Title
Advanced Sliding Mode Control and its Application to Autonomous Vehicle at the Limits of Handling

Author
Liao, Yi-Wen

Publication Date
2019

Peer reviewed|Thesis/dissertation

Advanced Sliding Mode Control and its Application to Autonomous Vehicle at the Limits of Handling

by

Yi-Wen Liao

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering - Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Francesco Borrelli, Chair
Professor Andrew Packard
Professor Sergey Levine

Spring 2019

Advanced Sliding Mode Control and its Application to Autonomous Vehicle at the Limits of Handling

Abstract

Advanced Sliding Mode Control and its Application to Autonomous Vehicle at the Limits of Handling

by

Yi-Wen Liao

Doctor of Philosophy in Engineering - Mechanical Engineering

University of California, Berkeley

Professor Francesco Borrelli, Chair

Model uncertainty is one of the most challenging areas of control theory. It is the key reason for using feedback control in order to achieve safety and performance. High-gain feedback is the simplest solution to improve performance. However, it comes at the price of instability and constraints violation. In particular, in sliding mode control, it manifests itself as an undesirable chattering. This dissertation addresses this issue by focusing on the development of advanced sliding mode control and demonstrating its effectiveness for autonomous vehicles under the limits of sensing and driving capabilities. First, a new adaptive sliding mode control (ASMC) strategy is proposed to reduce the control action to its minimum possible value while guaranteeing robust stability. Then, integral sliding model predictive control is introduced by merging the concept of ASMC with a robust model predictive control formulation for nonlinear constrained systems. Motivated by the ever-growing interests in autonomous vehicles, we apply the proposed control techniques to control an autonomous vehicle at the limits of handling. Different extreme driving scenarios such as large side slip angle estimation, path tracking with a large model mismatch and drifting maneuvers are considered and solved with the proposed control algorithms. Successful experimental results support the effectiveness of the developed control method.

To J. Karl Hedrick

There are certain people that walk into your life and shape you into a better person without asking for anything in return. Professor, you are one of them. Thank you for your guidance, wisdom and compassion as I continue growing. You have taught me so many lessons and influenced me so much from the academic to all aspects of life. It is truly an honor for me to be a part of VDLers and to have spent part of my life with you. Your spirit will stay alive in my mind.

Contents

List of Figures

List of Tables

List of Algorithms

Acknowledgments

I have received a great deal of support and assistance from so many people throughout the whole PhD journey. First, I would like to thank Professor Francesco Borrelli. You are an amazing advisor and mentor. Thank you for giving me a lot of freedom to pursue various research topics and, at the same time, kept me in the right direction before I drift far. You motivate me to challenge myself from within to think different. Without having been so blessed to have you, I would never have been so inspired to undertake this curriculum.

Next, I am very grateful to Professor Andrew Packard. Taking your classes is the most enjoyable thing in my PhD journey. You share a true love and excitement for teaching and motivating me to learn more all the times. Your attitude, strength and faith in life has set me an example to follow. You are my hero. I am especially thankful to Professor Sergey Levine for being my dissertation and qualifying exam committee. You are always responsive and supportive whenever I tried to reach you.

I gratefully acknowledge the financial support from the Hyundai Motor, without which the work would not be done. Several valuable discussions with employees enhanced this dissertation from the industrial perspective.

I am very fortunate to have many wonderful friends sharing precious moments with me. Grad school would not have been as much fun without them I made along the way. A heartfelt thanks to Chi Pang, Shuo-Yiin, Menjia, Chen-Yu and Leng-Chun for all the joy and laughter they brought into my life. You help me to stay positive when life gets rough and together we roll through the good and the bad times. I would like to thank Octavio, Minerva, Yoon, Wayne and Michael for providing their life and research experiences to me in the days when I first came to Berkeley.

Furthermore, I wish to give my thanks to all the VDL/MPC lab members, especially Chankyu, Andreas, Chang, Donghan, Fang-Chieh, Yongkeun, Yujia, Ashwin, Ugo, Monimoy, Jon, George, Tony, Jacopo and Jongsang for the help throughout this entire journey. It is always enjoyable to discuss with you on things that were foreign to me. Thanks you for providing me an opportunity to learn things from you. Without you, the CPG trips will be only boring and full of stress in the desert.

Also, I would like to thank my siblings Yu-Hsuan, Chieh-Chih, my brother-in-law Sung-Fu and two little cutest nephews, Ian and Ivan. Thank you for sharing all the life experiences and being there for me all the times. Sometimes a simple photo or video makes me laugh and helps me through the hard times.

Special thanks go to Chung-Yen for your endless love that helps me through each day. Thank you for putting up with my selfishness and temper. Thank you for letting me know you will always be with me no matter what I decide to do. Thank you for all the freedom you have given me and allowing me to delay our life living together by 3 years and patiently waiting for me to achieve my goals. I could not initiate and complete this without you.

Lastly, my sincerest words go to my parents, who have both sacrificed so much for me to be where I am today. Thank you for your unconditional love and care for the last 30 years. I am truly grateful for all the support you have shown me throughout my life.

Chapter 1

Introduction

1.1 Motivation

Model uncertainty is one of the most challenging areas of control theory. It is the key reason for using feedback in the control design that enables the system to adjust its output and meet a desired response. However, even with a feedback structure, there still exist certain limitations that will constrain the performance as not all the design objectives are achievable. For example, we should always consider system's robust stability and take a trade-off between control bandwidth and tracking precision into account while designing the control algorithm. Our goal for robust stability is to design a controller such that the performance or characteristics of the closed-loop system can be maintained for all allowable uncertainties. To deal with this problem, several control methods has been developed in the last three decades in the community.

Among all of the control methods, robust control is one of the powerful approaches that explicitly deals with unconstructed uncertainty and has been studied since 1980s. Note that unconstructed uncertainty is uncertainty about which there is no information available except the upper bound of its magnitude. The control goal is to achieve optimal performance and stability in the presence of modeling errors with a fixed control law. That is to say, once the controller is designed, the control policy will not change over the time and the tracking performance can be guaranteed. Several examples of modern robust control techniques including loop shaping, H_∞ control and sliding mode control (SMC) can be found in the literature [2–4]. The simplicity of their design procedures and comprehensive analysis approaches have attracted a wide variety of applications for robust control [5,6]. Especially for SMC, it is a relatively easily understandable nonlinear control technique that requires a low computational cost with the guarantee of robustness against uncertainties. Although robust control has been remarkably successful in dealing with uncertainty, it still has a drawback that the system usually does not work at optimal status under normal circumstances since the design is typically based on the worst-case condition of uncertainties. An alternative method to address this problem is adaptive control [7, 8]. Adaptive control is a control

technique that enables the control system to modify itself based on some system identification techniques to obtain a model of the process and its environment from input-output experiments. The main differences between adaptive and robust controls are that adaptive control does not require a priori information about the uncertainty and the control law is time-varying. The adaptive algorithm will help minimize the undesirable deviations from the prescribed closed-loop plant behavior and make the control input less aggressive.

Although robust and adaptive controls benefit the overall performance by considering uncertainty in the control design, they are challenging in dealing with input/state constraint satisfactions. On the other hand, model predictive control (MPC) [9] starts to gain more attentions since 1990s as it has the ability to account for complicated system dynamics, forecasted environment information, and system constraints. MPC is a form of control in which the current control action is obtained by solving an on-line constrained optimization problem. At each sampling instant, a finite horizon open-loop optimal control problem is solved for an optimal control sequence and the first control input is applied to the system. Since the open-loop prediction of the model is required in MPC design, the control algorithm that can provide deterministic guarantees on robustness for uncertain systems then becomes important. However, the optimization problem is usually computationally intractable when handling systems with uncertainties. This has driven research in robust model predictive control (RMPC) [10–12] and motivated the work of this dissertation.

As mentioned before, SMC has an advantage of a low computational cost guaranteeing the robustness against uncertainties and MPC has the potential for handling system constraints. Inspired by these complementary characteristics of each controller, this dissertation proposed an idea of merging these two control strategies together. The resulting control scheme is called adaptive discrete-time integral sliding model predictive control (ADISMPC). We first introduce a new adaptive sliding mode control (ASMC) for the purpose of reducing the control action to its minimum possible value and improving the robust performance. Then, expand the work to develop ADISMPC for constrained systems by combining the technique of MPC into an integral sliding mode control strategy (ISMC) with the proposed ASMC algorithm. Due to the sharp growth in autonomous vehicle market, the study of the vehicle at the limits of handling for emergency safety maneuver becomes more and more important. Different extreme driving scenarios such as large side slip angle estimation, path tracking with a large model mismatch and drifting maneuvers) are studied and solved with the proposed control algorithms. Successful experimental results support the effectiveness of the developed control method.

1.2 Main Contributions

This work contributes the developments of various adaptation algorithms in control system and studies its applications to different autonomous driving scenarios. The dissertation first introduces a new adaptive sliding mode control (ASMC) for the purpose of obtaining the minimum possible value of control action to address the well-known problem of chattering

phenomenon in the traditional sliding mode control. It has the advantages of allowing adjusting the control gains dynamically without knowledge of uncertainty bounds while guaranteeing the robustness performance in the same time. The design procedure is simple and easy to implement with nonlinear and MIMO systems. Part of this work was published in [13]. Real applications to autonomous driving for path tracking under extreme scenarios of large model mismatches and drifting maneuvers are conducted to verify the feasibility of the proposed control laws. A successful and good performance in each case study supports the attractiveness of the control algorithm used in a variety of practical applications.

Subsequently, expanding on the work of the proposed adaptive sliding mode control, this dissertation develops an approach of adaptive discrete-time integral sliding model predictive control for a wide class of nonlinear systems and exploits the robust property of a constrained system. It combines the technique of discrete-time model predictive control (MPC) into an integral sliding mode control strategy (ISMC). Under the structure of ISMC, the control signal is composed by two parts; one is generated by MPC and the other one is generated by ASMC. The first part assigns the original system trajectory controlled by MPC as a sliding surface and the second part is used to reject the effect of uncertainties or disturbances. Overall, the control has a better convergent performance compared with traditional robust MPC because of the adaptivity. It requires a simpler approach and a thinner amount in constraint tightening that results more flexibility to cope with any changing condition while doing the on-line optimization. Some relative works are published in [14–16]

Finally, the adaptive approach of vehicle's side slip angle, road bank angle and sensor bias estimations is proposed under the limit of only using the sensors that are available on current commercial vehicles. The algorithm is implemented into real vehicles and verified with several experimental tests. Part of this work has been accepted for publication in the IEEE Transactions on Vehicular Technology.

1.3 Dissertation Outline

The structure of this dissertation is organized as follows. It contains two main subjects that include a theoretical development of advanced sliding mode control (Part I) and its applications to autonomous vehicle driving (Part II).

For Part I, a new methodology of adaptive sliding mode control is introduced in Chapter 2. It shows that the adaptation algorithm based on the concept of the boundary layer enables the determination of an adequate gain with respect to the current uncertainty while keeping the property of finite-time convergence into the sliding band. Later, an extension of combining this algorithm with model predictive control (MPC) into a robust formulation is proposed in Chapter 3. According to the convergent property of adaptive sliding mode control proposed in Chapter 2, the new method simplifies the need of computing the robust invariant set for traditional robust MPC problem. For Part II, the applications of autonomous vehicle under limits of sensing and driving capacities are discussed. Chapter 4 considers the sensor-limited case for the side slip and bank angle estimations using currently

available sensors in production vehicles. Chapter 5 considers the case of path following with large model uncertainties under severe driving scenarios. Finally, Chapter 6 considers the extreme case, drifting, to the adaptive control design for steady state circling and path following.

Part I

Advanced Sliding Mode Control Theory

Chapter 2

Adaptive Sliding Mode Control

2.1 Introduction

Sliding mode control [17] has been recognized as one of the effective nonlinear control methods due to its robustness to uncertainties and its guarantee of finite time convergence. However, the design procedure requires the knowledge of the bound on the uncertainties, which, from a practical point of view, is usually hard to acquire. This results in an uncertainty bound that is often overestimated and hence leads to an undesirable large control gain in the discontinuous sliding term. Consequently, the system will suffer from large magnitude chattering behaviors [18].

To reduce this kind of "zig-zag" motion, several methods have been proposed, which include the boundary layer technique [19] and the "equivalent" control method [20, 21]. The first, proposed by Yao and Tomizuka, approximates the discontinuous signum function by a high-slope saturation function with a desired thickness of the boundary layer. The second, shown by Utkin as well as Tseng and Chen, replaces the discontinuous signum function with a low-pass filter. Although we can get a continuous sliding controller from these methods, the guarantee of global asymptotic stability is sacrificed [22]. In addition, both of these approaches require prior knowledge of the bound on the uncertainties. To avoid this, we can make use of the adaptive control strategy [23] to estimate the unknown parameters. Common methods of estimation include recursive least squares and gradient descent. A more direct way is to derive the update laws from Lyapunov stability theory and analyze the convergence performance. The update laws will use the current information to modify the control input in real time. Because of the advantage of not overestimating the bound on the uncertainties, many adaptation approaches combined with sliding mode control have been developed to tune the sliding gains. The adaptation law proposed in [24] is proportional to the tracking error. It shows that the system will converge to the sliding surface within a finite time. However, the sliding gain will approach infinity since the ideal sliding mode does not exist. In [25], neural networks model the uncertainties of the system and the resulting controller is implemented on a two-tank level control system. The results show that it

can enable a lower switching gain and eliminate the chattering with a thin boundary layer. However, it requires an off-line training process and cannot guarantee stability. Another gain-adaptation algorithm is proposed by using a sliding mode disturbance observer [26], but it has the drawback of requiring the knowledge of uncertainty bounds. The objective of this paper is to provide an adaptive control methodology for a class of nonlinear systems with uncertainties. Note that the uncertainties should be bounded but the prior knowledge of the bound is unknown.

This chapter is organized as follows. First, we review two adaptive sliding mode control strategies proposed in [27] in Section 2.2. Then, a new adaptive sliding mode control is introduced in Section 2.3 and the stability analysis is provided. Several practical implementation issues including each parameter tuning are discussed afterwards in Section 2.4. Finally, the performance of the proposed method is demonstrated with two examples and compared with one of the existing methods.

2.2 Preliminaries

2.2.1 Problem Statement

Consider a nonlinear system given by:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x},t) + \mathbf{l}(\mathbf{x},t)u(t) \\ y(t) = \mathbf{c}(\mathbf{x},t) \end{cases} \qquad \mathbf{x}(0) = \mathbf{x_0},\ t \geq 0 \tag{2.1}$$

where $\mathbf{x}(t) = [x_1(t), x_2(t), ..., x_n(t)]^{\mathrm{T}} \in \mathcal{X} \subset \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}$ is the control input and $y(t) \in \mathbb{R}$ is the system output. $\mathbf{f}(\mathbf{x},t)$ and $\mathbf{l}(\mathbf{x},t)$ are bounded and sufficiently smooth functions which describe the model of the system. Assume that both of them contain unmeasured model uncertainties which satisfy the "matching condition" for conventional sliding mode control [28]. Additionally, to guarantee controllability, $\mathbf{l}(\mathbf{x},t)$ should be $\neq \mathbf{0}$ for all $(\mathbf{x},t) \in \mathcal{X} \times \mathbb{R}^+$.

The common goal of the control problem is to guide the output $y(t)$ along a desired trajectory, $y_d(t)$, or around the origin. To design the sliding mode control, first we define a stable sliding surface $s(\mathbf{x},t)$ [4] with a relative degree equal to 1 with respect to $u(t)$. Then, we obtain the time derivative of $s(\mathbf{x},t)$ as

$$\begin{aligned} \dot{s}(\mathbf{x},t) &= \frac{\partial s(\mathbf{x},t)^T}{\partial \mathbf{x}}\dot{x} + \frac{\partial s(\mathbf{x},t)}{\partial t} \\ &= h(\mathbf{x},t) + g(\mathbf{x},t)u(t) \end{aligned} \tag{2.2}$$

where

$$\begin{aligned} h(\mathbf{x},t) &= \frac{\partial s(\mathbf{x},t)}{\partial t} + \frac{\partial s(\mathbf{x},t)^T}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x},t) \\ g(\mathbf{x},t) &= \frac{\partial s(\mathbf{x},t)^T}{\partial \mathbf{x}}\mathbf{l}(\mathbf{x},t). \end{aligned}$$

To handle the modeling uncertainties and unknown disturbances, we rewrite the model (2.2) with an addictive time-varying function, $\Delta f(\mathbf{x}, t)$:

$$\dot{s}(\mathbf{x}, t) = h(\mathbf{x}, t) + g(\mathbf{x}, t)u(t) + \Delta f(\mathbf{x}, t).$$

The term of $\Delta f(\mathbf{x}, t)$ represents the overall uncertainty of the system and satisfy the following inequality:

$$|\Delta f(\mathbf{x}, t)| \leq \varsigma(\mathbf{x}, t) \leq \mu$$

where μ is the *unknown* upperbound. The objective in this paper is to design a control law which can adapt the time-varying uncertainty, $\varsigma(\mathbf{x}, t)$, in order to reduce the chattering behavior in conventional sliding mode control, but still preserve its own strength in the guarantee of robustness and stability.

2.2.2 Adaptive Sliding Mode Control Review

As is common for sliding mode control, the controller is designed as

$$u = -K\text{sgn}(s) \tag{2.3}$$

where the controller gain, K, is the design parameter which should be greater than or equal to the uncertainty bound, μ.

$$\text{sgn}(s) \doteq \begin{cases} 1 & \text{if} \quad s > 0 \\ -1 & \text{if} \quad s < 0 \\ 0 & \text{if} \quad s = 0 \end{cases}$$

is the discontinuous switching function [29]. As mentioned in the introduction, having a poor estimation on the upperbound, μ, will lead to a larger chattering behavior in the system response. Thus, the main goal of the adaptive sliding mode control is to reduce the magnitude of the controller gain, K, to its minimum admissible value. In other words, the controller gain is not a constant anymore; instead, it can be tuned and modified with time. The method proposed in [27] is based on the use of "equivalent" control: once sliding mode occurs, the uncertainty magnitude can be evaluated and adequately tuned by a low-pass filter:

$$[\text{sgn}(s)]_{\text{eq}} \doteq z \in (0, 1) : \quad \tau\dot{z} + z = \text{sgn}(s(\mathbf{x}, t)). \tag{2.4}$$

To preserve sliding mode and minimize the chattering, the ideal gain $K(t)$ should tend to $\Delta f(t)$ and be slightly greater than $\Delta f(t)$. So, the design idea of the adaptation would be:

$$K(t) \approx |\Delta f(t)|/\alpha, \quad \alpha \in (0, 1)$$

where α is very close to 1. According to this, the minimal possible value of the gain K can be found using the following adaptation algorithm:

$$\dot{K} = \nu K \text{sgn}(\delta) - M[K - K^+]_+ + M[\epsilon - K]_+ \tag{2.5}$$

with

$$\delta \doteq |[\text{sgn}(s(\mathbf{x}, t))]_{\text{eq}}| - \alpha, \quad \alpha \in (0, 1)$$
$$[z]_+ \doteq \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0, \end{cases} \qquad M > \nu K^+, \ K^+ > \mu, \nu > 0.$$

$\epsilon > 0$ is a preselected minimal value of K and K^+ is the uncertainty bound. Once sliding mode with respect to $s(\mathbf{x}, t)$ is established, the adaptation law (2.5) allows the gain K to vary in the range of $[\epsilon, K^+]$ and to be slightly greater than the current uncertainty $\Delta f(t)$. This guarantees an ideal sliding motion.

Another strategy is proposed in [1] which is similar to what we have just introduced above. Instead of using the "equivalent" control method to estimate the boundary of the uncertainties, consider the adaptation law:

$$\dot{K} = \begin{cases} \bar{K}|s(\mathbf{x}, t)|\text{sgn}(|s(\mathbf{x}, t)| - \epsilon) & \text{if } K > \kappa \\ 0 & \text{if } K \leq \kappa \end{cases} \tag{2.6}$$

with $\bar{K} > 0$, $\epsilon > 0$ and a small enough value of $\kappa > 0$ that ensures a positive value of K. According to (2.6), K will decrease if $|s(\mathbf{x}, t)| < \epsilon$. In other words, the gain K will be kept at the smallest level that allows a given certain amount of accuracy which means we can only guarantee semi-global stability of the system. However, the big advantage of this method is that it does not require the knowledge of the uncertainty bound.

2.3 New Adaptive Sliding Mode Control

2.3.1 Motivation and Design Idea

Although the method proposed in [1] has a big advantage for not requiring the knowledge of the uncertainty bound, the adaptation algorithm (2.6) will introduce a discontinuous changing rate in control $u(t)$ at $|s(\mathbf{x}, t)| = \epsilon$, which is not realistic for many actuators. Another problem is that the adaptation law is a linear function of $|s(\mathbf{x}, t)|$, which makes the adaptation rate (increasing and decreasing) not even. In other words, the adaptive rate will become unreasonably large or too small when $s(\mathbf{x}, t)$ is in the reaching phase or converging around zero. As a result, the common problem of the sliding mode control, chattering behaviors, will be easily exhibited in the steady state. To overcome this, we propose another methodology incorporated with the concept of the boundary layer. Using the similar idea

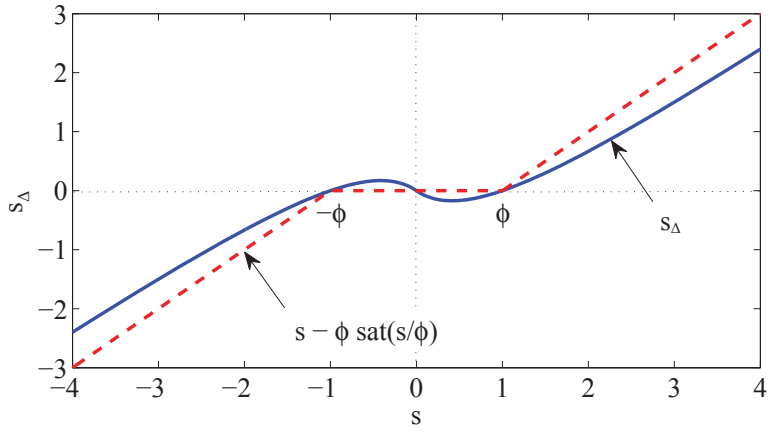


Figure 2.1: Comparison between the standard delta function with linear saturation function $\text{sat}(s/\phi)$ and the proposed delta function s_Δ.

that the gain will increase outside and decrease inside the small region around the sliding surface, we introduce a special delta function which "roughly" denotes the distance of s from the boundary layer. The function is defined as:

$$s_\Delta(\mathbf{x}, t) \doteq s - \frac{2s\phi}{|s| + \phi} \tag{2.7}$$

where $\phi > 0$ is a design parameter indicating the thickness of the boundary layer.

It is worth noting that, in comparison with the classical delta function defined by the saturation function $\text{sat}(s/\phi)$, the new delta function has a similar shape but with nonzero values inside the boundary layer (See Fig. 2.1). There are three main advantages of using $s_\Delta(\mathbf{x}, t)$ to derive the adaptation law. First, instead of blindly tuning a time constant τ of the low-pass filter in (2.4) or the adaptation gain $\bar{K}$ in (2.6), the new adaptation law provides a smooth adaptation process based on the feedback information from $s_\Delta(\mathbf{x}, t)$. Second, unlike the chattering behavior in many adaptive sliding mode control algorithms, it can alleviate the chattering with a simple parameter tuning method. Finally, the stability proof can be done in a clean and relatively easy way.

2.3.2 New Adaptation Control Law

Consider the same problem described in Section 2.2.1 with the sliding surface $s(\mathbf{x}, t)$ defined in the same way as listed in (2.2). The following theorems describe the stability property with the adaptation law based on the delta function we proposed.

Theorem 2.3.1. *Given the system (2.1) implemented with the following feedback control*

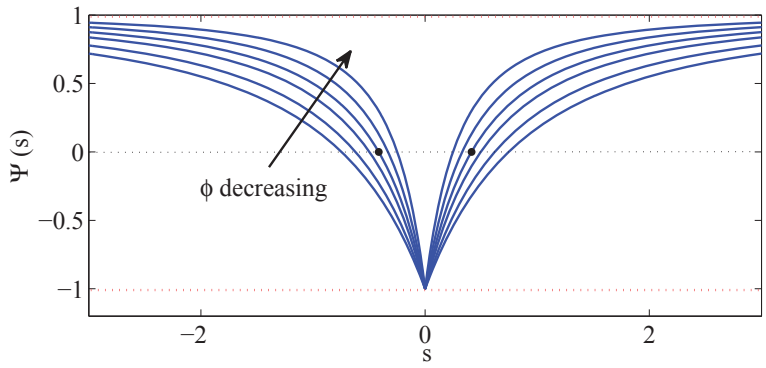


Figure 2.2: The plot of $\Psi(s)$ with varying thicknesses of the boundary layer ϕ. The dotted red lines indicate the upper and lower bounds of 1 and -1.

and adaptive update laws:

$$\begin{aligned} u &= -\frac{1}{g(\mathbf{x},t)}\left[h(\mathbf{x},t) + ks + \hat{\mu} sgn(s)\right] \\ \dot{\hat{\mu}} &= \begin{cases} \frac{1}{\rho}\left[1 - \frac{2\phi^2}{(|s|+\phi)^2}\right] & \text{if } \hat{\mu} \geq 0 \\ 0 & \text{if } \hat{\mu} < 0 \end{cases} \qquad \hat{\mu}(0) = \hat{\mu}_0 \end{aligned} \tag{2.8}$$

where $\rho > 0$ is the adaptation gain, $k > 0$ is the feedback control gain and $\hat{\mu}_0 > 0$ is the initial guess of the sliding gain. The closed-loop state s will approach the boundaries of the domain $\mathcal{S} = \{s \in \mathbb{R},\ |s| \geq \eta\}$ for $\eta = (\sqrt{2}-1)\phi$.

Proof. We first calculate the time derivative of the sliding surface s and s_Δ from (2.1) and (2.8) as follows:

$$\begin{aligned} \dot{s} &= \Delta f - \hat{\mu}\mathrm{sgn}(s) - ks \\ \dot{s}_\Delta &= \dot{s}\left[1 - \frac{2\phi^2}{(|s|+\phi)^2}\right]. \end{aligned}$$

Consider the following Lyapunov function candidate:

$$V(s,\hat{\mu}) = \mathrm{sgn}(s)s_\Delta + \frac{1}{2}\rho(\mu - \hat{\mu})^2. \tag{2.9}$$

We obtain the time derivative of V along the closed-loop system trajectories except $s = 0$ as

$$\begin{aligned} \dot{V}(s,\hat{\mu},t) &= \mathrm{sgn}(s)\dot{s}_\Delta - \rho\dot{\hat{\mu}}(\mu - \hat{\mu}) \\ &= (\Delta f \mathrm{sgn}(s) - \hat{\mu} - k|s|)\Psi(s) - \rho\dot{\hat{\mu}}(\mu - \hat{\mu}) \end{aligned} \tag{2.10}$$

where $\Psi(s)$ is defined by

$$\Psi(s) = 1 - \frac{2\phi^2}{(|s|+\phi)^2}$$

for the sake of simplicity in later expressions. Fig. 2.2 shows a plot of $\Psi(s)$ with varying thicknesses of the boundary layer. As we can see, the function $\Psi(s)$ intersects zero at the points $s = \pm\eta$. Moreover, it is positive when $s \in \mathcal{S}$ and negative outside. First, we consider the case of $\Psi(s) \geq 0$ which is $s \in \mathcal{S}$. $\dot{V}(s, \hat{\mu}, t)$ becomes

$$\begin{aligned}\dot{V}(s, \hat{\mu}, t) &\leq (\mu - \hat{\mu} - k|s|)\Psi(s) - \rho\dot{\hat{\mu}}(\mu - \hat{\mu}) \\ &= (\mu - \hat{\mu})\left[\Psi(s) - \rho\dot{\hat{\mu}}\right] - k|s|\Psi(s). \end{aligned} \tag{2.11}$$

By setting $\dot{\hat{\mu}} = \Psi(s)/\rho$, we can eliminate the first term on the right hand side of equation (2.11) and get the result of

$$\dot{V}(s, \hat{\mu}, t) \leq -k|s|\Psi(s) \leq 0 \quad s \in \mathcal{S}. \tag{2.12}$$

Notice that the result only gives us the update law of $\dot{\hat{\mu}} = \Psi(s)/\rho$ without the condition of $\hat{\mu}$ being non negative. If we substitute $\dot{\hat{\mu}} = 0$ into (2.10), $\dot{V}$ will be indefinite. However, since $\dot{\hat{\mu}} \geq 0$ with $\hat{\mu}_0 > 0$ for $s \in \mathcal{S}$, $\hat{\mu}$ can never be less than zero. In summary, we now have

- $V(s, \hat{\mu})$ is monotonically increasing and bounded from below.
- $\dot{V}(s, \hat{\mu}, t) \leq W(s, \hat{\mu}) = -k|s|\Psi(s) \leq 0$ is negative semidefinite.

for $(s, \hat{\mu}) \in \mathcal{S} \times \mathbb{R}^+$. Note that Barbalat's Lemma is not applicable since we did not make any assumption on the uniformly continuity of the uncertainty. Alternatively, we can apply LaSalle's invariance principle (Theorem 2.2) from Barkana [30] for the nonautonomous system. Based on satisfaction of assumption 1 in Section 2.3 in [30] for the boundedness of uncertainty (i.e. $|\Delta f| \leq \mu$), we can conclude that all system trajectories are bounded and contained within the domain $\Omega = \{(s, \hat{\mu}) \in \mathcal{S} |\ k|s|\Psi(s) = 0\}$ which implies

$$(s, \hat{\mu}) \to (\pm\eta,\ \mathbb{R}^+) \tag{2.13}$$

Now switch to the case of $\Psi(s) < 0$ for domain $\mathcal{S}' = \{s \in \mathbb{R},\ |s| < \eta\}$. Substituting $\dot{\hat{\mu}} = \Psi(s)/\rho$ into (2.10), we have:

$$\begin{aligned}\dot{V}(s, \hat{\mu}, t) &= (\Delta f \mathrm{sgn}(s) - k|s| - \mu)\Psi(s) \\ &\leq -(2\mu + k|s|)\Psi(s) \leq 2\mu + k|s| > 0\end{aligned}$$

where $\dot{V}$ is indefinite in the domain $\mathcal{S}'\backslash\{0\}$ and undefined at $s = 0$. Here, we cannot make any statement when $s \in \mathcal{S}'$. However, based on the result in (2.13), we can know that s will approach $|s| = \eta$ when it is in $\mathcal{S}$. □

2.3.3 Convergence and Stability Analysis

In Section 2.3, Theorem 2.1, we only prove the convergence of s to the boundary of $\mathcal{S}$ whenever $s \in \mathcal{S}$. However, there is no clear stability conclusion that can be drawn with respect to $\hat{\mu}$. Since $\Psi(s)$ is always positive in $\mathcal{S}$, it is possible that $\hat{\mu} \to \infty$ if $|s|$ never reaches the boundary within finite time. In this section, we show that $s(t)$ will reach the boundary of $\mathcal{S}$ with a finite $\hat{\mu}$ in finite time. Moreover, we can guarantee the trajectories of s and $\hat{\mu}$ are bounded in steady state.

