derived for car-like vehicles. The controller is an integration of two basic controllers, longitudinal control and lateral control, and based on human driving practice. A focus point is defined at a predetermined distance in front of or behind the vehicle to track the leader vehicle. If the controller manages to maneuver the focus point so that it tracks the leader vehicle, the performance target, which is that a vehicle follows another, can be achieved. The conditions of parameters l and p that guarantee the tracking stability are examined. It again shows that the driving practices that human drivers have been doing everyday such as focusing in front of the vehicle while driving forward or focusing on the back of the vehicle while driving in reserve. The two fundamental situations, driving on a straight line and driving on a circular path, that can make up any complex trajectory, are investigated. The conditions, with which not only the tracking stability but also the platooning stability (the string stability) are reinforced, have been identified. The analysis shows that with some selections of l and p , the velocity of the follower vehicle is smaller or equal to that of the leader vehicle, which may eliminate the tracking error amplified through the platoon. The derived conditions also point out that the design parameters l and p must be chosen properly to avoid the steering angle's saturation which leads to tracking instability. Especially while turning, due to the physical limits of the vehicle, parameter l can not be too big in magnitude and parameter p is bounded.

Chapter 3

A High-Fidelity Simulation Platform for Vehicle Control Algorithms

3.1 Introduction

A well-developed theory needs to be tested to prove its effectiveness and to ensure it works properly. Nevertheless, experiments normally require expensive equipments and lengthy process of part design and system integration. Extensive, comprehensive and near-reality simulations are helpful and often necessary. For a structural mechanical system like a vehicle, the simulations have been long done mathematically using many different simulation softwares, for example Mathematica [102], MathCAD [103] and especially Matlab [104], which now becomes the most widely used and powerful simulation software offering a general-purpose environment with quite a number of functions provided to study different types of systems. Furthermore, SIMULINK, a block diagram-based programming part of Matlab, simplifies much more the programming procedure with function blocks, making the developing control algorithm easy to read, to verify and to modify. The design of the controller can follow closely to its logics set by the developer.

In Matlab, vehicle dynamics are usually represented in differential or discrete equations, which have limitations in modeling a complex mechanical system. For examples, a car is usually modeled by a weighted point or rectangle. This simplified model is sufficient for controller design which concerns of generating control inputs to the actuators. However, to get more insights on what the behaviours of the system could be when the control inputs are applied, more detailed model of the system must be used. This is a really difficult task for a single devel-

oper using Matlab because of the complexity of the model derived from the physical system. For example, a car nowadays consists of hundreds of components and the dynamic model is so nonlinearly complex and multiple-parametric that it is almost impossible to identify precisely.

Fortunately, the fast increase of computers in terms of processing speed, memory size and graphic resolution and together with the application of an analysis method namely *Finite Elements* has brought up a series of simulation and analysis software for structural systems. Fundamentally, the Finite Elements methods break a structural system into a plenty but finite number of points, called *nodes*, which make up a grid, called *mesh*. This mesh is programmed to contain the material and structural properties which define how the structure will react to certain loading conditions. The mesh acts like a spider web where from each node, there extends a mesh element to each of the adjacent nodes. Many softwares for structural system analysis based on Finite Element methods have been developed including Abaqus [105], Ansys [106], Marc [107], and MSC/Nastran [108] to name a few. Moreover, the simulation platforms that even integrate kinematic/dynamic analysis include DADS from LMS International [109], SIMPACK from Intec [110] and, in particular, MSC/Adams from MSC.Software [111]. After having bought Marc [107], one of the best nonlinear finite element analysis software, and integrated with its current products, MSC Software becomes the leading company in the area of structural system analysis. Its product ADAMS (*Automatic Dynamic Analysis of Mechanical Systems*), a package of many simulation and analysis modules, has been extensively used especially in robotic and automotive industries [112–114]. One of its advantages is its ability to co-simulate ADAMS models with other simulation softwares like MATRIX, EASY and MATLAB.

In this chapter, a cooperative simulation platform is presented combining the advantages of two simulation softwares: ADAMS, version 12.0, for platoon modeling and Matlab/SIMULINK, version 6.5, for controller design, followed by an extensive study of simulation for the controllers developed in Chapter 2.

3.2 ADAMS®: vehicle modeling and environment design

3.2.1 Vehicle modeling

ADAMS is a software platform consisting of many software modules. *ADAMS/View* is a prototyping module that allows users to build a mechanical system using some provided common basic parts. A GUI (Graphic User Interface) exhibits all these basic parts and a user can *drag*

Figure 3.1: Body of a vehicle

and drop any required parts to build a complex mechanical system. The parts can be made of different materials and the user can choose various properties and parameter values, such as density, elasticity, stiffness and friction at contacts.

We consider a vehicle platoon system consisting of two vehicles moving on a ground plane. The ground is created using a 2D plane and placed at height level 0. The gravity is chosen with the normal value of $9.8m/s^2$. Both vehicles are car-like with four-wheel-driving and front-wheel-steering. The major mechanisms of a vehicle include the vehicle body, the tyres, 4 driving systems, and one steering systems. Their models in ADAMS as well as the integration are described as follows with most of the parameters such as sizes, weights, etc. being the same as those from the real vehicles that we have.

- **Vehicle body:** The vehicle body is modeled using an aluminium plate to support and hold all other parts. It weights about $250kg$ and has a shape as shown in Figure 3.1 with dimensions of $1.85m \times 1.21m \times 0.06m$.
- **Tyres:** Each tyre is modeled with a torus as shown in Figure 3.2. Its major and minor radii are $14cm$ and $6cm$, respectively, and its weight is $15kg$. The mass of a tyre is defined using a mass inertia tensor diagonal matrix. Table 3.1 shows a set of parameters for a tyre. The interaction with ground is defined as a contact between a circle and a plane. Contacting forces between the wheels and the ground are modeled and composed of two parts: normal force and friction force.

1. The normal force at a contact point is computed based on the IMPACT function from the library as a nonlinear spring-damper and is specified with parameters: stiffness, damping and penetration depth as shown in Table 3.1. When the vehicle motions reach steady status, all normal forces settle as constants which can be recorded for calculations of driving torques later.

Table 3.1: Torus’s parameter settings to model a tyre

Parameter	Description	Value
Tyre radius	The radius of the tyre	20cm
Weight	The weight of the tyre	11.82kg
Stiffness	The stiffness of the tyre	500kN/m
Damping	The damping coefficient when penetration occurs	1000Ns/m
Penetration Depth	The depth of penetration of the tyre	2cm
μ_s	Static friction coefficient	0.75
μ_d	Dynamic friction coefficient	0.6
v_s	Stiction-Transition velocity	3cm/s
v_d	Friction-Transition velocity	5cm/s

2. The friction force is defined using the Coulomb friction model. The friction coefficients vary depending on the slip velocity of the tyre as shown in Figure 3.3 with parameter values of μ_d , μ_s , v_d and v_s given in Table 3.1. All these parameters were chosen based on the general values for the type of the tyres (the actual values are not available at the moment).

- **Drive systems:** Each wheel is equipped with a driving system as shown in Figure 3.2. A revolute joint connects each of the four wheels to a cylindrical axis which in turn links to the vehicular body by either another revolute joint (for front wheels) or a fixed joint (for rear wheels). A torque is applied at each wheel to model driving motors. Since the vehicle-following controller can be either kinematics-based, which controls the velocity, or dynamics-based, which controls the acceleration, the driving torques can be generated as follows

Figure 3.2: Driving wheel model

Figure 3.3: Coulomb model of friction coefficient

- *Kinematics-based*: The input is the desired velocity v . Together with the steering information, the desired angular velocity at each wheel is computed using (2.35). Afterwards, the torque is generated utilizing a closed-loop proportional controller somewhat similar to the real low-level PID controller that is use on the real vehicle.

$$\tau_i = K_p * I_z * \left(\frac{v_{id}}{R} - \omega_i \right) \quad (3.1)$$

where τ_i is the driving torque at wheel i , I_z is the mass moment of inertia of a wheel (0.227 kgm^2), R is the radius of the wheel (0.2 m), ω_i is the current velocity of the wheel, v_{di} is the desired velocity for the that wheel, and K_p is a gain. The moment of inertia is computed by ADAMS immediately after all the parameters of the wheel such as the shape, the size and the mass density are provided. User can easily read that information. The current angular velocity of the wheel is obtained using function WZ of ADAMS/View that compute it during the simulation. The gain K_p must be chosen such that the desired velocity for each wheel is reached before the next numerical calculation step. In our case, tests show that $K_p = 5$ is sufficient. When the desired velocity is obtained, the total torque acting on the wheel will become zero. In other words, the driving torque will be equal to the total friction in terms of moments acting on the wheel.

- *Dynamics-based*: The control input to each driving system is the desired acceleration of the vehicle at the reference point. Therefore, the proportional-type torque generator can not be applied. For the sake of simplification, the reference point is generally chosen at the midpoint of the rear axle so that side-slip angle β is elimi-

nated. The external forces acting on the vehicle include the driving torque τ_m defined in (2.48), and four friction forces at the contact point between the wheels and the ground. By using the Newton's second law, the torque τ_m can be computed for a desired acceleration u_m as follows

$$\frac{1}{R}\tau_m - (fric_{rl} + fric_{rr}) - (fric_{fl} + fric_{fr}) \cos \gamma_f = M * u_m \quad (3.2)$$

where u_m is the desired acceleration, M is the total weight of the vehicle ($\sim 357.17kg$), γ_f is the front steering angle, $fric_{fl}$, $fric_{fr}$, $fric_{rl}$ and $fric_{rr}$ are the friction forces at the four wheels: front left, front right, rear left, and rear right, respectively, and $\frac{1}{R}\tau_m$ is the total driving force acting along the longitudinal axis of the vehicle that is required to achieve the desired acceleration. In general, this driving force is generated by applying a total torque, τ_m , at the two rear wheels. The driving total torque could be generated by an engine and applied on both rear wheels. Or two separate motors are used to generate equal torques at each wheel. The sum of the two torques are equal to that of the one-engine situation. The friction forces can be obtained by using the function *CONTACT* of ADAMS, that returns the contact force acting on an object at the contact point along a designated direction.

In fact, there are other resistant forces acting on the vehicle such as the air resistant force and internal friction of the joints. The air resistant force is normally proportional to the square of the speed of the vehicle and can be modeled as an external force. However, in practice, the vehicles do not run at high speeds, thus, this resistant force is normally sufficiently small and negligible. The internal friction of a joint can be defined when constructing the joint. Again, comparing to the friction between the ground and the tyres, these internal frictions are quite small and can be eliminated by the proportional controller in kinematics-based situation, or can be topped up to the total driving torque using function *FRICTION* of ADAMS.

- **Steering system:** The steering system is installed at the front axle of the vehicle. Its mechanism is shown in Figure 3.6. A main steering joint in the middle links with a steel jack as shown in Figure 3.4, which in turn joins to 2 other links. Each of these two links is connected to the wheel holder (i.e. the cylindrical axis mentioned above) of a wheel using a cylindrical joint as shown in Figure 3.5. When the main steering joint rotates, the two wheels linked to the main steering joint are accordingly steered.

Similar to the driving system, to model the steering actuator (for our vehicles, they are hydraulic actuators), the steering torque, τ_s is created that acting on the main steering joint to steer the entire steering system. The torque generator can be a proportional controller with steering rate as input for kinematics-based situations or a direct controller from the desired steering acceleration for dynamics-based situations. With the actual hydraulic actuator being very powerful compared with the steering friction, the desired steering rate or steering angle can be quickly obtained. Tests show that the actual maximum steering rate the hydraulic system can provide is about $27^\circ/s$, while the physical steering range is $\pm 22.5^\circ$. The steering friction is mainly constituted by the friction between the two front tyres and the ground, which are modeled as an external torque acting on the main steering joint. The internal friction of the steering joints and actuator is very small and normally negligible. Then the steering torque τ_s can be approximately proportional to the desired steering acceleration

$$\tau_s = I_z * u_s \tag{3.3}$$

where I_z is the inertia moment of the steering system and $u_s = \ddot{\gamma}_f$ is the desired steering acceleration.

Equations (3.2) and (3.3) are equivalent to equation (2.51) with the special selection of the torques in use.

With all the mechanisms built, a vehicle can be assembled. And a platoon system with two vehicles is shown in Figure 3.7. These vehicles, even though simple, are able to model the dynamic behavior of a car-like robot for the simulation study of dynamics and control design.

Figure 3.4: Main steering joint

Figure 3.5: Links to a wheel

Figure 3.6: Steering system model

Figure 3.7: ADAMS model of vehicle platooning

3.3 SIMULINK®: high-level controller design

Figure 3.8: Controller in SIMULINK

ADAMS can only model linearized controllers and thus is not a suitable platform for non-linear controllers such as (2.118). Thus, MATLAB is used to model this full-state tracking controller. MATLAB’s library of functions and toolboxes covers almost all the needs for developing and simulating sophisticated control algorithms. SIMULINK further simplifies the programming process of the controller and organizes it in terms of dataflow block diagrams.

Figure 3.8 shows an example of a controller, e.g. (2.118), being coded using several SIMULINK mathematical blocks. The left part is to compute the inverse of the matrix E and the right part to compute the matrix F based on the feedback from the vehicle model in ADAMS. The results of two matrices are then multiplied to generate the control input commands.

3.4 Closed-loop system modeling and co-simulation

3.4.1 Closed-loop system modeling

Figure 3.9: Closed-loop system in SIMULINK

The closed-loop system consists of the vehicle platoon system in ADAMS, the trajectory generator for the leader vehicle and the vehicle-following controller for the follower vehicle in SIMULINK, the measurement feedbacks from ADAMS to the controller (2.118) in SIMULINK and the control input commands from the trajectory generator and vehicle-following controller in SIMULINK to the vehicle models in ADAMS (Figure 3.9).

- **ADAMS block:** The vehicle platooning system in ADAMS is embedded in the SIMULINK as a block named **ADAMS block**. This can be accomplished in 2 steps:

1. *Export the vehicle platoon system model from ADAMS:* ADAMS allows the creation and sharing of state variables with other softwares. Utilizing this feature, several variables are created in the ADAMS model of the platooning system. These variables are grouped into two sets: inputs and outputs. The output variables serve as the measurement states, that simulate necessary sensors for the feedback purpose while the inputs serve as control inputs to the vehicle model. They will be presented shortly in the next paragraphs. The procedure to export an ADAMS model to MATLAB is as follows:

- (a) Export the ADAMS model to an ADAMS command file: from the menu of the software, we choose File/Export. A filename is set for the command file, usually the same as the model name, with an extension of *cmd*.

Figure 3.10: ADAMS blocks

(b) Export the ADAMS command file to MATLAB: ADAMS should be closed and then re-opened with the option “Import a file” to load the exported command file. After the file is imported completely, from the menu, we choose Tools/Plugins/Controls/Load. One more item named “Controls” will appear on the menu. Again, we choose Controls/Plant export. Then, a dialogue box is popped out and a file name can be set in the “File Prefix” box, by default it is **ad_2.csd**. The plant inputs and outputs are set in the next two boxes. In the “Control Package” list box, we choose MATLAB and click OK. a set of files will be generated including a **.m** file that will be called in MATLAB.

2. *Import ADAMS model to MATLAB*: The ADAMS model is imported to MATLAB with the call to the exported file name, e.g. **ad_2.csd**, followed by a call: **adams.sys**. A new SIMULINK block, namely *adams_sub*, representing the ADAMS model of the vehicle platooning system is created and ready to be added to a SIMULINK model (Figure 3.10).

- **Controller**: The high-level control algorithms, e.g. vehicle-following controller as well as the trajectory generator, are coded in SIMULINK.
- **Input commands**: These are the variables that are used to drive the vehicle platooning

Figure 3.11: Command Inputs to ADAMS

system. Two sets of command inputs consist of desired velocities/accelerations for driving and steering. One set is the outputs of the vehicle-following controller to drive the follower vehicle. The other is the inputs to the leader vehicle generated by a trajectory generator and unknown to the follower vehicle (Figure 3.11). Table 3.2 shows the typical inputs to the platooning model in ADAMS.

Table 3.2: Dataflow from SIMULINK to ADAMS

Name in SIMULINK	Description	Name in ADAMS
<i>steer_cmd</i>	front steering rate/angle of the follower vehicle	<i>DesFSteer1</i>
<i>steer_d_cmd</i>	front steering rate/angle of the leader vehicle	<i>DesFSteer2</i>
<i>force_cmd</i>	driving velocity/acceleration of the follower vehicle	<i>DesVelocity1</i>
<i>force_d_cmd</i>	driving velocity/acceleration of the leader vehicle	<i>DesVelocity2</i>

- **Feedbacks:** These are the simulation of the required measurements from the vehicle platooning system. They are the outputs from the ADAMS block for the feedback controller as well as those for display purpose, as shown in Figure 3.12. The details are as follows.
 1. *Feedback to controller:* These are the measurements of state variables of the vehicle platoon system as required by the feedback controller (2.118). They include the acceleration, velocity and steering angle from the follower vehicle, as well as

Figure 3.12: Feedbacks from ADAMS

the relative vehicle spacing and orientation angle. In total at least eleven variables are created as listed in Table 3.3. They are independent measurements except $MeaAngle_D$ ($\dot{\theta}$) and $MeaAngle_DD$ ($\ddot{\theta}$) that can be calculated by using (2.113).

2. *Variables for display*: These are for display analysis and comparison purpose only. They include the position and heading angle of the two vehicles in the generalized coordinates as well as the current states of the leader vehicle. There are eight such variables as listed in Table 3.4.

Finally, assembling all the parts altogether yields a SIMULINK model of the closed-loop system for the vehicle platoon system as shown in Figure 3.13. The biggest block is the **adams_sub** block that represents the ADAMS model.

3.4.2 Co-simulation

In this co-simulation platform, SIMULINK is the master and the ADAMS is the slave. Two models, the vehicle platoon system in ADAMS and the high-level controller in SIMULINK, will run in a synchronous manner. Data are passed from one to the other similarly to how it is implemented in a real system. In ADAMS, the vehicle platoon system model can run in either of two modes: *interactive* or *batch*. They are slightly different in terms of user-interface. In the

Table 3.3: Dataflow from ADAMS to SIMULINK

Name in ADAMS	Description	Name in SIMULINK
<i>MeaV1</i>	measured velocity of vehicle 1	<i>v</i>
<i>MeaAcc1</i>	measured acceleration of vehicle 1	<i>u_m</i>
<i>MeaFSteer1</i>	measured front steering angle of vehicle 1	<i>gamma</i>
<i>MeaFSteer1_D</i>	measured front steering rate of vehicle 1	<i>omega</i>
<i>MeaHead1_D</i>	measured heading rate of vehicle 1	<i>theta_dot</i>
<i>MeaAngle</i>	measurement of angle ϕ	<i>phi</i>
<i>MeaAngle_D</i>	measurement of angular velocity $\dot{\phi}$	<i>phi_dot</i>
<i>MeaAngle_DD</i>	measurement of angular acceleration $\ddot{\phi}$	<i>phi_dot_dot</i>
<i>MeaDis</i>	measurement of distance d	<i>d</i>
<i>MeaDis_D</i>	measurement of velocity $\dot{d}$	<i>d_dot</i>
<i>MeaDis_DD</i>	measurement of acceleration $\ddot{d}$	<i>d_dot_dot</i>

Table 3.4: Variables for display

Variable Name	Description
<i>MeaV2</i>	Velocity of the leader vehicle
<i>MeaAcc2</i>	Acceleration of the leader vehicle
<i>MeaFSteer2</i>	Front steering angle of the leader vehicle
<i>MeaHead1</i>	Heading angle of the follower vehicle
<i>MeaDis1_X</i>	Displacement along X-axis of the follower vehicle
<i>MeaDis1_Y</i>	Displacement along Y-axis of the follower vehicle
<i>MeaDis2_X</i>	Displacement along X-axis of the leader vehicle
<i>MeaDis2_Y</i>	Displacement along Y-axis of the leader vehicle

Figure 3.13: SIMULINK model

interactive mode, the ADAMS graphic motion simulation will be activated and displaying the simulation in 3D motions. On the other hand, the batch mode turns off the graphic interface and runs the simulation solely numerically (i.e. only ADAMS/Solver is called). Consequently, the simulation in the batch mode runs much faster.

All the intermediate data is stored both in MATLAB workspace and in ADAMS database. States and outputs can be displayed by plots. Furthermore, video clips can also be generated to show the whole process of motion, as shown in Figure 3.7. The maneuver performance of the platoon system can be illustrated clearly and effectively from different view points as well.

3.5 Simulation Study

The co-simulation platform that has been developed in the previous section is used for the study of the vehicle-following controller developed in the previous chapter.

The study in the previous chapter has revealed some effects of parameters l and p on the tracking performance. It also pointed out necessary conditions for the controller to be valid and able to converge. Simulations will be carried out to verify this study and to provide more insights about the behaviour of the system.

The physical information of the real vehicle, Cycab, and its sensors is used in the simulation.

$$\begin{aligned} 2a &= 1.2m \\ \gamma_{\max} &= 22.5^\circ = \frac{\pi}{8} rad \\ d_{\max} &= 20m \end{aligned}$$

The conditions (2.102) and (2.103) are realized

- *Look-ahead tracking:* $f = 1$

$$\begin{cases} 0 \ll l \ll 20 \\ 0 < p < 4 \end{cases} \quad (3.4)$$

- *Look-behind tracking:* $f = -1$

$$\begin{cases} -20 \ll l \ll 0 \\ -4 < p < 0 \end{cases} \quad (3.5)$$

To simplify, the reference point is chosen at the rear midpoint, $a_f = 2a$ and $a_r = 0$. This way of reference point selection for a car-like vehicle is very common due to the fact that the side-slip angle is always zero, $\beta = 0$.

Simulations were extensively carried out with different sets of parameters, e.g. l , p , λ and ξ . Some results are shown in this section. The simulations were set up in such a way that the leader vehicle always traveled on the same trajectory every time while the follower vehicle, with different sets of controller parameters, would show different tracking behaviours.

3.5.1 Kinematics-based

We examined the following situations where the leader vehicle started moving on a straight path. After having reached to a certain speed, it begun to turn until its steering angle reached a predetermined constant angle and continued to move at that constant speed and constant steering angle. The maximum steering angle was set at $|\gamma_d| = 0.2$ radian ($\sim 11.46^\circ$). For look-ahead tracking, its velocity increased from zero to $2m/s$. For look-behind tracking, it decreased from zero to $-2m/s$. The examined time interval was 100 seconds, which means the two vehicle traveled roughly 5 rounds. The initial error was zero for all the cases.

We also examined the situations with l priori chosen and those with p priori chosen. Parameter λ was fixed at 5 to facilitate the comparison.

Figure 3.14: Necessary conditions for circular trajectory look-ahead tracking when $\gamma_d = 0.2$

• **Look-ahead tracking**

With $\gamma_d = 0.2$, Figure 2.10 reduces to the figure of condition in Figure 3.14

– *p priori chosen:*

Based on the analysis in Section 2.3.3, one value was chosen from each of the three ranges of p : $0 < p \leq 1$, $1 < p < 2$ and $2 \leq p < 4$. They were $p = 0.5$, $p = 1.5$ and $p = 2.5$.

Figure 3.14 shows l can not be chosen too big, as shown in (2.130) or (2.131), i.e. $l < l_{\max}^+(p)$. And it also shows if $p \geq 2$, there may be one (when $p = 2$) or two values of l , namely l_1^+ and l_2^+ , that guarantee the two vehicles eventually move on the same circular path.

Table 3.5: Tracking results for selected sets of (l, p) for look-ahead tracking with p priori chosen

p	l			
	$2m$	$3.1m$	$4m$	$7m$
0.5	Figures 3.20-3.23			
1.5	Figures 3.24-3.27			
2.5	Figures 3.28-3.31			

3.5. Simulation Study

84

With $p = 2.5$,

$$\begin{aligned} l_{\max}^+ &= 2a \left(p - 1 + \sqrt{\frac{1}{\tan^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}} + (p - 1)^2} \right) \approx 7.1339 \\ l_1^+ &= 2a \frac{\sin(p-1)\gamma_d - \sqrt{\sin p \gamma_d \sin(p-2)\gamma_d}}{\sin \gamma_d} \approx 0.4636 \\ l_2^+ &= 2a \frac{\sin(p-1)\gamma_d + \sqrt{\sin p \gamma_d \sin(p-2)\gamma_d}}{\sin \gamma_d} \approx 3.1064 \end{aligned}$$

With $p = 1.5$,

$$l_{\max}^+ = 2a \left(\frac{\sin(p-1)\gamma_{\max}}{\sin \gamma_{\max}} + \sqrt{\frac{1}{\tan^2 \gamma_d} - \frac{\cos^2(p-1)\gamma_{\max}}{\sin^2 \gamma_{\max}}} \right) \approx 5.6699$$

With $p = 0.5$,

$$l_{\max}^+ = 2a \left(p - 1 + \sqrt{\frac{1}{\tan^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}} + (p - 1)^2} \right) \approx 4.4568$$

For safety reason, normally l should not be chosen too small, thus, the following values of l were taken for testing with each of the values of p : $l = 2m$, $l = 3.1m$, $l = 4m$ and $l = 7m$.

The results are shown in Figures 3.20-3.31 as listed in Table 3.5. For each set of (l, p) , four figures are generated which show the trajectories of the leader vehicle and that of the follower vehicle as in Figures 3.20, 3.24 and 3.28; the tracking error (difference between desired and actual spacing between vehicle) as in Figures 3.21, 3.25 and 3.29; and two important states: velocity, Figures 3.22, 3.26 and 3.30, and steering angle, Figures 3.23, 3.27 and 3.31, of the two vehicles. In these figures, the leader vehicle is always represented by a blue dotted line and its states are the same for every case. The states of the follower vehicle are always continuous and smooth and the tracking performance is good with the proposed sets of (l, p) .

– *l priori chosen:*

Similarly, when l is priori chosen, p must be chosen so that Theorem 2.3.14 is satisfied. From Figure 3.14, we note that for l greater than a certain value, $l \geq \bar{l}^+$, no matter what value is chosen for p , the tracking will be unstable. $\bar{l}^+$ is

$$\bar{l}^+ = 2a \left(\frac{\pi}{2\gamma_{\max}} - 1 + \sqrt{\frac{1}{\tan^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}} + \left(\frac{\pi}{2\gamma_{\max}} - 1 \right)^2} \right) \approx 9.7783$$

3.5. Simulation Study

85

For $l < \bar{l}^+$, p must be chosen greater than 0 for some smaller values of l , i.e. $0 < l \leq l_0^+$, and greater than a lower limit $p_{\min}^+(l)$ for the bigger values of l , i.e. $l_0^+ < l < \bar{l}^+$.

We can calculate l_0^+ as follows

$$l_0^+ = 2a \left(\sqrt{\frac{1}{\sin^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}}} - 1 \right) \approx 3.9625$$

Also, for each value of l such that $0 < l < \bar{l}^+$, we may/may not find one value of p , namely p^+ , that guarantee the tracking convergence of the two trajectories.

Then, we choose the following values of l : $2m$, $4m$, $6m$ and $7m$.