Theorem 2.3.2. *Given the system (2.1) implemented with (2.8) with initial conditions* ($s_0 \neq 0, \hat{\mu}_0 > 0$) *satisfying:*

$$|s_0| + \frac{1}{k}\,\hat{\mu}_0 = V_0' > \sigma/k, \qquad \sigma = \mu + \frac{1}{k\rho} \tag{2.14}$$

there exists a finite time T such that

$$|s(t)| + \frac{1}{k}\,\hat{\mu}(t) \leq b, \qquad \forall t \geq T = \frac{1}{k} ln \frac{V_0' - \sigma/k}{b - \sigma/k}$$

where b is any number such that $\sigma/k < b < V_0'$.

Proof. Select another Lyapunov candidate:

$$V'(t) = |s(t)| + \frac{1}{k}\,\hat{\mu}(t)$$

which is locally Lipschitz at $s = 0$ and $\hat{\mu} = 0$. Since $V'(t)$ is not differentiable everywhere, the upper right Dini derivative, $D^+V'(t)$, is introduced [31]. By the assumptions of $\mathbf{l}(x,t)$ and $\mathbf{f}(x,t)$ in Section II.A, we know that the solution of equations (2.2) and (2.8) exists and is absolutely continuous. Therefore, $D^+V'(t)$ is defined and the upper bound can be derived as:

$$\begin{aligned} D^+V'(t) &= D^+\left[|s(t)| + \frac{1}{k}\hat{\mu}(t)\right] \\ &\leq |\Delta f| - \hat{\mu} - k|s| + \frac{1}{k\rho}\max\{\Psi(s), 0\} \\ &\leq \mu - \hat{\mu} - k|s| + \frac{1}{k\rho} \end{aligned} \tag{2.15}$$

according to the fact that $\Psi(s)$ is always bounded within the range of $[-1, 1]$. Rewrite (2.15) into $D^+V'(t) \leq -kV'(t) + \sigma$. Then, the upper bound of the solution is given by

$$\begin{aligned} V'(t) &\leq e^{-kt}V_0' + \sigma \int_0^t e^{-k(t-\tau)}d\tau \\ \Longrightarrow |s(t)| + \frac{1}{k}\,\hat{\mu}(t) &\leq b, \quad \forall t \geq T = \frac{1}{k}\ln\frac{V_0' - \sigma/k}{b - \sigma/k} \end{aligned} \tag{2.16}$$

where b is any number such that $\sigma/k < b < V_0'$. □

Remark: As stated in Theorem 2.3.1, s is converging to the boundary $|s| = \eta$. From (2.16), we can know that the sum of $|s(t)|$ and $\hat{\mu}(t)$ is uniformly ultimately bounded [29] with ultimate bound b after T. Therefore, we conclude that the $s(t)$ will reach the boundary of $\mathcal{S}$ with a finite $\hat{\mu}$ in finite time. Also, choosing the initial conditions satisfying (2.14) is not an issue in the implementation, since $\Psi(s)$ is always positive in $\mathcal{S}$ and the condition (2.14) will be met eventually for any initial setting of s_0 and $\hat{\mu}_0$.

Theorem 2.3.3. *Given the system (2.1) implemented with the adaptation control law (2.8) the trajectories of s are bounded within $|s(\mathbf{x}, t)| < \delta$ after it first time reaches the domain $\mathcal{S}'$, where*

$$\delta = \sqrt{(2\eta)^2 + \frac{\mu^2}{m}} - \eta \tag{2.17}$$

and m can be any value satisfying the following inequalities:

$$m < \frac{\sqrt{2}}{\rho\phi} \quad \text{and} \quad \mu\sqrt{m} \leq \frac{1}{\rho}\Psi(\eta + \frac{\mu}{\sqrt{m}}). \tag{2.18}$$

Proof. According to the proof of Theorem 2.1 and 2.2, we get the result that s will reach the boundary of $\mathcal{S}$ with a finite $\hat{\mu}$ in finite time. To estimate the overshoot of s after the first time it reaches the domain $\mathcal{S}'$, without loss of generality, consider the scenario when $s_0 = \eta^+$. Then, we choose an affine function to lower bound the original nonlinear adaptation law $\dot{\hat{\mu}} = \Psi(s)/\rho$ between the range $s = (\eta, \eta + \mu/\sqrt{m})$ and set $\hat{\mu}_0 = 0$, $k = 0$ in order to get the worst case response of s. The system dynamics can be written as:

$$\begin{cases} \dot{s} = -\hat{\mu} + \mu \\ \dot{\hat{\mu}} = ms - m\eta \end{cases}. \tag{2.19}$$

This yields

$$\begin{aligned} s(t) &= (s_0 + \eta)\cos(\sqrt{m}t) + \frac{\mu - \hat{\mu}_0}{\sqrt{m}}\sin(\sqrt{m}t) - \eta \\ &\leq \sqrt{(2\eta)^2 + \frac{\mu^2}{m}} - \eta \leq \eta + \frac{\mu}{\sqrt{m}}. \end{aligned} \tag{2.20}$$

With this result, the requirements of m in (2.18) then are set. Because the slope of the adaptation law is equal to $\sqrt{2}/\rho\phi$ at $s = \eta$, the first requirement is set to allow the affine function to lower bound the nonlinear adaptation law between the range $s = (\eta, \eta + \mu/\sqrt{m})$. Then, the second requirement is set to ensure the validity of the dynamics (2.19) within the range we claim. □

2.4 Implementation Issues and Parameter Tuning

In Section 2.3, we have introduced three parameters ϕ, k, ρ for the new adaptation control strategy. Although the semi-global stability has been proven for all of them being positive in continuous-time, an adequate choice between each parameter is still needed for a good performance based on different scenarios. Another problem would be the implementation issue. Since nowadays many control algorithms are implemented using digital computers, an approximated discrete-time controller is commonly applied to the system. An inappropriate way of discretizing the control law may degrade the performance and even cause instability [32]. Therefore, to prevent the divergence of the adaptation gain in our algorithm after discretization, the trade off between the tracking precision and the control bandwidth is discussed.

2.4.1 Practical Implementation Issues

To guarantee the idea of the adaptation process works after discretization (i.e. forward Euler discretization), we need to make sure that s can reach inside the domain $\mathcal{S}'$ with sampling. Consider the case when s is approaching $\pm(\sqrt{2}-1)\phi$. At this moment, the system dynamics can be approximately described as

$$\frac{|\Delta s|}{\Delta t} \approx \hat{\mu} + k(\sqrt{2}-1)\phi. \tag{2.21}$$

Following the basic rule of thumb of allowing the system to sample roughly four times inside the domain $\mathcal{S}'$, we then substitute $\Delta s = \pm 2(\sqrt{2}-1)\phi$ as the total traveling distance across the domain $\mathcal{S}'$ into (2.21). This brings out the requirement for the sampling time

$$t_s \leq \frac{\Delta t}{4} = \frac{(\sqrt{2}-1)\phi}{2(\hat{\mu} + k(\sqrt{2}-1)\phi)}. \tag{2.22}$$

By rearranging (2.22) into

$$0 < \hat{\mu} \approx \mu \leq \frac{(1-2kt_s)(\sqrt{2}-1)\phi}{2t_s}, \tag{2.23}$$

we then obtain the upper bound of the adaptation gain.

2.4.2 On the ϕ-tuning

According to Theorem 2.3.1, the system trajectories will evolve around the boundary of the domain $\mathcal{S}$ (i.e. $\pm(\sqrt{2}-1)\phi$) which implies that ϕ is a design parameter of steady accuracy. Ideally, we can set ϕ as small as possible to have good tracking performance. However, in the real implementation, ϕ is limited by μ, t_s and k

$$\phi \geq \frac{2t_s\mu}{(\sqrt{2}-1)(1-2kt_s)}$$

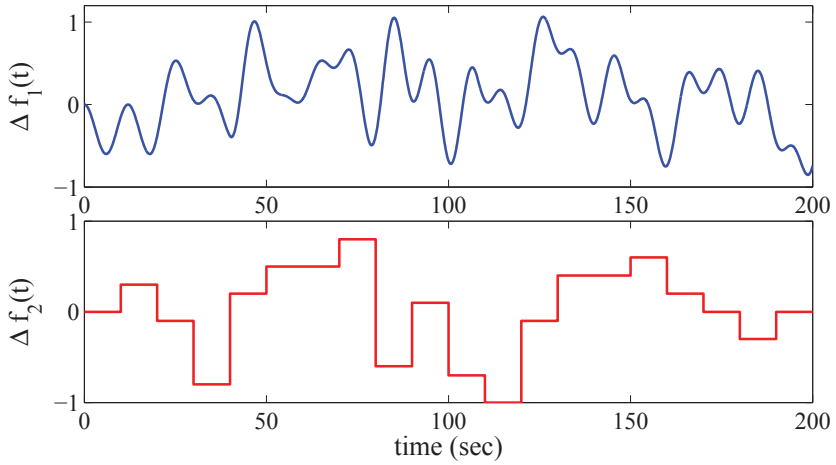


Figure 2.3: The uncertainty functions $\Delta f_1(t)$ and $\Delta f_2(t)$ vs. time

according to (2.23). Since the overall uncertainty, μ, is unknown, it is hard to check if the inequality is satisfied. In a lot of cases, the sampling time is limited by the bandwidth of sensors or actuators. So, if the closed-loop system is not robustly stable after discretization, the only thing we can do is to relax the accuracy requirement. Another thing that should be noticed is that setting ϕ too small will induce a large adaptation gain and cause a high frequency chattering in both state responses and the control input. We can observe that the slope of function $\Psi(s)$ around $\pm(\sqrt{2}-1)\phi$ becomes steeper as ϕ decrease in Fig. 2.2. Therefore, we also need to trade off between the performance and minimizing the chattering by choosing an adequate ϕ.

2.4.3 On the k-tuning

Compared with the standard sliding mode control law (2.3), the new one described in (2.8) has an additional term ks, where k is the design parameter for the feedback gain. Having this additional control term benefits the overall performance since it will help speed up the convergence and smooth out the adaptation process. Therefore, a higher value of k ideally would be desired. However, in practice it should be limited by both actuator/unmodeled dynamics and the boundary thickness of $\mathcal{S}$. A high-gain control can easily excite unmodeled dynamics that could adversely affect the stability. For the second limitation of the boundary thickness, the condition of

$$0 \leq k \leq \frac{1}{\eta}$$

is required, since having too large of a feedback gain may lead to the system trajectories becoming confined inside an even smaller region of $\mathcal{S}$. Under this condition, the adaptation process will be terminated eventually as $\hat{\mu}$ goes to zero.

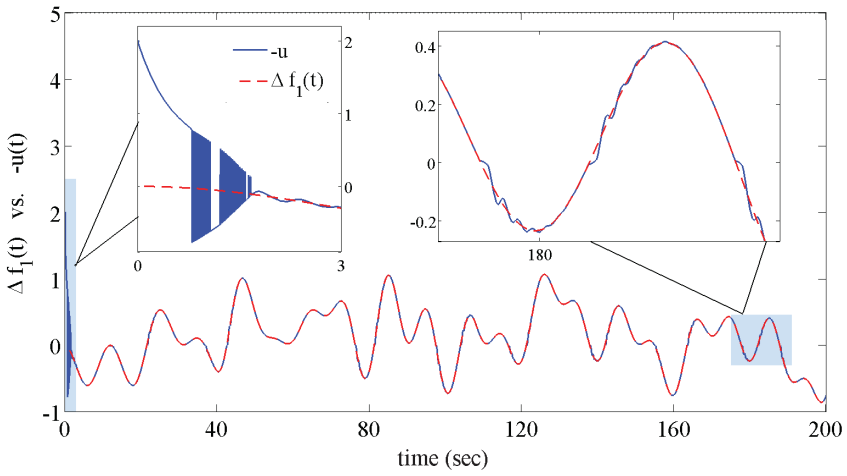


Figure 2.4: The adaptation performance for the smooth uncertainties $\Delta f_1(t)$

2.4.4 On the ρ-tuning

ρ is the adaptation gain which is tuned based on the varying speed of the unknown uncertainties. Choosing a smaller ρ allows a faster learning rate that can improve the adaptation process with high frequency uncertainties. However, we should notice that the smallest value of ρ is limited by the actuation rate in application.

In conclusion, having smaller or larger values in both ϕ and ρ or k may be preferable, but all of them should be carefully chosen with suitable values to effectively avoid high control activity during the reaching phase and the adaptation process.

2.5 Simulation

Two examples will be investigated in this section. First, we apply the adaptive control law given in (2.8) to a simple first-order system for a regulation problem in order to clearly demonstrate the properties of the new method. Then, we again apply the control law to a higher order system with both multiplicative and additive uncertainties to a tracking problem.

2.5.1 Regulation Problem

Consider the following system:

$$\dot{x} = \Delta f(t) + u \tag{2.24}$$

with $\Delta f(t)$ being bounded and unknown. Then, look for two different uncertainties (see in Fig. 2.3) applied to this system: one is a smooth continuous function $\Delta f_1(t)$; the other is a sequence of square signals $\Delta f_2(t)$.

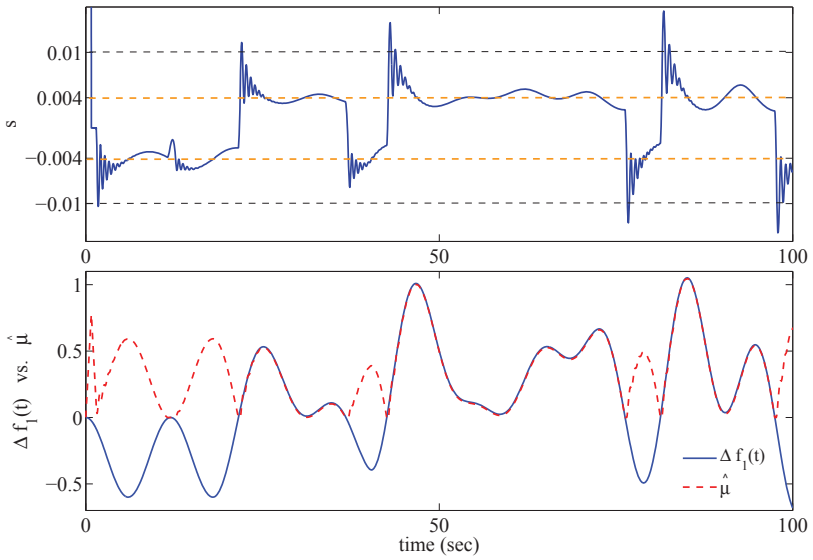


Figure 2.5: The convergence value of the sliding variables $s(t)$ vs. the adaptation gain $\hat{\mu}$ for the smooth uncertainties $\Delta f_1(t)$

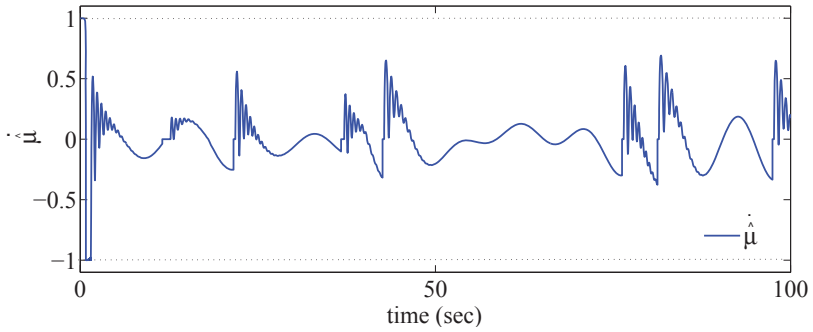


Figure 2.6: The changing rate of the adaptation gain, $\dot{\hat{\mu}}$ for the smooth uncertainties $\Delta f_1(t)$.

To implement the new adaptive control law, we first define the sliding surface as $s = x = 0$. Then, we choose the parameters $\phi = 0.01,\ \rho = 1,\ k = 2,\ \hat{\mu}_0 = 0.001,\ x_0 = 1$ for the case with $\Delta f_1(t)$ uncertainty and $\phi = 0.03,\ \rho = 0.7,\ k = 9,\ \hat{\mu}_0 = 0.001,\ x_0 = 0.1$ for the other case. Since the varying rates of the uncertainties are different, we choose a smaller ρ in order to have a faster learning rate for the case of the square uncertainties. The effect is clearly seen. Fig. 2.4 and Fig. 2.7 demonstrate the adaptation process works well under both low and high frequency uncertainties. The control input follows the external perturbation well particularly for slowly-varying uncertainties. Fig. 2.5 and Fig. 2.8 show the response of state trajectories and the adaptation gain. We can see that in steady state, the sliding variable will evolve around the boundary of $\mathcal{S}$ (dotted orange line) without infinitely high frequency chattering. Since there are two convergence values $\pm(\sqrt{2}-1)\phi$ for the sliding variable, a connection between the convergence value of the sliding variable and the adaptation gain

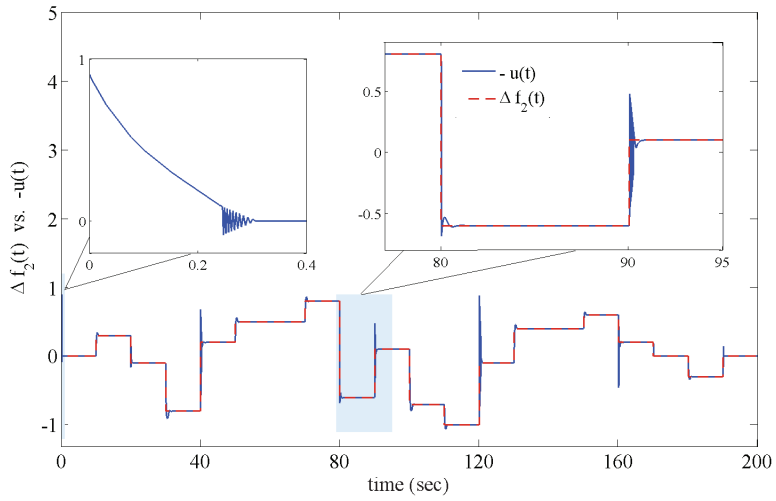


Figure 2.7: The adaptation performance for the square uncertainties $\Delta f_2(t)$

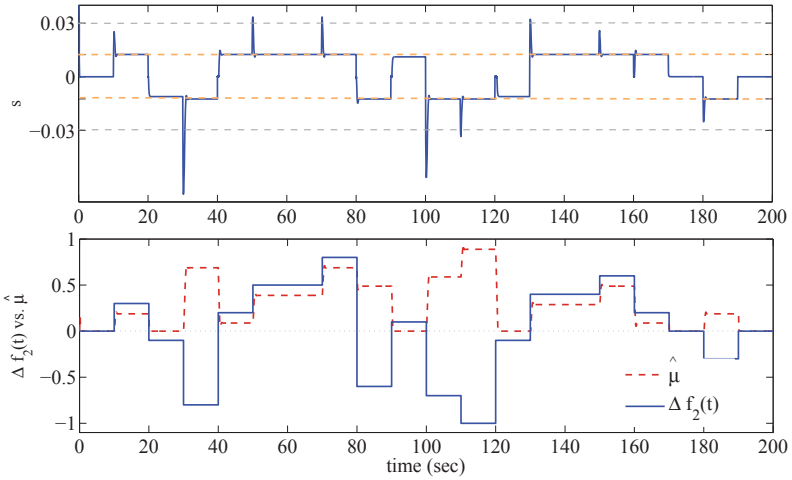


Figure 2.8: The convergence value of the sliding variables $s(t)$ vs. the adaptation gain $\hat{\mu}$ for the square uncertainties $\Delta f_2(t)$.

can be found in the figures. Because we restrict the adaptation gain to be always positive, the sliding variable will converge to the negative value of the boundary layer whenever the adaptation gain has the opposite sign of the current uncertainty.

The learning rates of the adaptation gain are shown in Fig. 2.6 and Fig. 2.9. We can notice that $\dot{\hat{\mu}}$ is always bounded within the region of $[-1/\rho, 1/\rho]$ since $|\Psi(s)| < 1 \ \forall s \in \mathbb{R}$. Actually, it is one of the advantages of the proposed adaptation law compared with other methods. The main difference is that the update law is not a linear feedback law with respect to the sliding variable. The learning rate is limited and can be tuned by ρ. During the reaching phase, the state will converge with the auxiliary feedback term ks instead of speeding up the increasing rate of the sliding gain. This can help us smooth out the

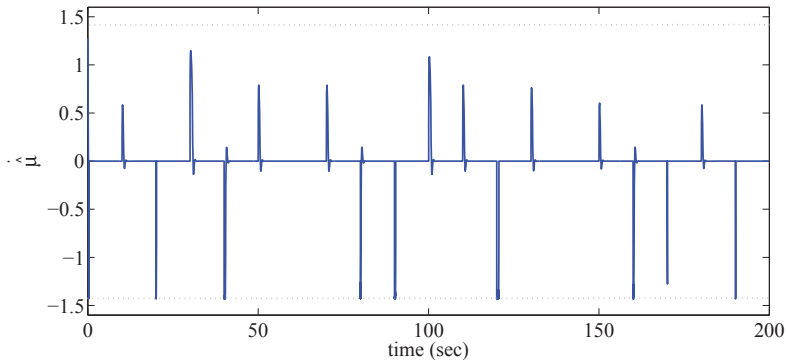


Figure 2.9: The changing rate of the adaptation gain, $\dot{\hat{\mu}}$ for the square uncertainties $\Delta f_2(t)$.

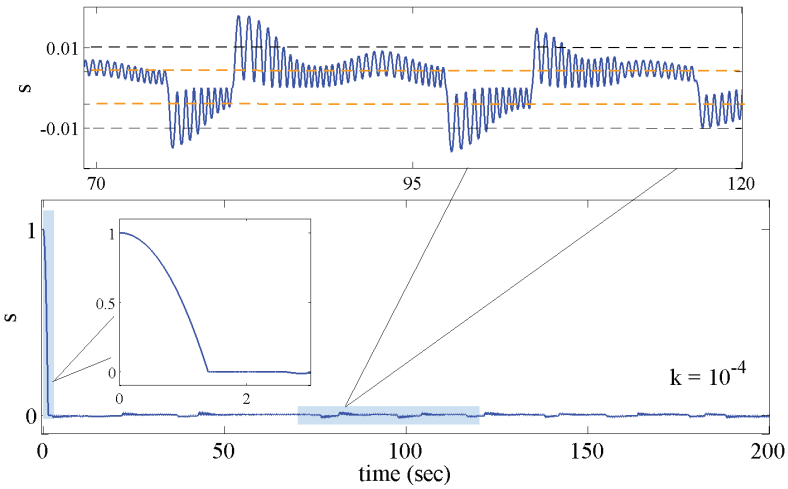


Figure 2.10: The sliding variable trajectories for the smooth uncertainties with a small feedback gain $k = 0.0001$

adaptation process and eliminate the oscillation behavior. We can verify the performance from Fig. 2.10 and Fig. 2.11 for the case of smooth uncertainties. The parameter setting is the same except $k = 0.0001$. The results show that both the state response and the control input signal perform worse when k is small.

2.5.2 Tracking Problem

Next, consider the following nonlinear system:

$$\begin{aligned} \dot{x_1} &= x_2 \\ \dot{x_2} &= [x_1 \Delta x_1(t)]x_2 + \sin(x_1 \Delta x_1(t)) + d_1(t) + u \\ y &= x_1 \end{aligned}$$

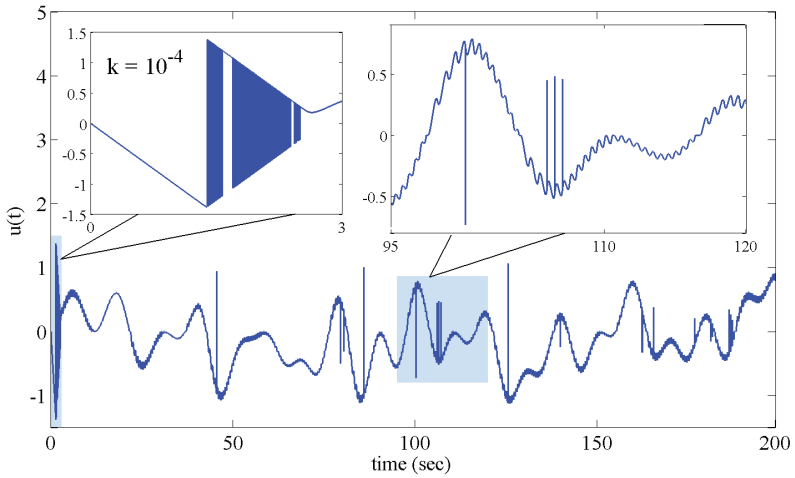


Figure 2.11: The control input for the smooth uncertainties with a small feedback gain $k = 0.0001$

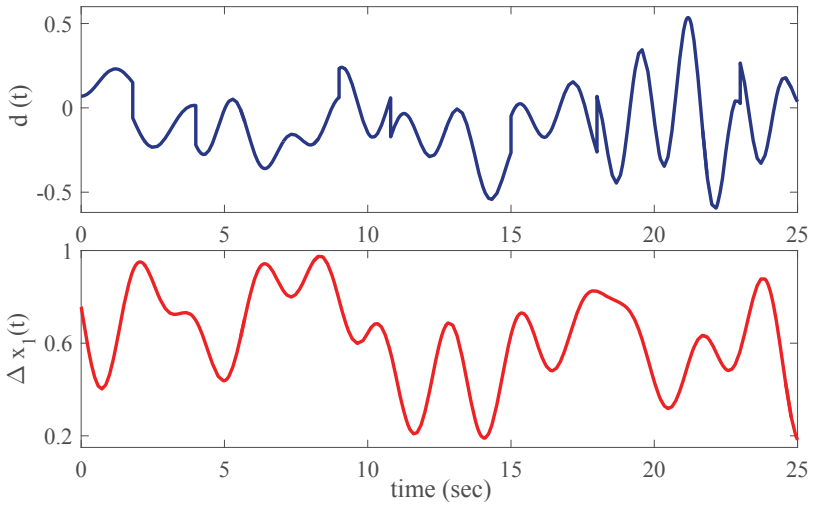


Figure 2.12: The multiplicative and additive uncertainties, $\Delta x_1(t)$ and $d(t)$.

with one multiplicative uncertainty, $\Delta x_1(t)$, and one additive uncertainty, $d(t)$, described in Fig. 2.12. The control objective is to apply the robust control law such that the output, y, tracks a reference signal, $y_{\mathrm{d}} = 3\sin(0.4\pi \mathrm{t})$. We first define $e = y - y_{\mathrm{d}}$ and design a stable sliding surface as

$$s = \dot{e} + \lambda e, \qquad \lambda = 6$$

Then, apply the adaptive control law as

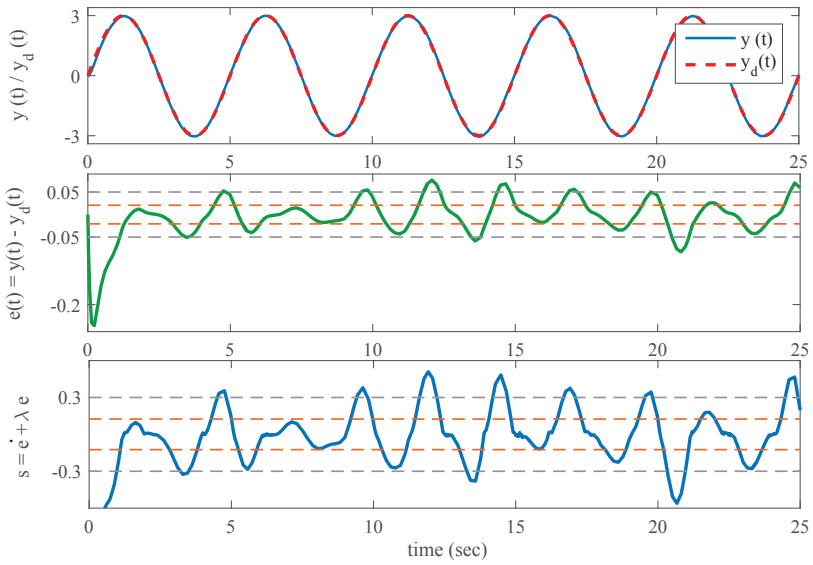


Figure 2.13: The tracking performance for the new proposed method.

$$u = -x_1 x_2 - \sin x_1 + \ddot{y}_{\mathrm{d}} - \lambda(x_2 - \dot{y}_{\mathrm{d}}) - ks - \hat{\mu}\mathrm{sgn}(s)$$

$$\dot{\hat{\mu}} = \begin{cases} \frac{1}{\rho}\left[1 - \frac{2\phi^2}{(|s|+\phi)^2}\right] & \text{if } \hat{\mu} \geq 0 \\ 0 & \text{if } \hat{\mu} < 0 \end{cases} \qquad \hat{\mu}(0) = \hat{\mu}_0.$$

Assume the initial conditions of the states are all zero, $x_1 = x_2 = 0$ and the parameters are set as $\rho = 0.7$, $k = 5$, $\hat{\mu}_0 = 0.001$ and $\phi = 0.3$

Fig. 2.13 demonstrates the tracking performance. As we can see in the second plot in Fig. 2.13, the sliding variable, s, evolves around the boundary of $\mathcal{S}$ after it reaches the domain $\mathcal{S}'$. Additionally, since we have the result of

$$s = 6e \approx (\sqrt{2} - 1)\phi$$

in the steady state for $\dot{e} \approx 0$, we can know that the tracking error, e, will exhibit similar behavior as the sliding variable but with a scale of $1/6$. Fig. 2.14 shows the simulation results of the adaptation gain and the control input. Although the overall uncertainty is unknown, we still can obtain a smooth adaptation process. Moreover, according to the analysis of the connection between the convergence value and the adaptation gain in the first example, we can even roughly reconstruct the overall uncertainty from the plots. The learning rate of the adaptation gain is shown in Fig. 2.15. We can clearly see that $\dot{\hat{\mu}}$ is smooth and always bounded within the range of $[-1/0.7,\ 1/0.7]$.

2.6 Comparison

In this section, we compare the proposed adaptive sliding mode control with the one introduced in [1] by using the regulation example in Section 2.5.1. Although the design concepts

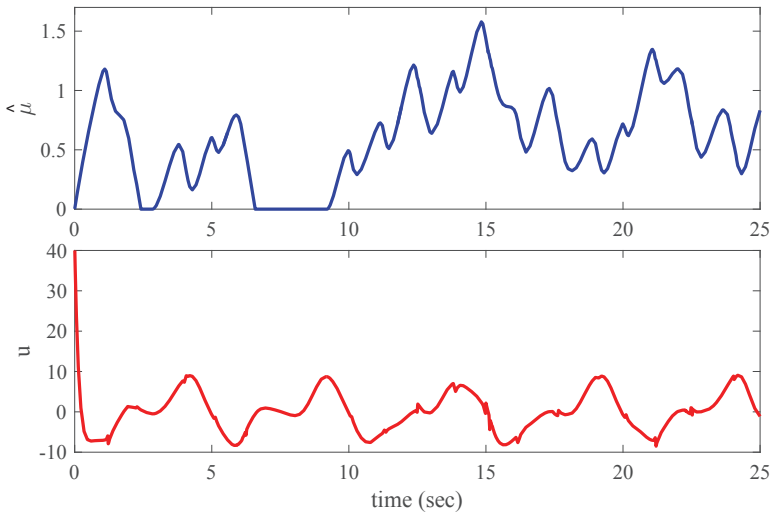


Figure 2.14: The adaptation gain $\hat{\mu}$ and the control input of the tracking problem for the new proposed method.

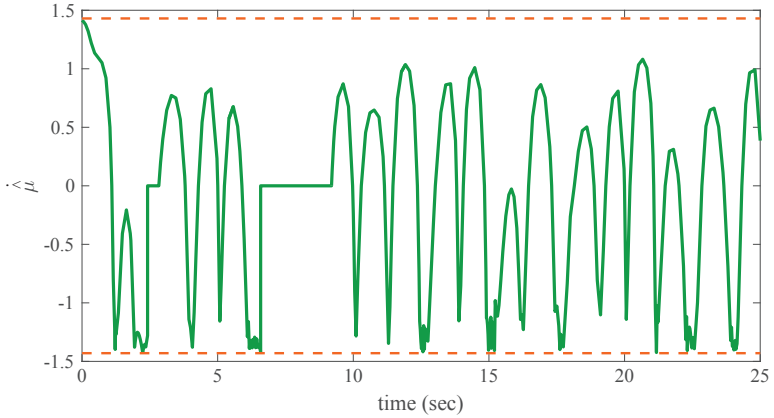


Figure 2.15: The changing rate of the adaptation gain, $\dot{\hat{\mu}}$, of the tracking problem for the new proposed method.

of these two methods are similar, the new one stands out for its smooth adaptation process without a high gain (i.e. $1/\rho$). We implement the adaptation law (2.6) in the case with the continuous uncertainty by setting the parameters $\bar{K} = 3000$ and $\kappa = 0.01$. Moreover, we choose $\epsilon = 0.01(\sqrt{2} - 1)$ in order to have the same convergence standard for the comparison of these two methods. The simulation results in Fig.2.16-2.18 display the closed-loop performance of the adaptive controller proposed in [1]. It appears that undesired chattering behaviors are introduced in both the control input and the sliding variable responses because of the large discontinuous switching rate in the adaptation gain (shown in Fig. 2.18).

One suggestion for the alleviation of the chattering is to set the parameter $\bar{K}$ small, but as stated in Section 2.3, the linear adaptation law with a smaller gain will make the

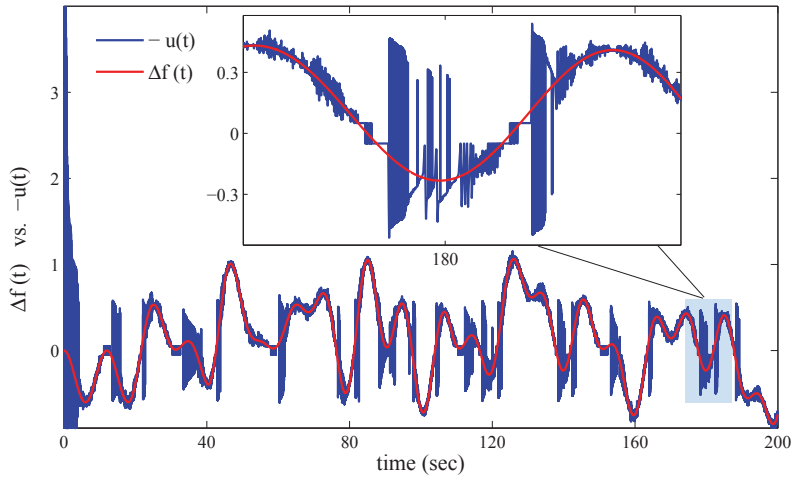


Figure 2.16: The adaptation performance for the method proposed in [1].

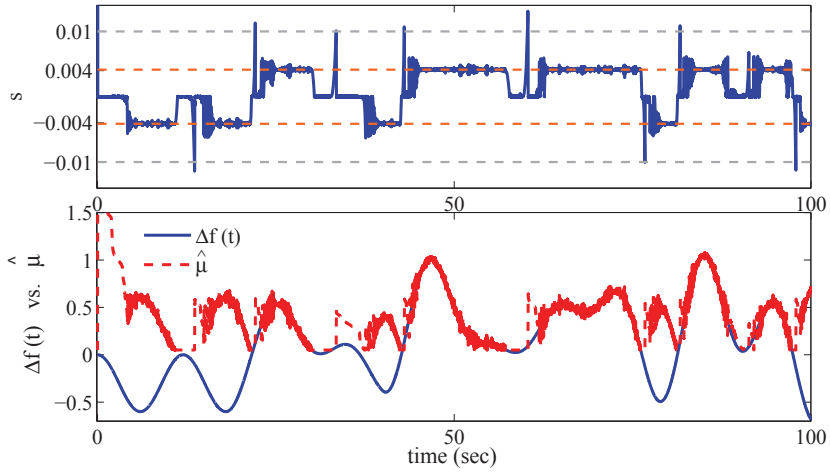


Figure 2.17: The convergence value of the sliding variables $s(t)$ vs. the adaptation gain $\hat{\mu}$ for the method proposed in [1].

decreasing rate even more insignificant inside the domain of $|s(\mathbf{x}, t)| < \epsilon$, which fails to address the problem. Fig. 2.19 and Fig. 2.20 show the control effects with $\bar{K} = 150$. Although the changing rate of the adaptation gain becomes much smaller (Fig. 2.20), the chattering behaviors of the state and control input responses are not suppressed and even made worse with the small $\bar{K}$ (Fig. 2.19).

2.7 Conclusion

This chapter proposed a new methodology of adaptive sliding mode control for a class of uncertain nonlinear systems. The algorithm utilizes the concept of the boundary layer. Based

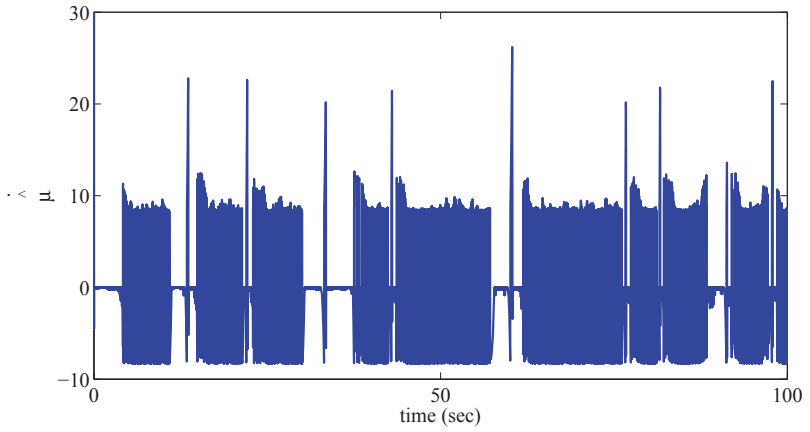


Figure 2.18: The changing rate of the adaptation gain, $\dot{\hat{\mu}}$ for the method proposed in [1].

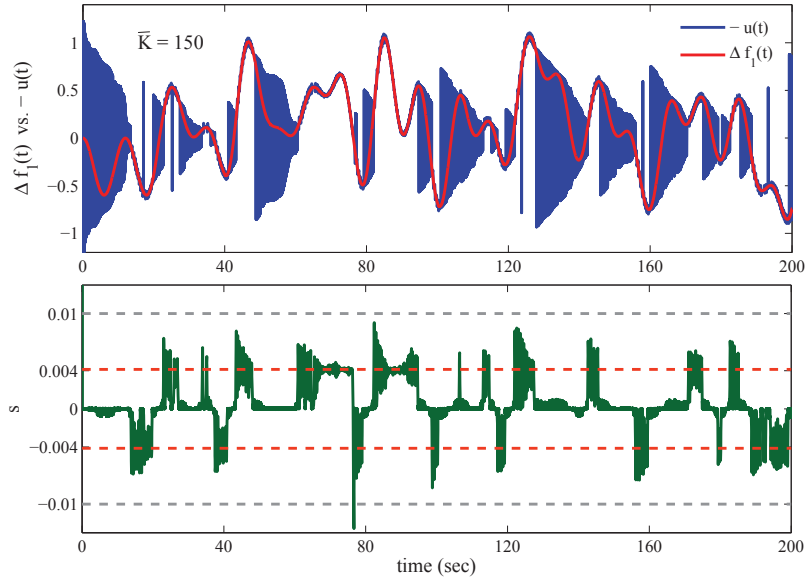


Figure 2.19: The performance of the control input canceling the uncertainty and the response of the $s(t)$ with a smaller $\bar{K}$.

on the property that the system will hover inside and outside around the boundary region of $\mathcal{S}$, the adaptation law is designed such that the sliding gain will decrease and increase accordingly. Numerical examples illustrated the effect of the adaptation process. The process enables the determination of an adequate gain with respect to the current uncertainty. Semi-global stability of the closed-loop system with the adaptation gain is also guaranteed. Overall, this method achieves the minimum possible value of time-varying sliding mode control input and reduces the high-frequency chattering behavior without requiring knowing any knowledge of the uncertainties.

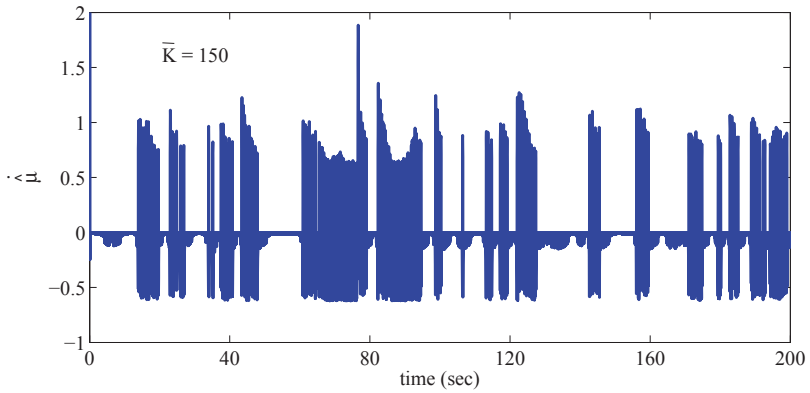


Figure 2.20: The changing rate of the adaptation gain, $\dot{\hat{\mu}}$ for a smaller $\bar{K}$.

Chapter 3

Adaptive Integral Sliding Model Predictive Control

3.1 Introduction

In many industrial applications, most of control systems require robust stability and optimal performance. Research in these areas continues to grow. Over the past few decades, a number of control strategies have been extended to address robustness and adaptation, especially, in model predictive control (MPC) and in sliding mode control (SMC). MPC is a well-known control method for optimizing the performance of constrained systems. However, it becomes computationally demanding when dealing with system uncertainty or unknown disturbances [10]. A number of approaches for robust MPC have been proposed in the literature [33, 34].

One popular approach is min-max MPC [35], where an optimal control problem is solved with respect to the worst case disturbance. This method is effective with systems that have a few number of states, or systems with slow dynamics. It demands a high computational cost for large systems. Another approach is constraint tightening MPC, which remains as computationally complex as the nominal problem by tightening the original constraints. Based on this idea, different tightening policies have been proposed [36, 37]. The drawback of this method is that the tightened sets have to be time-invariant, and thus may be too conservative because of considering the worst case disturbance off-line. To remove some of these restrictions, tube MPC [11, 34] introduces extra degrees of freedom in the controller design by using a sequence of time-varying "tubes" determined from the disturbance invariant set [38]. The disturbance invariant set is defined as the set of all possible state deviations introduced by disturbance with the stable feedback controller. The associated feedback controller introduced in tube MPC can ensure the actual states converging to the nominal states.

On the other hand, compared with MPC, SMC [39] has been recognized as one of the effective robust control strategies for systems with disturbances or uncertainty conditions. Its

main idea is to design a discontinuous control law that leads the system towards a custom-made stable sliding surface, with relative degree equal to one. Unlike robust MPC, SMC requires a low computational cost and has the guarantee of the finite-time convergence. However, SMC fails to address problems of optimal performance and constrained satisfactions. Due to the complementary strengths of MPC and SMC, there are many different approaches of combining these two control strategies. For example, in [40] and [41], the authors use the idea of robustness from discrete-time SMC in a constrained optimal control discrete-time MPC. Instead of regulating the system states directly to zero, the optimal control algorithm regulates the designed sliding surface near to zero. This is done by introducing the custom-made stable sliding surface into the objective function of MPC. Since the sliding surface is designed to be stable, the state trajectory will stay within the sliding band and converge into an area around the origin. Another approach introduced in [12] is to combine the technique of continuous-time MPC into integral SMC (ISMC) [42]. Under the structure of ISMC, the control signal is composed by two parts; one that is generated by MPC and one that is generated by SMC. The first part makes the order of sliding surface equal to the order of the original system and the second part is used to reject the effect of a bounded unknown disturbance. Due to the discrepancy between the nominal and perturbed evolutions of the system, the state and input constraints are tightened beforehand. Hence, the MPC strategy can be implemented into system without considering the disturbance.

Robust MPC has been extensively studied. However, many of the existing robust MPC approaches ([35], [11] and [12] etc.) require iterative computations for the robust invariant set. This may present a challenge for large systems. In this chapter, we introduce a simple approach to tighten the original constraints of the optimal problem without computing the robust invariant set. Based on the idea presented in [12], the robust control strategy in this paper is to modify the concept of discrete-time ISMC [43] into a discrete-time MPC problem. The chapter is organized as follows. We first briefly review over discrete-time ISMC and discrete-time robust MPC. Then, the method of discrete-time integral sliding model predictive control (DISMPC) and adaptive DISMPC are proposed with a simple constraint tightening approach. The properties of robustness and stability are discussed afterwards. Finally, the performances are compared with the method of robust invariant tube MPC with a numerical simulation.

3.2 Preliminaries

3.2.1 Problem Statement

Consider a nonlinear discrete-time system with an affine control input described as follow:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)(\mathbf{u}_k + \mathbf{d}_k) \tag{3.1}$$

where $\mathbf{x}_k \in \mathbb{R}^n$, $\mathbf{u}_k \in \mathbb{R}^m$ are the state and control input vectors at time step k. $\mathbf{f}(\mathbf{x}_k) \in \mathbb{R}^n$ and $\mathbf{g}(\mathbf{x}_k) \in \mathbb{R}^n$ are the state transition and input distribution functions. The vector $\mathbf{d}_k \in \mathbb{R}^m$

is an unknown matched disturbance/uncertainty which is bounded by

$$\mathcal{W} = \{\mathbf{d}_k \in \mathbb{R}^m,\ 0 < \|\mathbf{d}_k\|_\infty \leq \alpha\}.$$

The system (3.1) is subject to the following set of hard constraints:

$$\mathbf{u}_k \in \mathcal{U} \text{ and } \mathbf{x}_k \in \mathcal{X} \tag{3.2}$$

where $\mathcal{U}$, $\mathcal{X}$ are polytopes and contain the origin as an interior point. Assume that the system is locally accessible within the operation region $(\mathbf{x}_k, \mathbf{u}_k) \in \mathcal{X} \times \mathcal{U}$ and $(\mathbf{0}, \mathbf{0})$ is the corresponding desired equilibrium pair for the nominal system:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\mathbf{u}_k. \tag{3.3}$$

To control the system (3.1) and guarantee its robustness under uncertainties, several control approaches are proposed in the literature. Among all of these, an integral sliding mode control (ISMC) and robust model predictive control (RMPC) are the control strategies which have their own strength and weakness. The former one has the capability to handle a system subject to disturbances and uncertainties, but does not consider the problems of optimal performance and constraints. On the other hand, RMPC is a well-known control method for optimizing the performance of constrained systems but requires high computational cost when dealing with robustness [10]. We will briefly discuss each of methods in the following sections in order to motivate the proposed adaptive discrete-time integral sliding model predictive control (ADISMPC).

3.2.2 Discrete-Time Integral Sliding Mode Control (DISMC) Review

Discrete-time sliding mode control (DSMC) method is a mature technique for the robust controller design of uncertain system. The motion of DSMC includes two phases: the reaching phase and the quasi-sliding phase. During the reaching phase, the system is moving in the direction of the sliding surface. After the closed-loop system is driven into the quasi-sliding band, the control law will maintain the states within the quasi-sliding band. The design procedure for DSMC can be divided into two steps:

1. Choose a sliding surface

$$\mathbf{s}_k = C\mathbf{x}_k \in \mathbb{R}^m \tag{3.4}$$

   which has stable dynamics and $C\mathbf{g}(\mathbf{x}_k) \neq 0$.

2. Establish a sliding control term in the control law as [44]:

$$\mathbf{u}_k = -M\text{sgn}(\mathbf{s}_k)$$

   where $M \in \mathbb{R}^{m\times m}$ is defined to be positive definite [42]. This ensures that the closed-loop system is driven towards the sliding surface and stays within the quasi sliding band.

However, in DSMC, we can only guarantee the robustness of the system during the quasi-sliding phase. To improve the control performance, the concept of discrete-time integral sliding control (DISMC) is proposed in [43]. Different from the conventional DSMC, the reaching phase of DISMC is eliminated by choosing the dimension of the sliding surface equal to that of the state space. We can further initialize the system trajectory to the sliding surface. Therefore, the robustness can be guaranteed throughout an entire response of the system. The design procedure for DISMC is divided into two steps:

1. Choose the control law in the form of

$$\mathbf{u}_k = \bar{\mathbf{u}}_k + \mathbf{u}'_k$$

where $\bar{\mathbf{u}}_k$ is the ideal control which can stabilize the nominal system (3.3) and $\mathbf{u}'_k$ is an additional sliding control input designed for disturbance rejection.

2. Design a discrete-time integral sliding surface as

$$\mathbf{s}_k = \mathbf{s}_0(\mathbf{x}_k) + \sigma_k \quad \text{with} \quad \sigma, \mathbf{s} \text{ and } \mathbf{s}_0 \in \mathbb{R}^m \tag{3.5}$$

where the first part of the discrete-time integral sliding surface $\mathbf{s}_0(\mathbf{x}_k)$ is designed as the linear combination of the states, similar to (3.4); while the second part of the discrete-time integral sliding surface σ_k is introduced as a summation term, which is similar to the integral term $\mathbf{z}$ in [42].

3.2.3 Robust Model Predictive Control (RMPC) Revisit

Model predictive control (MPC) is an attractive control strategy for systems with input and state constraints. This strategy has been widely adopted in many industrial applications. The basic idea of MPC is to approximate an infinite-time optimal control problem by a finite horizon one (N-step look-ahead). At each time step k, the optimization is solved in real-time to obtain a sequence of control inputs, $\mathcal{U}_k = \{\mathbf{u}_{0|k}, \mathbf{u}_{1|k}, ..., \mathbf{u}_{N-1|k}\}$. However, only the control input at the current time step, $\bar{\mathbf{u}}_k = \mathbf{u}^*_{0|k}$, is implemented at time k. Define $\mathcal{I} = \{0, 1, ..., N-1\}$, the MPC law is obtained by solving:

$$\begin{aligned}
\min_{\mathcal{U}_k} \quad & \mathbf{x}^T_{N|k} P \mathbf{x}_{N|k} + \sum_{i=0}^{N-1} (\mathbf{x}^T_{i|k} Q \mathbf{x}_{i|k} + \mathbf{u}^T_{i|k} R \mathbf{u}_{i|k}) \\
\text{s.t.} \quad & \mathbf{x}_{i+1|k} = \mathbf{f}(\mathbf{x}_{i|k}) + \mathbf{g}(\mathbf{x}_{i|k})\mathbf{u}_{i|k}, \quad \forall i \in \mathcal{I} \\
& \mathbf{x}_{i|k} \in \mathcal{X}, \quad \forall i \in \mathcal{I} \\
& \mathbf{u}_{i|k} \in \mathcal{U}, \quad \forall i \in \mathcal{I} \\
& \mathbf{x}_{i|N} \in \mathcal{X}_f, \ \ \mathbf{x}_{0|k} = \mathbf{x}_k
\end{aligned} \tag{3.6}$$

where $Q = Q^T \succeq 0$, $P = P^T \succeq 0$ are positive semi-definite and $R = R^T \succ 0$ is positive definite. The subscript $i|k$ is used for representing the prediction steps of time $k + i$ at the

sampling time k. According to Lyapunov stability theory, the closed-loop system (3.3) is asymptotical stable under the condition of setting the terminal constraint, $\mathcal{X}_f$, as a control invariant set of the system (3.1) with constraints (3.2) and the terminal cost, $\mathbf{x}_{N|k}^T P \mathbf{x}_{N|k}$ is a strictly Lyapunov function as shown in Theorem 13.2 in [9] for system (3.1). The closed-loop system with MPC can be expressed as

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\bar{\mathbf{u}}_k \tag{3.7}$$

The above MPC algorithm assumes a perfect system without model uncertainties and external disturbances. To address the robustness, the finite-time constrained optimal control problem (3.6) can be rewritten into the following formulation by taking the worst case scenario into account for every admissible uncertainties/disturbance sequence:

$$\begin{aligned} \min_{\mathcal{U}_k} \quad & \mathbf{x}_{N|k}^T P \mathbf{x}_{N|k} + \sum_{i=0}^{N-1} (\mathbf{x}_{i|k}^T Q \mathbf{x}_{i|k} + \mathbf{u}_{i|k}^T R \mathbf{u}_{i|k}) \\ \text{s.t.} \quad & \left.\begin{array}{l} \mathbf{x}_{i+1|k} = \mathbf{f}(\mathbf{x}_{i|k}) + \mathbf{g}(\mathbf{x}_{i|k})(\mathbf{u}_{i|k} + \mathbf{d}_{i|k}), \\ \mathbf{x}_{i|k} \in \mathcal{X}, \\ \mathbf{u}_{i|k} \in \mathcal{U}, \\ \mathbf{x}_{i|N} \in \mathcal{X}_f, \;\; \mathbf{x}_{0|k} = \mathbf{x}_k. \end{array}\right\} \forall \mathbf{d}_{i|k} \in \mathcal{W}, \;\; \forall i \in \mathcal{I} \end{aligned} \tag{3.8}$$

Compared with the differences in the nominal case, the optimization is over the state and control constraints satisfied for all disturbance sequences. To ensure a robust stability and persistent feasibility, first we need to set the terminal constraint, $\mathcal{X}_f$, as a robust control invariant set and the terminal cost being a strictly Lyapunov function as defined in Theorem 16.9 in [9]. Second, we can choose the optimization being over either open-loop control policies or some feedback policies $\mathbf{u}_{i|k} = \pi_{i|k}(\mathbf{x}_{i|k}) \in \mathcal{U}$. Both of them can be solved as a nominal control problem with tighter constraints on states and inputs. The main differences between these two are that the open-loop formulation tries to counteract any feasible disturbance at time steps k+i with one single control sequence and the feedback formulation takes into account that the disturbance and the control play one move at a time. In practical point of views, the open-loop formulation is easier to implement but the unbounded propagation of uncertainties over the prediction horizon may easily lead the problem infeasible. On the other hand, the feedback formulation is more applicable, since the tightening is less conservative. However, the evaluation of the robust MPC law through optimization in space of feedback policies is general computationally intractable and not practical for complex systems.