For $l = 7m$, we have

$$p_{\min}^+ = 1 + \frac{l^2 - (2a)^2 \left(\frac{1}{\tan^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}} \right)}{4al} \approx 2.4160$$

For $l = 6m$,

$$p_{\min}^+ = 1 + \frac{1}{\gamma_{\max}} \arcsin \left(\frac{l^2 - (2a)^2 \left(\frac{1}{\tan^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}} \right)}{4al} \sin \gamma_{\max} \right) \approx 1.7405$$

$$p^+ = 1 + \frac{1}{\gamma_d} \arcsin \left(\frac{l^2 + (2a)^2}{4al} \sin \gamma_d \right) \approx 3.7140$$

For $l = 4m$,

$$p_{\min}^+ = 1 + \frac{l^2 - (2a)^2 \left(\frac{1}{\tan^2 \gamma_d} - \frac{1}{\sin^2 \gamma_{\max}} \right)}{4al} \approx 0.0405$$

$$p^+ = 1 + \frac{1}{\gamma_d} \arcsin \left(\frac{l^2 + (2a)^2}{4al} \sin \gamma_d \right) \approx 2.8462$$

For $l = 2m$, $p_{\min}^+ = 0$ and

$$p^+ = 1 + \frac{1}{\gamma_d} \arcsin \left(\frac{l^2 + (2a)^2}{4al} \sin \gamma_d \right) \approx 2.1355$$

Table 3.6: Tracking results for selected sets of (l, p) for look-ahead tracking with l priori chosen

l	p			
	1.5	2.1355	2.8462	3.95
$2m$	Figures 3.32-3.35			
$4m$	Figures 3.36-3.39			
$6m$	Figures 3.40-3.43			
$7m$	Figures 3.44-3.47

Figure 3.15: Necessary conditions for circular trajectory look-behind tracking when $\gamma_d = 0.2$

Thus, we choose the following values of p to test with each value of l : 1.5, 2.1355, 2.8462, and 3.95.

The results are displayed in Figures 3.32-3.47 as listed in Table 3.6. Similarly, each case is represented by a set of four figures including trajectories (Figures 3.32, 3.36, 3.40 and 3.44), tracking error (Figures 3.33, 3.37, 3.41 and 3.45), velocity (Figures 3.34, 3.38, 3.42 and 3.46) and steering angle (Figures 3.35, 3.39, 3.43, 3.47). For $l = 2m$ and $4m$, everything looks fine as predicted in Chapter 2. but with $l = 6m$ and $7m$, some sets of (l, p) inside the valid range result in undesirable performance (Figure 3.42 and Figure 3.46) where the steering angle saturation is experience. This is because in Chapter 2 the limit of the steering angle has not been considered for the transient period but for the stabilized/convergent period only.

• Look-behind tracking

With $\gamma_d = 0.2$, Figure 2.9 becomes Figure 3.15

– *p priori chosen:*

We chose $p = -1$ and $p = -2$ from $-4 < p < 0$ to test.

From Figure 3.15, for every p , l has a lower bound, i.e. $l > l_{\min}^-(p)$. There is also one value of l , namely l^- , such that the pair (l, p) can guarantee the tracking convergence of two circular trajectories,

3.5. Simulation Study

87

With $p = -1$, we have

$$l_{\min}^- = 2a \left(p - \sqrt{\frac{1}{p^2 + \sin^2 \gamma_d} - \frac{1}{\tan^2 \gamma_{\max}}} \right) \approx -6.6342$$

$$l^- = 2a \left(\frac{\sin p \gamma_d}{\tan \gamma_d} - \sqrt{\frac{\sin^2 p \gamma_d}{\tan^2 \gamma_d} + 1} \right) \approx -2.8563$$

With $p = -2$, we have

$$l_{\min}^- = 2a \left(p - \sqrt{\frac{1}{p^2 + \sin^2 \gamma_d} - \frac{1}{\tan^2 \gamma_{\max}}} \right) \approx -8.2182$$

$$l^- = 2a \left(\frac{\sin p \gamma_d}{\tan \gamma_d} - \sqrt{\frac{\sin^2 p \gamma_d}{\tan^2 \gamma_d} + 1} \right) \approx -4.9042$$

Then for each value of p , the following values of l were tested: $-1m$, $-2m$, $-2.85m$, $-4.9m$ and $-7m$.

The results are illustrated in Figures 3.48-3.55 as arranged in Table 3.8 with Figures 3.48 and 3.52 for trajectory, Figures 3.49 and 3.53 for tracking error, Figures 3.50 and 3.54 for velocity and Figures 3.51 and 3.55 for steering angle. Like the look-ahead cases, the performance in look-behind tracking of the controller is also good. More discussions and will be given shortly in the next section.

– l priori chosen:

From Figure 3.15, it is noted that when l is smaller than a certain value, namely $\bar{l}^-$, i.e. $l \leq \bar{l}^-$, the tracking will be unstable no matter what p is chosen. That $\bar{l}^-$ is

$$\bar{l}^- = -2a \left(\frac{\pi}{2\gamma_{\max}} + \sqrt{\left(\frac{\pi}{2\gamma_{\max}} \right)^2 + \frac{1}{\sin^2 \gamma_d} - \frac{1}{\tan^2 \gamma_{\max}}} \right) \approx -11.9600$$

When l is chosen such that $0 > l \geq l_0^-$ where

$$l_0^- = -2a \sqrt{\frac{1}{\sin^2 \gamma_d} - \frac{1}{\tan^2 \gamma_{\max}}} \approx -5.3001$$

Table 3.7: Tracking results for selected sets of (l, p) for look-behind tracking with p priori chosen

p	l				
	$-1m$	$-2m$	$-2.85m$	$-4.9m$	$-7m$
-1	Figures 3.48-3.51				
-2	Figures 3.52-3.55				

then any value of p in $(-4, 0)$ is acceptable. For l such that $\bar{l}^- < l < l_0^-$, p must be chosen so that $-4 < p < p_{\max}(l)$, where $p_{\max}$ is the upper bound for p depending on given l .

In addition, for any l that $\bar{l}^- < l < -2a = 1.2$, there may/may not be a value of p , namely p^- , such that the pair (l, p) could lead to the tracking convergence of the two circular trajectories.

Thus, the following values were chosen for l : $-1m$, $-4m$, and $-7m$.

For $l = -1m$ or $l = -4m$, we have $p_{\max} = 0$.

With $l = -4m$, we also have

$$p^- = \frac{1}{\gamma_d} \arcsin \left(\frac{l^2 - (2a)^2}{4al} \tan \gamma_d \right) \approx -1.5625$$

And for $l = -7m$,

$$p_{\max} = \frac{l^2 - (2a)^2 \left(\frac{1}{\sin^2 \gamma_d} - \frac{1}{\tan^2 \gamma_{\max}} \right)}{4al} \approx -1.2446$$

$$p^- = \frac{1}{\gamma_d} \arcsin \left(\frac{l^2 - (2a)^2}{4al} \tan \gamma_d \right) \approx -3.0561$$

Then, the following values of p were chosen to test with each l : -3.0561 , -2.5 , -1.5625 , and -0.5 .

Sets of 4 figures are produced to illustrate the tracking performance for $l = -1m$, $-4m$ and $-7m$ as in Table 3.8.

• **Tracking with different values of λ :**

One set of (l, p) was picked up for each of the two tracking modes to test with different values of λ from as small as 0.1 to as big as 100 in order to gain some insights on its

Table 3.8: Tracking results for selected sets of (l, p) for look-behind tracking with l priori chosen

l	p			
	-3.0561	-2.5	-1.5625	-0.5
$-1m$	Figures 3.56-3.59			
$-4m$	Figures 3.60-3.63			
$-7m$	Figures 3.64-3.67

Figure 3.16: Tracking error for $l = 4$ and $p = 2.8462$

Figure 3.17: Steering angle for $l = 4$ and $p = 2.8462$

Figure 3.18: Tracking error for $l = -4$ and $p = -1.5625$

Figure 3.19: Steering angle for $l = -4$ and $p = -1.5625$

influence on tracking performance. The two sets were $(l = 4m, p = 2.8462)$ and $(l = -4m, p = -1.5625)$. The tracking error and tracking result of the steering angle are shown in Figures 3.16-3.19. The other states are somewhat identical thus not displayed. In all cases, the tracking is successful but with different tracking error.

3.5.1.1 Discussions

The tracking of the follower vehicle and its state variables such as velocity and steering angle to those of the leader vehicle behaved similarly to what was discussed earlier in Chapter 2 in accordance with the selection of parameters l and p . If the tracking is successful, the states of the follower vehicle will converge to constant values. Moreover, when tracking a circular trajectory, depending on the selection of (l, p) , the resultant circular trajectory of the follower vehicle

may be identical to, inside or outside that of the leader vehicle as discussed and calculated in Chapter 2.

Especially, during the straight line runs (the first 10 seconds), the tracking was consistent and stable. In these cases, parameter p had no effects at all because the steering angle was zero. The tracking error was large mainly when the leader vehicle accelerated or decelerated. When its speed became constant, the follower vehicle could track with very small errors.

When the leader vehicle made circular runs, it took some time for the follower vehicle to engage on its convergent circle, if any. It is because it is the steering rate that was controlled directly not the steering angle which is directly related to the circle's radius. Therefore, the steering angle was unable to catch up the desired one as fast as the velocity did.

The simulation results also reveal some cases where even though the analysis in Chapter 2 says the tracking should be stable, it was actually not quite successful, for example, when $p = 2.42$ and $l = 7m$ (Figures 3.44-3.47). The overshoot of the response of the steering angle reached the physical limit of the steering angle and created saturation which leads to very large tracking errors or even instability. Thus, it would be advisable that the parameters l and p should not be chosen too close to their bounds.

The behaviour of the look-behind tracking is slightly different from that of the look-ahead tracking at the beginning. The follower vehicle turned to the other direction a bit before it turned back to the same direction as that of the leader vehicle, whereas in look-ahead tracking, the follower vehicle always turned to the same direction as that the leader vehicle turned to. This maneuver of the look-behind tracking controller can be explained by the desired point that was tracked. In look-ahead tracking mode, the rear midpoint of the leader vehicle is tracked while in look-behind tracking it is the front midpoint. When going reverse and attempting to turn, the front part of the leader vehicle moved slightly to the opposite side with respect to the follower vehicle before the rear part pulled it back to the same side. It is why the follower vehicle also turned a bit to that opposite side.

Simulations show that parameter λ of the closed-loop system (2.92) had some influence on the performance, especially during the transient period. Obviously, the bigger values of λ , the smaller tracking error. It is because with high natural frequencies, the closed-loop system can quickly regulate the tracking error to zero. Higher frequencies also result in higher peak response. Fortunately, for the case where the velocity and steering angle are not so big, the differences are very small (for example as in Figure 3.17 the difference of the peak steering

angle between $\lambda = 0.1$ and $\lambda = 100$ is 0.000116 radian or 0.0066°). In practice, the peak difference could be much higher due to the delay in the actuators. Therefore, it is recommended not to choose too big values of λ . During the turns, there was always a static error. It is because the closed-loop system in use belongs to the class of PD-like controllers. This static error depends on the steering angle and the natural frequency. When the vehicle runs on a straight path, this static error is eliminated. λ could be increased to reduce this error. And the static errors for the look-behind tracking are bigger than those for the look-ahead tracking.

3.5.2 Dynamics-based

The dynamics-based controller is also verified by simulations. A complex trajectory was implemented for the leader vehicle including a couple of turns and some speeding-up and slowing-down. The turns made by the leader vehicle were sharp ones with maximum steering angle of about 17° and its heading angle changed 53° in just about 5 seconds. Its speed and acceleration changed rapidly to slow down before a turn or speed up after the turn.

Several sets of (l, p) for look-ahead and look-behind tracking modes were tested. Similar to the kinematic-based tracking situations, a set of figures is generated to illustrate the tracking results of trajectory, tracking error, velocity, steering angle and additionally acceleration and heading angle as shown in Figures 3.68-3.91.

3.5.2.1 Look-ahead tracking

- p is priori set to 2 and l is in turn taking 1m, 2.5m, 4m and 5.5m: The figure set of Figures 3.68-3.73 shows the tracking results for this case.
- l is priori set 2.5m and p is taking 0.5, 1, 2 and 3: Results are shown in Figures 3.74-3.79.

3.5.2.2 Look-behind tracking

- p is priori set to -1.5 and l is in turn taking -1m, -2.5m and -4m: Figures 3.80-3.85 illustrates the tracking performance for this case
- l is priori set to -2.5m and p is taking -0.5, -1, -2 and -3: Tracking results such as trajectory, tracking error, velocity, acceleration, steering angle and heading angle are displayed in Figures 3.86-3.91, respectively.

3.5.2.3 Discussions

The results show that the follower vehicle successfully followed the leader vehicle (Figures 3.68, 3.74, 3.80 and 3.86) for both look-ahead and look-behind tracking situations. In general, the tracking behaviours were slightly similar to those of the kinematic-based controller.

The tracking errors were small especially along the straight path as shown in Figures 3.69, 3.75, 3.81 and 3.87. However, during turns, the value of parameter p clearly affects the tracking performance. Though starting at the same initial position and orientation, with smaller values of p , e.g. $0 < p < 2.5$ for look-ahead tracking, the follower vehicle tried to cut corners to catch up with the leader vehicle. In contrast, larger values of p resulted in overshooting before turning. This both maneuvers are not welcome as the car might penetrate into another lane. The reason is because the follower vehicle was trying to converge to a circle corresponding to the leader vehicle's circular path. And it was shown in the previous tests that depending on the chosen parameters l and p , the circle can be smaller, bigger or equal to the one of the leader vehicle. There are possibly sets of (l, p) with what the two circles could be identical. However, since the transient period could take some time, the leader vehicle may have changed its steering angle, which resulted in a different circular trajectory, before the follower vehicle could actually follow. This is shown clearly in the figures of tracking errors and steering angles (Figures 3.72, 3.78, 3.84 and 3.90). The tracking error was bigger than that of the kinematic-based controller since the turns were much sharper with the steering angle being about 17° compared to 11° in the previous cases.

The influence of l is also obvious. For larger values of l , the follower vehicle tried to take the shortcuts, while with smaller l , there were also overshoots (Figures 3.68, 3.80). This behaviour was predicted in the mathematic analysis for circular trajectories. One new finding is that in these special cases, even though parameter l may exceed the instability bound generated based on the maximum steering angle of the leader vehicle, the tracking was still successful. It is because the leader vehicle did not make full round turns. And before the instability begun, the leader vehicle had changed its course such that the tracking error was reduced. For other situations, it would not be that fortunate.

The look-behind tracking or backward tracking, which is difficult for other controllers, is solved completely using our controllers. This can be obtained since the controller inherits good driving maneuvers from human drivers who have been experiencing and solving the problems every day.

3.6 Conclusions

A co-simulation platform exploiting the advantages of two modeling and simulation softwares, ADAMS and SIMULINK, has been developed. The platform replaces the commonly used vehicle model in terms of differential equations by a near-reality 3D model whilst it still maintains the convenience and flexibility of using SIMULINK in designing a high-level controller. The development takes the full advantages of two individual environments to achieve higher fidelity. The platform is not just suitable for simulations of a platoon of vehicles, but also for other type of vehicle control algorithms like path planning, path following or auto parking. Moreover, the platform is able, with a simple replication of more vehicles in ADAMS, to simulate a system of more than 2 vehicles.

In order to verify and investigate the performance of the vehicle-following controller developed in the previous chapter, an extensive study using the simulation platform has been carried out. Different sets of parameters for two tracking situations, look-ahead and look-behind, have been used for testing and verifying the analysis discussed earlier. Simulations have also helped uncover some behaviours of the controller especially during the transient period or when the two vehicles travels on a complex path. The influences of design parameters l , p , λ and ξ are therefore identified.

3.6. Conclusions

Figure 3.20: Trajectories for $p = 0.5$

Figure 3.21: Tracking error for $p = 0.5$

Figure 3.22: Velocities for $p = 0.5$

Figure 3.23: Steering angles for $p = 0.5$

Figure 3.24: Trajectories for $p = 1.5$

Figure 3.25: Tracking error for $p = 1.5$

Figure 3.26: Velocities for $p = 1.5$

Figure 3.27: Steering angles for $p = 1.5$

Figure 3.28: Trajectories for $p = 2.5$

Figure 3.29: Tracking error for $p = 2.5$

Figure 3.30: Velocities for $p = 2.5$

Figure 3.31: Steering angles for $p = 2.5$

Figure 3.32: Trajectories for $l = 2$

Figure 3.33: Tracking error for $l = 2$

Figure 3.34: Velocities for $l = 2$

Figure 3.35: Steering angles for $l = 2$

Figure 3.36: Trajectories for $l = 4$

Figure 3.37: Tracking error for $l = 4$

3.6. Conclusions

Figure 3.44: Trajectories for $l = 7$

Figure 3.45: Tracking error for $l = 7$

Figure 3.46: Velocities for $l = 7$

Figure 3.47: Steering angles for $l = 7$

Figure 3.48: Trajectories for $p = -1$

Figure 3.49: Tracking error for $p = -1$

3.6. Conclusions

Figure 3.50: Velocities for $p = -1$

Figure 3.51: Steering angles for $p = -1$

Figure 3.52: Trajectories for $p = -2$

Figure 3.53: Tracking error for $p = -2$

Figure 3.54: Velocities for $p = -2$

Figure 3.55: Steering angles for $p = -2$

Figure 3.56: Trajectories for $l = -1$

Figure 3.57: Tracking error for $l = -1$

Figure 3.58: Velocities for $l = -1$

Figure 3.59: Steering angles for $l = -1$

Figure 3.60: Trajectories for $l = -4$

Figure 3.61: Tracking error for $l = -4$

Figure 3.62: Velocities for $l = -4$

Figure 3.63: Steering angles for $l = -4$

Figure 3.64: Trajectories for $l = -7$

Figure 3.65: Tracking error for $l = -7$

Figure 3.66: Velocities for $l = -7$

Figure 3.67: Steering angles for $l = -7$

3.6. Conclusions

Figure 3.68: Trajectories for $p = 2$

Figure 3.69: Tracking error for $p = 2$

Figure 3.70: Velocities for $p = 2$

Figure 3.71: Accelerations for $p = 2$

Figure 3.72: Steering angles for $p = 2$

Figure 3.73: Heading angles for $p = 2$

Figure 3.74: Trajectories for $l = 2.5m$

Figure 3.75: Tracking error for $l = 2.5m$

Figure 3.76: Velocities for $l = 2.5$

Figure 3.77: Accelerations for $l = 2.5$

Figure 3.78: Steering angles for $l = 2.5$

Figure 3.79: Heading angles for $l = 2.5$

3.6. Conclusions

Figure 3.80: Trajectories for $p = -1.5$

Figure 3.81: Tracking error for $p = -1.5$

Figure 3.82: Velocities for $p = -1.5$

Figure 3.83: Accelerations for $p = -1.5$

Figure 3.84: Steering angles for $p = -1.5$

Figure 3.85: Heading angles for $p = -1.5$

Figure 3.86: Trajectories for $l = -2.5m$

Figure 3.87: Tracking error for $l = -2.5m$

Figure 3.88: Velocities for $l = -2.5$

Figure 3.89: Accelerations for $l = -2.5$

Figure 3.90: Steering angles for $l = -2.5$

Figure 3.91: Heading angles for $l = -2.5$

Chapter 4

Implementation of a Vehicle-Following Controller

This chapter describes in detail the vehicles used for experiments of the vehicle-following controllers developed in Chapter 2. These vehicles were made at Robosoft company in France and called Cycab. In total, we have three of this type of vehicles painted in different colors: red, blue, and white. Therefore, we call them Red Cycab, Blue Cycab and White Cycab. They are almost identical. Blue and White Cycab vehicles are four-wheel-driving front-wheel-steering vehicles while Red Cycab is a four-wheel-driving four-wheel-steering vehicle. After being extensively used over few years, two of them, White Cycab and Red Cycab, broke down. Fortunately, we managed to repair the Red Cycab with a new design of low-level control system that will be presented in the next part. Then, a platoon of two vehicles can be implemented (Figure 4.1).

After the description of the Cycab vehicle, a laser-scanner-based sensing system is designed and developed in order to provide reliable and fast measurements of the relative distance and orientation between two vehicles, as well as their time derivative estimations. By utilizing this sensing system, the kinematics-based vehicle-following controller (2.99) is implemented. The experimental results and discussions will end the chapter.

4.1 The low-level Control System of the Blue Cycab

The Cycab vehicles are equipped with one direct-current brushed motor at each wheel for the driving purpose. For Blue Cycab, the two front wheels are linked with a hydraulic steering mechanism while its two back wheels are fixed straight. Hence, Blue Cycab is a 2-wheel-

Figure 4.1: The CyCab vehicles

steering vehicle and in most cases it works as a car-like vehicle. Red Cycab is however more complex. In addition to the front steering system the same as that of Blue Cycab, it possesses another steering system for the back two wheels. This additional steering system makes it a four-wheel-steering vehicle and offers more flexible maneuvers such as tightening turning and parallel translation. All these driving and steering motors of the vehicle are originally controlled by an onboard computer running on the *ALBATROSTM* real-time operating system.

Some of the specification of Cycab is listed in Table 4.1.

4.1.1 Hardware configuration

Figure 4.2 shows the original structure of the Cycab, which is still valid for Blue Cycab. This low-level control system is a computer-based controller utilizing a Motorola MC68LC040-based Embedded Controller and an 8-axis Motion Control board to control 4 driving motor controllers and 1 or 2 steering controllers.

For each driving motor, one feedback control loop is set up to drive the motor to a desired velocity at a desired acceleration. In fact, a closed-loop PID controller is used for each driving motor. The PID filter is implemented on the Motion Control board with its PID gains adjustable by software. The PID controller takes the feedback of the motor speed from an incremental encoder mounted coaxially with the motor (Figure 4.3). The output from the PID controller will be converted to a DC voltage ranging from 0V to 5V which is treated as the reference voltage

4.1. The low-level Control System of the Blue Cyclic

Table 4.1: Specifications of CyCab

Size	1.9m * 1.2m * 1.65m
Distance between wheels	1.2m * 1.1m
Weight	~ 350kg inclusive of batteries
Battery Power	48V, operational up to 2 hours
Load Capacity	2 persons
Maximum Speed	±5m/s
Speed resolution	0.1m/s
Maximum Steering	±22.5°
Steering resolution	0.1°

Figure 4.2: Original hardware configuration of CyCab

Figure 4.3: The mounting of a driving motor

4.1. The low-level Control System of the Blue Cycab

109

Figure 4.4: The mechanism of a steering system

to a motor controller. The motor controller actually works as an amplifier which generates a high-current PWM voltage proportional to the reference voltage to drive the motor.

A steering system is also a closed-loop system. Two wheels are mechanically linked together and an electro-hydraulic system is used to transform a reference voltage into a force acting on the mechanical links to steer them. A PID controller oversees the whole process and takes feedback of angular position from an absolute encoder. In Figure 4.4, the absolute encoder is hidden under the flat surface and firmly linked to the piston of the hydraulic cylinder. The translation of the piston is created under the pressure of the oil inside the cylinder, which in turn is pumped by a set of a DC motor and a hydraulic pump. Similarly to how to control the DC motor in driving part, the motion of the DC motor of the steering system is also generated by a current-modulated PWM voltage from a motor controller (or amplifier). This PWM voltage is generated proportionally based on a $\pm 10V$ reference voltage from the PID controller.

4.1.2 Software configuration

The onboard computer-based embedded controller runs on a real-time operating system, namely *ALBATROSTM*, specially designed for multi-axis devices real-time control. The features of *ALBATROSTM* include a real-time kernel, all I/O drivers, generalized PID regulators, trajectory generators, sensor read modules and a command interpreter. The real-time kernel is to generate the real-time clocks for the entire system to synchronize operations. The closed-loop PID regulators and the trajectory generators are two periodic tasks that base on the real-time clocks. A trajectory generator is to create a reference trajectory (position with respect to time) whenever a new desired position/velocity is set. A PID regulator is then to drive the motor to

Figure 4.5: Software configuration of CyCab

follow this reference trajectory. In general, the sampling period of the trajectory generator is a multiple of that of the PID regulator. By default, they are 10ms for PID controllers and 40ms for trajectory generators. A wide set of text-based commands is also provided through which users can use to take virtually all the controls to any part, even the kernel, of the system. Some examples of the commands are listed in Table 4.2.

4.1.3 Communication and control of the Cycab

Since the embedded controller works as a stand-alone computer, the only possible connection to outside is through serial communication channels (RS232). Therefore, to access to low-level of the vehicle, for example to control the motors, a connection from another computer, called the host computer, must be established and commands are sent over this connection to the command interpreter of the *ALBASTROSTM* as shown in Figure 4.5.

To develop an application that autonomously controls the vehicle, the high-level controller must be carried out on the host computer and able to issue control commands to the onboard controller as well as to receive encoder readings over the RS232 communication channel. The provided set of text-based commands makes the communication easy to set up and maintain. Virtually all computers nowadays are provided with one or two serial ports (COM1 and COM2). The software libraries for serial communication are also available, for example the Microsoft Communications Control MSCOMM which is integrated in all Microsoft Windows operating system. The communication can be quickly established by some settings and function calls. Then, a so-called command generator and interpreter module is also necessary to be developed on the host computer to communicate with the onboard controller using explicit text-based commands.

Though the control of the vehicle can be achieved easily over RS232 connection, the system

4.1. The low-level Control System of the Blue Cycab

Table 4.2: Frequently used commands of Blue Cycab

Command	Description
serv on t=bv d=1	Turn on PID control loops for velocities
serv of d=1	Turn off control loops for driving
serv on t=bp d=2	Turn on PID control loop for steering angle
serv of d=2	Turn off control loop for steering
move v ac=<fl>,<fr>,<rl>,<rr>	Set desired velocity for each driving motor with resolution of 0.1m/s <fl> : front-left desired velocity <fr> : front-right desired velocity <rl> : rear-left desired velocity <rr> : rear-right desired velocity
move v rc=<fl>,<fr>,<rl>,<rr> p=<n>	Set relative velocity with a preset time to complete → to control accelerations <n> : number of control periods to reach to the command
move p ac=<fs>,0 d=2	Set desired steering angle with resolution of 0.1° <fs> : desired steering angle
move v rc=<fs>,0 p=<n> d=2	Set relative steering angle with a preset time to complete → to control steering rate <n> : number of control periods to reach to the command
pos s=s d=1	Read values from the 4 incremental encoders
pos s=g d=2	Read value from the absolute encoder in 0.1°

might encounter two main problems involved in this kind of connection. If the communication channel is lost, the vehicle will definitely out of control. This is very disadvantageous. We experienced this problem before when the RS232 module of the onboard controller was burnt due to a sudden electrical surge and the vehicle was simply out of control. The other main drawback of the RS232 connection with concerns of autonomous control is its bandwidth limit. The transfer rate of the RS232 connection is always fixed at 38400 baud and if the host computer tries to send frequent commands and receive real-time feedbacks, communication bottlenecks will occur.

The Blue Cycab vehicle is, however, a good test-bed for car-like applications running at relatively low speeds. Some of our research experiments that were successfully carried out on this vehicle include GPS-based path-following and auto-parking.