Based on the above statements, we know that having uncertainties included in the optimal control design will vastly increase the complexity for the robust MPC. This motivates us to find a powerful predefined "inner loop" control law which can help eliminate or, at least, reduce the disturbance for system (3.1) to its minimal value so that we can rely on the inherent robustness of the nominal MPC [45]. As we mentioned in the introduction, sliding mode control stands out by its strength of a low computational cost for a robust stability guarantee. We then combine this technique into designing the control algorithm to overcome the disadvantage of large computational demands in RMPC.

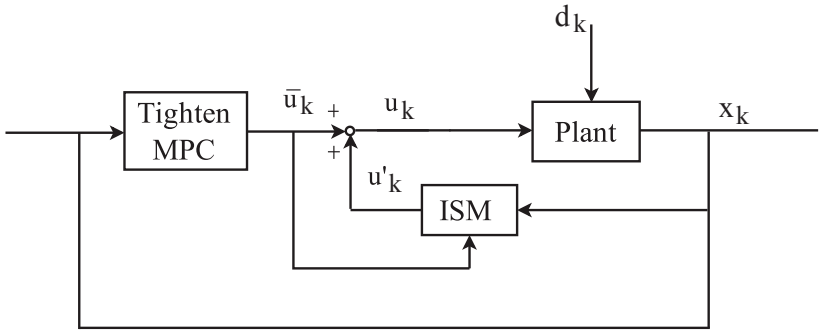


Figure 3.1: Block diagram of discrete-time integral model predictive control (DISMPC) strategy

3.3 Discrete-time Integral Sliding MPC (DISMPC)

3.3.1 Design Procedure

In this section, a discrete-time integral sliding model predictive control (DISMPC) algorithm for robust MPC is proposed. Since the state trajectory obtained by MPC is asymptotically stable for the nominal system without disturbance, it can be considered as a candidate of our discrete-time integral sliding surface in DISMC. With DISMC, the entire state response will not exceed the range of the quasi sliding band. According to this property, the constraint tightening in the open-loop formulation than can be simplified. Figure 3.1 shows the block diagram of the implementation for the control strategy and the procedure of designing DISMPC are listed as follows:

1. Recall that the Pontryagin difference of sets A and B is defined as $A \ominus B := \{x \mid x+B \subseteq A\}$. Calculate the tightened constraint sets

$$\bar{\mathcal{U}} = \mathcal{U} \ominus \mathcal{V}, \quad \bar{\mathcal{X}} = \mathcal{X} \ominus \mathcal{Z} \tag{3.9}$$

   where $\mathcal{Z}$ is the quasi sliding band along the boundary of $\mathcal{X}$ and $\mathcal{V}$ is the sliding input control set.

2. At each time k, according to the current feedback state, solve the following constrained

finite-time optimal control problem:

$$\begin{aligned}
\min_{\mathcal{U}_k} \quad & \mathbf{x}_{N|k}^T P \mathbf{x}_{N|k} + \sum_{i=0}^{N-1} (\mathbf{x}_{i|k}^T Q \mathbf{x}_{i|k} + \mathbf{u}_{i|k}^T R \mathbf{u}_{i|k}) \\
\text{s.t.} \quad & \mathbf{x}_{i+1|k} = \mathbf{f}(\mathbf{x}_{i|k}) + \mathbf{g}(\mathbf{x}_{i|k})\mathbf{u}_{i|k}, \quad \forall i \in \mathcal{I} \\
& \mathbf{x}_{i|k} \in \bar{\mathcal{X}}, \qquad \forall i \in \mathcal{I} \setminus \{0\} \\
& \mathbf{u}_{i|k} \in \bar{\mathcal{U}}, \qquad \forall i \in \mathcal{I} \\
& \mathbf{x}_{i|N} \in \mathcal{X}_f, \ \ \mathbf{x}_{0|k} = \mathbf{x}_k \in \mathcal{X}.
\end{aligned} \tag{3.10}$$

where the terminal constraint, $\mathcal{X}_f$, is set to be a robust control invariant set and the terminal cost is a strictly Lyapunov function as defined in Theorem 16.9 in [9]. Note that it is generally impossible to obtain an explicit characterization of P and $\mathcal{X}_f$, especially for a nonlinear system. Several approximation methods have been introduced in the literature [46, 47].

3. Then, based on the solution of tightened MPC, $\bar{\mathbf{u}}_k = \mathbf{u}_{0|k}^*$, design a discrete-time integral sliding surface as:

$$\begin{cases}
\mathbf{s}_k = C\mathbf{x}_k + \sigma_k \\
\sigma_{k+1} = \sigma_k + C(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_k) - \mathbf{g}(\mathbf{x}_k)\bar{\mathbf{u}}_k) \\
\sigma_0 = -C\mathbf{x}_0
\end{cases} \tag{3.11}$$

where $\sigma_k, \mathbf{s}_k \in \mathbb{R}^m$ and the matrix $C\mathbf{g}(\mathbf{x}_k) \succ 0$ [42].

4. Set the disturbance rejection control input as

$$\mathbf{u}_k' = -M\text{sgn}(\mathbf{s}_k)$$

where M is positive definite and the element of matrix M are large enough to reject the disturbance.

5. Implement the control law as

$$\mathbf{u}_k = \bar{\mathbf{u}}_k + \mathbf{u}_k' \tag{3.12}$$

6. Repeat the procedure 2) to 4) at next time instant.

3.3.2 Optimal Discrete-time Sliding Surface

It can be noticed that $\mathbf{s}_0 = 0$ in (3.11), which means the states start on the sliding surface at the beginning of the time step. Moreover, from (3.1) and (3.11), we can know that

$$\begin{aligned}
\mathbf{s}_{k+1} - \mathbf{s}_k &= C\mathbf{x}_{k+1} - C\mathbf{x}_k + \sigma_{k+1} - \sigma_k \\
&= C(\mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)(\mathbf{u}_k + \mathbf{d}_k)) - C\mathbf{x}_k + \sigma_{k+1} - \sigma_k \\
&= C\mathbf{g}(\mathbf{x}_k)(\mathbf{u}_k - \bar{\mathbf{u}}_k + \mathbf{d}_k).
\end{aligned} \tag{3.13}$$

To obtain the discrete-time integral sliding surface, we first set $\mathbf{s}_{k+1} - \mathbf{s}_k = 0$ and solve for the equivalent control law on the discrete-time integral sliding surface:

$$\mathbf{u}_{eq} = \bar{\mathbf{u}}_k - \mathbf{d}_k \tag{3.14}$$

By substituting (3.14) into (3.1), we then find out the discrete-time integral sliding surface:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\bar{\mathbf{u}}_k \tag{3.15}$$

By comparing (3.7) with (3.15), it is natural to see that the discrete-time integral sliding surface defined in (3.5) is the same as the state trajectories solved by optimal MPC problem (3.10). Therefore, we can guarantee the state response will asymptotically converge to certain area near the equilibrium point.

3.3.3 Sliding Mode Control Design and Stability Analysis

To analyze the stability property, we first substitute (3.12) into (3.13) to obtain the dynamics of sliding variables

$$\mathbf{s}_{k+1} = \mathbf{s}_k + C\mathbf{g}(\mathbf{x}_k)\mathbf{u}'_k + C\mathbf{g}(\mathbf{x}_k)\mathbf{d}_k \tag{3.16}$$

which provides a guidance on designing the sliding term. By observing (3.16), it is easy to tell that several control strategies can be implemented to guarantee the robust stability of (3.16). The most intuitive one would be the deadbeat control:

$$\mathbf{u}'_k = -(C\mathbf{g}(\mathbf{x}_k))^{-1}\mathbf{s}_k \tag{3.17}$$

which allows the smallest amount of tightening in the state constraints. However, it may introduce an unnecessary increasing of unmatched disturbances (if the system has one) and will require the largest amount of tightening in the control constraints. As we know that the nominal MPC has its own inherent robustness, therefore, to balance the performance, we then choose the control strategy as

$$\mathbf{u}'_k = -\alpha(C\mathbf{g}(\mathbf{x}_k))^{-1}\|C\mathbf{g}(\mathbf{x}_k)\|_\infty \operatorname{sgn}(\mathbf{s}_k) \tag{3.18}$$

which minimizes the tightening in $\mathcal{U}$ to allow a maximal control capacity in the nominal MPC.

Theorem 3.3.1. *To ensure the reachability of the specified discrete-time integral sliding surface (3.11), we design the DISMPC control law as:*

$$\mathbf{u}_k = \bar{\mathbf{u}}_k - \alpha(C\mathbf{g}(\mathbf{x}_k))^{-1}\|C\mathbf{g}(\mathbf{x}_k)\|_\infty \, sgn(\mathbf{s}_k) \tag{3.19}$$

where $\bar{\mathbf{u}}_k$ is the first optimal control signal derived from (3.10) with the tightened constraints. $\|C\mathbf{g}(\mathbf{x}_k)\|_\infty$ denotes the matrix ∞-norm of $C\mathbf{g}(\mathbf{x}_k)$ for all $\mathbf{x}_k \in \mathcal{X}$. With the control law

defined by (3.19), the system's state then is guaranteed to stay in the quasi-sliding band, $\triangle$, along the state trajectories defined by tighten MPC:

$$\triangle = \{\ |\mathbf{s}_{k(j)}| \leq \varsigma,\ \varsigma = 2\alpha \left\|C\mathbf{g}(\mathbf{x}_k)\right\|_\infty,\ \forall j \in \mathcal{J}\} \tag{3.20}$$

where $\mathcal{J} = \{1, ..., m\}$ is the set of sliding surface indices and $s_{k(j)}$ represents the j^{th} element of the vector $\mathbf{s}_k$.

Proof. Substituting (3.19) into (3.13) yields

$$\mathbf{s}_{k+1} = \mathbf{s}_k - \alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty \operatorname{sgn}(\mathbf{s}_k) + C\mathbf{g}(\mathbf{x}_k)\mathbf{d}_k \tag{3.21}$$

Since the initial state is already on the sliding surface, we can separate the proof into two cases.

1. $0 \leq \mathbf{s}_{k(j)} \leq \varsigma = 2\alpha \left\|C\mathbf{g}(\mathbf{x}_k)\right\|_\infty$:

   For any time step $k \geq 1$, since the disturbance is bounded as $0 \leq \|\mathbf{d}_k\|_\infty \leq \alpha$, the equation (3.21) becomes:

$$\begin{aligned}\mathbf{s}_{k+1(j)} &\leq \mathbf{s}_{k(j)} - \alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty + \|C\mathbf{g}(\mathbf{x}_k)\|_\infty\|\mathbf{d}_k\|_\infty \\ &\leq 2\alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty = \varsigma\end{aligned}$$

$$\begin{aligned}\mathbf{s}_{k+1(j)} &\geq \mathbf{s}_{k(j)} - \alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty - \|C\mathbf{g}(\mathbf{x}_k)\|_\infty\|\mathbf{d}_k\|_\infty \\ &= \mathbf{s}_{k(j)} - 2\alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty \geq -\varsigma\end{aligned}$$

2. $0 > \mathbf{s}_{k(j)} \geq -\varsigma = -2\alpha \left\|C\mathbf{g}(\mathbf{x}_k)\right\|_\infty$:

   For any time step $k \geq 1$, the equation (3.21) becomes:

$$\begin{aligned}\mathbf{s}_{k+1(j)} &\geq \mathbf{s}_{k(j)} + \alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty - \|C\mathbf{g}(\mathbf{x}_k)\|_\infty\|\mathbf{d}_k\|_\infty \\ &\geq -2\alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty = -\varsigma\end{aligned}$$

$$\begin{aligned}\mathbf{s}_{k+1(j)} &\leq \mathbf{s}_{k(j)} + \alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty + \|C\mathbf{g}(\mathbf{x}_k)\|_\infty\|\mathbf{d}_k\|_\infty \\ &= \mathbf{s}_{k(j)} + 2\alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty < \varsigma\end{aligned}$$

Combining cases 1) and 2), we can deduce the result of

$$\triangle = \{\ |\mathbf{s}_{k(j)}| \leq \varsigma,\ \varsigma = 2\alpha \left\|C\mathbf{g}(\mathbf{x}_k)\right\|_\infty,\ \forall j \in \mathcal{J}\} \tag{3.22}$$

□

3.3.4 Control Input Constraint Tightening

From the aforementioned argument, there is an additional disturbance rejection control input designed as:

$$\mathbf{u}'_k = -\alpha(C\mathbf{g}(\mathbf{x}_k))^{-1}\|C\mathbf{g}(\mathbf{x}_k)\|_\infty \operatorname{sgn}(\mathbf{s}_k) \tag{3.23}$$

Each element of this additional control input $\mathbf{u}'_{k(j)}$ has the upper bound of

$$\begin{aligned}\|\mathbf{u}'_k\|_\infty &= \alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty\|(C\mathbf{g}(\mathbf{x}_k))^{-1}\operatorname{sgn}(\mathbf{s}_k)\|_\infty \\ &\leq \alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty\|(C\mathbf{g}(\mathbf{x}_k))^{-1}\|_\infty\|\operatorname{sgn}(\mathbf{s}_k)\|_\infty \\ &= \alpha\|C\mathbf{g}(\mathbf{x}_k)\|_\infty\|(C\mathbf{g}(\mathbf{x}_k))^{-1}\|_\infty = \alpha\kappa(C\mathbf{g}(\mathbf{x}_k))\end{aligned}$$

where $\kappa(\bullet)$ is the condition number defined in ∞-norm. To minimize the amount of tightening, we can further choose $C = (\mathbf{g}^T(\mathbf{x}_k)\mathbf{g}(\mathbf{x}_k))^{-1}\mathbf{g}^T(\mathbf{x}_k)$ for the system with full rank matrix $\mathbf{g}(\mathbf{x}_k)$. This makes $C\mathbf{g}(\mathbf{x}_k) = \mathrm{I} \succ 0$, and hence $\kappa(C\mathbf{g}(\mathbf{x}_k)) = 1$. From this result, the disturbance rejection input control polytope can then be introduced as:

$$\mathcal{V} = \{\mathbf{u}'_k \mid |\mathbf{u}'_{k(j)}| \leq \alpha\,, \quad \forall j \in \mathcal{J}\} \tag{3.24}$$

Therefore, to satisfy the actual control input constraints (3.2), the new tightened control input constraint for optimal MPC problem is found as

$$\bar{\mathcal{U}} = \mathcal{U} \ominus \mathcal{V} \tag{3.25}$$

3.3.5 State Constraint Tightening

The actual state constraint, $\mathcal{X}$, is set to be a closed polytope, which can be expressed as the following inequality equations:

$$\mathcal{X} = \{\mathbf{x}_k \in \mathbb{R}^n \mid G\mathbf{x}_k \leq W,\ \forall k \in \mathbb{Z}^+\}$$

Since the motion equation in the integral sliding surface has a dimension of n, it can be arbitrarily assigned in the state space. Therefore, to tighten the state constraints, we consider the worst case of a sliding surface S lying on the boundaries of $\mathcal{X}$:

$$\mathcal{S} = \{\mathbf{x}_k \in \mathbb{R}^n \mid g_l\mathbf{x}_k - w_l = 0\,, \forall l \in \mathcal{L},\ \forall k \in \mathbb{Z}^+\}$$

where $\mathcal{L} = \{1, ..., p\}$ indicates the set of constraint indices. According to (3.22), the integral quasi-sliding domain $\mathcal{Z}$ for S can then be expressed as:

$$\begin{aligned}\mathcal{Z} &= \{\mathbf{x}_k \mid -\varsigma \leq\ g_l\mathbf{x}_k - w_l \leq \varsigma\,,\ \varsigma = 2\alpha\,\|G\mathbf{g}(\mathbf{x}_k)\|_\infty\,\} \\ &= \{\mathbf{x}_k \mid (w_l - \varsigma) \leq\ g_l\mathbf{x}_k \leq (w_l + \varsigma)\,\}\end{aligned} \tag{3.26}$$

$\forall l \in \mathcal{L},\ \forall k \in \mathbb{Z}^+$. As a result, we obtain the new tightened state constraint as:

$$\begin{aligned}\bar{\mathcal{X}} &= \{\mathbf{x} \in \mathbb{R}^n \mid g_l\mathbf{x}_k \leq (w_l - 2\alpha\,\|G\mathbf{g}(\mathbf{x}_k)\|_\infty)\,,\ ,\forall l \in \mathcal{L},\ \forall k \in \mathbb{Z}^+\} \\ &= \mathcal{X} \ominus \mathcal{Z}\end{aligned} \tag{3.27}$$

3.4 Adaptive Discrete-time Integral Sliding MPC (ADISMPC)

3.4.1 Algorithm improvement

Although the control law proposed in (3.19) simplifies the constraint tightening procedure in the robust MPC formulation (3.8), the "bang-bang" logic in the control law design may easily introduce a "zig-zag" motion in the state response. To overcome this drawback, we can increase the inner loop's sampling rate for the integral sliding controller, since the critical limitation in computation time comes from solving the MPC optimization problem. As we know that for an ideal sliding mode control designed with continuous time, we can guarantee asymptotic tracking with zero steady-state error. Therefore, under the assumptions of having an infinitely large sampling rate and fast actuators, we can completely eliminate the matched disturbance term so that the RMPC problem (3.8) becomes a nominal MPC problem (3.6) after tightening the control constraints. However, there is no such actuator exists in practice, and hence a chattering phenomenon will still be introduced when implemented using any discretization approximation. The controller can no longer guarantee zero steady-state error. Instead, it guarantees ultimate boundedness with a quasi-sliding band.

To improve the robust performance, we then incorporate the adaptive sliding mode control strategy proposed in Chapter 2 to replace the additional sliding term, u'_k, for disturbance rejection. We will use a forward Euler discretization to approximate the controller. Suitable parameters of t_s, ϕ, k and ρ are chosen to balance between actuator capacity and steady state performance based on the discussion in Section 2.4. This can help us guarantee not only a thin quasi-sliding band for state constraint tightening but also an avoidance of a large chattering phenomenon as it allows a higher sampling rate and the sliding gain is adapted. Assume t is the current time instant. Specify two sampling systems: t_m and t_s where $t = t_m k$ is for tightened MPC controller and $t = t_s n$ is for adaptive ISMC controller. The proposed DISMPC can be updated as follows:

1. Calculate the tightened constraint sets

$$\bar{\mathcal{U}} = \mathcal{U} \ominus \mathcal{V}', \quad \bar{\mathcal{X}} = \mathcal{X} \ominus \mathcal{Z}' \tag{3.28}$$

   where $\mathcal{Z}'$ is the new quasi sliding band along the boundary of $\mathcal{X}$ and $\mathcal{V}'$ is the adaptive sliding input control set.

2. At each sampling time k, according to the current feedback state, solve the following

constrained finite-time optimal control problem:

$$\begin{aligned}
\min_{\mathcal{U}_k} \quad & \mathbf{x}_{N|k}^T P \mathbf{x}_{N|k} + \sum_{i=0}^{N-1} (\mathbf{x}_{i|k}^T Q \mathbf{x}_{i|k} + \mathbf{u}_{i|k}^T R \mathbf{u}_{i|k}) \\
\text{s.t.} \quad & \mathbf{x}_{i+1|k} = \mathbf{f}(\mathbf{x}_{i|k}) + \mathbf{g}(\mathbf{x}_{i|k})\mathbf{u}_{i|k}, \quad \forall i \in \mathcal{I} \\
& \mathbf{x}_{i|k} \in \bar{\mathcal{X}}, \qquad \forall i \in \mathcal{I} \setminus \{0\} \\
& \mathbf{u}_{i|k} \in \bar{\mathcal{U}}, \qquad \forall i \in \mathcal{I} \\
& \mathbf{x}_{i|N} \in \mathcal{X}_f, \ \ \mathbf{x}_{0|k} = \mathbf{x}_k \in \mathcal{X}.
\end{aligned} \tag{3.29}$$

where the terminal constraint, $\mathcal{X}_f$ is set to be a robust control invariant set and the terminal cost is a strictly Lyapunov function as defined in Theorem 16.9 in [9].

3. Given the newest updated solution of tightened MPC, $\bar{\mathbf{u}}_k$, design a discrete-time integral sliding surface at sampling time step n:

$$\begin{cases}
\mathbf{s}_n = C\mathbf{x}_n + \sigma_n \\
\sigma_{n+1} = \sigma_n + C(\mathbf{x}_n - \mathbf{f}'(\mathbf{x}_n) - \mathbf{g}'(\mathbf{x}_n)\bar{\mathbf{u}}_k) \\
\sigma_0 = -C\mathbf{x}_0
\end{cases} \tag{3.30}$$

where $\sigma_n, \mathbf{s}_n \in \mathbb{R}^m$. $\mathbf{f}'(\mathbf{x}_n)$ and $\mathbf{g}'(\mathbf{x}_n)$ are the system's state transition and input distribution functions with a smaller sampling time, t_s. The matrix C is chosen to be equal to $(\mathbf{g}'^T(\mathbf{x}_n)\mathbf{g}'(\mathbf{x}_n))^{-1}\mathbf{g}'^T(\mathbf{x}_n)$ such that $C\mathbf{g}'(\mathbf{x}_n) = \mathrm{I}$.

4. Update the sliding gain followed by the discretized adaptation law (2.8):

$$\hat{\mu}_{n+1(j)} = \hat{\mu}_{n(j)} + \frac{t_s}{\rho_{(j)}} \left[1 - \frac{2\phi_{(j)}^2}{(|\mathbf{s}_{n(j)}| + \phi_{(j)})^2} \right] \ \forall j \in \mathcal{J}, \quad \hat{\mu}_{0(j)} = \hat{\mu}_{(j)}(0) \tag{3.31}$$

and saturate it between $[0, \ \bar{\mu}_{(j)}]$.

5. Set the additional sliding control input as

$$\mathbf{u}'_n = -N \in \mathbb{R}^{m \times 1}$$

with the j^{th} element equal to $\hat{\mu}_{n(j)}\text{sgn}(\mathbf{s}_{n(j)}) + k_{(j)}\mathbf{s}_{n(j)}$ and obtain the control law:

$$\mathbf{u}_n = \bar{\mathbf{u}}_k + \mathbf{u}'_n. \tag{3.32}$$

6. Repeat the procedure from 2) if a new MPC control input is updated, otherwise continue from 3) at next time instant.

3.5 Simulation

In this section, we apply both DISMPC and adaptive DISMPC control laws to a disturbed linear time-invariant system. To verify the effectiveness of the proposed control strategies, we further compare the simulation results of invariant tube MPC introduced in [48]. Each of the methods has its own pros and cons. Detail discussions are provided in the following sections. Note that it is our purpose to choose a linear system as an example to gain insight for the benefits of adaptive DISMPC. As a matter of fact, the method we proposed can be easily implemented into a nonlinear disturbed system as well.

3.5.1 Numerical Example

Consider a linear time-invariant continuous time system given by

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 10 \\ 0 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 10 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ 10 \end{bmatrix} d(t) \tag{3.33}$$

where $d(t) = 0.05\cos(25t) + 0.03\sin(5t) - 0.025\cos(10t+2) + 0.01\cos(t+2)$ such that the constraint of

$$d(t) \in \mathcal{D} = \{d(t) \mid \|d(t)\|_\infty \leq \alpha = 0.1\}$$

is satisfied. Assume that the system starts from $\mathbf{x}(0) = [-5, \ -2]^T$ where the state, control input constraints are bounded and shown as follows:

$$\mathbf{x}(t) \in \mathcal{X} = \{\mathbf{x}(t) \mid G\mathbf{x}(t) \leq 2, \ G = [0 \ 1]\}$$
$$u(t) \in \mathcal{U} = \{u(t) \mid -1 \leq u(t) \leq 1\}.$$

To implement all the control logics using invariant tube MPC, DISMPC, and adaptive DISMPC, starting from formulating the problem into a robust MPC (3.8), we first obtain a discrete time model by a zero order hold method with a sample time of $t_k = 0.1$ sec. The discrete time system can be described as follows:

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + B_k(u_k + d_k) \tag{3.34}$$

where

$$A_k = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}; \ B_k = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}. \tag{3.35}$$

Then, set the horizon length $N = 9$ and the corresponding positive definite weighting matrices

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; \qquad R = 0.1.$$

Next, to compare all the simulation results, we will first introduce tube MPC. Then, implement DISMPC and adaptive DISMPC control algorithms later in the following sections.

3.5.2 Disturbance Invariant Tube MPC

Starting from implementing disturbance invariant tube MPC, we first need to iteratively calculate a disturbance invariant set, $\mathcal{Z}^{tube}$, for the closed-loop system

$$\mathbf{x}_{k+1} = A_f \mathbf{x}_k + B_k d_k \tag{3.36}$$

where $A_f = A_k + B_k K$. The detail procedure is provided from Algorithm 19 in [49]. The feedback matrix $K = [0.6167,\ 1.2703]$ is chosen based on the solution of discrete algebraic Riccati equation for the unconstrained LQR problem to stabilize the matrix A_f. In other words, the eigenvalues of the matrix A_f should be all in the unit circle. Recall that the multiplication of a set by a matrix denotes the mapping of the elements in the set

$$A\mathcal{B} = \{c \mid \exists\, b \in \mathcal{B},\, c = Ab\}.$$

Define the disturbance invariant set as

$$\mathcal{Z}^{tube} = \{\mathbf{x}_k \mid A_f \mathbf{x}_k + B_k d_k \in \mathcal{Z}^{tube},\, \forall \mathbf{x}_k \in \mathcal{Z}^{tube} \text{ and } \forall\, B_k d_k \in \mathcal{W}\}.$$

where $\mathcal{W} = B_k \mathcal{D}$. With this definition, the tightened constraints of the disturbance invariant tube MPC then can be expressed as:

$$\begin{aligned} \bar{\mathcal{X}}^{tube} &= \mathcal{X} \ominus \mathcal{Z}^{tube} \\ \bar{\mathcal{U}}^{tube} &= \mathcal{U} \ominus K\mathcal{Z}^{tube} \\ \bar{\mathcal{X}}_f^{\ tube} &= \mathcal{X}_f \ominus \mathcal{Z}^{tube}. \end{aligned}$$

Then, having $\mathcal{X}_f$ as the control invariant set and the terminal cost being the value function calculated from the original nominal system, finally, we can implement the feedback control law of disturbance invariant tube MPC with the following form

$$u_k = \bar{u}_k + K(x_k - \bar{x}^*_{0|k})$$

where $\bar{u}_k = u^*_{0|k}$ is the first optimal control signal obtained from the tightened MPC problem which includes the initial state as a decision variable and $\bar{x}^*_{0|k}$ is the first solution of the associated optimal state sequence for the tightened MPC.

The simulation results of the phase trajectories from $(-5, 2)$ for tube MPC is shown in Fig. 3.2. The solid line is the actual trajectory which is simulated through out the continuous time model (3.33) with the control input holding at each sample value for one time interval. The dash line is the sequence of optimal initial states $\{\mathbf{x}^*_{0|k}(\mathbf{x}(kt_k))\}$ for $k = 0,\ 1,\ 2, \ldots$. As we can see, the set $\mathcal{Z}^{tube}$ is robustly exponentially stable for the controlled uncertain system. However, we should notice that the set of $\mathbf{x}^*_{0|4} \oplus \mathcal{Z}^{tube}$ violates the state constraint. This is because we didn't include the model mismatch introduced by discretization into the disturbance model. To avoid this, we should further increase the upper bound of $\mathcal{W}$.