4.2 The sensing system of relative measurements

In this section, a reliable, fast and accurate ranging measurement method is developed for the purpose of measuring the relative distance and orientation between two vehicles. The measuring proposal is designed such that it eases the setup and implementation.

4.2.1 Selection of sensor

As mentioned earlier in Chapter 1, to measure the relative position and orientation of the two vehicles, one might measure their absolute position and take the difference between them. The absolute position in a global coordinate system can be obtained by using GPS, or any other positioning systems. The disadvantage of this method is that the position of the leading/preceding vehicle must be then transmitted to the follower vehicle over a wireless communication channel which might cause delay or loss of the information. Therefore, a non-contacting sensing system installed onboard the follower vehicle to detect and measure the relative distance and orientation is more desirable.

There are also two approaches of non-contacting range measurement: passive sensing and active sensing.

Passive sensing means the environment is sensed and from what the sensor provides, the target is detected. The sensor is normally well calibrated beforehand to convert the sensed information of the target to the distance. Cameras or other image capturing devices belong to

4.2. The sensing system of relative measurements

113

this approach [115–119]. A camera consists of an array of photo diodes that convert the photon of light coming on it into a proportional amount of electron which, in turn, is converted to a voltage. The obtained image can be color or gray-scale depending on the type of photo diodes in use. The target must then be located on that image. Usually, by exploiting the symmetric feature of the back of the vehicle, one might be able to detect the portion belonging to the vehicle on the image. This process however is really time consuming in reality since the image contains lots of background objects. Lighting or shadows also affect the recognition process. That is why nowadays passive sensing is not widely used in car-following applications.

Active sensing, on the other hand, interacts with the environment by sending out a physical signal to the environment. The signal can be sound, light, or some kinds of modulated wave. When the signal arrives at the target, they are reflected back to the sensor. The range from the sensor to the target can be computed by either measuring the time traveled (time of flight TOF), by measuring the phase shift between transmitted and received signals, or by measuring the geometric point of arrival of the signal (triangulation). To detect not only the relative distance but also the relative orientation, the direction of the transmitting/receiving signal is required. Additionally, diffuse signals such as sound are not suitable for orientation detection. Thus, most of the ranging systems use visible or invisible lights and a optoelectronic mechanism to transmit and receive directional signals. With a scanning mechanism, the view of the sensor is sequentially directed to areas in the environment.

Active sensing improves the measurement accuracy compared to passive sensing. However, it still has to solve the problem of target detection, especially when other unwanted objects appear in the acquired data. It would be wise to somehow filter out all those background objects and retain only the target object in the “image”. One way of doing that is to specify purposely the wanted target using special marks that the sensor can “see”. This method improves dramatically the measurement accuracy because after being filtered, the “image” only contains the designed marks and probably some unexpected points that by any means also satisfy the filter characteristics. A well-designed set of marks can easily help detect the patterned marks among these points.

Two different designs have been tried as follows

- *The usage of a linear camera and infrared lights*
- *The usage of a laser scanner*

The first design utilizes a linear camera, the CIL2048 Line-Scan camera manufactured by LORD Ingnierie Company [120], to detect infrared lights mounted on the leader vehicle. The updating rate of this design is very fast, up to a thousand samples per second. But since it was difficult to create enough power to supply the infrared lights so that the linear camera could see them clearly, this design was finally given up.

4.2.2 Using a laser scanner and reflective tapes

A pulsed time-of-flight range finder has been developed by Erwin Sick and his company [121] (Figure 4.6). This sensor uses pulsed modulation to determine range and nowadays becomes very popular as it is robust and well engineered. The SICK sensor typically has a maximum range of 50m, with range resolution of better than 2cm, angular resolution of 0.5° and a scan time ranging from 13ms to 53ms. The sensor uses a rotating mirror to produce the scan, which can be over 180° (see Table 4.3). The scanning geometry employed by the sensor is shown in Figure 4.7.

A step motor controls the rotation of the mirror. At each position, a beam of light is transmitted and the laser waits for the reflection to arrive. The time interval between when starting to transmit and when receiving the reflection is measured and proportional to the distance to the target.

$$d = \frac{\Delta t}{2c} \quad (4.1)$$

where Δt is the measured time interval and c is the speed of light.

The light uses in the first laser scanners was infrared light but then replaced by a laser beam for better performance.

Due to its high accuracy and fast transmission, the SICK laser scanners are now widely used in industrial applications such as quality control and intrusion detection. In robotics, the sensor is used for navigation purposes.

Like the linear camera, the laser scanner provides 1-dimensional images. But the difference is now the images are in terms of distances with respect to the scanning angles other than in terms of intensity with respect to angles as for the linear camera. The information of intensity of the arrival light is still used for the signal detection purpose. This intensity information is normally not required by users since they have already got the distance and orientation information.

Without a special design, the laser scanner works just as a camera in the sense that it provides

4.2. The sensing system of relative measurements

Figure 4.6: The SICK laser scanner

Table 4.3: Specifications of the SICK LMS221 Type-6 laser scanner

Code	LMS 221-30206, outdoor
Color	grey
Power Supply	+24V
Heater included	yes
Range/Range resolution/Range accuracy	80m/10mm/ $\pm 60mm$
Scanning range/resolution/direction	180°/0.5°/left-to-right
Connection	RS232/RS422, up to 500Kbaud
Response time	13ms
Reflectivity telegrams	supported
Laser protection class	eye-safe
Enclosure rating	IP67
Weight	approx. 9kg

Figure 4.7: Scanning direction of the laser scanner

Figure 4.8: A scan from laser scanner

an image consisting of all the objects in its view (Figure 4.8). Black dots are the readings from the laser scanner at scanning angles from 0° to 180° . The leader vehicle is represented by the dark segment near the center of the view. How to filter out unnecessary objects? The answer is to consider the reflectivity, i.e. how much percentage of the incoming signal is reflected by a material, and set a higher threshold for the laser scanner so that it will only see strongly reflective objects. Usually, all the laser scanners have the intensity threshold fixed. And because most of the normal materials have a low reflectivity of less than 50%, the threshold is, by default, set as low as 5% so that the laser scanner can detect the object of the normal materials. Table 4.4 [121] lists the reflectivity of some materials

Following the idea of marking the leader vehicle as an strongly reflective object, a type of reflective tape developed by the 3M company. The convenience of using this tape is that we can just paste the tape wherever we wish to have a reflective surface. This special tape can reflect strongly of the light beaming on it back to the coming direction of the signal. From the laser scanner's view, the reflective tape looks strongly bright compared to other materials. From the user's view, however, in general, without the intensity information the tape looks normal like other objects. The tape may be even undistinguishable from the leader vehicle shape because it is very thin. Thus, if the distance information were solely used, it would be

Table 4.4: Reflectivity of some materials

Material	Reflectivity
Cardboard, matt black	10%
Cardboard, gray	20%
Wood (raw pine,dirty)	40%
PVC, gray	50%
Paper, matt white 80g/m ²	80%
Aluminium, anodised, black	110...150%
Steel, rust-free shiny	120...150%
Steel, very shiny	140...200%
Reflectors	> 2000%

impossible to detect the tapes. Fortunately, among different types of laser scanners developed by SICK company, one type, namely SICK laser scanner Type 6, can provide that additional information of the intensity (they use word ‘reflectivity’ for this). The distance information for each scanning angle is now integrated with 1, 2 or 3 bits of reflectivity information (making up up to 8 levels). The reflectivity information tells which of the 8 levels of intensity the received signal belongs to. All the reflections from normal materials result in level 0 (the lowest level) while those from the reflective tape normally fall in level 2 or level 3 depending on the distance to the laser scanner. Thus, the filtering job is now easily done using a simple software filter. In Figure 4.8, the little white-filled circles represent the center a group of points that have high level of reflectivity. The performance was amazing because we still maintain the resolution of scanning and yet able to filter out virtually all the background objects. The image now contains only few points representing the reflective tape.

4.2.2.1 Measurement of the relative distance and orientation (d, ϕ)

Because the laser scanner has been well calibrated, one reflective tape pasted on the leader vehicle is in fact sufficient to detect and measure the relative distance and orientation. However, if there is strong reflections from other materials, it is impossible to pinpoint which reflection is from the actual target. Moreover, with one point, we are unable to compute the difference in heading angle of the two vehicles, θ_e . At least two points (or two reflective tapes) must be pasted on the leader vehicle. The known distance between two tapes can be used as a feature to

4.2. The sensing system of relative measurements

118

distinguish them from other light sources. It is very likely that the scanner may miss one of the tapes due to some reasons such as signals being blocked or the tape being inside the blind range of the scanner (for example, the tape lies completely between two consecutive scanning beams from the scanner). In these cases, the scanner may provide inaccurate measurements. As shown in Lemma 2.3.2, when the tracking convergence is achieved, the distance between two vehicles is expectedly convergent to a predetermined distance l . If the range measurements are wrong, the actual distance between two vehicles will be different from what is expected. That is why a template composed of three pieces of strongly reflective tape is used to minimize the measurement error while the two vehicles are running as described below.

Three tapes are pasted on the leader vehicle: one at the left edge, one at the right edge and the third at somewhere in between of the first two tapes. The third tape is pasted such that its distances to the other tapes are different. This manner will help much the recovery of the missing point as shown later. Let us call l_1, l_2 and l_3 for the distances between the left and right tapes, the right and third tapes, and the left and the third tapes, respectively, $l_1 > \max(l_2, l_3)$. These three distances can be easily computed from the readings of the laser scanner.

$$\begin{aligned} l_1 &= \sqrt{d_{left}^2 + d_{right}^2} - 2d_{left}d_{right}\cos(\phi_{left} - \phi_{right}) \\ l_2 &= \sqrt{d_{right}^2 + d_{3rd}^2} - 2d_{right}d_{3rd}\cos(\phi_{right} - \phi_{3rd}) \\ l_3 &= \sqrt{d_{left}^2 + d_{3rd}^2} - 2d_{left}d_{3rd}\cos(\phi_{left} - \phi_{3rd}) \end{aligned} \quad (4.2)$$

where (d_{left}, ϕ_{left}) , $(d_{right}, \phi_{right})$ and (d_{3rd}, ϕ_{3rd}) are the readings of the left tape, right tape and third tape, respectively. When l_1, l_2 and l_3 are available, we can calculate the following terms of triangle ABC

- The angle δ_{left} at vertex A

$$\delta_{left} = \arccos\left(\frac{l_1^2 + l_3^2 - l_2^2}{2l_1l_3}\right) \quad (4.3)$$

- The angle δ_{right} at vertex B

$$\delta_{right} = \arccos\left(\frac{l_1^2 + l_2^2 - l_3^2}{2l_1l_2}\right) \quad (4.4)$$

Suppose the leader vehicle is referenced at the midpoint M of the left and right tapes, the relative distance and orientation as well as the heading difference can be easily computed using

4.2. The sensing system of relative measurements

119

Figure 4.9: The detection and measurement of the reflective tapes

the left and right measurements only.

$$\begin{aligned}
 d &= \frac{1}{2} \sqrt{2d_{\text{left}}^2 + 2d_{\text{right}}^2 - l_1^2} \\
 \phi &= \arcsin \left(\frac{d_{\text{left}} \cos \phi_{\text{left}} + d_{\text{right}} \cos \phi_{\text{right}}}{2d} \right) \\
 \theta_e &= \arcsin \left(\frac{d_{\text{right}} \sin \phi_{\text{right}} - d_{\text{left}} \sin \phi_{\text{left}}}{l_1} \right)
 \end{aligned} \tag{4.5}$$

Figure 4.8 shows an example when the reflective tapes are detected and then the relative distance and orientation as well as heading difference are all calculated.

Now, suppose the laser scanner can only 'see' the left tape and the 3rd tape, with the priori knowledge of the 3 lengths l_1 , l_2 and l_3 , the required information is still obtained

$$\begin{aligned}
 \theta_e &= \arcsin \left(\frac{d_{3rd} \sin \phi_{3rd} - d_{\text{left}} \sin \phi_{\text{left}}}{l_3} \right) + \delta_{\text{left}} \\
 d &= \sqrt{d_{\text{left}}^2 + \left(\frac{l_1}{2}\right)^2 + l_1 d_{\text{left}} \cos(\phi_{\text{left}} - \theta_e)} \\
 \phi &= \arcsin \left(\frac{2d_{\text{left}} \cos \phi_{\text{left}} + l_1 \cos \theta_e}{2d} \right)
 \end{aligned} \tag{4.6}$$

If the left tape is unable to be seen, the right and 3rd tapes can provide the measurements

$$\begin{aligned}
 \theta_e &= \arcsin \left(\frac{d_{\text{right}} \sin \phi_{\text{right}} - d_{3rd} \sin \phi_{3rd}}{l_2} \right) - \delta_{\text{right}} \\
 d &= \sqrt{d_{\text{right}}^2 + \left(\frac{l_1}{2}\right)^2 - l_1 d_{\text{right}} \cos(\phi_{\text{right}} - \theta_e)} \\
 \phi &= \arcsin \left(\frac{2d_{\text{right}} \cos \phi_{\text{right}} - l_1 \cos \theta_e}{2d} \right)
 \end{aligned} \tag{4.7}$$

Figure 4.10: The distance measurements in a stationary situation

4.2.2.2 Accuracy and resolution of the measurement

The accuracy test was done with the laser scanner and the tapes stationary. Measurements over a long period of time (about 5 minutes) were collected. Figure 4.10-Figure 4.12 show the accuracy of the measurements. Given that the accuracy of the laser scanner is $\pm 2\text{cm}$ and angular resolution of 0.5° , the method can provide an accuracy of $\pm 5\text{mm}$ for a distance of 2.525m and accuracy of $\pm 0.05^\circ$ for an angle of -25.26° . The accuracy is much improved when at least 2 of the three tapes are detected.

The sampling time of the method was measured over a period of 5 minutes. The laser scanner was set to transmit its scanning measurements continuously and at the fastest transfer rate of 500 KBaud. The computer, an Intel Pentium IV 1.9GHz PC, received these scanning measurements and did all the necessary computations while it also carried out all other tasks such as collecting measurements from other sensors and computing the control inputs to all the motors. The time difference between two consecutive measurements of (d, ϕ) were recorded as in Figure 4.13. Because Microsoft Windows is not a real-time operating system, the sample time may change from time to time. The maximum sample time was 33ms , fast enough for the controller.

Figure 4.11: The angle measurements in a stationary situation

Figure 4.12: The heading-difference measurements in a stationary situation

Figure 4.13: Sampling time of the method

4.2.2.3 Estimation of derivatives

Due to the fast updating capacity, the first and second derivatives of d and ϕ are then estimated using polynomial approximation.

Suppose over a period of time, we have collected N samples $d_1, d_2, \dots, d_N$ at time $t_1, t_2, \dots, t_N$, respectively, with $t_1 < t_2 < \dots < t_N$. We will approximate these sample with a polynomial $d(t)$ whose order is of m

$$d(t) = \sum_{i=0}^m a_i t^i$$

where $a_i, i = 0..m$, are the coefficients that need estimating.

The estimation error function J is simply defined as a sum of all the squares of the error of each sample.

$$J = \frac{1}{2} \sum_{j=1}^N (d(t_j) - d_j)^2 = \sum_{j=1}^N \left(\sum_{i=0}^m a_i t_j^i - d_j \right)^2$$

The best fitting of coefficients $a = [a_0 \ a_1 \ \dots \ a_m]^T$ are the ones that make all the partial differentiates of the estimation error function J with respect to coefficients a be zero.

$$\frac{\partial J}{\partial a_k} = \sum_{j=1}^N \left(\sum_{i=0}^m a_i t_j^i - d_j \right) t_j^k = \sum_{j=1}^N \left(\sum_{i=0}^m a_i t_j^{i+k} - d_j t_j^k \right) = \sum_{i=0}^m a_i \sum_{j=1}^N t_j^{i+k} - \sum_{j=1}^N d_j t_j^k = 0$$

where $k = 0, \dots, m$.

In the matrix form, we have

$$\frac{\partial J}{\partial a} = Ta - D = 0$$

4.2. The sensing system of relative measurements

123

where $a = [a_0 \ a_1 \ \dots \ a_m]^T$ is the coefficient vector;

$$T = \begin{bmatrix} \sum_{j=1}^N t_j^0 & \sum_{j=1}^N t_j^1 & \dots & \sum_{j=1}^N t_j^m \\ \sum_{j=1}^N t_j^1 & \sum_{j=1}^N t_j^2 & \dots & \sum_{j=1}^N t_j^{m+1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j=1}^N t_j^m & \sum_{j=1}^N t_j^{m+1} & \dots & \sum_{j=1}^N t_j^{2m} \end{bmatrix} \quad \text{and} \quad D = \begin{bmatrix} \sum_{j=1}^N d_j \\ \sum_{j=1}^N d_j t_j \\ \vdots \\ \sum_{j=1}^N d_j t_j^m \end{bmatrix}$$

Note that matrix T is a symmetric matrix whose rank is m when $N > m$ and whose rank is $N - 1$ if $N \leq m$. Thus, the number of samples must be greater than the order of the polynomial, $N > m$, in order to achieve a unique polynomial estimation. And because $t_1 < t_2 < \dots < t_N$, symmetric matrix T is regular and we have

$$a = T^{-1}D$$

To speed up the computing process of estimated coefficient a , well known methods such as Gaussian Elimination, Gauss-Jordan Elimination or LU Decomposition can be easily applied. In our work, we applied Gaussian Elimination method to estimate a from $Ta = D$. The equation can be represented by a $(m + 1)$ -by- $(m + 2)$ matrix as follows

$$H = \begin{bmatrix} T & D \end{bmatrix}$$

This method has two stages:

1. *Stage I*: To transform matrix H into a upper triangular form by eliminating all elements below the principle diagonal using two elementary operations
 - (a) Multiplying all elements of one row with a nonzero constant
 - (b) Replacing one row with its linear combination with another row.

Matrix H becomes $\check{H}$

$$\check{H} = \begin{bmatrix} \check{h}_{00} & \check{h}_{01} & \dots & \check{h}_{0m} & \check{h}_{0(m+1)} \\ 0 & \check{h}_{11} & \dots & \check{h}_{1m} & \check{h}_{1(m+1)} \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & \check{h}_{mm} & \check{h}_{m(m+1)} \end{bmatrix}$$

$m + 1$ steps are conducted. Step i , with i starting from 0 to m , begins with making sure element h_{ii} is 1, if possible. With notice that at step i , all the elements h_{jk} , with

4.2. The sensing system of relative measurements

124

$i \leq j \leq m$ and $0 \leq k < i$, are zero, we will look for a row j , with $i \leq j \leq m$, such that h_{ji} is nonzero.

- If there is one such row j , this row will be swapped with row i . This operation will not change the solution a as well as the rank of matrix H . Then, all the elements of this row (now becoming row i) are divided by h_{ii} resulting in $h_{ij} = 0$, with $0 \leq j < i$, and $h_{ii} = 1$. Subsequently, all the rows j , with j from $i + 1$ to m , will be replaced by its summation with $(-h_{ji})$ times row i resulting in $h_{ji} = 0$.
- If such a row j with $h_{ji} \neq 0$ could not be found, it means all the elements h_{ji} are zero and we continue to next step.

After these $m + 1$ steps, matrix H is transformed into upper triangular matrix $\check{H}$ with $\text{rank } \check{H} = \text{rank } H$.

Note that

- Since times t_i , with $i = 1..N$, are strictly increasing, if matrix H has full rank, so does matrix $\check{H}$, and $\check{h}_{00} = \check{h}_{11} = \dots = \check{h}_{mm} = 1$. If matrix H has a rank of $m' \leq m$, it because only m' samples are available for polynomial estimation. Therefore, matrix $\check{H}$ also has a rank of m' , and $\check{h}_{00} = \dots = \check{h}_{(m'-1)(m'-1)} = 1$ and $\check{h}_{m'm'} = \dots = \check{h}_{mm} = 0$, resulting in an $(m' - 1)$ -order polynomial.
- Row i of matrix $\check{H}$ refers to the following equation

$$\check{h}_{ii}a_i + \check{h}_{i(i+1)}a_{i+1} + \dots + \check{h}_{im}a_m = \check{h}_{i(m+1)}$$

or

$$\check{h}_{ii}a_i = \check{h}_{i(m+1)} - (\check{h}_{i(i+1)}a_{i+1} + \dots + \check{h}_{im}a_m)$$

2. Stage 2: To calculate a_i using back substitution by replacing values of $a_{i+1}, \dots, a_m$ into the i^{th} row with i starting from m down to 0.

$$\begin{aligned} \check{h}_{mm}a_m &= \check{h}_{m(m+1)} \\ \check{h}_{(m-1)(m-1)}a_{m-1} &= \check{h}_{(m-1)(m+1)} - \check{h}_{(m-1)m}a_m \\ &\vdots \\ \check{h}_{00}a_0 &= \check{h}_{0(m+1)} - (\check{h}_{01}a_1 + \dots + \check{h}_{0m}a_m) \end{aligned}$$

4.2. The sensing system of relative measurements

125

Obviously, if $\check{H}$ has full rank, then $\check{h}_{00} = \check{h}_{11} = \dots = \check{h}_{mm} = 1$. Then, we have

$$\begin{aligned} a_m &= \check{h}_{m(m+1)} \\ a_{m-1} &= \check{h}_{(m-1)(m+1)} - \check{h}_{(m-1)m} a_m \\ &\vdots \\ a_0 &= \check{h}_{0(m+1)} - (\check{h}_{01} a_1 + \dots + \check{h}_{0m} a_m) \end{aligned}$$

If $\check{H}$ has a rank of $m' \leq m$, then $\check{h}_{00} = \dots = \check{h}_{(m'-1)(m'-1)} = 1$ and $\check{h}_{m'm'} = \dots = \check{h}_{mm} = 0$. Then, we set

$$\begin{aligned} a_m &= 0 \\ &\vdots \\ a_{m'} &= 0 \\ a_{m'-1} &= \check{h}_{(m'-1)(m+1)} \\ &\vdots \\ a_0 &= \check{h}_{0(m+1)} - (\check{h}_{01} a_1 + \dots + \check{h}_{0(m'-1)} a_{m'-1}) \end{aligned}$$

After the coefficients a have been computed, the first and second derivatives can be estimated as follows

$$\begin{aligned} d'(t) &= \sum_{i=1}^m i a_i t^{i-1} \\ d''(t) &= \sum_{i=2}^m i(i-1) a_i t^{i-2} \end{aligned}$$

Because the measurements come in discrete time and at a time only N recent samples are needed to estimate the derivatives, when a new sample comes, the oldest sample among N current samples can be thrown away and the latest one can be inserted. Matrix T and D are then re-calculated and finally, so is the new set of coefficient a .

Initially, when there are not enough $m+1$ samples, the order of the estimated polynomial is less than m . Only when there are $N > m$ samples can the estimation be of the order of m . By following the two stages described above, it is shown that when the very first sample (t_1, d_1) comes, the estimated polynomial is a constant function $d(t) = d_1$, hence the derivatives are zero. The second sample come and we will have a line. This continues increasing the order of the estimated polynomial until m . Before the first sample comes, the derivatives are undetermined, thus, to void unexpected retrievals for the controller, a desired sample $(t_1 = 0, d_1 = d_{desired})$ is initiated. The initial estimated derivatives are, therefore, zero.

4.2. The sensing system of relative measurements

126

4.2.2.4 Summary

The procedure of implementing the laser scanner for vehicle following purpose is summarized as follows.

1. Step 1: Setup: Three reflectors are pasted vertically on the body of the leader vehicle: one at the left edge, one at the right edge and the third at somewhere in between that makes its distance to the right one half its distance to the left one. Then the laser scanner is mounted on the follower vehicle with its base parallel with the ground and its height the same as that of the reflectors.
2. Step 2: Template Store: The template, i.e. three reflectors, can be captured manually or automatically. Manual storing requires user to measure the three distances between any two of the three reflectors. The three distances are stored for template detection. The three distances can also be measured automatically by the laser scanner itself. All we need to do is to locate the template in front of the laser scanner and then click on the "Save Template" button. One notice is to make sure that the laser scanner can see only and all the three reflectors and the points representing them would rather be stable.
3. Step 3: Template Detection and Measurement: After the template has been stored, it will be detected whenever it appears in the view range of the laser scanner. (d, ϕ) as well as their derivatives are then quickly calculated

Advantages of the proposed measurement method:

1. *Fast sampling rate*: The sampling interval of $30ms$ is sufficiently fast for the vehicle-following controller to be implemented in real-time. This rate is also acceptably good for derivative estimates as well as filtering.
2. *High level of measurement accuracy*: The accuracy is dependent on many factors such as the flatness of the ground, the width and verticalness of the reflectors and the laser scanner and the distance between the laser scanner and the reflectors. By using 3 long vertical reflectors, the accuracy of the detection has been improved significantly to 1-centimeter level.
3. *Robust against noise*: Noise is unwanted addition to a signal that changes the magnitude of the signal in a unknown manner. Due to noise, the laser scanner sometimes recognizes

unpredictably a reflection from a normal material as a reflective point creating a fake reflector. Similarly, it sometimes can not see a reflector even though the reflector is in its view range. With the triangle template, all these disturbance can be reduced marvelously.

4. *Weather condition independence:* The laser scanner can work well day and night, and on cloudy or sunny days. Unlike image processing methods where lighting and shadow are two serious factors, the method is working perfectly well no matter where the leader vehicle and the reflectors are located. The laser scanner is only affected when there is fog, rain or snow in what the laser beam can be diffused.
5. *Extendibility to Multi-vehicle identification:* Since every vehicle is recognized based on the template being put on it, not the shape, color or whatsoever features of the vehicle, the template on different vehicles could be designed in such a way that they are different from one another. When a laser scanner sees several reflectors, it is able to distinguish one template from the others, and calculate exactly the corresponding vehicle's relative position.

And some disadvantages of the proposed measurement method:

1. *Reflective tape requirement:* the tapes pasted on the leader vehicle must be strongly reflective
2. *Ground flatness requirement:* the ground for testing must be flat. Even after being well aligned, an unbalance among the 4 wheels of the leading or follower vehicles may cause the laser scanner and the reflectors not to be horizontally aligned resulting in changes in measured lengths among reflectors. To deal with this, the accuracy of the measurements might be sacrificed.

4.3 Experimental results

4.3.1 Setting up

The vehicle-following controller developed in Chapter 2 is implemented on the Blue Cycab vehicle with the Red Cycab vehicle running as a leader vehicle. The place where the two vehicles operated was the car park in front of the lab. The Red Cycab vehicle was manually driven a couple of rounds around the car park. The Blue Cycab vehicle utilizing the sensing

4.3. Experimental results

128

system was to follow at a desired spacing. This desired spacing could be chosen at will or could be taken automatically from initial relative distance between two vehicles provided by the sensing system. Due to the structure and operation of the vehicle, only the kinematic-based controller is implementable.

The tests for the look-ahead tracking situation were conducted. For the look-behind tracking situation, the laser scanner needs to be mounted at the back of the vehicle. Currently, we are making a metal frame to firmly hold it like how it is for the look-ahead tracking. When it is ready, the experiments for look-behind can be implemented. Nevertheless, from the simulation results in Chapter 3, if the vehicle-following controller works properly for the look-ahead situation, it should work for the look-behind tracking situation, too.

Initially, the two vehicles were at rest (at position (0,0)). Their relative orientation was generally nonzero. It is because it was not easily to align the two vehicles as in the simulations. However, this misalignment would not pose any problem to the controller as the steering angle should be controlled accordingly. The initial position of the two vehicles were purposely set to be the same for all the tests to facilitate the comparison. However, it was almost impossible to achieve it precisely. Moreover, because the leader vehicle was driven manually, it was also impossible for a human driver to repeat exactly the maneuvers at exactly the same intervals of time. Thus, the tests were conducted independently just to illustrate the effectiveness of the controller.

Parameter λ was always fixed at 1. The leader vehicle was remotely controlled via the wireless communication. Its trajectory included long straight runs and several left and right turns. The velocity and steering angle of the leader vehicles were recorded for comparison purpose. The position of the leader vehicle was also numerically computed based on the heading angle and the recorded velocity. The position calculations might be not very accurate but sufficiently good to illustrate how the following performance is.

On the follower vehicle, where the vehicle-following controller is executed, another recording system was also developed to record with respect to time the velocity, the steering angle and the estimated positions of the vehicle, as well as the relative distance and orientation provided by the sensing system. These records and those from the leader vehicle are compared to provide the full picture of the performance. Furthermore, video of the vehicle-following maneuvers were also recorded to display the result.

4.3.2 Experimental results and discussions

The first run was conducted with $p = 1.5$ and $l \approx 3m$. Two vehicles traveled for about 10 minutes over the distance of about 300m. The tracking results are displayed with the trajectories in Figure 4.14, heading angles in Figure 4.15, velocities in Figure 4.16, steering angles in Figure 4.17, vehicular spacing in Figure 4.18 and relative angle in Figure 4.19.

The second run was conducted at a speed higher than that for the first run. $p = 2$ was chosen. l was also automatically set with the initial vehicular spacing and approximately $l = 2.2m$. The run was conducted for about 4 minutes and the length of the path the two vehicles traveled was about 240m. The resultant trajectories, heading angles, velocities, steering angles, vehicular spacing and relative angle ϕ are given in Figures 4.20-4.25, respectively.

The results show that the following performance was consistent and relatively good. The follower vehicle, Blue Cycab, was able to follow and maintain a distance of 3m away from the leader vehicle, Red Cycab. Though the tracking was successful, the spacing between two vehicles was sometime larger or smaller than 3m, it because the two following reasons:

- *Roughness of the ground:* The car park ground is not flat. Moreover, there are few humps in it which are used to limit the speed of the cars when moving in the car park. Whenever the Red or Blue vehicles run over an uneven ground, the sensing system generated an increase or decrease in the measured spacing (Figures 4.18 and 4.24) as the alignment of the laser scanner and the reflective tapes was violated, though the actual spacing may not have changed that much. In an industrial environment such as warehouses or seaports, the flatness of the ground is a lot improved.
- *Roughness of manual control:* The leader vehicle, Red Cycab, was driven manually by using a remote PC. It is therefore difficult to maintain constant speed and smooth steering angle. Nevertheless, the results show the follower vehicle was still able to catch up and follow when the conditions changed suddenly.

Above all, the experimental results again illustrate the efficiency and robustness of the proposed vehicle-following controller. All the noise and roughnesses discussed just now can be considered as disturbances to the controller and will be eliminated by the closed-loop system provided the measurements from the sensing system and from the encoders are available. It results in the much smoother trajectory of the follower vehicle though that of the leader vehicle is rough.

4.3. Experimental results

Figure 4.14: Trajectories for $p = 1.5$

Figure 4.15: Heading angles for $p = 1.5$

Figure 4.16: Velocities for $p = 1.5$

Figure 4.17: Steering angles for $p = 1.5$

Figure 4.18: Vehicular spacing for $p = 1.5$

Figure 4.19: Relative angle ϕ for $p = 1.5$

4.3. Experimental results

Figure 4.20: Trajectories for $p = 2$

Figure 4.21: Heading angles for $p = 2$

Figure 4.22: Velocities for $p = 2$

Figure 4.23: Steering angles for $p = 2$

Figure 4.24: Vehicular spacing for $p = 2$

Figure 4.25: Relative angle ϕ for $p = 2$

The platoon vehicle is somewhat maintained, especially when the vehicle moved on a straight line. Even on a circular arc, the velocity of the platoon can be guaranteed if the measurements are good.

In summary, the vehicle-following controller worked properly on the experimental setup. The following maneuver for both straight and curved trajectory was achieved. This is because of the good combination of the proposed controller with the idea of imitating human practices, and the fast and reliable implementation on the real vehicles, especially the newly developed sensing system.

4.4 Conclusions

The vehicle-following controller developed in Chapter 2 is successfully implemented on a platoon of two vehicles named Cycab. In order to do so, besides the onboard sensors which provide the velocity and steering angle information of the vehicle, a sensing system is designed and realized that uses a laser scanner and reflective tapes to set up an active sensing system to measure the relative distance and orientation of two vehicles. The relative information is fetched to the controller as required feedbacks. The experimental results again prove the effectiveness of the proposed controller and show the follower vehicle is actually able to follow the leader vehicle. Though the performance is not as good as that from simulation, it indicates that the idea of imitating the human driving practices, that have been examined everyday, is right and effective.

Part II

Fault-tolerant control system for a four-wheel-steering vehicle

Chapter 5

Fault-Tolerant Control Design for a Four-Wheel-Steering Vehicle

5.1 Introduction

In Part I of the thesis, a vehicle-following controller that utilizes the human driving practice has been developed. The controller has been proved to work properly for both look-ahead and look-behind tracking situations. Simulations and experiments have also been carried out to verify the performance and efficiency of the proposed controller. However, like all other controllers developed for car-like vehicles, this controller will not work properly when there is a fault in the driving or steering system of the vehicle. It is obvious because in general for such car-like vehicles they have one driving system, e.g. the engine, and one steering system. The failure of this driving system will cause the vehicle to stall and the failure of steering system results in the disorientation of the vehicle. In most cases, the vehicle will be solely able to move forth and back along a fixed straight or circular path. To tackle these faulty situations and help the vehicle continue its operation, or at least roll back to the garage for repair, the only solution has so far been to install additional actuators to create a redundancy level of the system such that when one of them fail to work, the remaining ones are able to carry on the assigned operation. For the driving part, this can be done by install a motor or engine for each pair or each single wheel, like what our Cycab vehicles are designed. Likewise, for the steering part, an additional steering system could be installed to steer the rear pair of wheels to obtain a configuration of four-wheel-steering (4WS). Although this strategy increases the cost of the vehicle, the benefits gained include the flexibility and the reliability of the vehicle against faults.

In this chapter, a fault-tolerant controller that drives a four-wheel-steering vehicle to follow another vehicle is developed. The controller takes into account the diagnostic information of the driving and steering systems to issue the commands. The diagnostic signals are generated by the low-level control system of the vehicle and are basically logic signals indicating the working condition of a component or a system.

This chapter will begin with the analysis of the operating principles of the driving and steering systems, especially when they are at fault. A suitable model is then proposed for this type of vehicles. The controller utilizing the proposed model and parallel to the tracking method presented in Part I is derived. Subsequently, the modifications for the co-simulation platform to simulate the faults and the operation of the vehicle when there are faults at driving and steering systems are discussed. The simulation results and discussions will end this chapter. The practical development of low-level control system that has fault-tolerance capacities for the four-wheel-steering vehicle, Red Cycab, as well as experimental results will be presented and discussed in the next chapter.

5.2 Vehicle operation and vehicle model

5.2.1 Operation principles of Cycab's driving and steering systems

The Cycab vehicle, described previously in Chapter 4, utilizes DC brushed motors for its driving system and hydraulic actuators for its steering system.

For the driving system, the voltage from the motion controller board is used as reference voltage by the motor controller to generate a PWM current that is fetched to the DC motor. The DC motor is normally composed of two parts: the stator where the permanent magnets are mounted and the coil (or the armature) that physically links to the motor's axis. The current, when crossing the motor's armature that is located in a magnetic field of the stator, will create a magnetic torque acting the motor's axis and make it rotate. The torque is proportional to the current by a motor-specific gain. Thus, when there is a current crossing the armature, the motor axis is forced to rotate by the magnetic torque. If there is no current, the axis and the stator are in fact rotating independently, with some friction on their contact surface. In other words, when there is no current from the amplifier, the wheel that is tightened to the motor's axis is like a caster. Hence, if a fault is detected to occur in one driving control such as encoder failure, broken wire or motor failure, it is reasonable to isolate the entire loop by shutting down the

amplifier. The wheel is then unable to rotate by the attached motor but by the friction with the ground while the vehicle is driven by other active motors. If the remaining working motor or motors are powerful enough to take the full load of the vehicle and generate the desired velocity, the vehicle is still able to move forth and back.

The steering mechanism is somewhat more complicated. It also has an electrical part that converts a reference voltage to a torque by utilizing an amplifier and a DC motor. To create a even more powerful force, a hydraulic pump and a hydraulic cylinder are used. The pump is linked to the motor's axis through a belt and it pumps oil to one of the two sides of the cylinder dependent on the rotating direction of the motor. The difference of oil pressure across the two sides of the cylinder's piston creates a hydraulic force that move the piston up or down. As a result, the wheels that connects to the piston with hard steel bars and joints are also oriented accordingly. Similar to the driving system, if there is no current crossing the DC motor due to failures in the electrical part, the motor's axis will not rotate. The pump does not push any oil into the cylinder. Instead, its internal valves will block the oil from returning to its reservoir. The oil is therefore trapped inside the cylinder and prevents the piston from moving. Finally, the wheels linked to the piston are firmly held at its current position. Likewise, if the pump is not working properly due to its internal failure or due to the belt being worn out, the wheels are also not steerable and this fault can be easily detected with the readings from the absolute encoder. Very unlikely, if the fault cause is from the metal-covered cylinder or mechanical part, the vehicle is actually experiencing a severe damage, the wheels might swing unpredictably when the vehicle moves. In these cases, none of the controllers can work properly and the vehicle must be stopped until the problem is sorted out.

5.2.2 Vehicle model

In this section, we will derive a suitable model for our four-wheel-driving four-wheel-steering vehicle, Red Cycab. In addition to the structure like that presented for Blue Cycab, two rear wheels of Red Cycab are also linked to a steering system. The rear steering wheels offer the vehicle more flexible maneuvers such as tight turns, smaller value of r_{min} , and parallel translations where the vehicle can change lanes without changing its heading angle. The same methodology of model derivation is used as that for the car-like vehicle.

The vehicle is illustrated as in Figure 5.1. The reference point P for the vehicle is chosen on the symmetric longitudinal axle with the distances from P to the front and rear axle centers,

Figure 5.1: The vehicle model

Table 5.1: Parameters of wheel configuration for 4WS vehicle

Wheel	L	α	γ
Front left	$\sqrt{a_f^2 + b^2}$	$\alpha_{fl} = \arctan \frac{b}{a_f}$	γ_{fl}
Front right	$\sqrt{a_f^2 + b^2}$	$\alpha_{fr} = -\arctan \frac{b}{a_f}$	γ_{fr}
Rear left	$\sqrt{a_r^2 + b^2}$	$\alpha_{rl} = \pi - \arctan \frac{b}{a_r}$	γ_{rl}
Rear right	$\sqrt{a_r^2 + b^2}$	$\alpha_{rr} = \arctan \frac{b}{a_r} - \pi$	γ_{rr}

a_f and a_r , respectively, $a_f, a_r \geq 0$ and $a_f + a_r = 2a > 0$. And $P = \zeta = (x, y)$ is the position of the vehicle in the global coordinate system. With the heading angle θ , the vehicle can be represented in the global coordinate system by the triplet $\psi = [x \ y \ \theta]^T$.

The assumptions of **pure rolling** and **no skidding** are made with the wheel parameters listed in Table 5.1. Note that the rear wheels are now steerable as well.

By applying these parameters to (2.1) and (2.2) and putting these resultant equation into matrix forms, we obtain

$$J_1(\gamma_{fl}, \gamma_{fr}, \gamma_{rl}, \gamma_{rr})R_3(\theta)\dot{\psi} - J_2\dot{\phi} = 0 \quad (5.1)$$

$$C_1(\gamma_{fl}, \gamma_{fr}, \gamma_{rl}, \gamma_{rr})R_3(\theta)\dot{\psi} = 0 \quad (5.2)$$

$$J_1 = \begin{bmatrix} \cos \gamma_{fl} & \sin \gamma_{fl} & \sqrt{a_f^2 + b^2} \sin(\gamma_{fl} - \alpha_{fl}) \\ \cos \gamma_{fr} & \sin \gamma_{fr} & \sqrt{a_f^2 + b^2} \sin(\gamma_{fr} - \alpha_{fr}) \\ \cos \gamma_{rl} & \sin \gamma_{rl} & \sqrt{a_r^2 + b^2} \sin(\gamma_{rl} - \alpha_{rl}) \\ \cos \gamma_{rr} & \sin \gamma_{rr} & \sqrt{a_r^2 + b^2} \sin(\gamma_{rr} - \alpha_{rr}) \end{bmatrix} \quad (5.3)$$

$$J_2 = r_\varphi I_4 \quad (5.4)$$

5.2. Vehicle operation and vehicle model

138

$$\begin{aligned}
C_1 &= \begin{bmatrix} -\sin \gamma_{fl} & \cos \gamma_{fl} & \sqrt{a_f^2 + b^2} \cos(\gamma_{fl} - \alpha_{fl}) \\ -\sin \gamma_{fr} & \cos \gamma_{fr} & \sqrt{a_f^2 + b^2} \cos(\gamma_{fr} - \alpha_{fr}) \\ -\sin \gamma_{rl} & \cos \gamma_{rl} & \sqrt{a_r^2 + b^2} \cos(\gamma_{rl} - \alpha_{rl}) \\ -\sin \gamma_{rr} & \cos \gamma_{rr} & \sqrt{a_r^2 + b^2} \cos(\gamma_{rr} - \alpha_{rr}) \end{bmatrix} \\
&= \begin{bmatrix} -\sin \gamma_{fl} & \cos \gamma_{fl} & a_f \cos \gamma_{fl} + b \sin \gamma_{fl} \\ -\sin \gamma_{fr} & \cos \gamma_{fr} & a_f \cos \gamma_{fr} - b \sin \gamma_{fr} \\ -\sin \gamma_{rl} & \cos \gamma_{rl} & -a_r \cos \gamma_{rl} + b \sin \gamma_{rl} \\ -\sin \gamma_{rr} & \cos \gamma_{rr} & -a_r \cos \gamma_{rr} - b \sin \gamma_{rr} \end{bmatrix} \quad (5.5)
\end{aligned}$$

Again, we assume that all the wheels of the vehicle are Ackermann steering wheels, which means their rotating axes are coincident at the *ICR*.

Property 5.2.1 *A 4WS vehicle with restricted steering angles, i.e. $|\gamma_{fl}, \gamma_{fr}, \gamma_{rl}, \gamma_{rr}| \leq \gamma_{\max} \ll \frac{\pi}{2}$, always has matrix C_1 in (5.5) of rank 2 when all four wheels are Ackermann steering wheels.*

Proof: Since $\text{rank}(C_1) < 3$ when the steering wheels are Ackermann steering wheels, we will prove that $\text{rank}(C_1)$ is also greater than 1.

Obviously, since all the steering angles are restricted and make column 2 of matrix C_1 always nonzero, $\text{rank}(C_1) \geq 1$,

Suppose that $\text{rank}(C_1) = 1$, which implies that there is only one independent nonzero column in matrix C_1 , i.e. column 2. The other two columns must be linearly dependent on this column.

$$\begin{aligned}
\begin{bmatrix} -\sin \gamma_{fl} & -\sin \gamma_{fr} & -\sin \gamma_{rl} & -\sin \gamma_{rr} \end{bmatrix}^T &= k_1 \begin{bmatrix} \cos \gamma_{fl} & \cos \gamma_{fr} & \cos \gamma_{rl} & \cos \gamma_{rr} \end{bmatrix}^T \\
\begin{bmatrix} a_f \cos \gamma_{fl} + b \sin \gamma_{fl} \\ a_f \cos \gamma_{fr} - b \sin \gamma_{fr} \\ -a_r \cos \gamma_{rl} + b \sin \gamma_{rl} \\ -a_r \cos \gamma_{rr} - b \sin \gamma_{rr} \end{bmatrix} &= \begin{bmatrix} (a_f - bk_1) \cos \gamma_{fl} \\ (a_f + bk_1) \cos \gamma_{fr} \\ (-a_r - bk_1) \cos \gamma_{rl} \\ (-a_r + bk_1) \cos \gamma_{rr} \end{bmatrix} = k_2 \begin{bmatrix} \cos \gamma_{fl} \\ \cos \gamma_{fr} \\ \cos \gamma_{rl} \\ \cos \gamma_{rr} \end{bmatrix}
\end{aligned}$$

where k_1, k_2 are two scalars. Then, we have

$$\begin{cases} a_f - bk_1 = k_2 \\ a_f + bk_1 = k_2 \\ -a_r - bk_1 = k_2 \\ -a_r + bk_1 = k_2 \end{cases}$$

5.2. Vehicle operation and vehicle model

139

There are obviously no values of k_1 and k_2 that satisfy all these four equations given that $a_f + a_r = 2a > 0$ and $2b > 0$. It implies matrix C_1 in (5.5) always has rank of $\text{rank}(C_1) > 1$.

Thus, with $\text{rank}(C_1) < 3$, it is concluded that $\text{rank}(C_1) = 2$. ■

5.2.2.1 Steering coordination

The fact that $\text{rank}(C_1) = 2$ also implies that there are only two independent rows in matrix C_1 . In other words, only two independent steering angles are independent and the four wheel steering angles are then related to these two angles. Similar to car-like vehicles, two virtual wheels steering about the front and rear midpoints can be set and their angles are chosen as the two corresponding steering angles, called the front and rear steering angles: γ_f and γ_r , respectively. Another reason to choose the corresponding steering angles in that way is because the absolute encoders used to measure the steering angle are also installed at the front and rear midpoints. Thus, we will have the measurements directly.

In order to compute the steering angle of each wheel with respect to the front and rear steering angles, we can form matrix C'_1 by adding to matrix C_1 two more rows that describe to the two virtual wheels.

$$C'_1 = \begin{bmatrix} -\sin \gamma_f & \cos \gamma_f & a_f \cos \gamma_f \\ -\sin \gamma_r & \cos \gamma_r & -a_r \cos \gamma_r \\ -\sin \gamma_{fl} & \cos \gamma_{fl} & a_f \cos \gamma_{fl} + b \sin \gamma_{fl} \\ -\sin \gamma_{fr} & \cos \gamma_{fr} & a_f \cos \gamma_{fr} - b \sin \gamma_{fr} \\ -\sin \gamma_{rl} & \cos \gamma_{rl} & -a_r \cos \gamma_{rl} + b \sin \gamma_{rl} \\ -\sin \gamma_{rr} & \cos \gamma_{rr} & -a_r \cos \gamma_{rr} - b \sin \gamma_{rr} \end{bmatrix} \quad (5.6)$$

With the restriction that matrix C'_1 must retain the property of rank 2 of matrix C_1 , the determinant of every 3-by-3 sub-matrix of matrix C'_1 must be zero. Thus, we have

$$\begin{aligned} 0 &= \begin{vmatrix} -\sin \gamma_f & \cos \gamma_f & a_f \cos \gamma_f \\ -\sin \gamma_r & \cos \gamma_r & -a_r \cos \gamma_r \\ -\sin \gamma_{fl} & \cos \gamma_{fl} & a_f \cos \gamma_{fl} + b \sin \gamma_{fl} \end{vmatrix} \\ &= \sin \gamma_{fl} [2a \cos \gamma_f \cos \gamma_r - b \sin(\gamma_f - \gamma_r)] - \cos \gamma_{fl} (2a \sin \gamma_f \cos \gamma_r) \end{aligned}$$

or we finally have

$$\tan \gamma_{fl} = \frac{\tan \gamma_f}{1 - \frac{b}{2a}(\tan \gamma_f - \tan \gamma_r)}$$

5.2. Vehicle operation and vehicle model

140

Similarly, we can compute the steering angles of other wheels in terms of the two corresponding angles

$$\begin{aligned}\tan \gamma_{fl} &= \frac{\tan \gamma_f}{1 - \frac{b}{2a}(\tan \gamma_f - \tan \gamma_r)} \\ \tan \gamma_{fr} &= \frac{\tan \gamma_f}{1 + \frac{b}{2a}(\tan \gamma_f - \tan \gamma_r)}\end{aligned}\quad (5.7)$$

$$\begin{aligned}\tan \gamma_{rl} &= \frac{\tan \gamma_r}{1 - \frac{b}{2a}(\tan \gamma_f - \tan \gamma_r)} \\ \tan \gamma_{rr} &= \frac{\tan \gamma_r}{1 + \frac{b}{2a}(\tan \gamma_f - \tan \gamma_r)}\end{aligned}\quad (5.8)$$

Again, the mechanisms that can steer the individual wheels based on the corresponding angles is called *Ackermann steering* wheels. The steering systems of Cycab with the restricted steering angles can be assumed to approximate sufficiently the Ackermann steering mechanism.

5.2.2.2 Posture kinematic model

The motion of the vehicle as a rotation around an *ICR*, where the velocity of every point on the vehicle is orthogonal to the line connecting the point to the *ICR*. Similar to that of car-like vehicles, the velocity at the reference point is also chosen as $v(t)$. Since $\text{rank}(C_1) = 2$, the velocity $R_3(\theta)\dot{\psi}$ is dependent on one time-varying velocity $v(t)$. The side-slip angle β from the vehicle's longitudinal axis is as follows

$$\tan \beta = \frac{a_r \tan \gamma_f + a_f \tan \gamma_r}{a_f + a_r} \quad (5.9)$$

Then, we have the relation

$$R_3(\theta)\dot{\psi} = \begin{bmatrix} Q(\gamma_f, \gamma_r) \\ \kappa(\gamma_f, \gamma_r) \end{bmatrix} v(t) \quad (5.10)$$

where

$$Q(\gamma_f, \gamma_r) = \begin{bmatrix} \cos \beta & \sin \beta \end{bmatrix}^T \quad (5.11)$$

$$\kappa(\gamma_f, \gamma_r) = \frac{\cos \beta (\tan \gamma_f - \tan \gamma_r)}{2a} \quad (5.12)$$

(5.10) leads to

$$\dot{\psi} = \begin{bmatrix} R^T(\theta)Q(\gamma_f, \gamma_r) \\ \kappa(\gamma_f, \gamma_r) \end{bmatrix} v(t) \quad (5.13)$$

5.2. Vehicle operation and vehicle model

141

Regarding the two steering angles, when both steering systems are working properly, we have

$$\begin{aligned}\dot{\gamma}_f &= \omega_f \\ \dot{\gamma}_r &= \omega_r\end{aligned}$$

where ω_f and ω_r are the front and rear steering rates, respectively. Unfortunately, when one of the steering systems is down, the steering angle is held at its current position, and thus the corresponding steering rate becomes zero. To model this faulty situation, we define two boolean variables k_{sf}, k_{sr} , which take a value of either 0 or 1 only, to indicate the working condition of the front and rear steering systems, respectively

$$k_{sf}, k_{sr} = \begin{cases} 0 & : \text{the steering system malfunctions} \\ 1 & : \text{the steering system is functioning} \end{cases} \quad (5.14)$$

If both steering systems are at fault, γ_f and γ_r are constants. Hence, κ is also a constant. Or the turning radius $r = \frac{1}{\kappa}$ is a constant. The only possible motion for the vehicle is to rotate around a fixed *ICR* (when the fixed *ICR* is at infinity, the vehicle travels on a straight path). This fault is too severe for the vehicle to carry on the operation. Thus, we assume at least one steering system is operational

$$\delta_s = k_{sf} + k_{sr} \geq 1 \quad (5.15)$$

where δ_s is called the *degree of steerability* of the vehicle. Clearly, δ_s can be 1 or 2, depending on values of k_{sf} and k_{sr} , and there are three possible configurations for the vehicle

1. *Front-wheel-steering*: $k_{sf} = 1, k_{sr} = 0$: only the front wheel is steerable, the rear steering is fixed ($\delta_s = 1$) at a constant value;
2. *Rear-wheel-steering*: $k_{sf} = 0, k_{sr} = 1$: only the rear wheel is steerable, the front steering is fixed ($\delta_s = 1$) at a constant value;
3. *Four-wheel-steering*: $k_{sf} = k_{sr} = 1$: both front and rear wheels are steerable ($\delta_s = 2$).

The model of the steering systems becomes

$$\begin{aligned}\dot{\gamma}_f &= k_{sf}\omega_f \\ \dot{\gamma}_r &= k_{sr}\omega_r\end{aligned} \quad (5.16)$$

5.2. Vehicle operation and vehicle model

142

(5.16) and (5.13) fully describe the behaviour of the vehicle even in the case where faults occur at steering systems. Then, we achieve the following kinematic model

$$\dot{q} = G_{4ws}(q)\nu \quad (5.17)$$

$$G_{4ws}(q) = \begin{bmatrix} R^T(\theta)Q(\gamma_f, \gamma_r) & 0 & 0 \\ \kappa(\gamma_f, \gamma_r) & 0 & 0 \\ 0 & k_{sf} & 0 \\ 0 & 0 & k_{sr} \end{bmatrix} \quad (5.18)$$

where $q = [x \ y \ \theta \ \gamma_f \ \gamma_r]^T$ and $\nu = [v \ \omega_f \ \omega_r]^T$.

It is to notice that the kinematic (5.17) is only irreducible, i.e. the dimension of its involutive closure is smaller than the dimension of the state vector q , when both steering systems are working well. If one of them malfunctions, the dimension of $\dot{q}$ is only 4 while the dimension of q is 5. Hence, to be able to derive a model that can cover all the three steering configurations, we need to use a model deduced from (5.17). The model must therefore have the degree of steerability of 1, which implies that the front and rear steering systems, if both are working properly, must be coordinated, instead of being independent.

The coordination can be defined at will. Note that the distance from the reference point to the *ICR* is

$$r = \frac{1}{\kappa(\gamma_f, \gamma_r)} = \frac{2a}{\cos \beta (\tan \gamma_f - \tan \gamma_r)}$$

and

$$|r| \geq r_{\min} = \frac{2a}{2 \tan \gamma_{\max}}$$

with $|r| = r_{\min}$ when the reference point is at the center of the vehicle and $\gamma_f = -\gamma_r$ and $|\gamma_f| = |\gamma_r| = \gamma_{\max}$. The minimum radius $r_{\min}$ for the 4WS vehicle is smaller than that of car-like vehicle (2.126), half of it to be precise. Thus, to exploit this tight steering capacity of the vehicle due to the additional steering system, we propose the following steering coordination

$$\begin{aligned} \omega_f &= k_{sf}\omega \\ \omega_r &= -k_{sr}\omega \end{aligned} \quad (5.19)$$

(5.19) indicates that when both steering systems are working, $k_{sf} = k_{sr} = 1$, then, $\omega_f = -\omega_r$, the front and rear steering rates are opposite in sign, it means the two steering systems steer in opposite directions. If one steering system is at fault, its steering rate becomes zero while the other's one is still controllable. The newly-introduced steering rate term ω has different physical

5.2. Vehicle operation and vehicle model

143

meanings in different configurations. It might be the front steering rate, the rear steering rate, or a common steering rate.