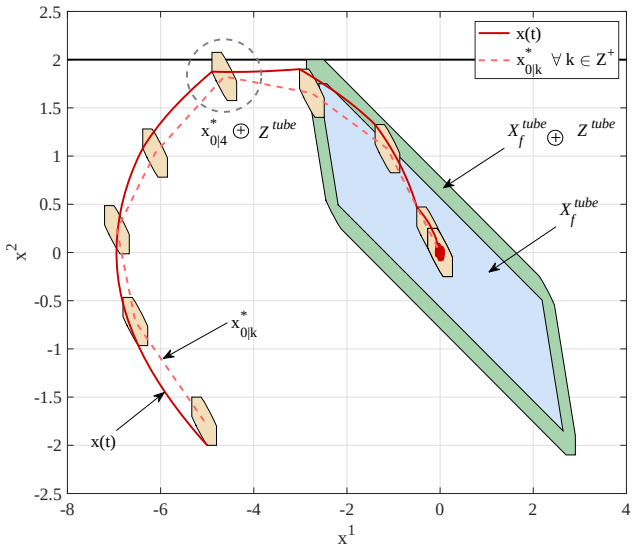


Figure 3.2: Phase trajectories from $(-5, 2)$ for tube MPC

3.5.3 Discrete-time Integral Sliding MPC

Following the design procedure listed in Section 3.3.1, to implement DISMPC, we need to first calculate the tightened constraints. As mentioned in Section 3.3.4, we will choose $C = (B_k^T B_k)^{-1} B_k^T = [0.4,\ 0.8]$ to minimize the amount of tightening for control input constraints. Then, according to equation (3.26), we know that $\varsigma = 2\alpha \|GB_k\|_\infty = 0.2$. Therefore, the sets of $\mathcal{V}$ and $\mathcal{Z}$ can be obtained from (3.24) and (3.26):

$$\mathcal{V} = \{ u'_k \,|\, -0.1 \leq u'_k \leq 0.1 \}$$
$$\mathcal{Z} = \{ x_k \,|\, -\varsigma \leq Gx_k - 2 \leq \varsigma,\ \varsigma = 0.2 \}$$

and the tightened state and control input constraints become

$$\bar{\mathcal{U}} = \{u \,|\, -0.9 \leq u_k \leq 0.9\} \tag{3.37}$$
$$\bar{\mathcal{X}} = \{x \,|\, Gx_k \leq 1.8\}. \tag{3.38}$$

The next step is solving the tightened MPC for $\bar{u}_k$. Before doing this, we need to design the terminal constraint, $\mathcal{X}_f$, as a robust control invariant set and the terminal cost being a strictly Lyapunov function for robust stability guarantee. Here, we choose $P = P_\infty$ as the solution of the algebraic Riccati equation for unconstrained LQR problem by knowing that there exists a sublevel set of $\mathbf{x}_k$ defined by positive constants c_1 and c_2 such that

$$c_1 |\mathbf{x}_k|^2 \leq \mathbf{x}_k^T P_\infty \mathbf{x}_k \leq c_2 |\mathbf{x}_k|^2 \tag{3.39}$$

will satisfy the assumption A3 listed in [9]. For the terminal constraint, we approximate the robust control invariant set by calculating the positive invariant set of the nominal system

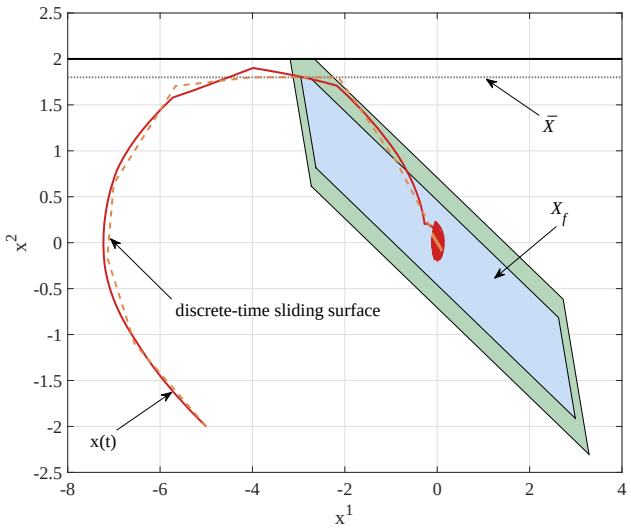


Figure 3.3: Phase trajectories from $(-5, 2)$ for DISMPC

under LQR control law with the original state and tightened control constraints, $\mathcal{X}$ and $\bar{\mathcal{U}}$. Then, tighten the set by $\mathcal{Z}$ afterwards. Once we obtain $\bar{u}_k$, the integral sliding surface can be defined as:

$$\begin{cases} s_k = C\mathbf{x}_k + \sigma_k \\ \sigma_{k+1} = \sigma_k + C(\mathbf{x}_k - A_k\mathbf{x}_k - B_k\bar{u}_k) \\ \sigma(0) = -C\mathbf{x}_0 \end{cases}$$

where $\mathbf{x}_k$ is the current state feedback of $\mathbf{x}(kt_k)$. Finally, we can implement the DISMPC control law as:

$$u_k = \bar{u}_k + 0.1\text{sgn}(s_k).$$

The simulation results are shown in Fig. 3.3. As already mentioned, we can see that the initial state starts on the integral sliding surface ($s_0 = 0$) and the state trajectory remains close to the discrete-time integral sliding surface represented by the orange dash line. The tightened state constraint is marked by the dotted line and the actual state trajectory (solid line) satisfies the original state constraint, $\mathcal{X}$.

3.5.4 Adaptive Discrete-time Integral Sliding MPC

Following from the result shown in Fig. 3.3, we can see that the convergence performance of DISMPC is worst than the one provided from the tube MPC. It is what we can expect as already mentioned in Section 3.4.1 and, therefore, we have the adaptive DISMPC proposed.

To implement adaptive DISMPC, we need to define another faster sampling system for the integral sliding mode controller. By choosing $t_s = 0.01$ sec which is 10 times faster than

$t_k = 0.1$ sec, we obtain another discrete time system described as follows:

$$\mathbf{x}_{n+1} = A_n\mathbf{x}_n + B_n(u_n + d_n) \tag{3.40}$$

where

$$A_n = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix};\ B_n = \begin{bmatrix} 0.005 \\ 0.1 \end{bmatrix}. \tag{3.41}$$

Then, we choose the matrix $C = (B_n^T B_n)^{-1} B_n^T = [0.4988,\ 9.9751]$, and tune the design parameters of $k = 0.1$, $\rho = 0.8$, $\phi = 0.2$ and $\bar{\mu} = 0.14$ for the adaptation law (3.31). To obtain these parameters, we first initialize $\phi = 0.2$ by inferring from s_n being bounded within the quasi-sliding band for $\varsigma = 2\alpha\|CB_n\|_\infty = 0.2$ without the adaptation. Since ϕ is set to be equal to 0.2, we can roughly guess the upper bound of s_n as 0.4 by equation (2.20) and further increase it for taking the errors induced from the discretization into account:

$$s_n \leq (\sqrt{2} - 1)\phi + \frac{\mu}{\sqrt{m}} \leq 0.4. \tag{3.42}$$

Then, the equivalent disturbance upper bound, α_{eq}, can be calculated as 0.2 according to

$$2\alpha_{eq}\|CB_n\|_\infty = 2\alpha_{eq} \approx 0.4$$

by knowing that C is chosen to let $\|CB_n\|_\infty = 1$. With this result, we have the new quasi-sliding band along the boundary of $\mathcal{X}$ and the adaptive sliding input control set as:

$$\mathcal{V}' = \{\, u_k' \,|\, -\bar{\mu} \leq u_k' \leq \bar{\mu},\ \bar{\mu} = \alpha + 2k\alpha_{eq} = 0.14\}$$
$$\mathcal{Z}' = \{\, x_k \,|\, -\varsigma \leq Gx_k - 2 \leq \varsigma,\ \varsigma = 2\alpha_{eq}\|GB_n\|_\infty = 0.04\}.$$

Therefore, the new tightened state and control input constraints become:

$$\bar{\mathcal{U}} = \{u \mid -0.86 \leq u_k \leq 0.86\} \tag{3.43}$$
$$\bar{\mathcal{X}} = \{x \mid Gx_k \leq 1.96\}. \tag{3.44}$$

Next, follow the same procedure to obtain the terminal constraint and the terminal cost as shown in Section 3.5.3. Design the integral sliding surface as:

$$\begin{cases} s_n = C\mathbf{x}_n + \sigma_n \\ \sigma_{n+1} = \sigma_n + C(\mathbf{x}_n - A_n\mathbf{x}_n - B_n\bar{u}_k) \\ \sigma(0) = -C\mathbf{x}_0 \end{cases}$$

where $\mathbf{x}_n$ is the current state feedback of $\mathbf{x}(nt_n)$ and $\bar{u}_k$ is the newest control input calculated from the tightened MPC. Finally, we can implement the adaptive DISMPC control law:

$$u_n = \bar{u}_k + \hat{\mu}_n \mathrm{sgn}(s_n) + ks_n \tag{3.45}$$

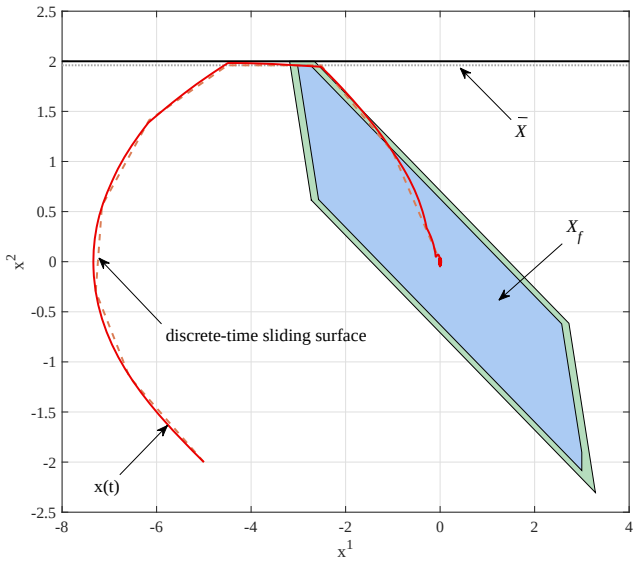


Figure 3.4: Phase trajectories from $(-5, 2)$ for adaptive DISMPC

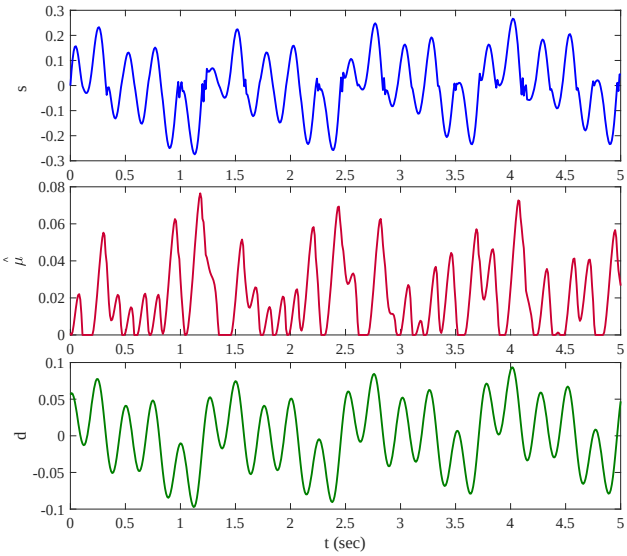


Figure 3.5: Sliding variable s, sliding gain $\hat{\mu}$ and additive disturbance for adaptive DISMPC

where $\hat{\mu}_n$ is the adaptation gain obtained from (3.31) with an initial value $\hat{\mu}_0 = 0.001$.

Again, we run the simulation with the control input holding at constant during the sampling time of 0.01 sec. Fig. 3.4 illustrates the performance of adaptive DISMPC. As expected, it has a smaller spread of state trajectories compared with the one obtained from DISMPC. The responses of sliding variable and the adaptation gain are shown in Fig. 3.5. We can see that the sliding variable is always bounded within the quasi-sliding band of ± 0.4 and the adaptation gain has a trend to follow the additive disturbance $d(t)$.

3.6 Comparison

For comparison, we have all corresponding simulation results of tube MPC, DISMPC and adpative DISMPC shown in Fig. 3.6 - 3.8. As displayed in Fig. 3.6, there is no doubt that the robust performance of DISMPC is the worst and adaptive DISMPC stands out from the others with smallest convergent region and less amount of tightening in state constraint.

However, according to Fig. 3.7, the state response of implementing adaptive DISMPC

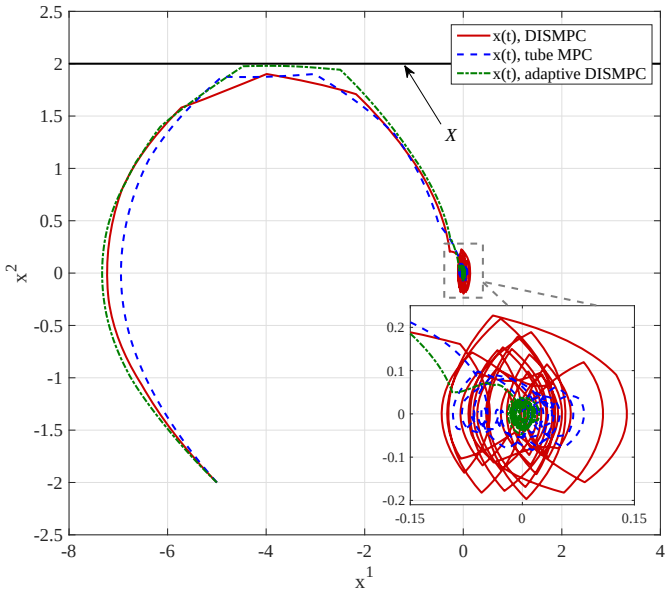


Figure 3.6: Performance comparison for DISMPC, tube MPC, and adaptive DISMPC

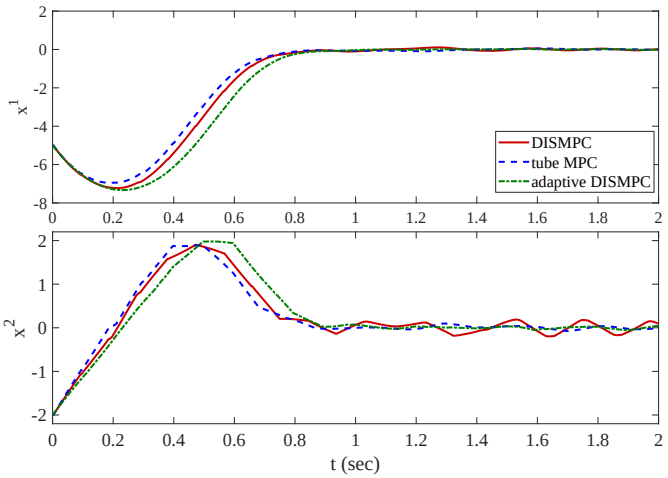


Figure 3.7: State response from $(-5, 2)$ for DISMPC, tube MPC, and adaptive DISMPC

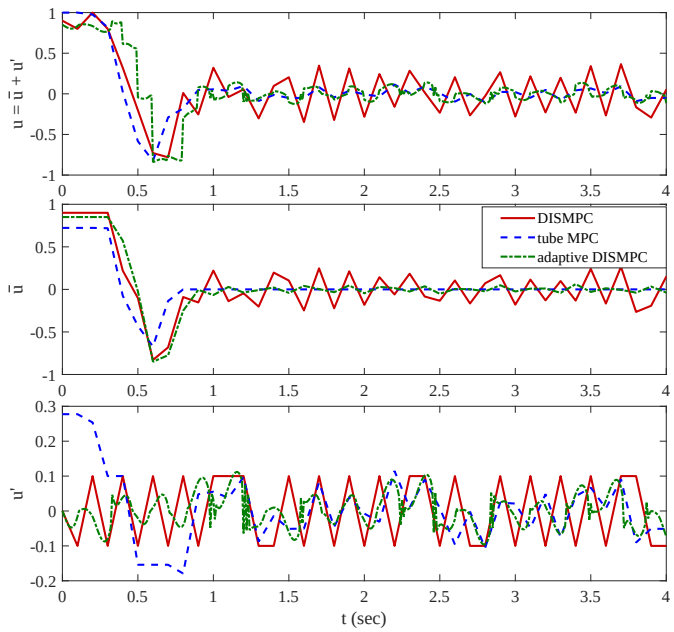


Figure 3.8: Control inputs for DISMPC, tube MPC, and adaptive DISMPC

converges slower than the others and the one using tube MPC is the fastest. This is because tube MPC is the only controller uses out the maximum control capacity in the beginning of the response. Comparing all the control input shown in Fig. 3.8, we can see that tube MPC achieves the maximum control input in the first quarter second. Since it incorporates the initial state as a decision variable in the optimal control problem, the current state feedback will always lie on the extreme point of $\mathbf{x}^*_{0|k} \oplus \mathcal{Z}^{tube}$ before the optimal initial state converges to zero. Therefore, the additional control input for disturbance rejection in tube MPC will not equal to zero at the starting time. Although this is the advantage of tube MPC, one thing we should bring out is that adaptive DISMPC allows a smaller control constraint tightening compared with tube MPC which gives more flexibility to cope with any changing condition during on-line optimization. Comparing the tightened constraints of

$$\bar{\mathcal{U}}^{tube} = \{u \mid -0.7226 \leq u_k \leq 0.7226\} \tag{3.46}$$

$$\bar{\mathcal{X}}^{tube} = \{x \mid Gx_k \leq 1.75\}. \tag{3.47}$$

to the one of DISMPC described in (3.43) and (3.44), we can see that tube MPC is more conservative than DISMPC even under the case of not considering the errors introduced by the discretization. Another advantage of the adaptive DISMPC is that it can effectively eliminate the disturbance for both linear and nonlinear system which further gains the inherit robustness of nominal MPC after the tightening. Although it does not simplify the analysis and the on-line implementation complexity into that of conventional MPC as tube MPC, it eliminates the need to compute the disturbance invariant set for a constrained system which is generally a significant practical issue in most of the problem.

3.7 Conclusion

In this Chapter, a method for robust model predictive control named adaptive DISMPC has been presented. Similar to the disturbance invariant tube MPC strategy, the feedback control law of adaptive DISMPC contains two parts: one is an optimal feedback control input solved from the tightened MPC problem; the other one is an additional adaptive sliding mode control input for disturbance rejection. By introducing the concept of quasi-sliding band, we need to tighten the constraints before solving the MPC problem. Compared with other alternative robust MPC formulations, adaptive DISMPC has a better convergence performance. Although it does not simplify the analysis and the on-line implementation complexity into that of conventional MPC as tube MPC, it eliminates the need to compute the disturbance invariant set for a constrained system. A simpler approach and a thinner amount in constraint tightening are its advantages. This results in having more flexibility to cope with any changing condition while doing the on-line optimization.

Part II

Applications

Chapter 4

Adaptive Sideslip Angle Estimation: Sensor-limited Conditions

4.1 Motivation

A number of active safety features have been introduced in the automotive industry in the past 30 years to prevent accidents such as braking assistance, traction and electronic stability control systems [50–53]. The main goals of these systems are to maintain vehicle stability and to improve vehicle handling. To implement these functions, vehicle states, parameters and road conditions need to be measured or estimated. Among all of these, sideslip angle, the angle between the longitudinal direction of the vehicle and the velocity vector, is one of the most important variables which heavily influences vehicle dynamics and is required by a number of active safety controllers. Although it can be directly measured by sensors such as optical sensors [54] or GPS sensors [55, 56], these solutions are not implemented by OEMs because of cost and reliability. Therefore, the estimation of sideslip angle based on the sensors available in production vehicles is an important topic that has been widely discussed in the literature [57–73]. Most of the approaches in the literature are model-based and can be classified into three main categories: kinematics model-based, dynamics model-based and a combination of the two.

The kinematics model-based approach proposed in [57, 58] constructs an observer based on the longitudinal and lateral translation kinematics of a point mass model. This method has the advantage of not requiring the vehicle parameters, tire model and road friction coefficient. It can provide an accurate sideslip estimate in a number of cases. However, it suffers from a drifting issue in small yaw rate maneuvers and the estimated result is sensitive to disturbance and measurement noise such as bank angles or sensor bias when the longitudinal and lateral accelerations are small. The dynamics model-based approach constructs an advanced state observer (i.e. an extended Kalman filter or an unscented Kalman filter) by using a bicycle model or its variations [59–65]. These models consider the effect of forces applied to vehicle mass and rotation inertia which provides a relatively robust estimate to acceleration

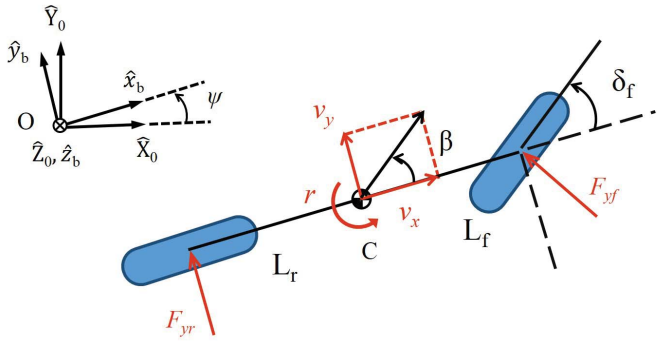


Figure 4.1: Lateral dynamics for bicycle model.

measurement noise compared to the one from the kinematics model. However, an estimation bias is often observed due to the model uncertainties associated to variations of vehicle mass and the tire cornering stiffnesses. Existing literature has also focused on developing algorithms for estimating the sideslip angle and vehicle model system parameters simultaneously. In [63–66], Lyapunov-based observers have been proposed for the tire cornering stiffness identification. Although these techniques can improve the estimation results, they require persistent input excitations and the adaptation performance becomes worse beyond the linear tire model region. Alternative studies have proposed learning-based techniques to assist the traditional adaptation methods [67–69]. However, the estimated performance is hard to validate in the region when data is limited. The third category of algorithms tries to merge kinematics and dynamics models into a hybrid solution [70–73]. The concept is to switch between these two estimators and to exploit their respective advantages. This method might look attractive, however the switching will cause a discontinuity in the sideslip estimation. Motivated by the idea of the hybrid solution [70–73] and parameter adaptation, in this paper, we develop a novel sideslip estimation algorithm which only relies on the dynamics model for the estimation but utilizes the strength of the kinematics observer to adapt the front and rear tire cornering stiffnesses. In this way, we maintain the advantage of the dynamics model-based observer and further improve the estimator performance in the nonlinear tire region. In addition, the proposed approach does not need rich input excitation as required in traditional adaptation methods.

The chapter is organized as follows. We first introduce commonly used models for kinematics and dynamics-based observer designs. Then, modifications with bank angle and sensor bias effects are considered and included into each of the models. A recursive adaptation algorithm is derived and the stability property is discussed afterwards. Finally, the performance is validated with different scenario tests and compared with existing methods.

4.2 Modeling

Many of the conventional sideslip angle estimation methods are model-based. In this section, we are going to introduce two different models which have been frequently used in the literature [68]: a 2-DOF point mass rigid body kinematics model and a bicycle dynamics model. Each of them has its own strengths and weaknesses in the observer design. Details are discussed in the following sections.

4.2.1 Kinematics model

Kinematics is the study of motion which treats the movement of components without considering the forces. To describe the general motion of a rigid body, we first introduce two coordinate systems followed by the ISO convention: one is fixed in the inertial space $\{\hat{X}_0, \hat{Y}_0.\hat{Z}_0\}$ and the other one is fixed to the body $\{\hat{x}_b, \hat{y}_b, \hat{z}_b\}$ (see Fig. 4.1). Then, considering the vehicle as a single rigid body constrained to move in the $\hat{X}_0\hat{Y}_0$-plane, the translation motion is described as:

$$\begin{aligned} a_x &= \dot{v}_x - r v_y \\ a_y &= \dot{v}_y + r v_x \end{aligned} \tag{4.1}$$

where ψ is the yaw angle, $r \triangleq \dot{\psi}$ is the yaw rate of the vehicle. a_x and v_x denote the inertial acceleration and velocity resolved in the longitudinal $\hat{x}_b$-direction. a_y and v_y denote the same physical quantities but resolved in the lateral $\hat{y}_b$-direction. Define the sideslip angle as

$$\beta = \tan^{-1}(v_y/v_x).$$

We write the system (4.1) into a standard state space form as:

$$\begin{aligned} \dot{\mathbf{x}}_k &= A_k(t)\mathbf{x}_k + B_k(t)\mathbf{u}_k \\ \mathbf{y}_k &= C_k(t)\mathbf{x}_k \end{aligned} \tag{4.2}$$

where $\mathbf{x}_k = [v_x,\ v_y]^T$ is the state vector, $\mathbf{u}_k = [a_x,\ a_y]^T$ is the control input vector, $\mathbf{y}_k = v_x$ is the measurement output vector and the system, input and output matrices are

$$\begin{aligned} A_k(t) &= \begin{bmatrix} 0 & r(t) \\ -r(t) & 0 \end{bmatrix}, \quad B_k(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ C_k(t) &= \begin{bmatrix} 1 & 0 \end{bmatrix}. \end{aligned} \tag{4.3}$$

As explained in [74], using the kinematics model is advantageous as it allows a sideslip angle estimation without requiring vehicle parameters. All we need is r, a_x, a_y and v_x which can be directly obtained from sensors available in commercial vehicles. However, the estimation is sensitive to sensor noise which is substantial for a_x and a_y. Moreover, the convergence of the estimation error can be guaranteed only when yaw rate is not equal to

zero. In fact, the system (4.2) is not observable when the yaw rate is equal to zero and the poor conditioning of the observability matrix causes a drifting problem. This can be avoided by resetting the estimated states to zero every time when yaw rate is less than a threshold value.

4.2.2 Dynamics model

A variety of dynamics models have appeared in the literature. The so-called lateral bicycle model shown in Fig. 4.1 is a widely used and rather simple model that neglects the coupling of the roll, pitch and longitudinal dynamics. By using Newton's law of motion, the lateral dynamics of the bicycle model is described as follows [75]:

$$\begin{aligned} ma_y &= m(\dot{v}_y + v_x r) = F_{yf}\cos\delta_f + F_{yr} \\ I_z\dot{r} &= L_f F_{yf}\cos\delta_f - L_r F_{yr} \end{aligned} \tag{4.4}$$

where m is the vehicle mass, I_z is the equivalent yaw moment of inertia, δ_f is the front steering angle and L_f, L_r are the distance from the vehicle center of gravity (COG) to the front and rear axles. To further simplify the model, we assume small tire slip and front steering angles. Then, the front and the rear lateral tire forces F_{yf}, F_{yr} can be approximated by a linear function:

$$\begin{aligned} F_{yf}\cos\delta_f \approx F_{yf} &= C_f\left(\delta_f - \frac{v_y + L_f r}{v_x}\right) \\ F_{yr} &= C_r\left(\frac{-v_y + L_r r}{v_x}\right) \end{aligned} \tag{4.5}$$

where C_f and C_r are the front and rear tire cornering stiffnesses. Substituting (4.5) into (4.4), we then obtain a nonlinear model. Assume that the vehicle is traveling with slowly varying longitudinal velocity. At each step, a linearization process will be applied to approximate the nonlinear system (4.4)-(4.5) with a linear time varying system shown as follows:

$$\begin{aligned} \dot{\mathbf{x}}_d &= A_d(t)\mathbf{x}_d + B_d(t)\mathbf{u}_d \\ \mathbf{y}_d &= C_d(t)\mathbf{x}_d + D_d(t)\mathbf{u}_d \end{aligned} \tag{4.6}$$

where $\mathbf{x}_d = [v_y,\ r]^T$ is the state vector, $\mathbf{u}_d = \delta_f$ is the control input vector and $\mathbf{y}_d = [a_y,\ r]^T$ is the measurement output vector.

$$\begin{aligned} A_d(t) &= \begin{bmatrix} \frac{-C_f - C_r}{m v_x(t)} & -v_x(t) - \frac{L_f C_f - L_r C_r}{m v_x(t)} \\ \frac{-L_f C_f + L_r C_r}{I_z v_x(t)} & \frac{-L_f^2 C_f - L_r^2 C_r}{I_z v_x(t)} \end{bmatrix}, \\ C_d(t) &= \begin{bmatrix} \frac{-C_f - C_r}{m v_x(t)} & -\frac{L_f C_f - LrC_r}{m v_x(t)} \\ 0 & 1 \end{bmatrix}, \\ B_d(t) &= \begin{bmatrix} \frac{C_f}{m} \\ \frac{L_f C_f}{I_z} \end{bmatrix}, \quad D_d(t) = \begin{bmatrix} \frac{C_f}{m} \\ 0 \end{bmatrix}. \end{aligned} \tag{4.7}$$

Using the dynamics bicycle model to estimate the sideslip angle has several advantages. First, the estimator can be tuned to be less sensitive to acceleration measurement noise compared to the one based on the kinematics model. Also, drifting and observability issues of the kinematics model are not present. However, the estimated accuracy is affected by the vehicle parameters in the matrices (4.7). First, since we use a linear tire model, the sideslip estimate will be accurate only in the linear tire region. Second, compared to m, I_z, L_f and L_r, it is hard to find a good initial condition for the tire cornering stiffness coefficients. To mitigate this issue, on-line adaptation algorithms have been introduced to identify the cornering stiffness [64, 65]. We will also use this idea in our method.

4.3 Sideslip Estimation Method

The method proposed in this paper relates to the idea of [71] which merges the kinematics and dynamics model observers into a hybrid solution. Since the estimated state from the kinematics model is unaffected by the parameter uncertainties, in [71], the observer is built to mainly rely on it but will switch to the dynamics model when the absolute value of the yaw rate is less than a threshold value r_t to avoid unobservability and the drifting issue. Although this method addresses the drifting issue, relying on the kinematics model leads to noisy estimates. Moreover, the switch between the kinematics and dynamics models for the observer often introduces a discontinuous estimate during the transition.

To overcome these issues and keep the benefits of hybrid models, we propose a new method which is based on a dynamics model but adapts on-line the front and rear tire cornering stiffnesses using information derived from the kinematics model. Compared with the traditional adaptation algorithm proposed in [64, 65], the proposed approach does not need persistent excitation in the control input and also improves the adaptation performance in the nonlinear tire region. For the observer design, we further include the road bank angle disturbance and lateral acceleration sensor bias into the system model in order to minimize possible modeling and estimation errors. This is discussed in next.

4.3.1 Augmented Models

4.3.1.1 Dynamics model augmented with the road bank angle and sensor bias

We consider the bicycle model and include the gravitational force to the lateral dynamics:

$$m(\dot{v}_y + v_x r) = F_{yf}\cos\delta_f + F_{yr} - mg\sin\phi \tag{4.8}$$

where ϕ is the road bank angle with the sign convention shown in Fig. 4.2. Then, combining the yaw dynamics in (4.4) with (4.8), we rewrite the first equation of (4.6) as:

$$\dot{v}_y = \frac{-(C_f + C_r)}{mv_x}v_y - (v_x + \frac{L_f C_f - L_r C_r}{mv_x})r - g\sin\phi + \frac{C_f}{m}\delta_f. \tag{4.9}$$

The measurement model should also be corrected with the bank disturbance and sensor bias as well. Note that the lateral accelerometer measures the right hand side of (4.8) divided by m and plus the component of gravity in $\hat{y}_b$ direction. We obtain the measurement model of the lateral acceleration as:

$$\begin{aligned} a_y^{sen} &= a_y + g\sin\phi + d = \dot{v}_y + v_x r + g\sin\phi + d \\ &= \frac{-(C_f + C_r)}{m v_x(t)} v_y - \frac{L_f C_f - L_r C_r}{m v_x(t)} r + d + \frac{C_f}{m}\delta_f. \end{aligned} \tag{4.10}$$

where d is the sensor bias. By augmenting the system with a constant bank angle disturbance and the sensor bias, the state vector and measurement output are $\mathbf{x}_d = [v_y,\ r,\ \sin\phi,\ \ d]^T$ and $\mathbf{y}_d = [a_y^{sen},\ r]^T$. The system state space matrices which replace the one in (4.7) are

$$\begin{aligned} A_d(t) &= \begin{bmatrix} \frac{-C_f - C_r}{m v_x(t)} & -v_x(t) - \frac{L_f C_f - L_r C_r}{m v_x(t)} & -g & 0 \\ \frac{-L_f C_f + L_r C_r}{I_z v_x(t)} & \frac{-L_f^2 C_f - L_r^2 C_r}{I_z v_x(t)} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \\ C_d(t) &= \begin{bmatrix} \frac{-C_f - C_r}{m v_x(t)} & -\frac{L_f C_f - L_r C_r}{m v_x(t)} & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \\ B_d(t) &= \begin{bmatrix} \frac{C_f}{m} \\ \frac{L_f C_f}{I_z} \\ 0 \\ 0 \end{bmatrix}, \qquad D_d(t) = \begin{bmatrix} \frac{C_f}{m} \\ 0 \end{bmatrix}. \end{aligned} \tag{4.11}$$

To implement the extended Kalman filter using a digital controller, we further discretized model (4.11) using a forward Euler method as:

$$\begin{aligned} \hat{\mathbf{x}}_d[k+1] &= (A_d[k]\Delta t + \mathrm{I}_4)\hat{\mathbf{x}}_d[k] + B_d[k]\Delta t \mathbf{u}_d[k] + \mathbf{w}_d[k] \\ \mathbf{y}_d[k] &= C_d[k]\hat{\mathbf{x}}_d[k] + D_d[k]\mathbf{u}_d[k] + \mathbf{v}_d[k] \end{aligned} \tag{4.12}$$

where Δt is the sampling period and $[\cdot]$ represent the discrete time instant. $\mathbf{w}_d[\cdot]$ and $\mathbf{v}_d[\cdot]$ are the process and measurement noises satisfying the typical assumptions of the extended Kalman filter.

4.3.1.2 Kinematics model augmented with the road bank angle

For the kinematics model, we only include the bank angle disturbance into the lateral motion by noting that the system is not observable if the model of the sensor bias is added. The model becomes:

$$\begin{aligned} a_x &= \dot{v}_x - r v_y \\ a_y^{sen} &= \dot{v}_y + r v_x + g\sin\phi. \end{aligned} \tag{4.13}$$

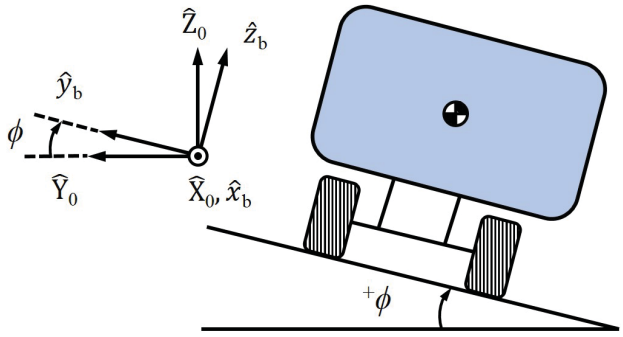


Figure 4.2: Sign convention for bank angle.

Then, having the same extended Kalman filter structure shown in (4.12), we re-define the state vector $\mathbf{x}_k = [v_x,\ v_y,\ \sin\phi]^T$, the output vector $\mathbf{u}_k = [a_x,\ a_y^{sen}]^T$ and the system state space matrices as

$$\begin{aligned} A_k(t) &= \begin{bmatrix} 0 & r(t) & 0 \\ -r(t) & 0 & -g \\ 0 & 0 & 0 \end{bmatrix}, \quad B_k(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \\ C_k(t) &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}. \end{aligned} \tag{4.14}$$

The above model is, again, discretized into:

$$\begin{aligned} \hat{\mathbf{x}}_k[k+1] &= (A_k[k]\Delta t + \mathrm{I}_3)\hat{\mathbf{x}}_k[k] + B_k[k]\Delta t\mathbf{u}_k[k] + \mathbf{w}_k[k] \\ \mathbf{y}_k[k+1] &= C_k[k+1]\hat{\mathbf{x}}_k[k+1] + \mathbf{v}_k[k+1]. \end{aligned} \tag{4.15}$$

4.3.2 Adaptation for the tire cornering stiffness

In the previous section, we have introduced two observer models (4.12) and (4.15). Next, we will show how we merge both observers by using $\hat{v}_{y,k}$, the lateral velocity estimated from (4.15), to adapt the front and rear tire cornering stiffnesses in the dynamics model (4.12). The sideslip estimation will then calculate by using this updated dynamics model.

4.3.2.1 Regression model

The adaptation is formulated as a regularized weighted least square (RWLS) problem [76,77]. To build up the adaptation algorithm, we first specify the regression model as

$$Y = \Phi^T\theta \tag{4.16}$$

where θ is the parameter to be estimated; Φ and Y are the input and output measurements. Substituting equation (4.13) into (4.9), the a_y measurement can be expressed as follows:

$$\begin{aligned} a_y^{sen} &= \dot{v}_y + v_x r + g\sin\phi \\ &= \frac{-(C_f + C_r)}{m v_x} v_y - \frac{L_f C_f - L_r C_r}{m v_x} r + \frac{C_f}{m}\delta_f. \end{aligned} \tag{4.17}$$

Then, combining (4.17) with the yaw rate dynamics, we define the regression model as:

$$\begin{aligned} \Phi^T &= \begin{bmatrix} \frac{-L_f^2 r - L_f \hat{v}_{y,k}}{v_x} + L_f\delta_f & \frac{-L_r^2 r + L_r \hat{v}_{y,k}}{v_x} \\ \frac{-L_f r - \hat{v}_{y,k}}{v_x} + \delta_f & \frac{L_r r - \hat{v}_{y,k}}{v_x} \end{bmatrix}, \\ Y &= \begin{bmatrix} I_z \dot{r} \\ m a_y^{\text{sen}} \end{bmatrix}, \qquad \theta = \begin{bmatrix} C_f \\ C_r \end{bmatrix} \end{aligned} \tag{4.18}$$

where the unknown lateral velocity is replaced by $\hat{v}_{y,k}$ estimated from the kinematics model. Observe that all the other time-varying variables in the input and output measurements can be directly obtained from the standard sensors for yaw stability control system. The angular acceleration is obtained by differentiating the yaw rate: $(r[k] - r[k-1])/\Delta t$ with a low-pass filter.

4.3.2.2 Adaptation algorithm

Considering all the input and output data sampled at time instant $i\Delta t$, where $i = 1, 2, ...k$ is the time step, we want to minimize the sum of the squared prediction errors:

$$J(\theta_k) = \sum_{i=1}^{k} \lambda^{(k-i)} \left\| Y_i - \Phi_i^T \theta_k \right\|_2^2 + \delta \left\| \theta_k - \theta^+ \right\|_2^2 \tag{4.19}$$

where $0 \ll \lambda < 1$ is the forgetting factor and $\theta^+ = [C_f^+ \; C_r^+]^T$ is the nominal values of the front and rear tire cornering stiffnesses. Comparing (4.19) with a standard least square problem, we have included an additional 2-norm regularized term with $\delta > 0$ in order to improve the estimate robustness when the data is less informative or too noisy. By setting the partial derivative of $J(\theta_k)$ with respect to θ_k to zero, the optimal solution, θ_k^*, can be derived as follows:

$$\theta_k^* = \left(\sum_{i=1}^{k} \lambda^{k-i} \Phi_i \Phi_i^T + \delta \mathrm{I}_2 \right)^{-1} \left(\delta\theta^+ + \sum_{i=1}^{k} \lambda^{k-i} \Phi_i Y_i \right)$$

which implies

$$\tilde{\theta}_k^* = \left(\sum_{i=1}^{k} \lambda^{k-i} \Phi_i \Phi_i^T + \delta \mathrm{I}_2 \right)^{-1} \sum_{i=1}^{k} \lambda^{k-i} \Phi_i \tilde{Y}_i \tag{4.20}$$

$$\begin{aligned} \text{where} \qquad & \tilde{\theta}_i := \theta_i - \theta^+, \qquad \tilde{\theta}_k^* := \theta_k^* - \theta^+ \\ & \tilde{Y}_i := Y_i - \Phi_i^T \theta^+ = \Phi_i^T \tilde{\theta}_i. \end{aligned}$$

The expression in (4.20) is called the batch formulation since it processes the available data set all at once. For simplicity, we can further rewrite the solution in a recursive way as:

$$\tilde{\theta}_k^* = \tilde{\theta}_{k-1}^* + (R_k + \delta \mathrm{I}_2)^{-1} \left[\delta(\lambda - 1)\tilde{\theta}_{k-1}^* + \Phi_k e_k \right]. \tag{4.21}$$

$$\text{where} \quad R_k = \sum_{i=1}^{k} \lambda^{k-i} \Phi_i \Phi_i^T = \lambda R_{k-1} + \Phi_k \Phi_k^T \tag{4.22}$$

$$e_k = \tilde{Y}_k - \Phi_k^T \tilde{\theta}_{k-1}^*$$

and $k = 1, 2, ..., \infty$. Notice that $(R_{k+1} + \delta \mathrm{I}_2)^{-1}$ in (4.21) is a simple 2-by-2 matrix inversion and the existence of the solution is guaranteed by the regularization term. The recursive formula in (4.22) of the adaptation gain R_k will help us better understand the stability properties of the adaptation algorithm [78]. More details will be discussed in Section 5. In the next section, we summarize the new proposed algorithm for the sideslip angle estimation.

4.3.3 Proposed sideslip angle estimation algorithm

We have presented the discrete-time dynamics and kinematics observer models in (4.12) and (4.15), respectively. At each time step, both of the estimated states will be updated using the extended Kalman filters [79]. The dynamics model is used for estimating the sideslip angle and the kinematics model is used for estimating the tire cornering stiffnesses by applying the adaptation law (4.21). Notice that we will enable the adaptation process only when the absolute value of the yaw rate is greater than a certain threshold, r_t, in order to have a valid estimated $\hat{v}_{y,k}$ from the kinematics model. The pseudo code of the estimation algorithm is provided in Algorithm 1.

4.4 Stability and Convergence Analysis

In this section, we study the stability of the proposed adaptation algorithm. In other words, we want to study the convergence property of the tire cornering stiffness estimation error. The energy-based hyperstability approach [80] is used as it addresses the problem nonlinearity. The analysis will follow three steps.

A. Formulate the adaptation algorithm (4.21) as a nonlinear feedback systems shown in Fig. 4.3 where v_k will be linked to the error between the measured output Y_k listed in (4.18) at time k and the one predicted according to the adaptation law $\Phi_k^T \theta_k^*$.

B. Prove that v_k converges to 0 for $k \to \infty$ by using the sufficient conditions of the hyperstability listed in Appendix A.

C. Use the previous results to prove the convergence of the estimated tire cornering stiffnesses to a neighbourhood of the true ones.

Algorithm 1 Sideslip Angle Estimation (Algorithm 1)

initialize:
  $\hat{\mathbf{x}}_k[0] \leftarrow [v_x[0]\ v_y[0]\ 0]^T$, $\hat{\mathbf{x}}_d[0] \leftarrow [v_y[0]\ r[0]\ 0\ \ 0]^T$, $\tilde{\theta}_0^* \leftarrow 0$, $R_0 \leftarrow 0$, $\theta_0^* \leftarrow \theta^+ + \tilde{\theta}_0^*$,
  $P_k[0] \leftarrow P_{k,0}$, $P_d[0] \leftarrow P_{d,0}$ *// initialize prior means and estimate error covariance matrices for* EKF
for i = 1 to k+1 **do**
  $\hat{\mathbf{x}}_d[i] \leftarrow \text{EKFupdate}(\hat{\mathbf{x}}_d[i-1], \mathbf{u}_d[i-1], \mathbf{u}_d[i], \mathbf{y}_d[i], P_d[i-1], \theta_{i-1}^*)$ *// dynamics model (4.12)*
  $\hat{\mathbf{x}}_k[i] \leftarrow \text{EKFupdate}(\hat{\mathbf{x}}_k[i-1], \mathbf{u}_k[i-1], \mathbf{y}_k[i], P_k[i-1]))$ *// kinematics model (4.15)*
  if $|r_i| \geq r_t$ **then**
    $R_i = \lambda R_{i-1} + \Phi_i \Phi_i^T$ *// obtain the input measurement Φ_i from (4.18)*
    $\theta_i^* \leftarrow \theta^+ + \text{AdaptationUpdate}(R_i, \tilde{\theta}_{i-1}^*)$ *// apply a recursive update law (4.21)*
  else
    $\theta_i^* \leftarrow \theta_{i-1}^*$
    $\hat{\mathbf{x}}_k[i] \leftarrow [v_x[i]\ \hat{v}_{y,d}[i]\ \sin\hat{\phi}_d[i]]^T$ *// update the state estimates of the* EKF *for model (4.15)*
    $P_k[i] \leftarrow \text{diag}(0, P_d[i](1,1), P_d[i](3,3))$ *// update the error covariance matrix for model (4.15)*
  end if
  $\beta[i] \leftarrow \tan^{-1}(\hat{v}_{y,d}[i]/v_x[i])$ *// calculate the sideslip angle*
end for

4.4.1 Nonlinear feedback formulation of the adaptation algorithm

We start by formulating the adaptation algorithm (4.21) as a nonlinear feedback system. We first derive a recursive law for the adaptation gain, $(R_k + \delta \mathrm{I}_2)^{-1}$. Define $F_k = (R_k + \delta \mathrm{I}_2)^{-1}$ and substitute R_k as defined in (4.22). We obtain:

$$F_k = [\lambda F_{k-1}^{-1} + \delta(1-\lambda)\mathrm{I}_2 + \Phi_k \Phi_k^T]^{-1}. \tag{4.23}$$

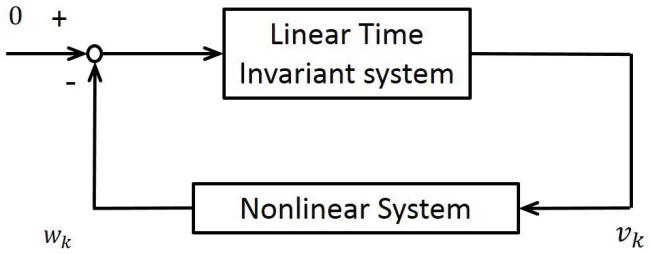


Figure 4.3: Nonlinear feedback system.

Then, denote $(\sigma_{1,k}, \boldsymbol{u}_{2,k})$, $(\sigma_{2,k}, \boldsymbol{u}_{2,k})$ as the first and the second pairs of the singular value and the left singular vector of the matrix Φ_k. We can rewrite $\Phi_k\Phi_k^T$ as

$$\Phi_k\Phi_k^T = \sum_{j=1}^{2} \sigma_{j,k}^2 \boldsymbol{u}_{j,k}\boldsymbol{u}_{j,k}^T = \sum_{j=1}^{2} \boldsymbol{\phi}_{j,k}\boldsymbol{\phi}_{j,k}^T$$

and further represent $\delta(1-\lambda)\mathrm{I}_2 + \Phi_k\Phi_k^T$ as:

$$\delta(1-\lambda)\mathrm{I}_2 + \Phi_k\Phi_k^T = \sum_{j=1}^{2} \mu_{j,k}\boldsymbol{\phi}_{j,k}\boldsymbol{\phi}_{j,k}^T \tag{4.24}$$

$$\text{with} \qquad \mu_{j,k} = \frac{\sigma_{j,k}^2 + \delta(1-\lambda)}{\sigma_{j,k}^2} \geq 1 \quad \text{for } j = 1, 2.$$

Here, $\mu_{j,k}$ is guaranteed to be finite and always exist because $\sigma_{j,k} > 0$. From the expression of Φ_k^T in (4.18), we observe that Φ_k^T is always full rank with exception of singular cases which can be easily discarded in real applications. Combing (4.23) and (4.24), we obtain a measurement updated law of the adaptation gain by applying the matrix inverse lemma:

$$\begin{aligned} F_k &= [\lambda F_{k-1}^{-1} + \mu_{1,k}\boldsymbol{\phi}_{1,k}\boldsymbol{\phi}_{1,k}^T + \mu_{2,k}\boldsymbol{\phi}_{2,k}\boldsymbol{\phi}_{2,k}^T]^{-1} \\ &= F_k' - \frac{F_k'\boldsymbol{\phi}_{2,k}\boldsymbol{\phi}_{2,k}^T F_k'}{\mu_{2,k}^{-1} + \boldsymbol{\phi}_{2,k}^T F_k'\boldsymbol{\phi}_{2,k}} \end{aligned} \tag{4.25}$$

$$\text{where} \quad F_k' = \frac{1}{\lambda}\left(F_{k-1} - \frac{F_{k-1}\boldsymbol{\phi}_{1,k}\boldsymbol{\phi}_{1,k}^T F_{k-1}}{\lambda\mu_{1,k}^{-1} + \boldsymbol{\phi}_{1,k}^T F_{k-1}\boldsymbol{\phi}_{1,k}}\right). \tag{4.26}$$

Notice that the updated law of the adaptation gain (4.25)-(4.26) contains two parts. First, F_{k-1} is updated with the first singular vector of the input measurement data, $\phi_{1,k}$ to yield F_k'. Then F_k' is updated based on the second singular vector, $\phi_{2,k}$. For this reason, the original sampling time $k = 1, ..., T$ is now converted into $n = 1, ..., 2T$, where $k = \lceil \frac{n}{2} \rceil$. This will allow us to use the hyperstability theorem which is formulated for SISO systems (Appendix). Substitute (4.25) into (4.21) to obtain:

$$\begin{aligned} \tilde{\theta}_n^* &= \tilde{\theta}_{n-1}^* - \beta_n \boldsymbol{f}_n\boldsymbol{\phi}_n\boldsymbol{\phi}_n^T\tilde{\theta}_{n-1}^* + \boldsymbol{f}_n\boldsymbol{\phi}_n\tilde{y}_n \\ &= \tilde{\theta}_{n-1}^* + \boldsymbol{f}_n\boldsymbol{\phi}_n(\tilde{y}_n - \beta_n\boldsymbol{\phi}_n^T\tilde{\theta}_{n-1}^*) \end{aligned} \tag{4.27}$$

$$
\begin{aligned}
\text{where} \quad & \boldsymbol{f}_n^{-1} = \alpha_n \boldsymbol{f}_{n-1}^{-1} + \beta_n \phi_n^T \phi_n, \qquad \boldsymbol{f}_0^{-1} = 0 \\
& \boldsymbol{f}_n = \frac{1}{\alpha_n}\left(\boldsymbol{f}_{n-1} - \frac{\boldsymbol{f}_{n-1}\boldsymbol{\phi}_n\boldsymbol{\phi}_n^T\boldsymbol{f_{n-1}}}{\alpha_n\beta_n^{-1} + \boldsymbol{\phi}_n^T\boldsymbol{f}_{n-1}\boldsymbol{\phi}_n}\right) \\
& \tilde{y}_n = \phi_n^T\tilde{\theta}_n, \qquad \tilde{\theta}_0^* = 0, \qquad \alpha_1 = 1.
\end{aligned} \tag{4.28}
$$

$$
\begin{aligned}
\boldsymbol{\phi}_n &= \begin{cases} \boldsymbol{\phi}_{1,\lceil \frac{n}{2} \rceil}, \\ \boldsymbol{\phi}_{2,\lceil \frac{n}{2} \rceil}, \end{cases} \qquad \beta_n = \begin{cases} \mu_{1,\lceil \frac{n}{2} \rceil}, \\ \mu_{2,\lceil \frac{n}{2} \rceil}, \end{cases} \\
\alpha_n &= \begin{cases} \lambda, & \text{if } n \text{ is odd} \\ 1, & \text{if } n \text{ is even.} \end{cases}
\end{aligned} \tag{4.29}
$$

Notice that $\tilde{y}_n$ is the measured output and $\boldsymbol{\phi}_n^T\tilde{\theta}_{n-1}^*$ is the predicted one according to (4.27). We then define

$$
\begin{aligned}
\varepsilon_n &= \tilde{y}_n - \beta_n\boldsymbol{\phi}_n^T\tilde{\theta}_n^* \\
\varepsilon_n^o &= \tilde{y}_n - \beta_n\boldsymbol{\phi}_n^T\tilde{\theta}_{n-1}^*
\end{aligned} \tag{4.30}
$$

as a "*scaled*" a-posteriori and a "*scaled*" a-priori predicted measurement errors, respectively. Right multiplying $\boldsymbol{\phi}_{n+1}$ to $\boldsymbol{f}_n$ in equation (4.28), we obtain

$$
\boldsymbol{f}_n\phi_n = \frac{\boldsymbol{f}_{n-1}\boldsymbol{\phi}_n}{\alpha_n + \beta_n\boldsymbol{\phi}_n^T\boldsymbol{f}_{n-1}\boldsymbol{\phi}_n} \tag{4.31}
$$

and the adaptation law (4.27) becomes

$$
\tilde{\theta}_n^* = \tilde{\theta}_{n-1}^* + \frac{\boldsymbol{f}_{n-1}\boldsymbol{\phi}_n}{\alpha_n + \beta_n\boldsymbol{\phi}_n^T\boldsymbol{f}_{n-1}\boldsymbol{\phi}_n}\varepsilon_n^o. \tag{4.32}
$$

Then, again left multiplying $-\beta_n\boldsymbol{\phi}_n^T$ to (4.32) and adding $\tilde{y}_n$ to both sides of the equation lead to:

$$
\varepsilon_n = \frac{\alpha_n}{\alpha_n + \beta_n\boldsymbol{\phi}_n^T\boldsymbol{f}_{n-1}\boldsymbol{\phi}_n}\varepsilon_n^o. \tag{4.33}
$$

With this relation, we can express the adaptation law (4.32) using the a-posteriori predicted measurement error ε_n as follows:

$$
\tilde{\theta}_n^* = \tilde{\theta}_{n-1}^* + \frac{1}{\alpha_n}\boldsymbol{f}_{n-1}\boldsymbol{\phi}_n\varepsilon_n. \tag{4.34}
$$

Define the "scaled" parameter estimation error as

$$
\Delta\tilde{\theta}_n = \beta_n\tilde{\theta}_n^* - \tilde{\theta}_n, \tag{4.35}
$$

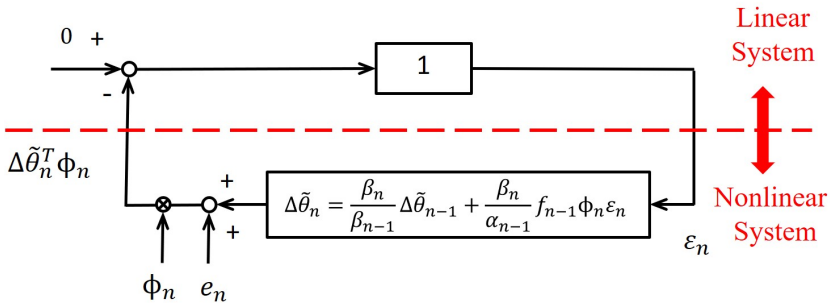


Figure 4.4: Block diagram of the adaptation algorithm for stability analysis.

with $\tilde{\theta}_0 = 0$ and $\beta_0 = 1$. We can rewrite (4.30) and (4.34)-(4.35) into the following error dynamics:

$$\begin{cases} \varepsilon_n = \phi_n^T\tilde{\theta}_n - \beta_n\boldsymbol{\phi}_n^T\tilde{\theta}_n^* = -\phi_n^T\Delta\tilde{\theta}_n \\ \Delta\tilde{\theta}_n = \frac{\beta_n}{\beta_{n-1}}\Delta\tilde{\theta}_{n-1} + \frac{\beta_n}{\alpha_{n-1}}\boldsymbol{f}_{n-1}\phi_n\varepsilon_n + e_n \end{cases} \tag{4.36}$$

where $e_n = \frac{\beta_n}{\beta_{n-1}}\tilde{\theta}_{n-1} - \tilde{\theta}_n$ is treated as an external bounded disturbance. Finally, we can represent this error dynamics into the block diagram of Fig. 4.4 which is equivalent to the nonlinear feedback system as shown in Fig. 4.3.

4.4.2 Hyperstability analysis

Theorem 4.4.1. *The nonlinear feedback system depicted in the block diagram of Fig. 4.4 with the error dynamics described in (4.36) without the external disturbance term e_n is asymptotically hyperstable (i.e. $\varepsilon_n \to 0$) if*

$$\frac{2-\alpha_{n+1}}{\beta_n} - \frac{1}{\beta_{n-1}} \geq 0 \quad \forall n = 1, 2, ...\infty. \tag{4.37}$$

Proof. To prove the theorem we use the concept of hyperstability [80] briefly summarized in Appendix. Next, we prove that the sufficient conditions listed in Theorem 4.8.1 are satisfied. First, we notice that the forward linear system is the identity, which is obviously strictly positive real. However, the nonlinear block of the adaptation algorithm does not satisfy the Popov inequality. Therefore, we further modify the system and rewrite it as the one shown in the block diagram of Fig. 4.5. Since the same signals have been added and subtracted in the feedback system, the stability property will not change.