By combining (5.17) and (5.19), we obtain a unified kinematic model:

$$\dot{q} = G(q)\mu \quad (5.20)$$

$$G(q) = G_{4ws}(q) \begin{bmatrix} 1 & 0 \\ 0 & k_{sf} \\ 0 & -k_{sr} \end{bmatrix} = \begin{bmatrix} R^T(\theta)Q(\gamma_f, \gamma_r) & 0 \\ \kappa(\gamma_f, \gamma_r) & 0 \\ 0 & k_{sf} \\ 0 & -k_{sr} \end{bmatrix} \quad (5.21)$$

where $\mu = [v \ \omega]^T$.

5.2.2.3 Velocity coordination

From (5.1), we have

$$\dot{\varphi} = J_2^{-1} J_1 R_3(\theta) \dot{\xi} \quad (5.22)$$

where $\varphi = [\varphi_{fl} \ \varphi_{fr} \ \varphi_{rl} \ \varphi_{rr}]^T$ are the rotating velocities of the front left, front right, rear left and rear right wheels, respectively.

From the kinematic model (5.20), we have

$$\begin{aligned} \dot{\varphi} &= J_2^{-1} J_1 R_3(\theta) \dot{\xi} = J_2^{-1} J_1 \begin{bmatrix} Q(\gamma_f, \gamma_r) \\ \kappa(\gamma_f, \gamma_r) \end{bmatrix} v(t) \\ &= \begin{bmatrix} \left(1 - \frac{b\kappa}{\cos\beta}\right) \cos\gamma_{fl} + \tan\gamma_f \sin\gamma_{fl} \\ \left(1 + \frac{b\kappa}{\cos\beta}\right) \cos\gamma_{fr} + \tan\gamma_f \sin\gamma_{fr} \\ \left(1 - \frac{b\kappa}{\cos\beta}\right) \cos\gamma_{rl} + \tan\gamma_r \sin\gamma_{rl} \\ \left(1 + \frac{b\kappa}{\cos\beta}\right) \cos\gamma_{rr} + \tan\gamma_r \sin\gamma_{rr} \end{bmatrix} \frac{v \cos\beta}{r_\varphi} \end{aligned} \quad (5.23)$$

Substituting (5.7) and (5.8) into (5.23) produces

$$\dot{\varphi} = \begin{bmatrix} \dot{\varphi}_{fl} \\ \dot{\varphi}_{fr} \\ \dot{\varphi}_{rl} \\ \dot{\varphi}_{rr} \end{bmatrix} = \begin{bmatrix} \sqrt{\left(1 - \frac{b\kappa}{\cos\beta}\right)^2 + \tan^2\gamma_f} \\ \sqrt{\left(1 + \frac{b\kappa}{\cos\beta}\right)^2 + \tan^2\gamma_f} \\ \sqrt{\left(1 - \frac{b\kappa}{\cos\beta}\right)^2 + \tan^2\gamma_r} \\ \sqrt{\left(1 + \frac{b\kappa}{\cos\beta}\right)^2 + \tan^2\gamma_r} \end{bmatrix} \frac{v \cos\beta}{r_\varphi} \quad (5.24)$$

(5.24) shows the relationship between the velocity at each wheel with the velocity v at the reference point. Hence, whenever a command of v is available, it can be distributed to the four

wheels provided the steering angles are measurable. Even if one driving motor is faulty, its corresponding amplifier is shut down at once so that the wheel becomes free from the motor rotating axis. The reference voltage to that faulty driving control loop can be withheld for safety by sending out zero voltage instead. Obviously, if all 4 driving motors are down, the vehicle is unable to move. Furthermore, due to the possible limitation in power, using only one driving motor is not practically encouraged. Hence, if more than 2 driving motors malfunction, it is recommended to stop the operation immediately and call for repair.

For the purpose of having velocity feedback for the high-level controller, the velocity v might be required from either measurements or estimations. Since the velocity measurements are available at the four wheels by the incremental encoders, the velocity at the reference point can be estimated based on the relations (5.24) with the usage of measurements from the absolute encoders as well.

$$v = r_\varphi \frac{\frac{k_{m_{fl}} \dot{\varphi}_{fl}}{\sqrt{(1 - \frac{b\kappa}{\cos \beta})^2 + \tan^2 \gamma_f}} + \frac{k_{m_{fr}} \dot{\varphi}_{fr}}{\sqrt{(1 + \frac{b\kappa}{\cos \beta})^2 + \tan^2 \gamma_f}} + \frac{k_{m_{rl}} \dot{\varphi}_{rl}}{\sqrt{(1 - \frac{b\kappa}{\cos \beta})^2 + \tan^2 \gamma_r}} + \frac{k_{m_{rr}} \dot{\varphi}_{rr}}{\sqrt{(1 + \frac{b\kappa}{\cos \beta})^2 + \tan^2 \gamma_r}}}{(k_{m_{fl}} + k_{m_{fr}} + k_{m_{rl}} + k_{m_{rr}}) \cos \beta} \quad (5.25)$$

where $k_{m_{fl}}$, $k_{m_{fr}}$, $k_{m_{rl}}$ and $k_{m_{rr}}$ are boolean values indicating the working condition of the front left, front right, rear left and rear right incremental encoders, respectively. They can take either 0 or 1 with the meaning similar to that of k_{sf} and k_{sr} . And we should have

$$k_{m_{fl}} + k_{m_{fr}} + k_{m_{rl}} + k_{m_{rr}} \geq 1$$

5.3 Fault-tolerant vehicle-following controller

The vehicle following scenarios are defined similarly to those for car-like vehicles where the leader vehicle is either moving forward in front of or moving backward behind the follower vehicle. Parameter f that takes either 1 or -1 is used to represent the designated tracking mode.

In Part I, a focus point is defined with two parameters l and p . The focus point is depending on the front steering angle to track the moving desired point. The controller that manages to drive the focus point to track the desired point is developed.

For 4WS vehicles, if the front steering system is working, the vehicle-following controller developed in Part I, should work with the rear wheels fixed straight. However, if the front steering system is at fault, the vehicle-following controller is no longer able to work. Nevertheless,

5.3. Fault-tolerant vehicle-following controller

145

it is promising that the vehicle-following controller integrating the control of the rear steering system should be able to carry on the operation when the front steering system is at fault. To follow this idea, the focus point must be redefined such that it relies on both steering angles as follows

$$z(q) = \zeta + R^T(\theta)d_\gamma \quad (5.26)$$

$$d_\gamma = \begin{bmatrix} \bar{a} + l \cos \gamma_p \\ l \sin \gamma_p \end{bmatrix} \quad (5.27)$$

$$\gamma_p = p_f \gamma_f + p_r \gamma_r \quad (5.28)$$

$$\bar{a} = \frac{1+f}{2}a_f - \frac{1-f}{2}a_r \quad (5.29)$$

where l, p_f, p_r are design parameters. l still has the physical meaning of the desired distance between two vehicles. The difference from the focus point defined in Part I is that the focus point in this case is dependent on both front and rear steering angles through two coefficients p_f and p_r , respectively. The focus point is defined by the coordinate (l, γ_p) in a polar coordinate system of the vehicle.

The desired point on the leader vehicle is relative to the reference point of the follower vehicle as expressed in (2.56). And the new tracking error is

$$\tilde{z} = z - z_d = R^T(\theta)(d_\gamma - d_\phi) = R^T(\theta) \begin{bmatrix} l \cos \gamma_p - f d \cos \phi \\ l \sin \gamma_p - f d \sin \phi \end{bmatrix} \quad (5.30)$$

Lemma 5.3.1 *Consider a 4WS vehicle with restricted steering angles, $|\gamma_f, \gamma_r| \leq \gamma_{\max} < \frac{\pi}{2}$, and a vehicle tracking problem formulated as forward tracking or backward tracking such that the focus point z defined in (5.26) tracks the leading point z_d in (2.56). If parameters p_f and p_r are chosen such that $|p_f| + |p_r| < \frac{\pi}{2\gamma_{\max}}$ and l is a finite constant, then the following two statements are equivalent:*

1. The tracking error $\tilde{z}(t) = z(t) - z_d(t)$ is convergent to zero, i.e.,

$$\lim_{t \rightarrow \infty} \|\tilde{z}(t)\| = 0$$

2. The relative orientation angle ϕ converges to γ_p , i.e.,

$$\lim_{t \rightarrow \infty} (\phi - \gamma_p) = 0$$

and the relative intervehicular spacing d converges to fl , i.e.,

$$\lim_{t \rightarrow \infty} (d - fl) = 0$$

5.3. Fault-tolerant vehicle-following controller

146

Proof: From (5.30), we have

$$||\tilde{z}||^2 = l^2 + d^2 - 2fdl \cos(\phi - \gamma_p) = [d - fl \cos(\phi - \gamma_p)]^2 + [fl \sin(\phi - \gamma_p)]^2 \quad (5.31)$$

With the fact that l is a finite constant, statement 1 is equivalent to

$$\lim_{t \rightarrow \infty} \sin |\phi - \gamma_p| = 0 \quad (5.32)$$

$$\lim_{t \rightarrow \infty} |d - fl \cos(\phi - \gamma_p)| = 0 \quad (5.33)$$

(a) If statement 1 is true, we will prove that statement 2 is also true.

Since $|\gamma_f, \gamma_r| \leq \gamma_{\max} < \frac{\pi}{2}$ and $|p_f| + |p_r| < \frac{\pi}{2\gamma_{\max}}$ then

$$|\gamma_p| = |p_f \gamma_f + p_r \gamma_r| \leq (|p_f| + |p_r|) \gamma_{\max} < \frac{\pi}{2}$$

Additionally, the relative angle ϕ is also bounded, $|\phi| \leq \frac{\pi}{2}$. Thus, we have $|\phi - \gamma_p| \leq |\gamma_p| + |\phi| < \pi$. As a result, (5.32) leads to

$$\lim_{t \rightarrow \infty} (\phi - \gamma_p) = 0 \quad (5.34)$$

Combining (5.34) with (5.33) produces

$$\lim_{t \rightarrow \infty} (d - fl) = 0 \quad (5.35)$$

(b) If statement 2 is true, that statement 1 is also true is obvious with $||\tilde{z}(t)||$ in the form of (5.31). ■

Like Lemma 2.3.2, Lemma 5.3.1 also reveals that l must be chosen such that fl lies in the valid range of the measurable distance d , i.e. $0 \ll fl \ll d_{\max}$. It also states the new selection of the focus point also leads to the vehicle-following achievement if the tracking error $\tilde{z}$ is regulated to zero. Therefore, the closed-loop system (2.92), or its rewritten version (2.93), is again applied on the new tracking error $\tilde{z}$ in (5.30). To derive the control laws from the closed-loop system (2.93), the output function z is related to the control input μ as follows

$$\dot{z} = \frac{\partial z}{\partial q} \dot{q} = E(q) \mu \quad (5.36)$$

5.3. Fault-tolerant vehicle-following controller

147

where

$$\begin{aligned}
 E(q) &= \frac{\partial z(q)}{\partial q} G(q) \\
 &= \begin{bmatrix} I_2 & R^T(\theta) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} d_\gamma & R^T(\theta) \frac{\partial d_\gamma}{\partial \gamma_f} & R^T(\theta) \frac{\partial d_\gamma}{\partial \gamma_r} \end{bmatrix} \begin{bmatrix} R^T(\theta) Q(\gamma_f, \gamma_r) & 0 \\ \kappa(\gamma_f, \gamma_r) & 0 \\ 0 & k_{s_f} \\ 0 & -k_{s_r} \end{bmatrix} \\
 &= R^T(\theta) \left[\left\{ Q(\gamma_f, \gamma_r) + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} d_\gamma \kappa \right\} (k_{s_f} p_f - k_{s_r} p_r) \frac{\partial d_\gamma}{\partial \gamma_p} \right] = R^T(\theta) \bar{E}(\gamma_f, \gamma_r)
 \end{aligned} \tag{5.37}$$

with

$$\bar{E}(\gamma_f, \gamma_r) = \begin{bmatrix} \cos \beta \left(1 - \frac{l\kappa}{\cos \beta} \sin \gamma_p \right) & -l(k_{s_f} p_f - k_{s_r} p_r) \sin \gamma_p \\ \cos \beta \left(\tan \gamma + \frac{l\kappa}{\cos \beta} \cos \gamma_p \right) & l(k_{s_f} p_f - k_{s_r} p_r) \cos \gamma_p \end{bmatrix} \tag{5.38}$$

$$\gamma = \frac{1+f}{2} \gamma_f + \frac{1-f}{2} \gamma_r \tag{5.39}$$

Lemma 5.3.2 Consider a 4WS vehicle with restricted steering angle, $|\gamma_f, \gamma_r| \leq \gamma_{\max} < \frac{\pi}{2}$, and a vehicle tracking problem formulated as forward tracking or backward tracking. A control input μ exists for (5.36) if the design parameters l , p_f and p_r are chosen so that the following conditions are satisfied

1. $l \neq 0$
2. $p_f \neq 0; p_r \neq 0; p_f \neq p_r$
3. $\left| p_f - \frac{1+f}{2} \right| + \left| p_r - \frac{1-f}{2} \right| < \frac{\pi}{2\gamma_{\max}}$

Proof: The existence of the input μ is guaranteed if and only if matrix $E(\theta, \gamma_f, \gamma_r)$ or, equivalently, matrix $\bar{E}(\gamma_f, \gamma_r)$ is nonsingular. This is equivalent to the determinant of matrix $\bar{E}(\gamma_f, \gamma_r)$ is nonzero

$$\det \bar{E} = l(k_{s_f} p_f - k_{s_r} p_r) \cos \beta \frac{\cos(\gamma_p - \gamma)}{\cos \gamma} \neq 0 \tag{5.40}$$

The restricted steering angles implies $\cos \gamma \geq \cos \gamma_{\max} > 0$. Also we have

$$|\tan \beta| = \left| \frac{a_r \tan \gamma_f + a_f \tan \gamma_r}{2a} \right| \leq \frac{a_r}{2a} |\tan \gamma_f| + \frac{a_f}{2a} |\tan \gamma_r| \leq \tan \gamma_{\max} \Rightarrow |\beta| \leq \gamma_{\max} < \frac{\pi}{2}$$

thus, $\cos \beta > 0$. Inequality (5.40) is equivalent to

$$\begin{cases} 1. & l \neq 0 \\ 2. & k_{s_f} p_f \neq k_{s_r} p_r \\ 3. & |\gamma_p - \gamma| < \frac{\pi}{2} \end{cases} \tag{5.41}$$

5.3. Fault-tolerant vehicle-following controller

148

Condition (5.41-2), with k_{sf}, k_{sr} taking values of 0 or 1 only, leads to

$$p_f \neq 0; p_r \neq 0 \text{ and } p_f \neq p_r \quad (5.42)$$

Condition (5.41-3) leads to

$$\begin{aligned} & \left| p_f \gamma_f + p_r \gamma_r - \left(\frac{1+f}{2} \gamma_f + \frac{1-f}{2} \gamma_r \right) \right| = \left| \left(p_f - \frac{1+f}{2} \right) \gamma_f + \left(p_r - \frac{1-f}{2} \right) \gamma_r \right| \\ & \leq \left| p_f - \frac{1+f}{2} \right| \gamma_{\max} + \left| p_r - \frac{1-f}{2} \right| \gamma_{\max} < \frac{\pi}{2} \Rightarrow \left| p_f - \frac{1+f}{2} \right| + \left| p_r - \frac{1-f}{2} \right| < \frac{\pi}{2\gamma_{\max}} \end{aligned} \quad (5.43)$$

Conditions (5.41), (5.42) and (5.43) produce the conditions for $\det(\bar{E})$ to be nonzero as stated in the lemma's statement. ■

The term $\dot{z}_d$ is also computed by differentiating (2.56) with the use of the new kinematic model (5.20)

$$\begin{aligned} \dot{z}_d &= \dot{\zeta} + R^T(\theta) \left\{ \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta} d_\phi + \dot{d}_\phi \right\} \\ &= R^T(\theta) \left\{ Q(\gamma_f, \gamma_r) v + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \dot{\theta} d_\phi + \dot{d}_\phi \right\} \\ &= R^T(\theta) \left\{ \begin{bmatrix} v \cos \beta \\ v \cos \beta \tan \gamma \end{bmatrix} + f R^T(\phi) \begin{bmatrix} \dot{d} \\ d(\dot{\theta} + \dot{\phi}) \end{bmatrix} \right\} \end{aligned} \quad (5.44)$$

As a result

$$F_{kin}(q, d, \dot{d}, \phi, \dot{\phi}) = \dot{z}_d - \lambda \tilde{z} = R^T(\theta) \bar{F}_{kin}(v, \gamma_f, \gamma_r, d, \dot{d}, \phi, \dot{\phi}) \quad (5.45)$$

where

$$\bar{F}_{kin} = \begin{bmatrix} v \cos \beta \\ v \cos \beta \tan \gamma \end{bmatrix} - \lambda l \begin{bmatrix} \cos \gamma_p \\ \sin \gamma_p \end{bmatrix} + f R^T(\phi) \begin{bmatrix} \dot{d} + \lambda d \\ d(\dot{\theta} + \dot{\phi}) \end{bmatrix} \quad (5.46)$$

The closed-loop system (2.93) becomes

$$E(q) \mu = F_{kin}(q, d, \dot{d}, \phi, \dot{\phi}) \quad (5.47)$$

$$\Leftrightarrow \bar{E}(\gamma_f, \gamma_r) \mu = \bar{F}_{kin}(v, \gamma_f, \gamma_r, d, \dot{d}, \phi, \dot{\phi}) \quad (5.48)$$

With parameters l, p_f and p_r satisfying Lemma 5.3.2, matrix $\bar{E}(\gamma_f, \gamma_r)$ is invertible and the control input μ can be acquired

$$\mu_{kin} = \bar{E}^{-1}(\gamma_f, \gamma_r) \bar{F}_{kin}(v, \gamma_f, \gamma_r, d, \dot{d}, \phi, \dot{\phi}) \quad (5.49)$$

The feedback control law μ_{kin} is also independent of the global coordinate measurements (x, y, θ) .

5.3. Fault-tolerant vehicle-following controller

149

The explicit form of $\mu_{kin} = [v_{kin} \ \omega_{kin}]^T$ in terms of measurable parameters $v, \gamma_f, \gamma_r, d, \dot{d}, \phi$ and $\dot{\phi}$ is

$$\begin{aligned} v_{kin} &= \frac{\cos \gamma}{\cos \beta \cos(\gamma_p - \gamma)} \begin{bmatrix} \cos \gamma_p & \sin \gamma_p \end{bmatrix} \bar{F}_{kin} \\ &= v + \frac{\cos \gamma \left[-\lambda l + f(\dot{d} + \lambda d) \cos(\gamma_p - \phi) + f d(\dot{\theta} + \dot{\phi}) \sin(\gamma_p - \phi) \right]}{\cos \beta \cos(\gamma_p - \gamma)} \end{aligned} \quad (5.50)$$

$$\begin{aligned} \omega_{kin} &= \frac{\cos \gamma}{(k_{sf} p_f - k_{sr} p_r) \cos(\gamma_p - \gamma)} \left[-\left(\frac{1}{l} \tan \gamma + \frac{\kappa}{\cos \beta} \cos \gamma_p \right) \frac{1}{l} - \frac{\kappa}{\cos \beta} \sin \gamma_p \right] \bar{F}_{kin} \\ &= -\frac{\dot{\theta}}{k_{sf} p_f - k_{sr} p_r} + \frac{-\lambda \sin(\gamma_p - \gamma) + f \frac{\dot{d} + \lambda d}{l} \sin(\phi - \gamma) + f \frac{d}{l} (\dot{\theta} + \dot{\phi}) \cos(\phi - \gamma)}{(k_{sf} p_f - k_{sr} p_r) \cos(\gamma_p - \gamma)} \\ &\quad - \frac{\kappa \cos \gamma \left[-\lambda l + f(\dot{d} + \lambda d) \cos(\gamma_p - \phi) + f d(\dot{\theta} + \dot{\phi}) \sin(\gamma_p - \phi) \right]}{(k_{sf} p_f - k_{sr} p_r) \cos \beta \cos(\gamma_p - \gamma)} \end{aligned} \quad (5.51)$$

It is obvious that the fault at one steering system could affect only the steering control while it has no effect on the velocity control.

In fact, a 4WS vehicle can become a car-like vehicle with one steering system fixed straight. Thus, all the conditions derived in Chapter 2 for vehicle-following controller developed for car-like vehicles can be applied for the controller for 4WS vehicles with special settings as below

- *Front-wheel-steering configuration:* when the rear wheels are fixed straight, $\gamma_r = 0$, and the front steering wheels are steerable, we obtain the configuration the same as that of car-like vehicle. The focus point relies on the front steering angle only, thus, parameter p_r has no effects at all on the controller. Then the conditions for this situations are derived from (2.102) for look-ahead tracking and (2.103) for look-behind tracking

$$\begin{cases} 0 \ll fl \ll d_{\max} \\ lp_f > 0 \\ |p_f| < \frac{\pi}{2\gamma_{\max}} \end{cases} \quad (5.52)$$

- *Rear-wheel-steering configuration:* when the front wheels are fixed straight, $\gamma_f = 0$, and the rear steering wheels are steerable, the vehicle is actually a car-like vehicle except its steerable wheels are now mounted at the back of the vehicle. Hence, its going forward is just like that the virtual car-like vehicle goes reverse. Similarly, going-backward is the same as that the virtual car-like vehicle's going forward. Thus, the vehicle's polar coordinate system of the 4WS vehicle and that for the virtual car-like vehicle are 180° different. In other words, parameter l for the 4WS vehicle's tracking should be $(-l)$

5.3. Fault-tolerant vehicle-following controller

150

for the tracking of the virtual car-like vehicle. The tracking mode in the sense for 4WS vehicle is also changed in sign when the vehicle is looked at from the virtual car-like vehicle's point of view. It means $f = -1$ will indicate the virtual vehicle is in the look-ahead tracking mode and $f = 1$ for the look-behind tracking mode. Consequently, Theorem 2.3.8 states the necessary conditions are

$$\begin{cases} 0 \ll (-f)(-l) \ll d_{\max} \Leftrightarrow 0 \ll fl \ll d_{\max} \\ (-l)p_r > 0 \Leftrightarrow lp_r < 0 \\ |p_r| < \frac{\pi}{2\gamma_{\max}} \end{cases} \quad (5.53)$$

- *Four-wheel-steering configuration:* if the center of the vehicle is chosen as the reference point, $a_f = a_r = a$, with the steering coordination (5.19) and the initial steering angles being zero, the two steering angles will be opposite in sign which results in the side-slip angle β 's becoming zero. Eventually, the vehicle model (5.20) is equivalent to the kinematic model of a car-like vehicle whose length is half the length of the 4WS vehicle. Thus, all the conditions developed for the car-like vehicle are applicable to the 4WS vehicle.

The conditions on (l, p_f) or (l, p_r) are then combined to guarantee that the tracking can continue even if suddenly a steering system stops working due to a fault. Furthermore, the conditions for the the situations where the leader vehicle travels on a circular path developed in Chapter 2 are most likely applicable.

The only problem may be if one steering system is at fault, its steering angle may be fixed at a **nonzero** angle which creates the asymmetry in steering maneuvers for the vehicle when it turns left or right. For example, if the rear steering angle is fixed at 15° , if the vehicle needs to turn left, i.e. $\dot{\theta} > 0$, from the equation of $\dot{\theta}$ in (5.20), the front steering angle must be greater than 15° , while to turn right, even if the front steering wheels are slightly turned left, the vehicle is still turning right. This asymmetry in steering maneuvers narrows the performance of the vehicle as well as the controller to avoid the saturation of the steering angles. The follower vehicle may be moving in an abnormal way. For example, to follow the leader vehicle moving on a straight path, the follower will run on another straight path parallel to that of the leader vehicle with the heading angle being not aligned with that of the leader vehicles. In contrast, the newly proposed controller (5.49) is ready for bad situations where there are faults in the vehicle system. It is understandable that when there is a fault, the overall performance should be

degraded. However, if the fault is contained and the operation can be carried on, the degradation would be acceptable.

Summary of conditions

The necessary conditions are combined from the conditions of Lemma 5.3.1, Lemma 5.3.2, (5.52) and (5.53)

$$\left\{ \begin{array}{ll} 0 \ll fl \ll d_{\max} & \text{(i)} \\ |p_f| + |p_r| < \frac{\pi}{2\gamma_{\max}} & \text{(ii)} \\ l \neq 0 & \text{(iii)} \\ p_f \neq 0; p_r \neq 0; p_f \neq p_r & \text{(iv)} \\ |p_f - \frac{1+f}{2}| + |p_r - \frac{1-f}{2}| < \frac{\pi}{2\gamma_{\max}} & \text{(v)} \\ lp_f > 0 & \text{(vi)} \\ |p_f| < \frac{\pi}{2\gamma_{\max}} & \text{(vii)} \\ lp_r < 0 & \text{(viii)} \\ |p_r| < \frac{\pi}{2\gamma_{\max}} & \text{(ix)} \end{array} \right. \quad (5.54)$$

For each tracking situation, we deduce

- *Look-ahead tracking:* $f = 1$

$$\left\{ \begin{array}{ll} 0 \ll l \ll d_{\max} & \text{(i)} \\ -\frac{\pi}{2\gamma_{\max}} < p_r < 0 & \text{(ii)} \\ \max\left(0, 1 - p_r - \frac{\pi}{2\gamma_{\max}}\right) < p_f < p_r + \frac{\pi}{2\gamma_{\max}} & \text{(iii)} \end{array} \right. \quad (5.55)$$

- *Look-behind tracking:* $f = -1$

$$\left\{ \begin{array}{ll} -d_{\max} \ll l \ll 0 & \text{(i)} \\ -\frac{\pi}{2\gamma_{\max}} < p_f < 0 & \text{(ii)} \\ \max\left(0, 1 - p_f - \frac{\pi}{2\gamma_{\max}}\right) < p_r < p_f + \frac{\pi}{2\gamma_{\max}} & \text{(iii)} \end{array} \right. \quad (5.56)$$

5.4 Simulation studies

Simulations are carried out to verify the effectiveness of the proposed control scheme, especially when faults occur at the 2 steering systems and the 4 driving motors. Since the vehicle can travel forth and back with front and rear steering systems, the two tracking modes are basically symmetrical, especially with the center of the vehicle chosen as the reference point. Therefore, simulations for look-ahead tracking ($f = 1$) are sufficient to examine the proposed controller.

5.4.1 Simulation faults on the co-simulation platform

The co-simulation platform described in Chapter 3 can be used to simulate faults at driving and steering systems. From the design of the vehicle model in ADAMS, we notice that the wheels are under control, i.e. being driven or steered, only when there is an active torque acting on it. This is very much the same as the real situations. Thus, to simulate the faults at the driving system, we just simply nullify the driving torque

$$\tau_{driving_i} = k_{d_i} * K_p * I_z * \left(\frac{v_{i_d}}{R} - \omega_i \right) \quad (5.57)$$

where k_{d_i} is the boolean value indicating that the driving system for wheel i is working properly or not. When $k_{d_i} = 1$, the driving system is fine and drives the wheel to the desired speed v_{i_d} . When $k_{d_i} = 0$, the torque is zero and the wheel's rotation is independent of the vehicle body.

For the steering system, the steering torque is generated to steer the wheels at a desired steering rate through a closed-loop control. When there is a fault, the similar strategy, i.e. to set the torque to zero, can be applied. However, in practice, when the steering system is at fault, it is very likely that the wheels are firmly held at their current steering position due to the oil trapped in the hydraulic cylinder. Therefore, the steering torque is modeled in such a way that it will regulate the steering rate to zero

$$\tau_{steering_i} = K * I_y * (k_{s_i} \omega_{i_d} - \omega_i) \quad (5.58)$$

Normally, $k_{s_i} = 1$ and the closed-loop steering system works as usual to keep the steering rate at ω_{i_d} . When a fault occurs, $k_{s_i} = 0$, the desired steering rate becomes 0. The steering torque now plays the role of the torque generated by the trapped oil to hold the wheels at their current steering position.

In total, there are 6 fault indicators: 2 for the steering systems and 4 for the driving systems. Their values, or the occurrence of the faults, are set randomly by the SIMULINK model. Thus, to ADAMS model, these indicators are treated as inputs.

5.4.2 Setting-up

Without loss of generality, we choose the reference point at the center of the vehicle ($a_f = a_r$). Since l is the expected distance between two vehicles, it is application-dependent and chosen in advance. In the following study, we focus on the effects of system parameters p_f and p_r , which are introduced in the proposed vehicle-following controller.

Figure 5.2: Values of p_f, p_r to guarantee tracking stability

The closed-loop system's parameters λ is set as a constant of 5. The desired spacing is chosen to be $l = 2.5m > 0$. The real specifications of the Cycab vehicles are used: $a_f = a_r = 0.6m$, $2b = 1.1m$, $\gamma_{\max} = \frac{\pi}{8}$. The range of the sensor is $20m$. Then the necessary conditions are listed in (5.59).

$$\begin{cases} 0 \ll l \ll 20m & \text{(i)} \\ -4 < p_r < 0 & \text{(ii)} \\ \max(0, -p_r - 3) < p_f < p_r + 4 & \text{(iii)} \end{cases} \quad (5.59)$$

(5.59-ii) and (5.59-iii) are illustrated in the shaded area in Figure 5.2

The trajectory of the leader vehicle is maintained the same for every simulation. The leader vehicle is modeled as a car-like vehicle, running in front to generate a desired trajectory. During a simulation, faults are made occur at random at the two steering systems and the four driving motors with the condition that at any time there is at least one operational steering system and at least 2 operational driving motors. Figure 5.3 shows the timing diagram of the simulated faults. The upper four lines indicate the working condition of the four driving motors (high level for working fine and low level for faulty state). The two lower lines represent the status of the two steering systems: front steering k_{sf} and rear steering k_{sr} , respectively.

The vehicle starts with both working steering systems and zero initial steering angles. After 15.5 seconds, the rear steering gets fault and is disabled. The vehicle runs with the front steering only. At the 23rd second, the front steering is recovered. However, 5 seconds later, it is the front steering system that malfunctions this time. The vehicle becomes a rear steering vehicle. For the driving motors, the faults are designed in such a way that there are always at least 2 operational motors at all time. Initially, all the motors are working. After 8 seconds, faults begin to occur, first at the front left wheel, then at the front right wheel at the 15th second. Although the front

Figure 5.3: Faults occurred against time

left motor is recovered at the 21st second, the rear right motor is at fault 2 seconds later. The final error comes at second 33 at wheel rear left. Wheel rear right is recovered at the same time.

5.4.3 Results

The results of tracking performance are shown in Figures 5.4-5.7. The follower vehicle successfully follows the leader vehicle at the desired distance of 2.5m with the presence of one or more faults at the subsystems. The vehicle smoothly moves when its steering systems switch from working to faulty conditions and vice versa. Whenever a fault occurs, there is a small peak in the absolute tracking error. It implies a new initial condition for the tracking performance. It will then be pulled down immediately by the proposed close-loop system. When the leader vehicle changes its speed or its steering angle(s), the error also increases. But the error will start converging to zero immediately after the velocity and/or the steering angles of the leader vehicle are stable. This manner has been proved for the car-like tracking situations. Figure 5.6 shows the vehicle’s velocity is also smooth and can track the leader vehicle’s velocity. Though faults are on/off at one or two driving motors, the vehicular velocity is still maintained continuous and smooth. The steering angles in Figure 5.7 illustrate how flexible the vehicle maneuvers could be. With the additional steering for the rear wheels, to track a steering angle of about 17° of the leader vehicle, the follower vehicle only needs to steer its front and rear steering systems to angles of $\sim 3^\circ$ and -3° , respectively. Moreover, parallel maneuvers have been introduced. It

Figure 5.4: Trajectory ζ of the vehicles

Figure 5.5: Absolute output tracking error $\|z\|$

Figure 5.6: Velocity v of the vehicles

Figure 5.7: Steering angles γ_f, γ_r of the vehicles

means the vehicle translates without changing its heading orientation but the moving direction is different from the heading orientation. It happens when the two steering angles are the same. The final 10 seconds of the simulation clearly show this maneuver. The leading moves straight while the follower vehicle translates with both of its steering angles being about 5° .

5.5 Conclusions

This chapter presents a fault-tolerant vehicle-following controller for four-wheel-steering vehicles. The controller is an extended version of that developed for car-like vehicles in Part I. The key idea of this controller is that the focus point is now dependent on both steering angles rather than just the front steering angle as for car-like vehicles. Based on a suitable vehicle model that accurately represents the vehicle even when there are faults at its components or subsystems, the controller is able to carry on driving the vehicle to follow the leader vehicle without having to stop. The faulty component or subsystem is isolated from other working ones to prevent the fault from propagating.

In the next chapter, the implementation of the controller on a real vehicle will be presented. The development of the fault-tolerant low-level control system for such a 4WS vehicle will be described.

Chapter 6

Experiments on a Distributed Four-Wheel-Driving Four-Wheel-Steering Vehicle

6.1 Introduction

This chapter describes the design and development of a decentralized low-level control system for the Red Cycab vehicle, followed by the experimental results of the controller developed in the previous chapter.

6.2 Design of fault-diagnostic low-level control system

Having examined the structure and operating principles of all the driving motors and steering systems, it is noted that in fact, the vehicle is able to operate with two driving motors and one steering system. This reduced configuration is made possible by disabling the amplifier of the unused motors and steering system. Therefore, the idea of developing 2 independent but cooperative controllers - one for the two front driving motors and the front steering system, called the front controller, and the other, namely the rear controller, for the rear subsystems - became reasonable. The new system is hopefully able to carry on the operation even in the case where one controller fails to work.

Each controller is designed as a PC-based controller that consists of motion control boards to interface with actuators and their feedback encoder. When being combined with proper control

strategies, the two front and rear low-level controllers will offer better robustness against partial failures of the actuators and/or sensors. For example, if there is a fault at the front part, the vehicle is still able to operate using its rear part and vice versa. When there is no fault, the two controllers cooperate with each other to provide the full functionalities of the four-wheel-steering four-wheel-driving vehicle. A high-level controller that plays the role of a driver can be developed and executed on another computer, called the host PC, to issue desired velocity and steering angles to the low-level controllers via communication channels such as wireless network, Ethernet or RS232.

Due to the limitation in space of the existing case and to the limitation in power of the onboard batteries, each low-level controller is designed and built using a set of PC/104 boards. The boards in the PC/104 form are very compact in size, 10cm-by-10cm, and they require very little power solely from a 5VDC power supply. They are stacked on one another to make up a tower-like computer, that can be placed in a small closed box (Figure 6.1). We even do not need to worry about heat-disposal problems for these boards.

Each controller is composed of one all-in-one CPU board [122], which supports serial and Ethernet communications, and 2 motion control boards, one for controlling the two driving motors and the other for controlling the attached steering system. The motion control board is the POSYS 802 from servo-Halbeck, Germany [123]. This board is able to handle 2 control axes in either open-loop mode or closed-loop feedback control mode by its built-in PID filters. The same boards are used for both driving and steering controls. Though it costs a bit more compared to the usage of a 4-axis motion control board to handle both driving and steering controls, using two similar 2-axis motion control boards for driving and steering offers an option to recover easily the operation should one board break down. Moreover, if even if one motion controller is not working, e.g. driving is lost, the other controller still provides its assigned function, e.g. steering control.

By using open-loop control mode, the user is able to set directly the output analog voltage which is, in general, the reference to the actuator/motor. Closed-loop control, on the other hand, takes the feedback from the sensor/encoder and compare with the desired value, in terms of position or velocity, set by the user. The error is treated as the input to a PID filter. Depending on its adjustable gains, the PID filter will generate the output voltage to control the actuator/motor to that desired position or velocity.

Figure 6.1: Red Cycab low-level controllers: The two towers

Figure 6.2: Block diagram of the driving system

6.2.1 Driving control

As shown in Figure 6.4, velocity commands are issued by the host PC to the low-level controllers through communication channels and then implemented by the closed-loop PID filter of the POSYS controller card. The reference voltage generated by the POSYS card, ranging from -10V to 10V, must be mapped to the valid input range of the amplifier [124] which is from 0.1V to 4.9V. To achieve that an operational amplifier is used to build the mapping circuit. The schematic design of the circuit is shown in Figure 6.3.

To further protect the amplifier in the case where the op-amp may be at fault, the output from the op-amp passes through a clamp circuit composed of two diodes to make sure it falls in the valid range of 0.1V-4.9V. The two resistors $R5$ and $R6$ are purposely to limit the current passing through the Zener diode $D2$. Their resistance is very small compared to other resistors

6.2. Design of fault-diagnostic low-level control system

160

Figure 6.3: Schematic design of the mapping circuit

Table 6.1: The resistors used in the mapping circuit

Resistor	Value ($k\Omega$)
$R1$	6.2
$R2$	1.5
$R3$	1.2
$R4$	4.7

in the circuit. Then, the voltage is sent to a photo-relay. The photo-relay plays a role of a gate to the amplifier and is controlled by a logic signal through the use of function SetAmpEnable of the motion control board. When this photo-relay is open, the output voltage is sent to amplifier. When it is closed, its output becomes high-resistance. The amplifier is able to detect this high-resistant state and consider it as an input fault and immediately disable itself and shut down its PWM driver. When the high-resistant state is over, the amplifier will automatically enable itself.

Mathematically, the relation between the input and output voltages of the mapping circuit is

$$V_{out} = -\frac{R2}{R1}V_{in} + \left(1 + \frac{R2}{R1}\right) \frac{R3}{R3 + R4}12^V - 0.7^V \tag{6.1}$$

The resistors are chosen such that

$$\begin{aligned} \frac{R2}{R1} &= \frac{2.4}{10} \\ R3 &= R1 // R2 \\ \left(1 + \frac{R2}{R1}\right) \frac{R3}{R3 + R4}12^V - 0.7^V &= 2.5^V \end{aligned}$$

and listed in Table 6.1.

Then Equation (6.2) becomes

$$V_{out} = -0.24 \cdot V_{in} + 2.5^V \tag{6.2}$$

Table 6.2: Voltage Mapping of the driving system

V_{in} from POSYS	V_{out} to the amplifier	Motion direction
$-10V \rightarrow -0.87V$	$4.25V \rightarrow 2.7V$	anticlosewise/backward
$-0.87V \rightarrow 0.87V$	$2.7V \rightarrow 2.3V$	deadzone/stopping
$0.87V \rightarrow 10V$	$2.3V \rightarrow 0.45V$	closewise/forward

Table 6.3: PID gains used for the driving system

K_p	3
K_i	20
K_d	500
K_{out}	65535 (full-scale)

The testing result shows the mapping circuit works as expected (Table 6.2).

The amplifier, in turn, generates a corresponding high-current PWM signal to the DC motor. The current velocity is measured and fetched to the PID filter by an incremental encoder [125] at the resolution of 500 pulses per revolution. The desired velocity is maintained until a new velocity command is issued. The PID gains were obtained using trial-and-error method. The requirements were to stabilize the motor to any desired velocity within the working range of $\pm 4m/s$ as quickly as possible and with an acceptably small overshoot. The currently used set of gains for a driving system is shown in Table 6.3 with the PID filter is defined as follows

$$Output_n = \left(K_p * E_n + K_d * (E_n - E_{n-1}) + K_i * \sum_{j=0}^n E_j \right) * K_{out} \tag{6.3}$$

where E_n is the accumulated error term, K_p , K_d and K_i are the adjustable gains, and K_{out} is the scale factor of the output command.

6.2.2 Steering control

Similar to the driving control, a command is issued from the host PC to set the desired steering angle of a steering system. When the command is received, the PID filter on the POSYS compares it with the current steering angle to generate the reference voltage. The steering amplifier [126] converts this $\pm 10V$ reference signal to a PWM signal and sends it to a DC motor. The rotation of the motor is transferred through a belt to a hydraulic pump [127] which creates a force to steer the mechanical system of the steering system linking to the two wheels. The steer-

Figure 6.4: Block diagram of the steering system

ing angle is measured by a 13-bit absolute encoder [128] at the resolution of $360^\circ/2^{13} \sim 0.044^\circ$. The POSYS board, besides being able to read from incremental encoders, is able to read from other sensors through its optional SSI daughter card, the SSI800 card [129]. This small card mounted directly on top of the POSYS board is equipped to interface with 4 SSI (Synchronous Serial Interface) channels. The SSI synchronous serial interface is an industry standard output and is supported by most suppliers of automation systems. Those sensors/encoders that support the SSI interface can be connected directly to the SSI800 card. The card converts the synchronous serial input signal into a 16-bit word and fetches it to the POSYS board as a feedback. The clock speed is programmable and can be up to $1.1MHz$. Unfortunately, the existing absolute encoder has a 13-bit parallel output that does not match with the SSI interface. Moreover, the encoder output is in binary code which is again not suitable for the SSI800 card that only supports the Gray code. Therefore, a converter circuit is designed to convert the output of the absolute encoder to the acceptable form of the SSI800 card. The schematic diagram of the circuit is shown in Figure 6.5. The converter relies on the clock pulses generated by the SSI800 card to transmit the 13 output bits of the encoder, one by one. The conversion from binary code to Gray code [130, 131] is also accomplished simultaneously.

With the feedback from the absolute encoder through the converter card and the SSI800 card, the PID filter is able to calculate the necessary reference voltage for the steering system. The behaviour of the steering performance is dependent on the adjustable PID gains. Trial-and-error method was used to chose the appropriate PID gains, as shown in Table 6.4, that works well with any desired steering angle in the working range of $\pm 22.5^\circ$. The gain sets for front and rear steering systems are slightly different because the center of gravity of the vehicle is not

6.2. Design of fault-diagnostic low-level control system

163

Figure 6.5: Schematic diagram of the encoder output converter

Table 6.4: PID gains used for the steering systems

	Front steering	Rear Steering
K_p	200	300
K_i	20	150
K_d	500	500
K_{out}	65535 (full-scale)	65535 (full-scale)

exactly at the center of the vehicle, which results in the unbalance of loads on the front and rear parts.

6.2.3 Emergency stop subsystem

During experiments, it is very necessary and useful to have a mechanism that can stop the vehicle immediately especially when an autonomous controller is under testing. Thus, an Emergency Stop subsystem is designed and installed. It is equipped with four electrically-driven brakes installed at every wheel and controlled by a contacting relay (Finder Type 55.32 9048 0040, 10A, 48V DC, see [132]). The relay works on two states connecting its output either to +24V or to the ground of the vehicle batteries. When +24V is applied to the brake, its two skids are pulled out of the wheels’s rotating axis leaving the wheel under the control of the motor. When no voltage is applied on the brake, the skids are released from the magnetic force created by the voltage, but then are pulled close to each other by strong springs to hold the rotating axis resulting in an emergency stop of the wheel.

Since all the four brakes are connected to the relay, by controlling the state of the relay we can control all the four brakes. For the safety, when the brakes are on, the driving motors must be disabled. This is to protect the brakes and the motors from being overheated by the friction when the motor tries to rotate and the brake tries to hold. This manner is done by linking the INHIBIT signal (active low) of the amplifier to the logic control signal of the relay. When the control signal is low or when the power is cut off, the brakes are on and the amplifiers are disabled (inhibited). And when the control signal is high, the relay output switches to +24V and the brakes are off the rotating axes. The amplifiers are also enabled and work as discussed earlier in Section 6.2.1.

There are three ways to control the control signal of the relay: two for human operator and one for computer-based controllers. The first two are in fact two buttons: one is fixed onboard

6.2. Design of fault-diagnostic low-level control system

165

the vehicle and the other is a remote button that control an onboard ON/OFF circuit via radio frequency connection. The third way is a logic output signal controlled by the two onboard controllers. Any one of the three ways must be able to stop the vehicle. In addition, only when all the three are on should the brakes be off the rotating axes. To give human operator higher privileged control of the vehicle, the first two are to open or close the main relay which supplies +48V battery power to all the vehicle's components. When this main relay is open, all the amplifiers, DC-DC converters, and computer-based controllers are shut down. The onboard controllers, with lower control privilege, are to open or close the brake relay only. A problem arises here because there are two controllers and only one logic control signal. It is necessary to build a logic circuit that generates the control signal with the following requirements:

- *One of the two controllers is down:* the other will take the full control of the brake relay
- *Both controllers are working:* either of the two controllers is able to set the brake on. In other words, the brakes are pulled off the rotating axes only when both controllers agree to do so. This is to make sure they both are ready for the motion and velocity coordination.
- *Both controllers are not working:* obviously the brakes must be holding the rotating axes in this case for the safety purpose.

Therefore, each of the two controllers use two digital output signals to indicate its working status, CPU_f and CPU_r , and its brake control signal, $BRAKE_f$ and $BRAKE_r$. CPU_f and CPU_r are high when the softwares running on the onboard controllers are working normally and they are low when the respective controller is off or the software is not ready. The two $BRAKE$ signals indicate the controllers wish to pull the brakes off the axes (when high) or let them hold the axes (when low). A logic circuit was designed to generate the brake control signal based on these 4 signals from the onboard controllers. A high level of the output means the brakes should be off the axes and when it is low, the brakes are on. We have the following truth table (Table 6.5)

The logic equation is then deduced

$$\begin{aligned} Brake_off &= \overline{CPU_f} \cdot \overline{CPU_r} + CPU_f \cdot BRAKE_f + CPU_r \cdot BRAKE_r \\ &= \overline{CPU_f + CPU_r} \cdot \overline{CPU_f \cdot BRAKE_f + CPU_r \cdot BRAKE_r} \end{aligned} \quad (6.4)$$

And the logic circuit is implemented using only 2 ICs: 74LS02 and 74LS08. The emergency stop subsystem is finally completed and shown in Figure 6.6.

6.2. Design of fault-diagnostic low-level control system

166

Table 6.5: Truth table of the logic brake control circuit

$(CPU_r, BRAKE_r)$	$(CPU_f, BRAKE_f)$			
	(L, L)	(L, H)	(H, H)	(H, L)
(L, L)	L	L	L	H
(L, H)	L	L	L	H
(H, H)	L	L	L	L
(H, L)	H	H	L	H

Figure 6.6: The Emergency Stop subsystem

Figure 6.7: Transmission Path

6.2.4 Communication among controllers

The configuration for the transmission path between the host PC and the motion controllers is shown in Figure 6.7. It operates in a master/slave arrangement. This is to prevent the two low-level controllers from overloading the host PC with an abundance of feedback information. The host PC has to communicate with other devices and do high-level calculations, thus it is advisable to transmit a reasonable amount of data.

The Host PC transmits all the commands to the master controller, controller *A*, solely. If there is a command for controller *B* (the slave), controller *A* will forward it to controller *B*. For feeding back to the host PC where the high-level controller is running, all the measurements are gathered at controller *A* before being sent to the host PC.

Though not in use, the communication channel between the host PC and controller *B* is reserved for the situations when the link between controller *A* and the host PC is at fault. In these cases, controller *A* and controller *B* will swap their role: controller *A* becomes the slave and controller *B* becomes the master. The communication channels from the host PC to the two controllers are monitored based on a send/acknowledgment protocol with a preset timeout.

Two types of physical connections are set up and used simultaneously: Ethernet and RS232. The Ethernet network, with a transfer rate of 100Mbps, links all the three computers together with a hub while an RS232 connection is established between any pair of computers (transfer rate of 112Kbps). The Ethernet network is much preferable due to its magnificently high transfer rate. The RS232 connections are much slower and hence used as the alternative channel when the Ethernet network is down (very unlikely).

The low-level control architecture of Red Cycab can be summarized in Figure 6.8.

6.2.5 Software configuration for Red CyCab

The software systems running on two low-level controllers and the host PC are programmed using Microsoft Visual C++ and run in Microsoft Windows operating system. Though Microsoft Windows is not a real-time operating system, unlike the embedded controller of the Blue Cycab, all the motion controls are carried out in real-time by a DSP (Digital Signal Processor) in conjunction with an ASIC (Application-Specific Integrated Circuit) on the POSYS card. The PC/104 CPU board actually plays a less important role since its major task is to establish and maintain the communication among the two low-level controllers and the host PC as well as to collect/send out the logic signal indicating the working condition of the system components.

Figure 6.8: Red Cycab's low-level control system

Motion control commands are sent to and implemented by the POSYS cards themselves. Application software written for one controller are provided with a wide range of commands in terms of function calls to get access to functionalities of the POSYS card. Some frequently used functions are listed in Table 6.6

6.2.6 Design of a fault detection and isolation system for the vehicle

The importance of having a fault diagnostic system has become essential after the vehicle was down for quite some time. The purpose is to keep monitoring the working condition of all the important components in the system. The fault diagnostic system is to tell whether a component is working normally or not. The fault isolating system is to immediately isolate the faulty component and disable the involved components as well to protect them from abnormal operations.

Table 6.6: Frequently called functions of the POSYS card

Function	Description
SetMotorMode(axis,mode)	Determine the mode of motor operation ON: closed-loop with a PID filter in used OFF: open-loop and the output voltage is set directly
SetMotorCommand(axis)	Set the output voltage (only in open-loop mode)
GetMotorCommand(axis)	Get the current output voltage
SetKi/SetKp/SetKd (axis,value)	Change the 3 gains of the PID filter
SetPosition(axis,value)	Set the desired position for the PID filter
SetVelocity(axis,value)	Set the desired velocity for the PID filter
GetActualPosition(axis)	Get the reading from incremental encoder
GetActualVelocity(axis)	Get speed estimate of the incremental encoder
GetAbsoluteSSIPosition(axis)	Get the reading from absolute encoder

The isolation may affect the operation of the vehicle especially when it is in autonomous mode controlled by a high-level controller. Therefore, the information about the faulty and isolated components must be reported to the high-level controller and the controller should be able to adjust itself to work with a new physical configuration of the vehicle. This process is called “*fault tolerant control*”.

6.2.6.1 Driving system

A driving system is actually a closed-loop control for a driving motor as described earlier. Any fault at one component in the loop will cause the whole loop to work improperly. For example, if the incremental encoder is faulty, the feedback of the PID filter is wrong resulting in wrong output voltage to the motor which, in turn, apparently cause wrong motions of the motor and the wheels. Likewise, if a fault occurs at the DC motor making it partially damaged, the PID will have to issue much higher reference voltage to request the amplifier to inject much higher current to the DC motor so that the motor can hopefully rotate at the desired angular velocity. This will cause over-voltage and over-temperature at the motor and very soon the motor will be totally burnt. That is why when a fault is detected, it is advisable to turn off the amplifier.

Since the vehicle is a rigid object, the velocities at different positions on its body are all coordinated. A simple example is that when the vehicle runs on a straight line at a constant

6.2. Design of fault-diagnostic low-level control system

170

speed, all the four wheels must run at that constant speed too. Now if one driving motor is not working properly, the speed at that wheel is not coordinated with the other three, and the motion of the vehicle becomes unpredictable. If somehow the faulty motor can be stopped from working and the attached wheel released from the control of the motor. This is made possible by disabling the amplifier that control the DC motor. It is well known that a DC brushed motor, like what Cycab vehicles have, is composed of two parts: the stator where the stationary magnets are mounted and the current-carrying coil, namely the armature windings or the rotor. When an electric current passes through the coil, which is placed in a magnetic field, a magnetic force, or torque for this case, is generated and perpendicular to the magnetic field and the electric current. The total torque acting on the coil is the subtraction of the internal friction and the external torque from the magnetic torque. The internal friction is normally so small that it can be neglected. The external torque, in this case, normally is in fact generated by the friction between the tire and the ground. Thus, when the magnetic force is large enough, it actually turns the coil and creates the motion of the motor. As it is stated, only when there exists a current in the coil can the DC motor turn. If there is no current crossing the coil, there will be no magnetic torque and the motion of the coil is dependent solely on external torque acting on the wheel. Indeed, the wheel is released from the control of the motor and rotates independently. Since the current crossing the coil of the motor is generated and controlled by the amplifier, to cut off this current, we just simply disable the amplifier.

Having examined the amplifier, it was found out that the amplifier [124] has its own fault diagnostic system. This system can automatically detect faults at its input reference voltage, its internal components, and its output current to the motor. Whenever a fault is detected, the amplifier will shut down the output and signal the fault code via a fault diagnostic signal. During normal operation, with no faults, the diagnostic signal is steadily on (high level). If the amplifier detects a fault, the signal provides two types of information. First, it shows a slow flash ($2Hz$) or a fast flash ($4Hz$) to indicate the severity of the fault. Slow-flash faults are self-clearing; as soon as the fault is corrected, the amplifier will operate normally. Fast-flash faults are considered to be more serious in nature and a manual reset must be done to resume the operation after the fault is corrected. After the severity indication has been active for 5 seconds, the signal flashes a 2-digit fault code continuously until the fault is corrected. Table 6.7 lists all the faults that the amplifier's self-diagnostic system is able to detect. Some faults are to detect the condition of the input signals such as fault (3,3) for input open-circuit fault and fault (2,1) for input short-circuit

6.2. Design of fault-diagnostic low-level control system

fault. Also, the normal operation of the amplifier is monitored such as fault (1,2), fault (1,4), etc. And another group of fault is for output diagnosis like fault (1,1) (short in the motor), fault (1,3), or fault (4,4) which can be interpreted as the motor is overloaded.