We now again check the sufficient conditions for the modified system. Start with the requirement of the nonlinear feedback block satisfying Popov inequality. Considering System

A in the block diagram of Fig. 4.5 with input (s_n) and output (w_n) signals, we have

$$\begin{aligned} w_n &= \Delta\tilde{\theta}_n^T \boldsymbol{\phi}_n, \quad s_n = \varepsilon_n + \frac{\alpha_{n+1}}{2}\Delta\tilde{\theta}_n^T \boldsymbol{\phi}_n \\ \boldsymbol{\phi}_n \boldsymbol{\phi}_n^T &= \beta_n^{-1}(\boldsymbol{f}_n^{-1} - \alpha_n \boldsymbol{f}_{n-1}^{-1}) \\ \boldsymbol{\phi}_n \varepsilon_n &= \frac{\alpha_n}{\beta_n}\boldsymbol{f}_{n-1}^{-1}\Delta\tilde{\theta}_n - \frac{\alpha_n}{\beta_{n-1}}\boldsymbol{f}_{n-1}^{-1}\Delta\tilde{\theta}_{n-1} \end{aligned} \tag{4.38}$$

from equations (4.28) and (4.36) without considering the external disturbancee_n. Define

$$\eta_n = \frac{2-\alpha_{n+1}}{\beta_n} - \frac{1}{\beta_{n-1}}$$

for the sake of simplicity in later expression. The sum of the product of w_n and s_n can be calculated as:

$$\begin{aligned} \sum_{n=1}^{2k} w_n s_n &= \sum_{n=1}^{2k} \Delta\tilde{\theta}_n^T \boldsymbol{\phi}_n (\varepsilon_n + \frac{\alpha_{n+1}}{2}\Delta\tilde{\theta}_n^T \boldsymbol{\phi}_n) \\ =& \sum_{n=1}^{2k} \frac{\alpha_n}{2}\frac{1}{\beta_{n-1}}(\Delta\tilde{\theta}_n^T - \Delta\tilde{\theta}_{n-1}^T)\boldsymbol{f}_{n-1}^{-1}(\Delta\tilde{\theta}_n - \Delta\tilde{\theta}_{n-1}) \\ &+ \sum_{n=1}^{2k} \frac{\alpha_{n+1}}{2\beta_n}\Delta\tilde{\theta}_n^T \boldsymbol{f}_n^{-1}\Delta\tilde{\theta}_n - \frac{\alpha_n}{2\beta_{n-1}}\Delta\tilde{\theta}_{n-1}^T \boldsymbol{f}_{n-1}^{-1}\Delta\tilde{\theta}_{n-1} \\ &+ \sum_{n=0}^{2k} \frac{\eta_n \alpha_n}{2}\Delta\tilde{\theta}_n^T \boldsymbol{f}_{n-1}^{-1}\Delta\tilde{\theta}_n \\ =& \sum_{n=1}^{2k} \frac{\alpha_n}{2\beta_{n-1}}(\Delta\tilde{\theta}_n^T - \Delta\tilde{\theta}_{n-1}^T)\boldsymbol{f}_{n-1}^{-1}(\Delta\tilde{\theta}_n - \Delta\tilde{\theta}_{n-1}) \\ &+ \frac{\alpha_{2k+1}}{2\beta_{2k}}\Delta\tilde{\theta}_{2k}^T \boldsymbol{f}_{2k}^{-1}\Delta\tilde{\theta}_{2k} - \frac{\alpha_1}{2\beta_0}\Delta\tilde{\theta}_0^T \boldsymbol{f}_0^{-1}\Delta\tilde{\theta}_0 \\ &+ \sum_{n=1}^{2k} \frac{\eta_n \alpha_n}{2}\Delta\tilde{\theta}_n^T \boldsymbol{f}_{n-1}^{-1}\Delta\tilde{\theta}_n. \end{aligned} \tag{4.39}$$

Since all the variables of α_n, β_n are positive as defined in (4.29), we can know that the sum of the product of w_n and s_n will have a lower bound

$$\sum_{n=1}^{2k} w_n s_n \geq -\frac{\alpha_1}{2\beta_0}\Delta\tilde{\theta}_0^T \boldsymbol{f}_0^{-1}\Delta\tilde{\theta}_0 = 0$$

and satisfy the Pupov inequality with a condition of

$$\eta_n \geq 0 \quad \forall n = 1, 2, ..., 2k. \tag{4.40}$$

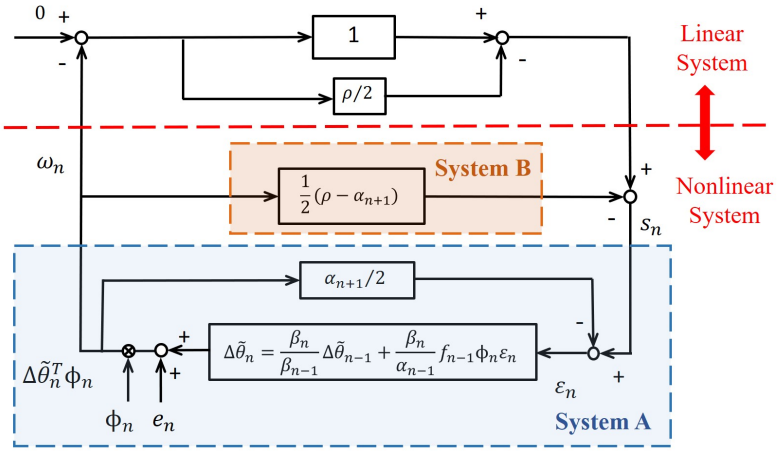


Figure 4.5: Equivalent system of the block diagram in Fig. 4.4 for stability analysis.

Next, considering the time varying linear System B in the block diagram of Fig. 4.5, we find that it also satisfies the Popov inequality since

$$\frac{1}{2}(\rho - \alpha_{n+1}) \geq 0 \quad \text{for choosing} \quad 1 \leq \rho < 2. \tag{4.41}$$

Then, the overall nonlinear feedback system as shown in the block diagram of Fig. 4.5 satisfies the Popov inequality since it is made by a feedback connection of two passive systems, A and B. Finally, the linear feedforward system for the modified system, $1 - \rho/2$, is strictly positive real for having $1 \leq \rho < 2$. Now, we know that the adaptation system is hyperstable. In other words, $|(1 - \rho/2)w_n| < \infty$ is bounded. This will further imply that the output of the nonlinear feedback system, $w_n < \infty$, is bounded as well. Therefore, having all of three requirements listed in Appendix A, we can conclude that the adaptation system without the external disturbance is asymptotic hyperstable $\varepsilon_n \to 0$. □

We start from the analysis in Theorem 4.4.1 and consider the effect of the external disturbance e_n.

Theorem 4.4.2. *Assume that the two norm of the difference of the true parameter $\tilde{\theta}_k$ between two consecutive steps is bounded:*

$$\|\tilde{\theta}_{k-1} - \tilde{\theta}_k\|_2 \leq \Lambda \qquad \forall k = 1, 2, ..., \infty.$$

Consider the regularized weighted least square problem (4.19) with the regression model described in (4.18). Then, there exists a set of parameters $\delta > 0, 0 \ll \lambda < 1$ satisfying the condition:

$$\sigma_{1,k}^2 \sigma_{2,k}^2 + \delta(2 - \lambda)\sigma_{2,k}^2 - \delta\sigma_{1,k}^2 \geq 0 \quad \forall k = 1, 2, ..., \infty \tag{4.42}$$

which guarantees $\varepsilon_{2k} \in R(\Lambda)$ for $R(\Lambda)$ being a ball of radius Λ centred in the origin.

Proof. We use the same steps as in Theorem 4.4.1. Consider the input signal ε_n of System A with

$$\phi_n \varepsilon_n = \frac{\alpha_n}{\beta_n} \boldsymbol{f}_{n-1}^{-1}(\Delta\tilde{\theta}_n + \tilde{\theta}_n) - \frac{\alpha_n}{\beta_{n-1}} \boldsymbol{f}_{n-1}^{-1}(\Delta\tilde{\theta}_{n-1} + \tilde{\theta}_{n-1})$$

deriving from (4.36). We can derive the same Popov inequality as shown in (4.39) for System A but with an extra term of

$$\sum_{n=1}^{2k} \alpha_n \Delta\tilde{\theta}_n^T \boldsymbol{f}_{n-1}^{-1}(\frac{1}{\beta_n}\tilde{\theta}_n - \frac{1}{\beta_{n-1}}\tilde{\theta}_{n-1}). \tag{4.43}$$

Therefore, the same requirement of $\eta_n = \frac{2-\alpha_{n+1}}{\beta_n} - \frac{1}{\beta_{n-1}} \geq 0$ in Theorem 4.4.1 is necessary for hyperstability. From this, we can easily infer the condition of (4.42) by expanding out β_{n-1}, β_n and α_{n+1} using (4.29). Next, combining the first term in the right hand side of (4.39) together with (4.43), we can conclude that System A will satisfy Pupov inequality under the conditions:

$$\begin{aligned} &\frac{\eta_n}{2}\Delta\tilde{\theta}_n^T \boldsymbol{f}_{n-1}^{-1}\Delta\tilde{\theta}_n + \Delta\tilde{\theta}_n^T \boldsymbol{f}_{n-1}^{-1}(\frac{1}{\beta_n}\tilde{\theta}_n - \frac{1}{\beta_{n-1}}\tilde{\theta}_{n-1}) \geq 0 \\ &\qquad\qquad \forall n = 1, 2, ..., 2k \\ \implies &\|\Delta\tilde{\theta}_n\|_2 \geq \frac{2\kappa(\boldsymbol{f}_{n-1}^{-1})}{\eta_n} \left\| \frac{1}{\beta_n}\tilde{\theta}_n - \frac{1}{\beta_{n-1}}\tilde{\theta}_{n-1} \right\|_2 \\ &\qquad\qquad \forall n = 1, 2, ..., 2k \end{aligned} \tag{4.44}$$

where $\kappa(\cdot)$ denotes the condition number of the positive definite matrix $\boldsymbol{f}_{n-1}^{-1}$. Since the rate of $\tilde{\theta}$ is bounded by the assumption, the existence of the right hand side in (4.44) is guaranteed. Then, based on passivity theorem [81], we know that there exists a time-varying energy function which is positive definite and is dissipating over time in the region of

$$\|\Delta\tilde{\theta}\|_2 \geq \max_n \frac{2\kappa(\boldsymbol{f}_{n-1}^{-1})}{\eta_n} \left\| \frac{1}{\beta_n}\tilde{\theta}_n - \frac{1}{\beta_{n-1}}\tilde{\theta}_{n-1} \right\|_2. \tag{4.45}$$

Equation (4.45) implies the boundedness of the predicted measurement error ϵ_n in the adaptation algorithm. □

Remark: Some considerations can be drawn from the analysis in Theorem 4.4.1 and Theorem 4.4.2. First, the boundedness of the adaptation error depends on the time-varying rate of change of $\tilde{\theta}$. According to the result shown in (4.45), a larger value in $\left\| \frac{1}{\beta_n}\tilde{\theta}_n - \frac{1}{\beta_{n-1}}\tilde{\theta}_{n-1} \right\|_2$ will lead to a larger bound in the predicted measurement error. Therefore, we can expect a better adaptation performance under non-extreme driving scenarios. Second, the boundedness of the adaptation error shrinks as the condition number of $\boldsymbol{f}_{n-1}^{-1}$ decreases. This highlights the importance of input measurement matrices, Φ_i^T, being well-conditioned in the

regression model. By observing (4.18), we can infer that the conditional number of Φ^T is roughly equal to the ratio of it's (2, 1) and (2, 2) components, since $L_f \approx L_r$ for a vehicle. In order to avoid a bad adaptation performance, we should add an additional condition of

$$\frac{1}{c_t} \leq \left| \frac{\Phi^T(2,1)}{\Phi^T(2,2)} \right| \leq c_t \tag{4.46}$$

to enable the adaptation process in our proposed sideslip angle algorithm where $c_t > 0$ is the maximum allowed conditional number of measurement data. Third, according to (4.39), η_n is the energy dissipation rate of the system. Therefore, a larger value of η_n results in a faster convergence rate. Based on an further analysis in the condition of $\eta_n \geq 0$, we can obtain a good starting value of the regularization weight

$$\delta \approx 1 \,/\, (\frac{1}{\sigma_{2,k}^2} - \frac{1}{\sigma_{1,k}^2}) \tag{4.47}$$

This is derived by rewriting (4.42) into the following form:

$$2 - \lambda \geq \sigma_{1,k}^2 \left(\frac{1}{\sigma_{2,k}^2} - \frac{1}{\delta} \right) \quad \text{for} \;\; 0 \ll \lambda < 1. \tag{4.48}$$

4.4.3 Convergence of the estimated cornering stiffness

Theorem 4.4.3. *The asymptotical hyperstability of the nonlinear feedback system depicted in Fig. 4.4 with $e_n = 0$ guarantees that the estimated parameters converge to $\frac{1}{\beta_n}\tilde{\theta}_n$*

$$\lim_{n\to\infty} \tilde{\theta}_n^* \to \frac{1}{\beta_n}\tilde{\theta}_n$$

Proof. In Theorem 4.4.1, the stability proof shows the convergence of ε_n without the disturbance term.

$$\varepsilon_n = \boldsymbol{\phi}_n^T(\tilde{\theta}_n - \beta_n\tilde{\theta}_n^*) \to 0. \tag{4.49}$$

Then, recalling from (4.33), we know that ε_n^0 will also converge to zero for a bounded $\boldsymbol{\phi}_n$. We have

$$\varepsilon_{n+1}^0 = \boldsymbol{\phi}_{n+1}^T(\tilde{\theta}_{n+1} - \beta_{n+1}\tilde{\theta}_n^*) \to 0$$

which can be further rewritten as

$$\boldsymbol{\phi}_{n+1}^T(\tilde{\theta}_n - \beta_n\tilde{\theta}_n^*) \to 0 \tag{4.50}$$

by substituting $\tilde{\theta}_{n+1} = \frac{\beta_{n+1}}{\beta_n}\tilde{\theta}_n$ under the assumption of no external disturbance.

Combining the results of (4.49) and (4.50) and using the fact that $\tilde{\theta}_n - \beta_n\tilde{\theta}_n^*$ cannot be orthogonal to $\boldsymbol{\phi}_n$ and $\boldsymbol{\phi}_{n+1}$ since $\boldsymbol{\phi}_n$ and $\boldsymbol{\phi}_{n+1}$ span the whole state space, we can conclude that $\tilde{\theta}_n - \beta_n\tilde{\theta}_n^*$ will approach zero as $n \to \infty$. □

Table 4.1: Model Parameters

vehicle mass	m	2300.132	kg
vehicle rotational inertia	I_z	4400	kgm^2
distance from COG to front axle	L_f	1.505	m
distance from COG to rear axle	L_r	1.504	m
front tire conering stiffness	C_f	160776	N/rad^2
rear tire conering stiffness	C_r	254100	N/rad^2
gravity	g	9.80665	m/s^2

Table 4.2: Estimator Parameters (Algorithm 1)

sampling time	Δt	0.01 sec
covariance matrix of $w_k[k]$	W_k	$\text{diag}([0.2, 0.6, 0.05])$
covariance matrix of $v_k[k]$	V_k	0.05
covariance matrix of $w_d[k]$	W_d	$\text{diag}([6, 0.5, 0.1, 0.0002])$
covariance matrix of $v_d[k]$	V_d	$\text{diag}([0.1, 0.01])$
forgetting factor	λ	0.975
regularized term weighting	δ	0.02
yaw rate threshold	r_t	0.1 rad/s

From Theorem 4.4.2 and Theorem 4.4.3 we can conclude that the estimated tire cornering stiffness coefficients will converge to a neighbourhood of the true values when we include the external disturbance e_n term in the proof.

4.5 Algorithm Improvement

In this section, the proposed estimation algorithm is evaluated with real experimental tests. We first conducted a slalom and a severe single lane changing tests at Hyundai-Kia Motors California Proving Ground. The first test setting consists of eleven lined up cones, separated by 18 m. The vehicle is driven through the course in a slalom pattern at constant speed, 50 km/hr. The second one is a standardized maneuver which generates a peak lateral acceleration of approximately 0.6g. A further analysis and a small modification of Algorithm 1 are provided based on the estimation results.

4.5.1 Experimental Setup

Our experimental vehicle is a 5^{th} generation Hyundai Genesis equipped with a differential global positioning system (dGPS) Oxford TR3000. A real-time kinematics (RTK) technology is adopted to allow an accuracy down to 2-4 cm for position measurement. We will consider the measured sideslip angle provided from dGSP as a ground truth to validate the estimated

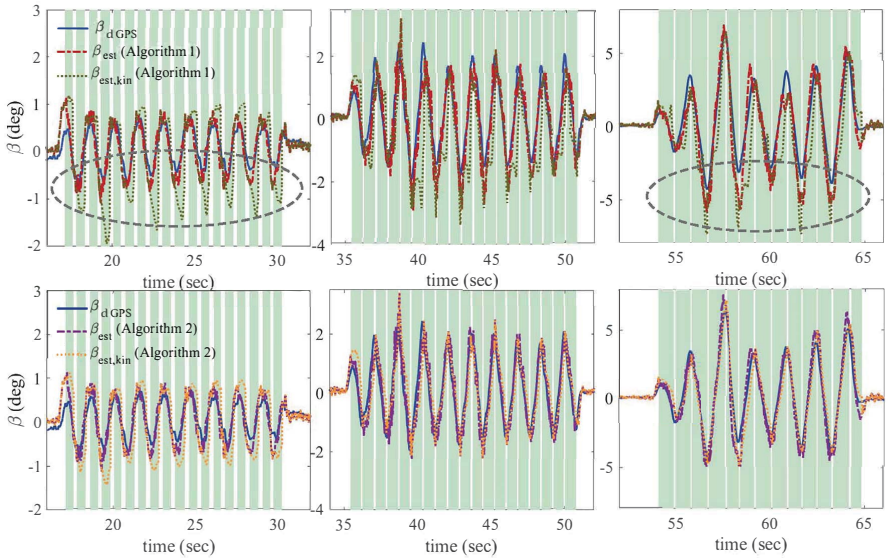


Figure 4.6: Performance comparison of Algorithm 1 and Algorithm 2 for a slalom test.

performance. The realtime computations are performed on a dSPACE DS1401 Autobox system which consists of a IBM PowerPC 750GL processor running at 900 MHz. The aforementioned hardware components communicate through a CAN bus and the estimation algorithm is executed at 100 Hz.

Table 4.1 shows the nominal model parameters of the test vehicle and Table 4.2 shows the estimation parameters for Algorithm 1. We initialize the measurement noise covariances by processing the measurement outputs while they are held constant. Since the values of the noise covariances are all small, we then apply a reasonable scaling factor to avoid the numerical issue before the tuning. The process noise covariance matrix is picked based on the unmodeled dynamics. According to the results shown in Table 4.2, we can see that the process noise covariance of the v_y equation is chosen to be relative bigger than other states since the coupling of the roll dynamics has been ignored and the gravity effect causes more influence on v_y dynamics. Similarly, we choose the process noise covariance of the sensor bias, d, to be significantly small because we believe that the offset is "nearly" constant. In other words, we can treat the dynamics of d as arbitrarily-slowly time-varying. For the forgetting factor, since it determines the rate of change of the weighting factors of the regression errors, we start with the value vary close to 1 for the fact that the tire cornering stiffness varies with the maneuver and our sampling rate is way fast enough to capture its varying speed. Then, we gradually decrease the value to allow more weighting on recent data to improve the performance.

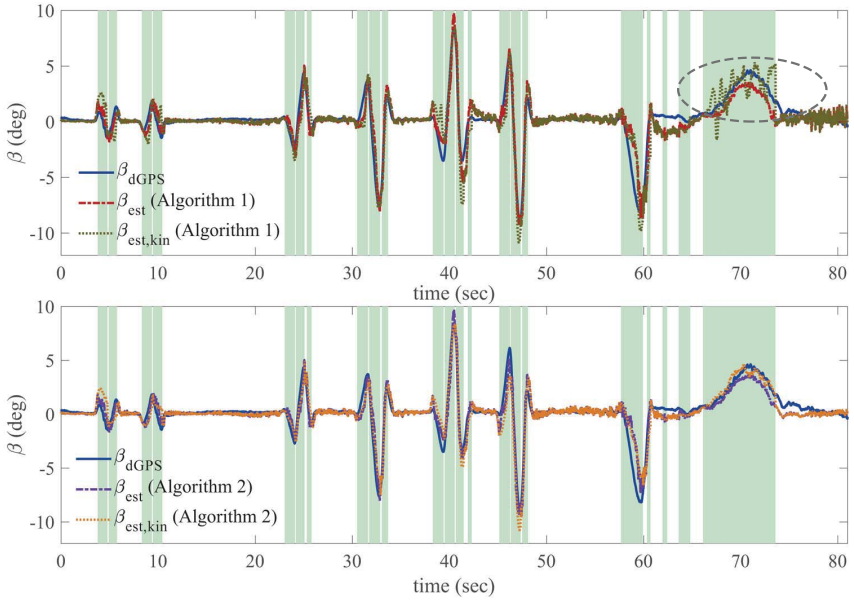


Figure 4.7: Performance comparison of Algorithm 1 and Algorithm 2 for a severe single lane changing.

4.5.2 Motivation

The estimation results of a slalom and a severe single lane changing tests are shown in the upper plots of Fig. 4.6 and Fig. 4.7 respectively. Note that the light green background represents the condition of $|r| > r_t$ indicating that the adaptation algorithm is active. Comparing with the solid blue (β_{dGPS}) and the dashdotted red (β_{est}) lines, we can see that the proposed method performs well. However, there is still room for improvement in the region highlighted with gray dashed lines. In these regions the estimated sideslip angle ($\hat{v}_{y,k}$) provided from the kinematics model (4.15) is noisy. This affects the output measurements Φ in the regression model which is used for the cornering stiffness adaptation. To address this issue, we proposed a small modification for Algorithm 1 which is described next.

4.5.3 Modification to Algorithm 1

In this section, we improve Algorithm 1 proposed in the previous section. We will show the performance of the new algorithm. However, the convergence analysis is harder to establish because of the tightly coupling between two observers.

According to the discussion above, we want to improve the estimation of the kinematics model by considering the bank angle effect. Start by deriving from the lateral dynamics (4.9) and the lateral acceleration models (4.10). We can get the following relation:

$$\dot{v}_y = -v_x r + a_y^{sen} - g \sin\phi - d. \tag{4.51}$$

Algorithm 2 Sideslip Angle Estimation (Algorithm 2)

initialize:
  $\hat{\mathbf{x}}_k[0] \leftarrow [v_x[0]\ v_y[0]\ 0]^T$, $\hat{\mathbf{x}}_d[0] \leftarrow [v_y[0]\ r[0]\ 0\ \ 0]^T$, $\tilde{\theta}_0^* \leftarrow 0$, $R_0 \leftarrow 0$, $\theta_0^* \leftarrow \theta^+ + \tilde{\theta}_0^*$,
  $P_k[0] \leftarrow P_{k,0}$, $P_d[0] \leftarrow P_{d,0}$ *// initialize prior means and estimate error covariance matrices for* EKF
for i = 1 to k+1 **do**
  $\hat{\mathbf{x}}_d[i] \leftarrow \text{EKFupdate}(\hat{\mathbf{x}}_d[i-1], \mathbf{u}_d[i-1], \mathbf{u}_d[i], \mathbf{y}_d[i], P_d[i-1], \theta_{i-1}^*)$ *// dynamics model (4.12)*
  $\hat{\mathbf{x}}_k[i] \leftarrow \text{EKFupdate}(\hat{\mathbf{x}}_k[i-1], \mathbf{u}_k[i-1], \mathbf{y}_k[i], P_k[i-1]))$ *// kinematics model (4.15)*
  $\mathbf{u}_k[i] \leftarrow [a_x[i]\ \ a_y^{sen}[i] - g\sin\hat{\phi}_d[i] - \hat{d}_d[i]]^T$
  if $|r_i| \geq r_t$ **and** $1/c_t \leq |\Phi_i^T(2,1)/\Phi_i^T(2,2)| \leq c_t$ **then**
    $R_i = \lambda R_{i-1} + \Phi_i \Phi_i^T$ *// obtain the input measurement Φ_i from (4.18)*
    $\theta_i^* \leftarrow \theta^+ + \text{AdaptationUpdate}(R_i, \tilde{\theta}_{i-1}^*)$ *// apply a recursive update law (4.21)*
  else
    $\theta_i^* \leftarrow \theta_{i-1}^*$
    $\hat{\mathbf{x}}_k[i] \leftarrow [v_x[i]\ \hat{v}_{y,d}[i]\ \ \sin\hat{\phi}_d[i]]^T$ *// update the state estimates of the* EKF *for model (4.15)*
    $P_k[i] \leftarrow \text{diag}(0, P_d[i](1,1), P_d[i](3,3))$ *// update the error covariance matrix for model (4.15)*
  end if
  $\beta[i] \leftarrow \tan^{-1}(\hat{v}_{y,d}[i]/v_x[i])$ *// calculate the sideslip angle*
end for

Then, based on the result shown in (4.51), the original kinematics model (4.15) in Algorithm 1 can be modified into:

$$A_k(t) = \begin{bmatrix} 0 & r(t) \\ -r(t) & 0 \end{bmatrix}, \quad B_k(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C_k(t) = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{4.52}$$

with the estimated state and the input vectors defined as:

$$\hat{\mathbf{x}}_k = \begin{bmatrix} v_x \\ v_y \end{bmatrix}, \quad \mathbf{u}_k = \begin{bmatrix} a_x \\ a_y^{sen} - g\sin\hat{\phi}_d - \hat{d}_d \end{bmatrix}.$$

We can see that the measured lateral acceleration in $\mathbf{u}_k$ is added with an additional term, $-g\sin\hat{\phi}_d - \hat{d}_d$, where $\sin\hat{\phi}_d$ and $\hat{d}_d$ are the estimated values from the dynamics model (4.12). With this modification, the kinematics model (4.52) does not remain unaffected by the vehicle parameters anymore. However, we can claim that the estimated term of $-g\sin\hat{\phi}_d - \hat{d}_d$ from the dynamics model is relatively less sensitive to the model error in the normal driving situations for $\dot{v}_y$ being small and slowly varying since

$$-g\sin\hat{\phi}_d - \hat{d}_d = \hat{\dot{v}}_{y,d} + v_x\hat{r} - a_y^{sen}. \tag{4.53}$$

By observing (4.53), we can expect that $-g\sin\hat{\phi}_d - \hat{d}_d$ will mostly depend on the error of $\hat{\dot{v}}_{y,d}$ because $\hat{r}$ and a_y^{sen} are directly relevant to the values measured from the sensors. Although

it will be bias more when $\dot{v}_y$ is large, we still can expect it with a similar trend and without too much difference from the true value.

The modified version of the estimator is provided in Algorithm 2. Moreover, we add an additional condition listed in (4.46) to guarantee well-conditioned measurement data for tire cornering stiffness adaptation. The estimated performance of Algorithm 2 is shown in the bottom plots of Fig. 4.6 and Fig. 4.7. We can see a significant improvement in the estimated value of $\hat{v}_{y,k}$.

4.6 Experimental Verification

Having a modified version of the estimator (Algorithm 2), to evaluate its robustness, three more different tests of severe and normal steering maneuvers under different road conditions are conducted and all the tests are listed as follows:

1) a slalom test on a low friction flat road,

2) a severe single lane changing on a normal flat road,

3) a steady circular motion test on a normal flat road,

4) a double lane changing test on a road with significant bank angle, and

5) a stop-N-turn test on a normal road.

To display the advantage of Algorithm 2, we further compare the experimental results with other two methods:

1) Dynamics observer: a dynamics estimator with a state augmented with bank angle and sensor bias without cornering stiffness adaptation, and

2) Hybrid observer: a hybrid estimator switching between the dynamics model and a kinematics model described in Algorithm 2.

All parameters required in Algorithm 2 are the same as Algorithm 1 listed in Table 4.2 except that the covariance matrix of $w_k[k]$ is set to be $\text{diag}([0.2, 0.6])$ and the maximum conditional number, c_t, is 20.

4.6.1 Experimental results for Algorithm 2

The experimental results are shown in Fig. 4.9-4.18 and the comparison of RMS error performances can be found in Fig. 4.8. As we can see, both the hybrid and dynamics observers exhibit a large RMS value under some driving situations. Algorithm 2 provides superior performances in all scenario tests.

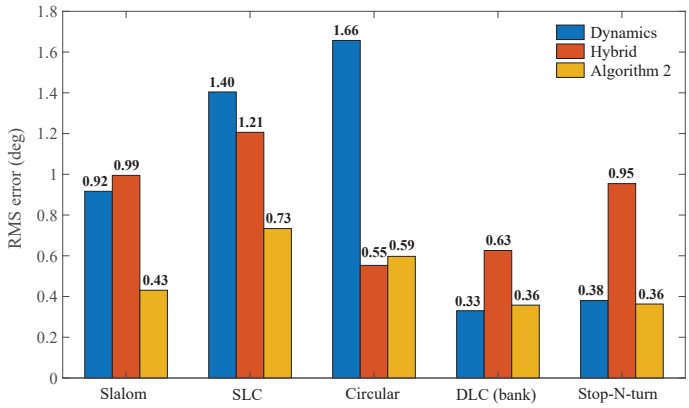


Figure 4.8: The root mean square errors of the proposed method compared with the existing methods.