Based on the given information of the diagnostic signal, an electronic circuit was designed to monitor the diagnostic signal. This circuit tests whether the signal is solid on for a sufficient interval of time to decide if the amplifier is working normally or not. The logic of the circuit is simple due to we did not intend to recognize the 2-digit fault code coming out when a fault occurs. Whenever the signal goes low, it implies there is a fault. When there a fault, the signal flashes at a frequency of either $2Hz$ or $4Hz$. It means the signal may be on for $0.5s$ and then off for another $0.5s$. An interval of 2 seconds is therefore enough for the on-state duration test. If the signal is on for at least 2 seconds, it is considered to be solid on. The schematic diagram of the circuit is shown in Figure 6.9. An opto-coupler is used to optically isolate the high-voltage input signal from the circuit. When the input signal is off (low), the photo-transistor of the opto-coupler is open, resulting in the base of the NPN transistor being connected to $+5V$ though the resistor R3. Therefore, the collector and the emitter of the NPN transistor are short and the capacitor C1 discharges quickly through the resistor R4 (about $150ms$). The NOT gate will output a high state indicating there is a fault. When the input signal goes back to high level, the photo-transistor of the opto-coupler is close and connecting the base of NPN transistor to the ground. Hence, the collector and the emitter of the NPN transistor are open. The capacitor C1 is charged by the power supply $+5V$ through a $100k\Omega$ potentiometer R2 that can be adjusted to obtain the required time interval to charge the capacitor to the high level. Normally, if the desired interval is 2 seconds, R2 should be about $8k\Omega$. The NOT gate will then invert the voltage over the capacitor and output a low-level, meaning no fault is detected, when the capacitor is fully charged.

Figure 6.9: Schematic diagram of fault diagnostic circuit for Curtis amplifier

With the amplifier being able to detect a wide range of faults including those for the mo-

Table 6.7: Status fault codes of the amplifier

Codes			Explanation
	solid off		no power or defective controller
	solid on		operational, no faults
*	1,1	◇ ◇	current limit fault
*	1,2	◇ ◇◇	EEPROM fault
*	1,3	◇ ◇◇◇	motor voltage fault
*	1,4	◇ ◇◇◇◇	output section fault
	2,1	◇◇ ◇	5kΩ – 0 or throttle wiper input fault
	2,2	◇◇ ◇◇	static return-to-off fault
	2,3	◇◇ ◇◇◇	high pedal disable fault
	3,1	◇◇◇ ◇	emergency reverse wiring fault
*	3,2	◇◇◇ ◇◇	high pedal disable fault for more than 5 sec.
*	3,3	◇◇◇ ◇◇◇	main contactor fault
*	3,4	◇◇◇ ◇◇◇◇	electromagnetic brake driver fault
	4,1	◇◇◇◇ ◇	battery under-voltage
	4,2	◇◇◇◇ ◇◇	battery over-voltage
*	4,3	◇◇◇◇ ◇◇◇	precharge fault
	4,4	◇◇◇◇ ◇◇◇◇	thermal cutback, due to over/under temperature
* = Fast-flash faults			

tor, monitoring the diagnostic signal of the amplifier simplifies dramatically the fault detection and isolation process for one driving motor. Another important component in the driving control loop that also requires much attention is the incremental encoder which is responsible for measuring the current angular position of the motor for the feedback purpose.

Table 6.8: Commands in C++ to read the diagnostic results of driving system

	Commands
Left wheel's amplifier diagnosis	<i>GetSignalStatus(0)&0x40</i>
Right wheel's amplifier diagnosis	<i>GetSignalStatus(1)&0x40</i>
Left wheel's incremental encoder diagnosis	<i>_inp(0x440)&0x01</i>
Right wheel's incremental encoder diagnosis	<i>_inp(0x440)&0x02</i>

The encoder in use belongs to a commonly used class of encoders of what can measure the relative angular position of a rotating axis. This encoder utilizes a transmitter-receiver set to count the opaque lines and thus the angular increment. Multiple transmitter-receiver sets may be arranged to provide multiple counts per line. One common technique is to offset two sets a half line-width apart. This results in four counts per line. This technique of enhancing resolution via out-of-phase signals is known as quadrature. Based on the additional counts, the encoder interface is able to detect the direction of the rotation by checking the phase-shift. The output signals from the incremental encoder on the Cycab vehicles are in the form of digital square waves and generated by an RS422 driver. The RS422 driver provides an ability to transmit the signal over a distance of up to 1.2km. When a fault occurs at the incremental encoder (internal short-circuit, open-circuit, malfunctioning), its output signals will be out of the valid range. A special IC has been developed by MAXIM [133] to detect the faulty encoder. The IC is able to detect several common-mode faults based on the examination of the RS422 signals. The output from this IC is a logical signal indicating where the encoder is working normally (low-state) or at fault (high-state).

The fault diagnostic signals of the amplifier and the encoder are fetched back to the low-level controller. The diagnostic signals of the left and right wheels' amplifier are connected to the two corresponding AxisIn ports, pin 16 and pin 33, of the POSYS motion control board, while the diagnostic signals of the two encoders are sent to digital input ports *DIO* and *DII* of the CPU board. Table 6.8 shows the commands to read the diagnostic signals for driving system.

6.2.6.2 Steering system

Table 6.9: Commands in C++ to read the diagnostic signals of steering system

	Commands
Steering amplifier diagnosis	<i>GetSignalStatus(0)&0x40</i>
Absolute encoder diagnosis	<i>_inp(0x440)&0x04</i>

A steering system is more sophisticated than a driving system as described in Section 6.2.2. Thus, more possible faults can occur, and any of them can make the closed-loop control unstable. The steering system is composed of three parts: electrical part that consists of an amplifier and a DC motor, hydraulic part that includes a hydraulic pump, oil tubes and a cylinder, and a mechanical part composed of several hard-metal links and joints to convert the translation of the piston into changes in angle of the two wheels. In most cases, the hydraulic and mechanical parts are designed in such a way they are reliable and virtually unbreakable. Thus, we only acquire diagnostic signals from the electrical part. Like that of the driving system, the amplifier [126] of the steering system also provides a self-diagnostic signal. This 5V logic signal and the similar one from the absolute encoder [128] are sent to the low-level controller to determine whether there is a fault or not. The signal from the absolute encoder is connected to digital input port *DI2* of the CPU board and that from the amplifier connected to the *AxisIn* signal of the POSYS board. Then the similar commands can be issued to read these signals as display in Table 6.9. Being aware of the range of the steering angle, $\pm 22.5^\circ$ or equivalent to ± 512 absolute encoder units, a fault of the feedback system including the encoder, the Parallel-to-SSI card and the SSI-800 interface card is certain if the reading is out of the valid range. This limit-checking method can be implemented by using software command.

6.2.6.3 Software system

The softwares running on the two controllers could be halted due to unknown reasons. When they are not working properly or not working at all, the whole vehicle system works improperly and could lead to abnormal behaviours. Thus, it would be better to have a way that can automatically reset the controllers back to working condition. The watchdog mechanism is therefore useful.

A watchdog is a device used to protect a system from specific software or hardware failures

that may cause the system to stop responding. The application is first registered with the watchdog device. Once the watchdog is running on your system the application must periodically send information to the watchdog device. If the device doesn't receive this signal within the set period of time it would execute the proper keystrokes to reboot the machine or restart the application. – *www.WebOpedia.com*

The watchdog timers in use are built-in on the CPU boards. There are two options when the watchdog is up: either to reboot the controller or to generate an hardware interrupt request. If the latter is selected, an interrupt handler must be activated beforehand and stay residentially in the CPU's memory to capture the interrupt request and to deal with the faulty situation accordingly. A simpler way to handle software failures, which is applied in our controllers, is to reboot the whole controller and make sure the controller is back to normal after rebooting. When the controllers are reset, all the output logic signals will switch to low level. As a consequence, the brake/motor inhibit control signal will be under full control of the other controller, if it is operational, or the vehicle will be stopped immediately if both controllers are down.

The software activates the watchdog every time it is on. The interval to set is 3 seconds. The software updates to the watchdog every 300ms so that the watchdog will not reboot the controller even if the software fails to report or the updating signal is lost somehow for a few times. The software will run for ever if no faults occur.

6.3 Experiments

The test bed vehicle developed in the previous section is used to verify the proposed controller developed in Chapter 5.

6.3.1 Fault simulation

From the design of the Red Cycab, the fault at a driving system can be done by disabling the amplifier. The command *SetAmpEnable*, which is used to open/close the photo-relay, can also be used to simulate the fault. When the photo-relay is open, the amplifier will immediately switch to inhibit mode and shut down its PWM driver. The inhibit mode is detected by the fault detection and isolation system and then reported to the controller.

For a steering system, there is no such control signal to turn on/off the its amplifier. The amplifier's inhibit signal is tied with all other inhibit signals and controlled by the same signal

that control the brakes. Thus, we can not control individually the inhibit signal to the amplifier. It would be better if a photo-relay, similar to that for a driving system, would have been inserted between the steering amplifier and the motion control board. In that case, the amplifier could be able to detect the high-resistant state of the photo-relay (equivalent to the wire-broken fault). Anyway, we can simulate the fault at the steering system by directly setting the fault-diagnostic result that is fetched to the controller. A value of 0, corresponding to the fault indicator from the steering system, is reported to the controller instead of actual signal from the fault detection and isolation system. The controller should accordingly issue the zero steering rate command to the steering system, which results in no change in steering angle.

6.3.2 Experimental results

To be comparable with simulation results, the center of the follower vehicle, Red Cycab, was also chosen as the reference point. Thus, the side-slip angle is

$$\tan \beta = \frac{1}{2}(\tan \gamma_f + \tan \gamma_r)$$

The initial steering angles were always zero since we wish the vehicle to start with both steering systems working. Thus, when the two steering system were still operational, the side-slip angle, β , was about zero due to the oppose in steering of the front and rear steering angles.

Parameter λ was set to 1. We also investigated the look-ahead tracking situation only, $f = 1$. The look-behind tracking situation is somewhat similar due to the symmetry of the 4WS vehicle structure. Thus, $\gamma = \gamma_f$.

Parameters l , p_f and p_r were chosen satisfying (5.55) with $d_{\max} = 20m$, $\gamma_{\max} = \frac{\pi}{9}$

$$\begin{cases} 0 < l \leq 20m & \text{(i)} \\ -4.5 < p_r < 0 & \text{(ii)} \\ \max(0, -p_r - 3.5) < p_f < p_r + 4.5 & \text{(iii)} \end{cases} \quad (6.5)$$

Therefore, $p_r = -2$ and $p_f = 2$ were chosen.

The Red Cycab started running with all the components operational. From time to time, one of its subsystems might be simulated to be faulty (Figure 6.10). In detail, the 4WS vehicle started to run with both steering systems operational. It then involved in a right turn where its front and rear wheels steered to negative and positive angles, respectively. At the 74th second, a fault at the front steering system was triggered. The vehicle continued to follow the leader vehicle with the rear steerable wheels working. The vehicle could be seen as working in the

Figure 6.10: Timing of simulated faults

look-behind mode in the car-like vehicle’s point of view. In fact the front wheels were not exactly fixed at zero but a small angle of about 1° . Until second 153, the fault at the front steering system was recovered and the vehicle was back to be a 4WS vehicle. Note that though the two steering angles were opposed each other, their magnitudes were slightly different due to the condition when the front steering system was back to the working condition. About 50 seconds later, it was the rear steering system that went down. The vehicle could only steer its front wheels to keeping the pursuit. It was a bit more difficult for the controller since the rear wheels were fixed at an angle of 2 degrees. Only about 20 seconds before the experiment ended was the rear steering system repaired and back to normal.

The tracking result are shown in Figures 6.11-6.16 with the trajectory shown in Figure 6.11, the vehicular spacing in Figure 6.12, the velocity in Figure 6.13, the steering angles in Figure 6.14, the heading angles in Figure 6.15 and the relative angle in Figure 6.16.

The two vehicle traveled over a distance of 270m within about 300 seconds. When both steering systems were working (during the first 74 seconds), the angle each steering system had to steer to track the steering angle of the leader vehicle was relatively small, roughly less than half the steering angle of the leader vehicle . This again shows the effectiveness of a 4WS vehicle. When there was a fault at one steering system, with the other steering angle successfully tracked that of the leader vehicle (Figure 6.14). The velocity was continuously tracked (Figure 6.13). And the vehicle also managed to follow the leader vehicle roughly at the desired distance though the leader vehicle was often changing its course while the follower vehicle suffered one or more faults (Figure 6.11). With the target performance (2.92), though the leader vehicle was manually driven and its trajectory was a bit rough and difficult to follow

6.3. Experiments

178

Figure 6.11: Trajectory ζ of the vehicles

Figure 6.12: Relative spacing d

Figure 6.13: Velocity v of the vehicles

Figure 6.14: Steering angles γ_f, γ_r of the vehicles

Figure 6.15: Heading angle θ of the vehicles

Figure 6.16: Relative angle ϕ

for some other controllers, our proposed controller is able to drive the follower vehicle to follow smoothly the leader vehicle.

Similarly, Figures 6.17-6.22 show the tracking performance when none of the vehicle's subsystems was at fault. The vehicle ran as a 4WS vehicle with the two steering angle in opposite directions. The results were much better compared to those of the faulty situation. It implies that when the whole system is operational, the proposed controller can utilize all the capacities of the vehicle to provide the best performance.

6.4 Conclusion

A distributed low-level control system has been developed for a four-wheel-driving four-wheel-steering vehicle. The system is in charge of controlling the 4WS vehicle's driving motors and steering systems. Two separate but cooperative controllers are designed and built to control all the motors. The two controllers and the host PC utilize a high-speed Ethernet communication channel to exchange commands, vehicular states and feedbacks. In addition, a simple fault detection and identification system is embedded to immediately detect and isolate any fault that occur on the vehicular system. The faults are also reported to the high-level controller for it to decide what necessary actions should be taken.

Together with the development on designing the low-level control system for the vehicle, a unified platoon tracking controller is derived for both look-ahead and look-behind maneuvers. The controller is designed in such a way that it can continue to work with the presence of faults, especially those at the two steering systems, which result in vehicle configuration changes. Simulations have proved the proposed vehicle model and platoon controller work impressively well under necessary conditions of design parameters l , p_f and p_r . These conditions are useful for system design to secure the existence and stability of the platooning controller.

6.4. Conclusion

180

Figure 6.17: Trajectory ζ of the vehicles

Figure 6.18: Relative spacing d

Figure 6.19: Velocity v of the vehicles

Figure 6.20: Steering angles γ_f, γ_r of the vehicles

Figure 6.21: Heading angle θ of the vehicles

Figure 6.22: Relative angle ϕ

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

Vehicle-following controllers have been successfully developed and implemented on an advanced co-simulation platform as well as practical vehicles. The achievements include integrated vehicle-following controllers for car-like (or front-wheel-steering) and four-wheel-steering vehicles; a cooperative simulation platform for vehicle control algorithms; an experimental vehicle-following system that incorporates a laser scanner-based vehicle-detecting and measuring system, and a complete fault-tolerant vehicle-following system for a four-wheel-steering four-wheel-driving vehicle.

The developed integrated vehicle-following controllers do combine not only the two key controls, driving and steering controls, into one but also two basic tracking situations, the look-ahead tracking and the look-behind tracking. With a proper selection of some design parameters, the controllers can be easily applied for different tracking situations. In addition, the controllers generate driving and steering control inputs simultaneously rather than one-by-one like most of the other vehicle-following controllers. Since the dynamic motions of the vehicle rely strongly on both control inputs, controlling them at the same time provides more accurate and reliable performances of the vehicle. Moreover, the nonlinear controllers are developed based on nonlinear kinematic/dynamic vehicle models that describe closely the actual vehicle dynamics, thus, the accuracy and reliability of the behaviours of the vehicles themselves and the entire vehicle-following system are much improved. Consequently, being able to control both steering angle driving makes the controllers possible to drive and orient the follower vehicle to follow the leader vehicle even when the leader vehicle moves on a complex trajectory with

turns. This is really suitable and useful for applications in industrial environments like seaports, airports and warehouses or in local transportation services such as shuttle bus systems.

The second achievement relates to the development of a cooperative simulation platform that integrates the advantages of two simulation softwares: ADAMS and SIMULINK. SIMULINK provides an advanced environment with useful tools to design a controller. Its programming method using block diagrams makes the controller design easy-to-implement while reflecting clearly and precisely the control algorithm. One drawback of SIMULINK and/or most other mathematics-based simulation softwares is that it is not easy to use them to model a physical system composed of many components different in shapes, sizes, or materials. The system dynamics is normally represented by some equations that omit many hidden dynamic features and effects of the system. ADAMS simulation software helps replace the simplified model by a much more comprehensive one whose characteristics and behaviours are practically copied from those of the actual system. The integration of two softwares provides a simulation platform where the vehicles are modeled closely to the reality and the control algorithms are designed and executed comfortably with powerful mathematic tools provided by SIMULINK. The simulation platform is also able to support not just the vehicle-following controllers but any other vehicle control systems.

The third achievement is about the experimental platform. To enable the implementation of the vehicle-following controller, a sensing system has been designed using a laser scanner mounted on the follower vehicle and reflective tapes pasted on the leader vehicle. This accurate, reliable and fast-updating vehicle-detecting and measuring system can provide the leader vehicle's relative position and orientation as well as the derivatives at a frequency of up to 30Hz, that not many of the vision-based systems have ever achieved, whereas it still maintains high accuracy of the measurements. Moreover, the laser scanner offers a 180° scanning range which is much wider than a CCD camera-based system usually does. The low-level control system of the vehicle has also been modified in order to offer more reliable performance with fault diagnosis capacities which, when integrated with a fault-tolerant controller like the fault-tolerant vehicle-following controller we proposed in this thesis, can provide much more reliable and non-disrupted operations.

The last but definitely not least achievement is a complete fault-tolerant vehicle-following system. To make the vehicle-following system able to continue its operations even when there is a fault at one of its steering or driving subsystems. The operating principles are investigated

and then the vehicle behaviours are modeled with concerns with the occurrences of the possible faults. This new vehicle model is used in the development of a fault-tolerant vehicle-following controller that is able to drive a four-wheel-steering vehicle to follow the leader vehicle with the presence of a fault in the vehicle system. Finally, in order to implement this controller, a fault-diagnostic system is embedded in the reconstructed decentralized low-level control system of the vehicle to continuously provide the working condition of the vehicle's components. The entire vehicle-following system is implemented successfully and the results show the effectiveness of the controller in smoothly and flexibly orienting front and rear steering systems to cope with the fault.

Above all, the thesis has dealt with the vehicle-following maneuver of autonomous vehicles from the methodology proposal and theoretic development to simulation and experimental implementation of the controllers. All relevant issues such as the influence of the design parameter selection and the hands-on works in the implementation have been discussed in detail. In conclusion, the proposed vehicle-following controller and experimental system are ready to be applied in an actual application.

7.2 Recommendations for further research

The works in this thesis have also identified some research issues that can be recommended for future works.

- As being defined, the focus point is always l -meter away from the vehicle. However, it is well known that a constant spacing strategy can guarantee the successful following maneuver of a small fleet of vehicles, i.e. the number of vehicles is small. For a large number of vehicles, the spacing between two consecutive vehicles must rely on the relative velocity as well as the preceding vehicle's and the leader vehicle's velocity. Thus, a more flexible spacing scheme should be applied for a fleet with a large number of vehicles. One of the simple schemes is to use headway time constant where the desired spacing is linear to the velocity.
- The vehicle models in the thesis require an important assumption: the wheel must not slip or skid on the ground. In practice, the wheel may be slightly slipping and/or skidding and the vehicle is still able to operate. The slippage and skidding of the wheel create some

7.2. Recommendations for further research

184

kind of model errors for the controllers developed in the thesis. Considering these errors with a more accurate model will definitely improve the performance.

- The following-vehicle controllers proposed in this thesis do not consider road conditions such as road bounds or flatness. Road bounds can be converted into control inputs limitation, for example by using "Artificial Potential Field" method. Together with their physical limitations they narrow the range of feasible trajectories for the vehicle. Studying and developing control algorithms with regards to saturation of control inputs is always a challenging issue that in this thesis we only can solve partially. Furthermore, the actuator dynamics had better be integrated in the controller to achieve even more accurate and robust controls. The controller may therefore become very complex.
- The proposed controllers are designed for two-vehicle scenarios. It may be also applicable for a small group of few vehicles working in an industrial environment. For a large platoon of vehicles, the issue of string stability becomes important. It was argued in the literature review that to guarantee the platoon's string stability, information of the leader vehicle must be transmitted to all follower vehicles in the platoon over a wireless communication channel. A new vehicle-following controller is required that takes into account this additional information.
- The fault detection and isolation developed in this thesis is a simple one. A more sophisticated fault management system is needed to oversee the entire vehicle health. In turn, the fault-tolerant controller must be able to accommodate these complex faults to offer optimal controls and still prevent the faults from propagation. In addition, the four-wheel-steering four-wheel-driving vehicle in practice is able to operate if both of its steering systems are down and the non-skidding constraint is violated. The four powerful motors mounted the wheels can operate in differential-driving mode to steer the vehicle. A new vehicle model concerning side-slip angles at the wheels is required and the control algorithms utilizing this model will be able to work even when no steering systems are operational.

Author's Publications

- [1] Danwei Wang, Guangyan Xu, and Minhtuan Pham. Look-ahead/behind control of a car-like vehicle. In *Proc. Robotics and Mechatronics Congress*, pages 101–106, Singapore, June 2001.
- [2] Bo He, Danwei Wang, Minhtuan Pham, and Tieniu Yu. Position and orientation estimation with high accuracy for a car-like vehicle. In *Proc. IEEE 5th International Conference on Intelligent Transportation Systems*, pages 528–533, Singapore, September 2002.
- [3] Minhtuan Pham and Danwei Wang. Dynamics-based full-state tracking for a car-like mobile robot. In *Proc. 7th International Conference on Conference on Control, Automation, Robotics and Vision (ICARCV 2002)*, pages 752–756, Singapore, December 2002.
- [4] Danwei Wang, Minhtuan Pham, and Cat T. Pham. Simulation study of vehicle platooning maneuvers with full-state tracking control. In *Proc. International Conference on High Performance Scientific Computing: Modelling, Simulations and Optimization of Complex Processes*, Hanoi, Vietnam, March 2003.
- [5] Danwei Wang, Minhtuan Pham, and Cat T. Pham. Simulation study of vehicle platooning maneuvers with full-state tracking control. In H. G. Bock, E. Kostina, H. X. Phu, and R. Rannacher, editors, *Modeling, Simulation and Optimization of Complex Processes*, pages 539–548. Springer, Germany, 2004.
- [6] Minhtuan Pham and Danwei Wang. A unified nonlinear controller for a platoon of car-like vehicles. In *Proc. American Control Conference (ACC2004)*, volume 3, pages 2350–2355, Boston, Massachusetts, USA, June 2004.
- [7] Danwei Wang, Minhtuan Pham, and Cat T. Pham. A vehicle-following controller with fault tolerant capabilities. In *Proc. 8th International Conference on Mechatronics Technology (ICMT2004)*, pages 245–251, Hanoi, Vietnam, November 2004.

- [8] Danwei Wang, ChangBoon Low, Bo He, and Minhtuan Pham. Accurate positioning for real-time control purpose: Integration of GPS, NAV200 and encoder data. In *Proc. 8th International Conference on Conference on Control, Automation, Robotics and Vision (ICARCV 2004)*, pages 161–166, Kunming, China, December 2004.
- [9] Danwei Wang and Minhtuan Pham. A platooning controller robust to vehicular faults. In *Proc. 8th International Conference on Conference on Control, Automation, Robotics and Vision (ICARCV 2004)*, pages 999–1004, Kunming, China, December 2004.
- [10] Danwei Wang and Minhtuan Pham. *Autonomous Mobile Robots: Sensing, Control, Decision-Making, and Applications*, chapter Unified Control Design for Autonomous Car-Like Vehicle Tracking Maneuvers, page to be published. CRC Press, 2005.
- [11] Danwei Wang, Minhtuan Pham, ChangBoon Low, JingBing Zhang, Ming Luo, DanHong Zhang, and YiZhi Zhao. State-of-art fault diagnosis/prognosis for wheeled mobile robots. In *Proc. 31st Annual Conference of the IEEE Industrial Electronics Society*, page to be presented, Raleigh (NC), USA, November 2005.

Bibliography

- [1] Charles Thorpe. The CMU rover and the FIDO vision navigation system. In *Proc. 3rd International Symposium on Unmanned Untethered Submersible Technology*, volume 3, pages 103–115, June 1983.
- [2] Roger W. Brockett. Asymptotic stability and feedback stabilization. In R. S. Millman R. W. Brockett and H. J. Sussmann, editors, *Differential Geometric Control Theory*, pages 181–191. Birkhauser, Boston, 1983.
- [3] J. B. Pomet, B. Thuilot, G. Bastin, and G. Campion. A hybrid strategy for the feedback stabilization of nonholonomic mobile robots. In *Proc. 1992 IEEE International Conference on Robotics and Automation*, volume 1, pages 129–134, Nice, France, May 1992.
- [4] Richard M. Murray and S. Shankar Sastry. Nonholonomic motion planning: Steering using sinusoids. *IEEE Transactions on Automatic Control*, 38(5):700–716, 1993.
- [5] Ole Jakob Sørдалen and Olav Egeland. Exponential stabilization of nonholonomic chained systems. *IEEE Transactions on Automatic Control*, 40(1):35–49, January 1995.
- [6] Claude Samson. Control of chained systems application to path following and time-varying point-stabilization of mobile robots. *IEEE Transactions on Automatic Control*, 40(1):64–77, January 1995.
- [7] Jin-Oh Kim and Predeep K. Khosla. Real-time obstacle avoidance using harmonic potential functions. *IEEE Transactions on Robotics and Automation*, 8(3):338–349, June 1992.
- [8] Elon Rimon and Daniel E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, October 1992.

- [9] Jürgen Guldner and Vadim I. Utkin. Sliding mode control for gradient tracking and robot navigation using artificial potential fields. *IEEE Transactions on Robotics and Automation*, 11(2):247–254, April 1995.
- [10] E. D. Dickmanns and A. Zapp. Autonomous high speed road vehicle guidance by computer vision. In *Proc. 10th IFAC World Congress*, pages 221–226, Muchen, Denmark, 1987.
- [11] M. Sampei, T. Tamura, T. Itoh, and M. Nakamichi. Path tracking control of trailer-like mobile robot. In *1991 IEEE/RSJ International Workshop on Intelligent Robots and Systems*, pages 193–198, 1991.
- [12] R. M. DeSantis. Path-tracking for car-like robots with single and double steering. *IEEE Transactions on Vehicular Technology*, 44(2):366–377, May 1995.
- [13] L. Panizza and R. Frezza. Paths of bounded curvature with minimal number of maneuvers. In *Proc. IEEE Intelligent Vehicles Symposium 2000 (IV'2000)*, pages 204–209, Dearborn, MI, USA, October 2000.
- [14] Thierry Fraichard and Alexis Scheuer. From Reeds and Shepp's to continuous-curvature paths. *IEEE Transactions on Robotics and Automation*, 20(6):1025–1035, December 2004.
- [15] Alexander Kirchner and Christian Ameling. Integrated obstacle and road tracking using a laser scanner. In *Proc. IEEE Intelligent Vehicles Symposium 2000*, pages 675–681, Dearborn (MI), USA, October 2000.
- [16] Keith A. Redmill, Srinivasa Upadhyaya, Ashok Krishnamurthy, and Ümit Özgüner. A lane tracking system for intelligent vehicle applications. In *Proc. 2001 IEEE Intelligent Transportation System Conference*, pages 273–279, Oakland (CA), USA, August 2001.
- [17] A. DeLuca, G. Orilo, and C. Simson. *Robot Motion: Planning and Control*, chapter Feedback Control of a Nonholonomic Car-Like Robot, pages 171–254. Springer, 1998.
- [18] Claude Samson. Path following and time-varying stabilization of a wheeled mobile robots. In *Proc. 2nd International Conference on Automation, Robotics, and Computer Vision (ICARCV92)*, volume 3, pages RO–13.1.1 – RO3.1.5, Singapore, September 1992.

- [19] G. Walsh, D. Tilbury, S. Sastry, R. Murray, and J. P. Laumond. Stabilization of trajectories for systems with nonholonomic constraints. *IEEE Transactions on Automatic Control*, 39(1):216–222, January 1994.
- [20] Luca Caracciolo, Alessandro de Luca, and Stefano Iannitti. Trajectory tracking control of a four-wheel differentially driven mobile robot. In *Proc. IEEE International Conference on Robotics and Automation*, volume 4, pages 2632–2638, Detroit, Michigan, USA, May 1999.
- [21] Jorge Hermosillo and Sepanta Sekhavat. Feedback control of a bi-steerable car using flatness application to trajectory tracking. In *Proc. American Control Conference 2003 (ACC2003)*, volume 4, pages 3567–3572, Denver, Colorado, USA, June 2003.
- [22] Danwei Wang and Guangyan Xu. Full state tracking and internal dynamics of nonholonomic wheeled mobile robots. In *Proc. American Control Conference (ACC2000)*, pages 3274–3277, Chicago (IL), USA, June 2000.
- [23] Danwei Wang and Guangyan Xu. Full state tracking and internal dynamics of nonholonomic wheeled mobile robots. *IEEE/ASME Transaction on Mechatronics, Focused Section on Advances in Robot Dynamics and Control*, 8(2):203–214, 2003.
- [24] Pravin Varaiya. Smart cars on smart roads: problems of control. *IEEE Transactions on Automatic Control*, 38(2):195–207, February 1993.
- [25] Almut Hochstädter and Michael Cremer. Investigating the potential of convoy driving for congestion dispersion. In *Proc. IEEE Conference on Intelligent Transportation System 1997 (ITSC97)*, volume 6, pages 735–740, Boston, USA, November 1997.
- [26] Pete Seiler, Aniruddha Pant, and Karl Hedrick. Disturbance propagation in vehicle strings. *IEEE Transactions on Automatic Control*, 49(10):1835–1841, October 2004.
- [27] W. S. Levine and M. Athans. On the optimal error regulation of a string of moving vehicles. *IEEE Transactions on Automatic Control*, 11(3):355–361, July 1966.
- [28] Steven E. Shladover, Charles A. Desoer, J. Karl Hedrick, Masayoshi Tomizuka, Jean Walrand, Wei-Bin Zhang, Donn H. McMahon, Huei Peng, Shahab Sheikholeslam, and Nick McKeown. Automatic vehicle control developments in the PATH program. *IEEE Transactions on Vehicular Technology*, 40(1):114–130, February 1991.

- [29] J. Karl Hedrick, Masayoshi Tomizuka, and Pravin Varaiya. Control issues in automated highway systems. *IEEE Control Systems Magazine*, 14(6):21–32, December 1994.
- [30] Steven E. Shladover. Longitudinal control of automotive vehicles in close-formation platoons. *Journal of Dynamic Systems, Measurement, and Control*, 113:231–241, 1991.
- [31] Sean C. Warnick and Armando A. Rodriguez. Longitudinal control of a platoon of vehicles with multiple saturating nonlinearities. In *Proc. American Control Conference 1994 (ACC94)*, pages 403–407, Baltimore, Maryland, USA, June 1994.
- [32] S. Huang and W. Ren. Longitudinal control with time delay in platooning. *IEE Proceedings on Control Theory and Applications*, 145(2):211–217, March 1998.
- [33] G. D. Lee, S. W. Kim, Y. U. Yim, J. H. Jung, S. Y. Oh, and B. S. Kim. Longitudinal and lateral control system development for a platoon of vehicles. In *Proc. 1999 IEEE/IEEE/JSAP International Conference on Intelligent Transportation Systems*, pages 605–610, Tokyo, Japan, October 1999.
- [34] Darbha Swaroop and J. Karl Hedrick. Direct adaptive longitudinal control of vehicle platoons. In *Proc. 33rd Conference on Decision and Control (CDC94)*, volume 1, pages 684–689, Lake Buena Vista, Florida, USA, December 1994.
- [35] Darbha Swaroop, J. Karl Hedrick, and S.B. Choi. Direct adaptive longitudinal control of vehicle platoons. *IEEE Transactions on Vehicular Technology*, 50(1):150–161, January 2001.
- [36] Sei-Bum Choi and J. Karl Hedrick. Vehicle longitudinal control using an adaptive observer for automated highway systems. In *Proc. American Control Conference 1995 (ACC95)*, volume 5, pages 3106–3110, Seattle, Washington, USA, June 1995.
- [37] Hyeoncheol Lee and Masayoshi Tomizuka. Adaptive vehicle traction force control for intelligent vehicle highway systems (IVHSs). *IEEE Transactions on Industrial Electronics*, 50(1):37–47, February 2003.
- [38] A. C. Brown, E. J. Cullen, J. Wu, M. Brackstone, D. J. Gunton, and M. McDonald. Vehicle to vehicle communication outage and its impact on convoy driving. In *Proc. IEEE Intelligent Vehicles Symposium 2000 (IV2000)*, pages 528–533, Dearborn (MI), USA, October 2000.

- [39] Xiangheng Liu, Andrea Goldsmith, Sunider Sonia Mahal, and J. Karl Hedrick. Effects of communication delay on string stability in vehicle platoons. In *Proc. 2001 IEEE Intelligent Transportation Systems Conference*, pages 625–630, Oakland (CA), USA, August 2001.
- [40] Diana Yanakiev and Ioannis Kanellakopoulos. Longitudinal control of heavy-duty vehicles for automated highway systems. In *Proc. American Control Conference 1995 (ACC95)*, volume 5, pages 3096–3100, Seattle, Washington, USA, June 1995.
- [41] Pascal Daviet and Michel Parent. Longitudinal and lateral servoing of vehicles in a platoon. In *Proc. IEEE Intelligent Vehicles Symposium (IV'96)*, pages 41–46, Tokyo, Japan, September 1996.
- [42] Hans Fritz. Longitudinal and lateral control of heavy duty trucks for automated vehicle following in mixed traffic: experimental results from the chauffeur project. In *Proc. 1999 IEEE International Conference on Control Applications*, volume 2, pages 1348–1352, Hawaii, USA, August 1999.
- [43] Darbha Swaroop and K. R. Rajagopal. A review of constant time headway policy for automated vehicle following. In *Proc. 2001 IEEE Intelligent Transportation Systems Conference*, pages 65–69, Oakland (CA), USA, August 2001.
- [44] Lydie Nouvelière and Saïd Mammar. Experimental vehicle longitudinal control using second order sliding modes. In *Proc. American Control Conference 2003 (ACC2003)*, volume 6, pages 4705–4710, Denver, Colorado, USA, June 2003.
- [45] Petros A. Ioannou and C. C. Chien. Autonomous intelligent cruise control. *IEEE Transactions on Vehicular Technology*, 42(4):657–672, November 1993.
- [46] C. C. Chien, Petros Ioannou, and M. C. Lai. Entrainment and vehicle following controllers design for autonomous intelligent vehicles. In *Proc. American Control Conference 1994 (ACC94)*, volume 1, pages 6–10, Baltimore, Maryland, USA, June 1994.
- [47] Diana Yanakiev and Ioannis Kanellakopoulos. Variable time headway for string stability of automated heavy-duty vehicles. In *Proc. 34th IEEE Conference on Decision and Control 1995 (CDC95)*, pages 4077–4081, New Orleans (LA), USA, December 1995.

- [48] Diana Yanakiev and Ioannis Kanellakopoulos. Nonlinear spacing policies for automated heavy-duty vehicles. *IEEE Transactions on Vehicular Technology*, 47(4):1365–1377, November 1998.
- [49] Diana Yanakiev and Ioannis Kanellakopoulos. Longitudinal control of automated CHVs with significant actuator delays. *IEEE Transactions on Vehicular Technology*, 50(5):1289–1297, September 2001.
- [50] Alexander Stotsky, C. C. Chien, and Petros Ioannou. Robust platoon-stable controller design for autonomous intelligent vehicles. In *Proc. 33th IEEE Conference on Decision and Control 1994 (CDC94)*, volume 3, pages 2431–2436, Lake Buena Vista (FL), USA, December 1994.
- [51] Alexander Stotsky and Vadim Harlamov. Variable structure platoon stable controller for autonomous intelligent vehicle. In *Proc. 34th IEEE Conference on Decision and Control 1995 (CDC95)*, volume 4, pages 4075–4076, New Orleans (LA), USA, December 1995.
- [52] Tae Soo No, Kil-To Chong, and Do-Hwan Roh. A Lyapunov function approach to longitudinal control of vehicles in a platoon. In *Proc. 2000 IEEE 51st Vehicular Technology Conference VTC2000-Spring*, volume 1, pages 336–340, Tokyo, Japan, May 2000.
- [53] Tae Soo No, Kil-To Chong, and Do-Hwan Roh. A Lyapunov function approach to longitudinal control of vehicles in a platoon. *IEEE Transactions on Vehicular Technology*, 50(3):116–124, January 2001.
- [54] Thanh M. Nguyen, James W. Sinko, and Randal C. Galijan. Using differential carrier phase GPS to control automated vehicles. In *Proc. IEEE 40th Midwest Symposium on Circuits and Systems*, volume 1, pages 493–496, California, USA, August 1997.
- [55] Jin Wang, Stefan Schroedl, Klaus Mezger, Roland Ortloff, Armin Joos, and Thomas Passegger. Centimeter vehicle positioning and lane keeping. In *Proc. 2003 IEEE Intelligent Transportation Systems*, volume 1, pages 649–654, Shanghai, China, October 2003.
- [56] V. Cerone, A. Chinu, and D. Regruto. A simple vision system for lane keeping. In *Proc. IEEE Conference on Intelligent Transportation System 1997 (ITSC97)*, pages 212–217, Boston (MA), USA, November 1997.

- [57] V. Cerone, A. Chinu, and D. Regruto. Experimental results in vision-based lane keeping for highway vehicles. In *Proc. American Control Conference 2002 (ACC2002)*, volume 2, pages 869–874, Anchorage (AK), USA, May 2002.
- [58] Sadayuki Tsugawa. Vision-based vehicles in japan: machine vision systems and driving control systems. *IEEE Transactions on Industrial Electronics*, 41(4):398–405, August 1994.
- [59] Jeng-Yu Wang and Masayoshi Tomizuka. Robust H_∞ lateral control of heavy-duty vehicles in automated highway system. In *Proc. American Control Conference 1999 (ACC99)*, volume 5, pages 3671–3675, Seattle, Washington, USA, June 1999.
- [60] Guang Lu and Masayoshi Tomizuka. Vehicle lateral control with combined use of a laser scanning radar sensor and rear magnetometers. In *Proc. American Control Conference 2002 (ACC2002)*, volume 5, pages 3702–3707, Anchorage (AK), USA, May 2002.
- [61] Stefen K. Gehrig and Fridtjof J. Stein. A trajectory-based approach for the lateral control of a car following system. In *Proc. IEEE International Conference on Systems, Man, and Cybernetics (SMC'98)*, volume 4, pages 3596–3601, San Diego (CA), USA, October 1998.
- [62] Ryan White and Masayoshi Tomizuka. Autonomous following lateral control of heavy vehicles using laser scanning radar. In *Proc. American Control Conference (ACC2001)*, volume 3, pages 2333–2338, Virginia, USA, June 2001.
- [63] İbrahim Haskara, Cem Hatipoğlu, and Ümit Özgüner. Combined decentralized longitudinal and lateral controller design for truck convoys. In *Proc. IEEE Conference on Intelligent Transportation System (ITSC'97)*, pages 123–128, Boston, Massachusetts, USA, November 1997.
- [64] Rajesh Rajamani, Han-Shue Tan, Boon Kait Law, and Wei-Bin Zhang. Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons. *IEEE Transactions on Control Systems Technology*, 8(4):695–708, July 2000.
- [65] Hung Pham, Karl Hedrick, and Masayoshi Tomizuka. Combined lateral and longitudinal control of vehicles for IVHS. In *Proc. American Control Conference 1994 (ACC94)*, volume 2, pages 1205–1206, Baltimore, Maryland, USA, June 1994.

- [66] Kazuya Tamura and Yoshimi Furukawa. Autonomous vehicle control system of ICVS City Pal: electrical tow-bar function. In *Proc. IEEE Intelligent Vehicles Symposium 2000 (IV2000)*, pages 702–707, Dearborn (MI), USA, October 2000.
- [67] D. N. Godbole, J. Lygeros, E. Singh, and A. E. Lindsey A. Deshpande. Design and verification of communication protocols for degraded modes of operation of AHS. In *Proc. 34th IEEE Conference on Decision and Control, 1995 (CDC '95)*, pages 427–432, New Orleans (LA), USA, December 1995.
- [68] Satyajit Patwardhan, Han-Shue Tan, Jürgen Guldner, and Masayoshi Tomizuka. Lane following during backward driving for front wheel steered. In *Proc. American Control Conference (ACC'97)*, volume 5, pages 3348–3353, New Mexico, USA, June 1997.
- [69] Doh-Hyun Kim and Jun-Ho Oh. Experiments of backward tracking control for trailer system. In *Proc. IEEE International Conference on Robotics and Automation (ICRA'99)*, volume 1, pages 19–22, Detroit, Michigan, USA, May 1999.
- [70] Claudio Altafini, Alberto Speranzon, and Bo Wahlberg. A feedback control scheme for reversing a truck and trailer vehicle. *IEEE Transactions on Robotics and Automation*, 17(6):915–922, December 2001.
- [71] Masami Saeki. Path following control of articulated vehicle by backward driving. In *Proc. IEEE International Conference on Control Applications (CCA 2002)*, volume 1, pages 421–426, Mexico City, Mexico, September 2002.
- [72] A. Astolfi, P. Bolzern, and A. Locatelli. Path-tracking of a tractor-trailer vehicle along rectilinear and circular paths: a Lyapunov-based approach. *IEEE Transactions on Robotics and Automation*, 20(1):154–160, February 2004.
- [73] Mogens Blanke, Michel Kinnaert, Jan Lunze, and Marcel Staroswiecki. *Diagnosis and Fault-Tolerant Control*, chapter 1. Introduction to diagnosis and fault-tolerant control, page 2. ISBN 3-540-01056-4. Springer, 2003.
- [74] Elias N. Skoundrianos and Spyros G. Tzafestas. Finding fault - fault diagnosis on the wheels of a mobile robot using local model neural networks. *IEEE Robotics and Automation Magazine*, 11(3):83–90, September 2004.

- [75] Jacob A. Crossman, Hong Guo, Yi Lu Murphey, and John Cardillo. Automotive signal fault diagnostics - part i: signal fault analysis, signal segmentation, feature extraction and quasi-optimal feature selection. *IEEE Transactions on Vehicular Technology*, 52(4):1063–1075, July 2003.
- [76] Yi Lu Murphey, Jacob A. Crossman, ZhiHang Chen, and John Cardillo. Automotive fault diagnosis - part ii: a distributed agent diagnostic system. *IEEE Transactions on Vehicular Technology*, 52(4):1076–1098, July 2003.
- [77] Helmut Mayer. Comparative diagnosis of tyre pressures. In *Proc. 3rd IEEE Conference on Control Applications, 1994*, volume 1, pages 627–632, Glasgow, Scotland, August 1994.
- [78] L. Le Letty. Parameter estimation in analytical models of automotive and fault diagnosis. In *Proc. IEEE American Control Conference 1995 (ACC'95)*, volume 2, pages 1050–1054, Seattle, Washington, USA, June 1995.
- [79] Marcus Börner, Mina Zele, and Rolf Isermann. Comparison of different fault detection algorithms for active body control components: automotive suspension system. In *Proc. IEEE American Control Conference 2001 (ACC2001)*, volume 1, pages 476–481, Arlington (VA), USA, June 2001.
- [80] Giorgio Rizzoni and Paul S. Min. Detection of sensor failures in automotive engines. *IEEE Transactions on Vehicular Technology*, 40(2):487–500, May 1991.
- [81] Vasanth Krishnaswami and Giorgio Rizzoni. Model based health monitoring of vehicle steering system using sliding mode observers. In *Proc. American Control Conference, 1995*, volume 3, pages 1652–1656, Seattle, Washington, USA, June 1995.
- [82] Vivek Garg and A. E. Lindsey. Fault detection for combined lateral and longitudinal control of vehicles for AHS. In *Proc. 35th IEEE Conference on Decision and Control, 1996 (CDC'96)*, volume 2, pages 2301–2302, Kobe, Japan, December 1996.
- [83] Pascal Bouron and Dominique Meizel. Fault detection and isolation using interval analysis: application to vehicle monitoring. In *Proc. 3rd International Conference on Information Fusion, 2000 (FUSION 2000)*, volume 2, pages WEC5/17 – WEC5/22, Paris, France, July 2000.

- [84] Jingang Yi, Luis Alvarez, Adam Howell, Roberto Horowitz, and Karl Hedrick. A fault management system for longitudinal vehicle control in AHS. In *Proc. American Control Conference (ACC'2000)*, volume 3, pages 1514 – 1518, Illinois, Chicago, USA, June 2000.
- [85] Rajesh Rajamani, Adam S. Howell, Chieh Chen, J.Karl Hedrick, and Masayoshi Tomizuka. A complete fault diagnostic system for automated vehicles operating in a platoon. *IEEE Transactions on Control Systems Technology*, 9(4):553–564, July 2001.
- [86] Domenico Capriglione, Consolatina Liguori, and Antonio Pietrosanto. Analytical redundancy for sensor fault isolation and accommodation in public transportation vehicles. *IEEE Transactions on Instrumentation and Measurement*, 53(4):993–999, August 2004.
- [87] A. E. Lindsey and Pryia B. Viswanath. Design, verification and failure diagnosis of wireless communication protocols for the AHS. In *Proc. IEEE Conference on Intelligent Transportation System, 1997 (ITSC'97)*, pages 52–57, Boston, Massachussets, USA, November 1997.
- [88] Hidayet Tunç Şimşek, Raja Sengupta, Sergio Yovine, and Farokh Eskafi. Fault diagnosis for intra-platoon communications. In *Proc. 38th IEEE Conference on Decision and Control, 1999 (CDC'99)*, volume 4, pages 3520–3525, 1999.
- [89] John Lygeros, Datta N. Godbole, and Mireille Broucke. A fault tolerant control architecture for automated highway systems. *IEEE Transactions on Control Systems Technology*, 8(2):205–219, March 2000.
- [90] Gianluca Antonelli and Stefano Chiaverini. Fault tolerant kinematic control of platoons of autonomous vehicles. In *Proc. 2004 IEEE International Conference on Robotics and Automation (ICRA'04)*, volume 4, pages 3313–3318, New Orleans (LA), USA, April 2004.
- [91] Jeffrey T. Spooner and Kevin M. Passino. Fault tolerant longitudinal and lateral control for automated highway systems. In *Proc. 1995 American Control Conference (ACC'95)*, volume 1, pages 663–667, Seattle, Washington, USA, June 1995.
- [92] Jeffrey T. Spooner and Kevin M. Passino. Fault-tolerant control for automated highway systems. *IEEE Transactions on Vehicular Technology*, 46(3):770–785, August 1997.

- [93] Shashikanth Suryanarayanan, Masayoshi Tomizuka, and Tatsuya Suzuki. Design of simultaneously stabilizing controllers and its application to fault-tolerant lane-keeping controller design for automated vehicles. *IEEE Transactions on Control Systems Technology*, 12(3):329–339, May 2004.
- [94] Guy Campion, Georges Bastin, and Brigitte D’Andréa-Novel. Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 12(1):47–62, February 1996.
- [95] Juergen Ackermann. Robust decoupling of car steering dynamics with arbitrary mass distribution. In *Proc. American Control Conference (ACC’94)*, volume 2, pages 1964–1968, Maryland, USA, June 1994.
- [96] Juergen Ackermann. Robust control prevents car skidding. *IEEE Control Systems Magazine*, 17(3):23–31, June 1997.
- [97] Pierre Rouchon, Michel Fliess, Jean Lévine, and Philippe Martin. Flatness, motion planning and trailer systems. In *Proc. 32nd Conference on Decision and Control (CDC’93)*, pages 2700–2705, Texas, USA, December 1993.
- [98] Michel Fliess, Jean Lévine, Philippe Martin, and Pierre Rouchon. Nonlinear control and Lie-Bäcklund transformations: towards a new differential geometric standpoint. In *Proc. 33rd Conference on Decision and Control (CDC’94)*, volume 1, pages 339–344, Lake Buena Vista, Florida, USA, December 1994.
- [99] Michel Fliess, Jean Lévine, Philippe Martin, and Pierre Rouchon. Flatness and defect of nonlinear systems: Introductory theory and examples. *International Journal of Control*, 61(6):1327–1361, 1995.
- [100] Chun-Yi Su, Yury Stepanenko, and A. A. Goldenberg. Reduced order model and robust control architecture for mechanical systems with nonholonomic Pfaffian constraints. *IEEE Transactions on Systems, Man, and Cybernetics*, 29(3):307–313, October 1999.
- [101] Saïd Mammar and Mariana Netto. Integrated longitudinal and lateral control for vehicle low speed automation. In *Proc. 2004 IEEE International Conference on Control Applications*, volume 1, pages 350–355, September 2004.
- [102] Wolfram Research, Inc., <http://www.wolfram.com/>. *Mathematica*, 2005.

- [103] Mathsoft Engineering & Education Inc., <http://www.mathcad.com/>. *MathCAD*, 2005.
- [104] The MathWorks, Inc., <http://www.mathworks.com/>. *MATLAB*, 2005.
- [105] ABAQUS, Inc., <http://www.hks.com/>. *ABAQUS*, 2005.
- [106] ANSYS Inc., <http://www.ansys.com/>. *ANSYS*, 2005.
- [107] Marc Analysis Research Corp, <http://www.marc.com/>. *MARC*, 2005.
- [108] MSC.Software Corporation, <http://www.mscsoftware.com/>. *MSC/Nastran*, 2005.
- [109] LMS International, <http://www.lmsintl.com/>. *DADS*, 2004.
- [110] INTEC GmbH, Germany, <http://www.simpack.de/>. *SIMPack*, 2004.
- [111] MSC.Software Corporation, <http://www.mscsoftware.com/>. *ADAMS*, 2004.
- [112] Bulent Ozdalyan and Michael V. Blundell. Anti-lock braking system simulation and modelling in adams. In *Proc. International Conference on Simulation '98*, pages 140–144, Jan 1998.
- [113] M. Renault and F. B. Ouezdou. Dynamic simulation of hand-forearm system. In *Proc. 10th IEEE International Workshop on Robot and Human Interactive Communication*, pages 20–25, September 2001.
- [114] Ben Hale-Heighway, C. Murray, S. Douglas, and M. Gilmartin. Multi-body dynamic modelling of commercial vehicles. *Computing and Control Engineering Journal*, 13(1):11–15, February 2002.
- [115] Nasser Kehtarnavaz, Norman C. Griswold, and Juck S. Lee. Visual control of an autonomous vehicle (BART)-the vehicle-following problem. *IEEE Transactions on Vehicular Technology*, 40(3):654–662, August 1991.
- [116] Michael Schwarzingier, Thomas Zielke, Detlev Noll, Michael Brauckmann, and Werner von Seelen. Vision-based car-following: detection, tracking, and identification. In *Proc. Intelligent Vehicles '92 Symposium (IV'92)*, pages 24–29, June 1992.
- [117] Thomas Zielke, Michael Brauckmann, and Werner von Seelen. CARTRACK: computer vision-based car following. In *Proc. IEEE Workshop on Applications of Computer Vision*, pages 156–163, November 1992.

- [118] Nasser Kehtarnavaz, Norman C. Griswold, Kelly Miller, and Paul Lescoe. A trans-portable neural-network approach to autonomous vehicle following. *IEEE Transactions on Vehicular Technology*, 47(2):694–702, May 1998.
- [119] Yasushi Omura, Shigeyuki Funabiki, and Toshihiko Tanaka. A monocular vision-based position sensor using neural networks for automated vehicle following. In *Proc. IEEE 1999 International Conference on Power Electronics and Drive Systems (PEDS'99)*, volume 1, pages 388–393, Hong Kong, July 1999.
- [120] LORD Ingénierie. *CIL 2048 Smart Line Scan Camera*. <http://www.lord-ing.com/eng/cilgb1.htm>, 2004.
- [121] SICK. *LMS200 / LMS211 / LMS220 / LMS221 / LMS291 Laser Measurement Systems - Technical Description*, 2000.
- [122] Axis Control (S). *ACM-631 VIA Eden 667M Embedded PC/104 Board*. <http://www.axiscontrol.com.sg/index.cfm?GPID=46&GBID=2&GBSID=15>, 2004.
- [123] servo-Hallbeck GmbH & Co.KG. *POSYS®8xx: Hardware Manual*, 2004.
- [124] Curtis Instruments Inc. *CurtisPMC 1223/33 1225/35 1227/37 MultiMode Motor Controller: Manual*, 1996.
- [125] IVO GmbH & Co. KG. *Incremental Encoder with hollow shaft GI338*. http://www.ivo.de/pdf_e/gi338.pdf, 2004.
- [126] Copley Controls Corporation. *300 Series Amplifier: User's Guide*, 1996.
- [127] Vetus DEN OUDEN Inc. *Hydraulic steering systems (with axial piston pump)*. http://www.vetus.com/products/hydr_steer/hydraulic_steering-options.cfm.
- [128] IVO GmbH & Co. KG. *Optical Absolute Encoder GA240/GA241 Parallel*. http://www.ivo.de/pdf_e/ga240_241_par.pdf, 2004.
- [129] servo-Hallbeck GmbH & Co.KG. *SSI 800/900 Synchronous Serial Interface for POSYS®800 and POSYS®900 Series Motion Controllers*, 2004.
- [130] Frank Gray. *Pulse Code Communication*, U. S. Patent 2 632 058, Mar 1953.

Bibliography

200

- [131] Paul Horowitz and Winfield Hill. *The art of electronics*. Cambridge University Press, second edition, 1989.
- [132] Finder®S.p.A. *FINDER 553290480040 Miniature General Purpose Relay*. <http://www.findernet.com/en/products/detail.php?codice=553290480040&lang=en&gruppo=gruppo1>, 2004.
- [133] Maxim Integrated Products Inc. *MAX3097E/MAX3098E Triple RS-422/RS-485 Receivers with Fault Detection*. http://www.maxim-ic.com/quick_view2.cfm/qv_pk/2273, 2004.