Starting from slalom, severe single lane changing and steady circular motion tests, we can see that Fig. 4.9, Fig. 4.11 and Fig. 4.13 demonstrate the effectiveness of the proposed estimator. For the dynamics model-based approach, it is obvious that there is a big disparity between the true and the estimated sideslip angle when the vehicle enters the nonlinear tire region. As expected, for the method switching between dynamics and kinematics models, we can see a discontinuous estimating during the transition. The longitudinal velocity, adapted tire cornering stiffnesses, estimated bank angle and sensor bias for all scenario tests are shown in Fig. 4.10, Fig. 4.12 and Fig. 4.14. The adapted cornering stiffnesses becomes smaller for a low friction road condition or entering the nonlinear tire region. Since the estimated bank angle is affected by the vehicle roll angle, we can conclude that all the estimated bank angle resulting within -4°- 4° may be questionable. However, we still trust the estimation for large bank angles. Fig. 4.15 and Fig. 4.16 shows the experimental results of double lane change tests on a road with a large bank angle. The estimate performance of the switching algorithm is poor since the kinematics model is sensitive to the lateral acceleration measurement disturbance introduced from the bank angle. Fig. 4.16 confirms the ability of the proposed algorithm to estimating the bank angle, with the estimated value of the bank angle converging to the true value of 14°. The adapted cornering stiffnesses remain unchanged because of the mild driving condition. Finally, we evaluate the performance of Algorithm 2 by conducting a stop-N-turn test for a varying low speed condition (Fig. 4.17 and Fig. 4.18). Again, the results are very promising.

In summary, the proposed algorithm stands out for its robustness in model error and measurement disturbance. It can be used for any driving situation with different road conditions. In addition, reliable estimates for bank angle and sensor bias are also available.

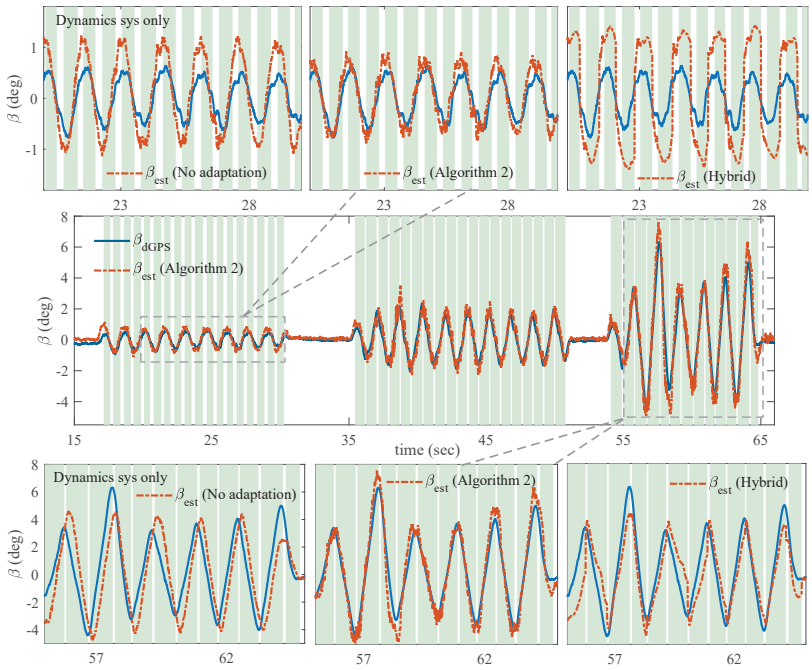


Figure 4.9: Comparison of the sideslip angle estimation for a slalom test.

4.7 Conclusion

This paper developed a real-time algorithm for estimation of sideslip angle using inexpensive sensors normally available for electronic stability control (ESC) applications. The algorithm utilizes a kinematics observer to improve the estimation based on a vehicle dynamics model. It also provides estimates of road bank angles, lateral acceleration sensor bias and tire cornering stiffness. The algorithm performance is evaluated through several experimental tests and the results indicate that the algorithm provides a good estimate of the vehicle sideslip angle both in normal and extreme maneuvers with different road conditions.

4.8 Appendix

4.8.1 Asymptotic Hyperstability and Strictly Positive Real [82]

Definition 1. *The feedback system shown in Fig. 4.3 is asymptotically hyperstable if the state x_k of the linear time invariant system converges to zero for $k \to \infty$*

Theorem 4.8.1. *The feedback system shown in Fig. 4.3 is asymptotically hyperstable if and only if*

1. *the linear time invariant system is strictly positive real.*

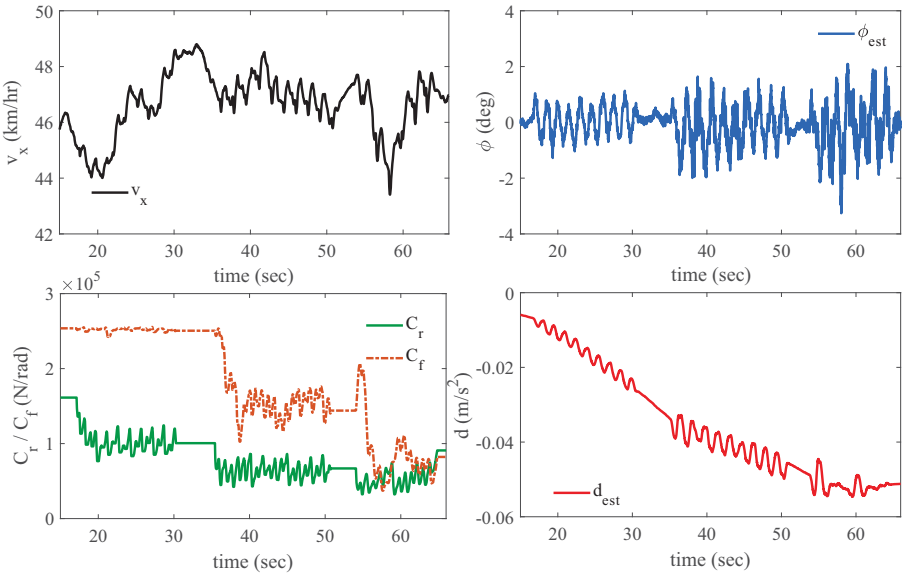


Figure 4.10: Slalom test results for Algorithm 2: longitudinal velocity; adapted cornering stiffnesses; estimated bank angle and sensor bias.

2. *the nonlinear feedback block satisfies Popov inequality:*

$$\exists\ \gamma > 0, \quad \sum_{k=1}^{k_1} w_k^T v_k \geq -\gamma^2 \quad \forall k_1 \geq 0$$

3. *the output signal, w_k, of the nonlinear block is bounded.*

Theorem 4.8.2. *A single input single output discrete-time system, $G(z)$, is strictly positive real if*

1. *the system does not possess any pole outside of or on the unit circle on z-plane.*
2. $\forall\ |\omega| < \pi,\ G(e^{-j\omega}) + G(e^{j\omega}) > 0$

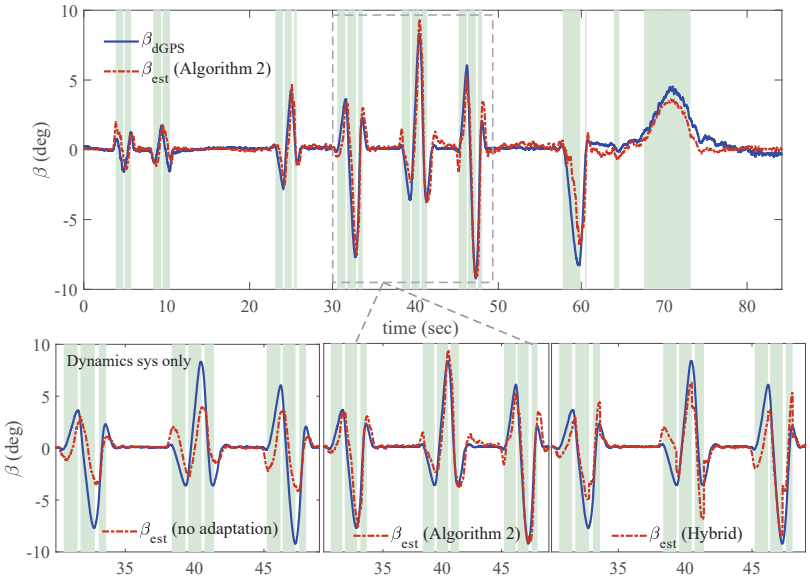


Figure 4.11: Comparison of the sideslip angle estimation for a severe single lane changing maneuver.

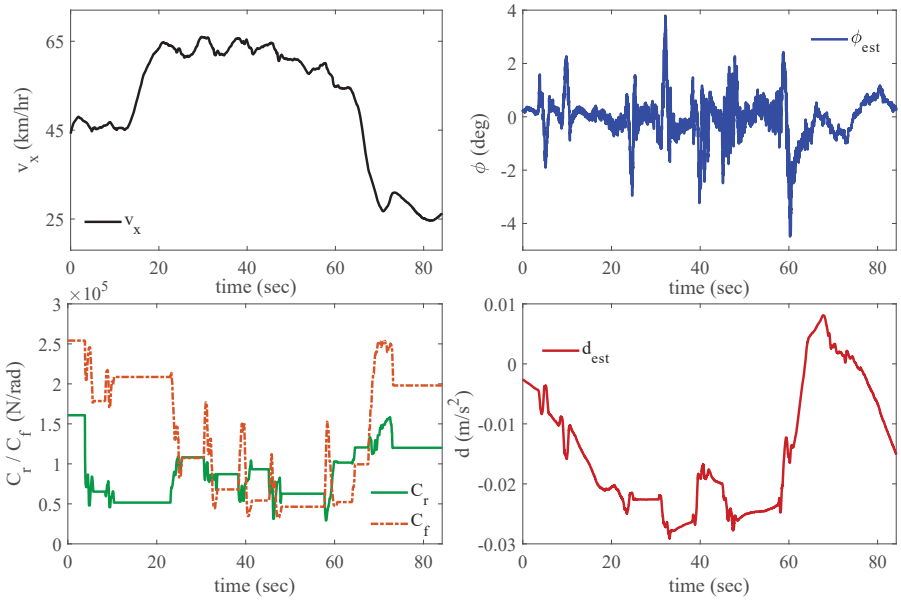


Figure 4.12: Severe single lane changing test results for Algorithm 2: longitudinal velocity; adapted cornering stiffnesses; estimated bank angle and sensor bias.

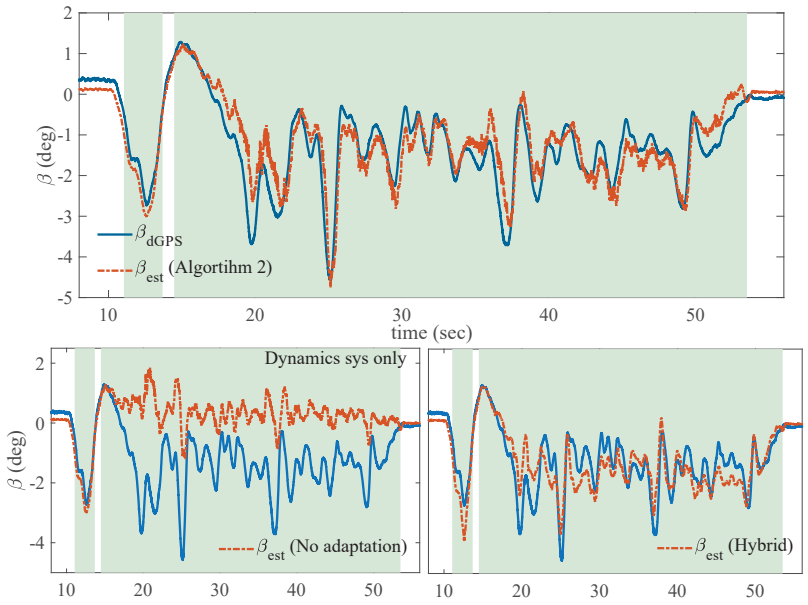


Figure 4.13: Comparison of the sideslip angle estimation for a steady circular motion.

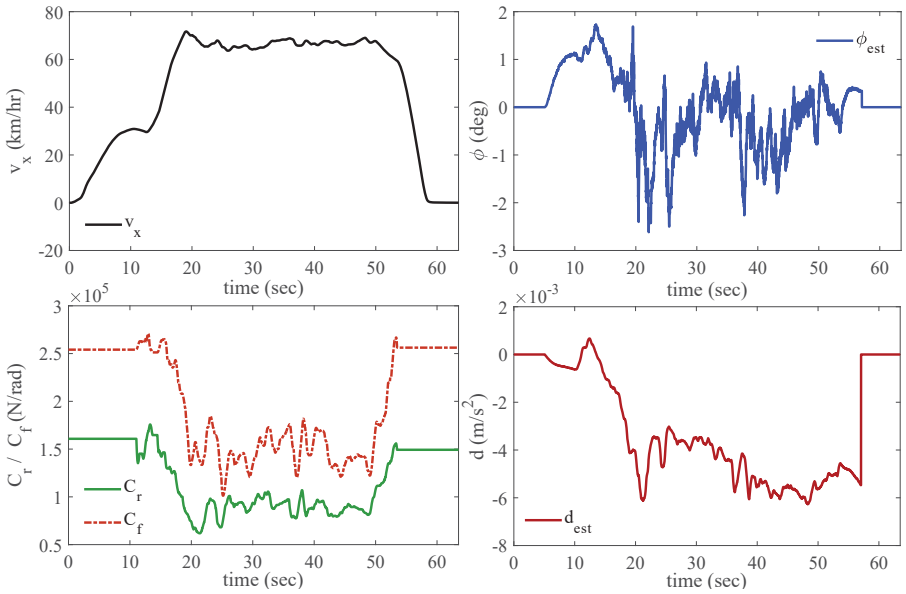


Figure 4.14: Steady circular motion test results for Algorithm 2: longitudinal velocity; adapted cornering stiffnesses; estimated bank angle and sensor bias.

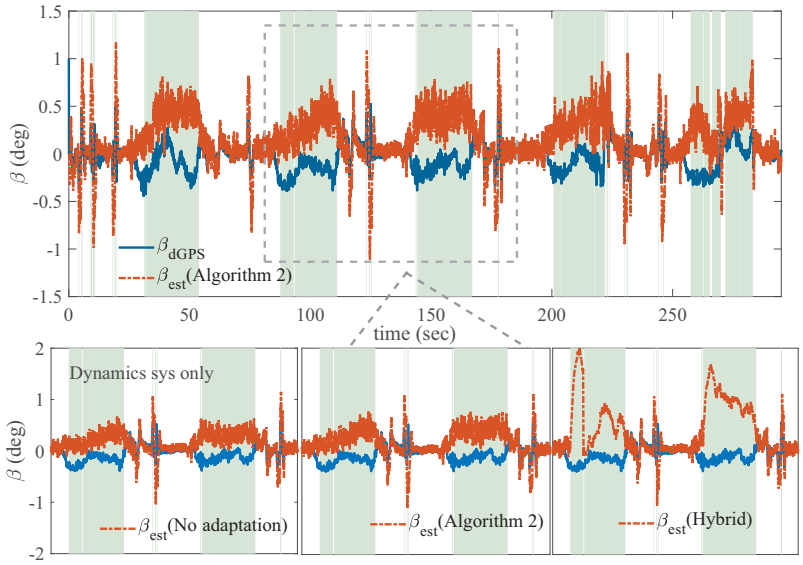


Figure 4.15: Comparison of the sideslip angle estimation for double lane changing on a bank.

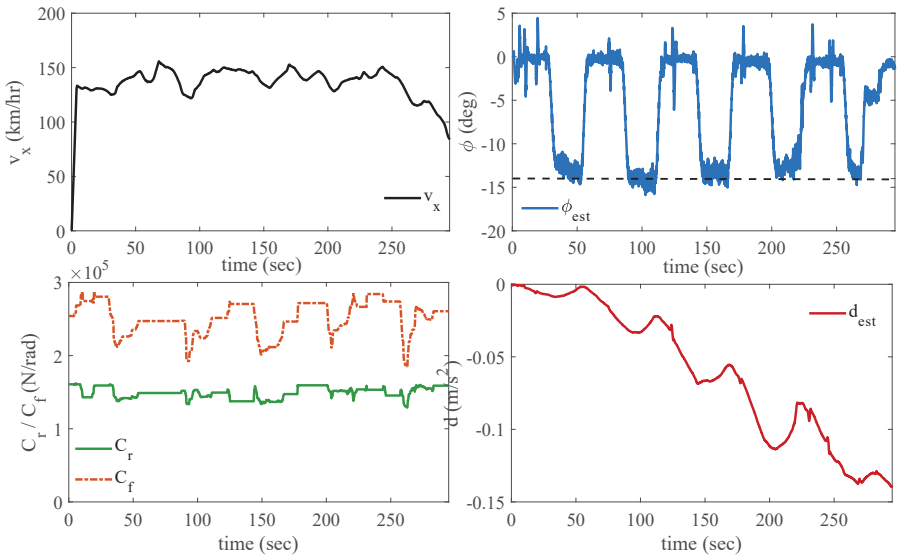


Figure 4.16: On-bank double lane changing test results for Algorithm 2: longitudinal velocity; adapted cornering stiffnesses; estimated bank angle and sensor bias.

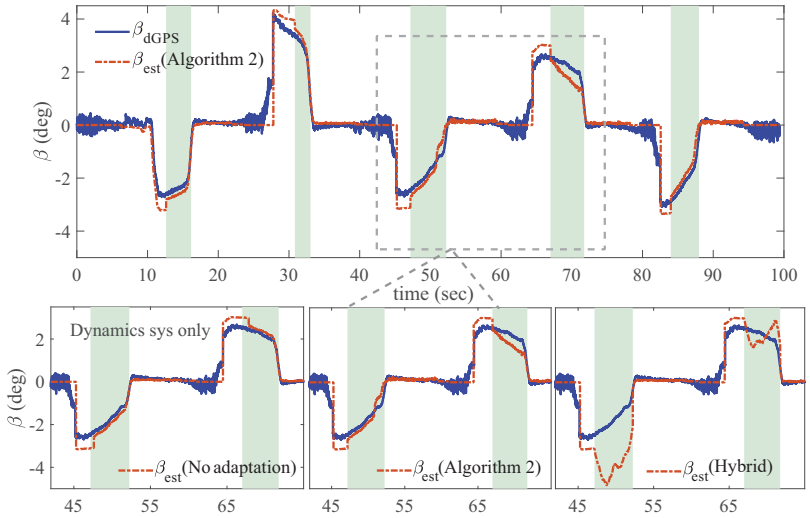


Figure 4.17: Comparison of the sideslip angle estimation for a stop-N-turn motion.

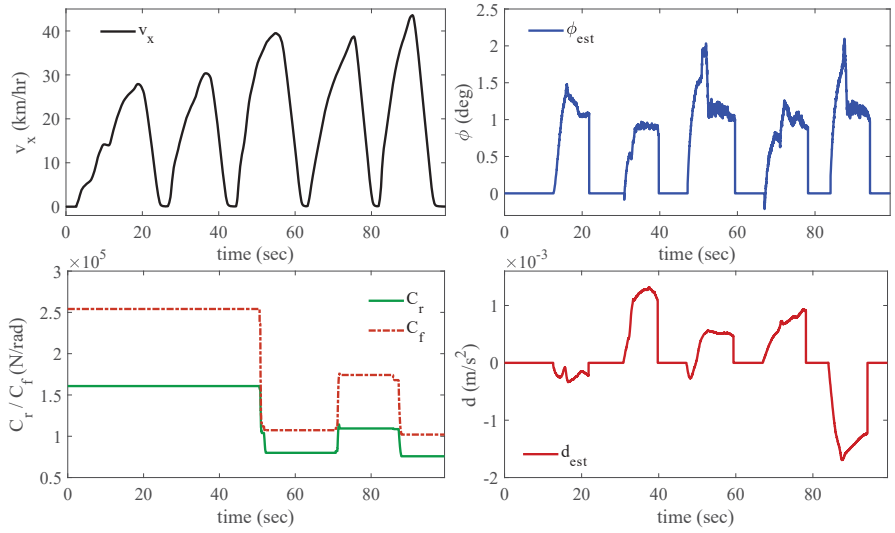


Figure 4.18: Stop-N-turn test results for Algorithm 2: longitudinal velocity; adapted cornering stiffnesses; estimated bank angle and sensor bias.

Chapter 5

Autonomous Figure-8 Tracking

This chapter presents a performance assessment of the adaptive sliding control algorithm introduced in Chapter 2 with an example of Figure-8 tracking.

5.1 Modeling and Trajectory Planning

This section introduces a vehicle model and a trajectory planning of Figure-8 tracking that will be utilized throughout the remainder of this chapter.

5.1.1 Vehicle Model in Terms of Body-fixed and Inertial Coordinates

Consider a model of a "front-wheel-drive" vehicle with two degrees of freedom depicted in Fig. 5.1. Given an inertial and a body-fixed coordinate systems defined as $\{\hat{X}_0, \hat{Y}_0.\hat{Z}_0\}$ and $\{\hat{x}_b, \hat{y}_b, \hat{z}_b\}$, we can describe a dynamics bicycle model using the following nonlinear differential equations:

$$\dot{v}_x = \frac{1}{m}(F_{xf}\cos\delta_f - F_{yf}\sin\delta_f) + rv_y \tag{5.1}$$

$$\dot{v}_y = \frac{1}{m}(F_{yf}\cos\delta_f + F_{xf}\sin\delta_f + F_{yr}) - rv_x \tag{5.2}$$

$$\dot{r} = \frac{1}{I_z}(L_f F_{yf}\cos\delta_f + L_f F_{xf}\sin\delta_f - L_r F_{yr}) \tag{5.3}$$

$$\dot{\psi} = r \tag{5.4}$$

$$\dot{X} = v_x\cos\psi - v_y\sin\psi \tag{5.5}$$

$$\dot{Y} = v_x\sin\psi + v_y\cos\psi \tag{5.6}$$

where m and I_z denote the vehicle mass and the equivalent yaw moment of inertia, respectively. L_f and L_r denote the distances from the vehicle's center of gravity (CoG) to the front

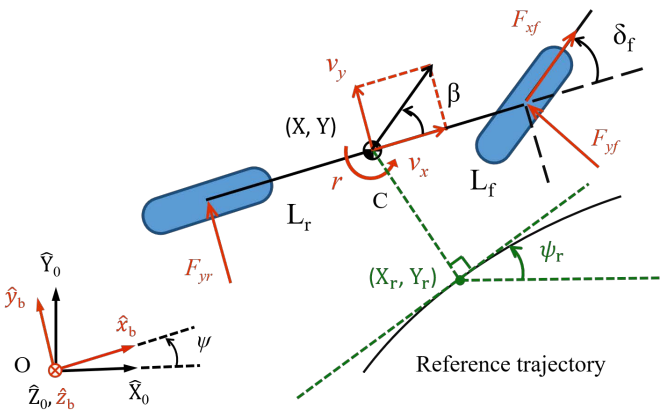


Figure 5.1: Dynamics bicycle model in terms of body-fixed and inertial coordinates of a front-wheel-drive vehicle.

and rear axles, respectively, and δ_f denotes the front steering angle. v_x and v_y denote the longitudinal and lateral velocities represented in the body fixed coordinate system, respectively, and r denotes the yaw rate. X and Y denote the inertial coordinates of the vehicle, and ψ is the angular heading. The longitudinal tire force is F_{xf} and the corresponding front and rear lateral tire forces are represented by F_{yf} and F_{yr}.

To simplify the model, we assume that the terms of $F_{yf} \sin \delta_f$ and $F_{xf} \sin \delta_f$ are negligible because they are relatively small to others for a small front steering angle. Then, the equations (5.1), (5.2) and (5.3) can be rewritten into:

$$\begin{aligned} \dot{v}_x &= \frac{1}{m} F_{xf} \cos \delta_f + r v_y \\ \dot{v}_y &= \frac{1}{m} (F_{yf} \cos \delta_f + F_{yr}) - r v_x \\ \dot{r} &= \frac{1}{I_z} (L_f F_{yf} \cos \delta_f - L_r F_{yr}). \end{aligned} \tag{5.7}$$

The magic formula are used to model the lateral tire forces described as follows [83]:

$$F_{y\star} = 2D_\star \sin \left(C_\star \tan^{-1} \left((1 - E_\star) B_\star \alpha_\star + E_\star \tan^{-1} \left(B_\star \alpha_\star \right) \right) \right) \tag{5.8}$$

where the symbol $\star$ is either f or r representing the front or rear tire. $B_\star$, $C_\star$, $D_\star$, and $E_\star$ are the corresponding empirical coefficients that characterize the tire model. Finally, the front and rear tire side slip angles are defined as

$$\alpha_f = \delta_f - \tan^{-1}\left(\frac{v_y + L_f r}{v_x}\right) \tag{5.9}$$

$$\alpha_r = -\tan^{-1}\left(\frac{v_y - L_r r}{v_x}\right). \tag{5.10}$$

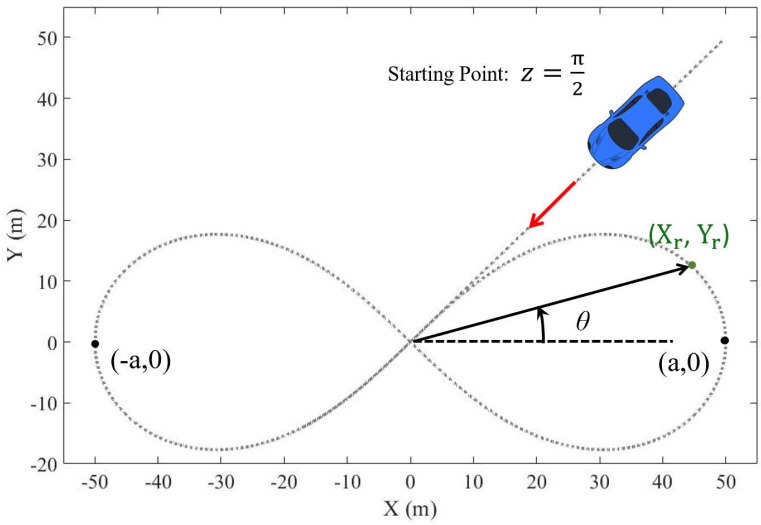


Figure 5.2: The lemniscate for Figure-8.

5.1.2 Trajectory Planning (Figure-8)

The test scenario is a Figure-8 path following maneuver which has the form of a lemniscate as shown in Fig. 5.2. In algebraic geometry, a lemniscate can be any of several Figure-8 or ∞-shaped curves [84]. Here, we just choose the most common one that has the form of:

$$(X_r^2 + Y_r^2)^2 = a^2(X_r^2 - Y_r^2) \tag{5.11}$$

represented in Cartesian space where a is the width in the major axis. Note that choosing this type of track has several advantages. One is that it provides greater complexity than a steady state cornering and a double lane changing as it contains more featured maneuvers. Another one is its continuity in curvature. Since our work is not focusing on the motion planning, having this property can help us reduce a significant amount of work in smoothing out the path [85]. Finally, the most important one is that it can be parameterized by one parameter, z, which helps simplify the process of finding the vehicle's projection point on the path (reference) in real time. To obtain the parametric equation of (5.11), we fist switch it to polar coordinate system, (r, θ) as:

$$r^2 = a^2 \cos(2\theta). \tag{5.12}$$

where the angles, θ, is defined in the ranges of $-\frac{\pi}{4} < \theta < \frac{\pi}{4}$ and $\frac{3\pi}{4} < \theta < \frac{5\pi}{4}$. Then, transform the variable $\tan\theta = \sin z$ for $z = [\frac{\pi}{2}, \frac{5\pi}{2}]$. The parametric equations of the lemniscate become

$$X_r(z) = \frac{a\cos(z)}{1 + \sin^2(z)} \tag{5.13}$$

$$Y_r(z) = \frac{a\sin(z)\cos(z)}{1 + \sin^2(z)} \tag{5.14}$$

and the arc length can be given by

$$s(z) = \sqrt{2}a \int_0^z \Big(3 - \cos(2z)\Big)^{-1/2} dz = aF(z, -1) \tag{5.15}$$

where $F(z, -1)$ is an elliptic integral of the first kind. Generally, this integration cannot be expressed as an explicit function. However, the entire arc length of Figure-8 can still be approximated by

$$s(2\pi) = aF(2\pi, -1) \approx 5.244a. \tag{5.16}$$

The curvature and tangential angle can also be expressed as a function of z:

$$\kappa_r(z) = \frac{3\sqrt{2}\cos(z)}{a\sqrt{3 - \cos(2z)}} \tag{5.17}$$

$$\psi_r(z) = \begin{cases} \tan^{-1}\big(\sin z\big) + \frac{\pi}{2}\pi & \text{if } \frac{3\pi}{2} \le z \le \frac{5\pi}{2} \\ \tan^{-1}\big(\sin z\big) - \frac{5\pi}{2}\pi & \text{if } \frac{\pi}{2} \le z < \frac{3\pi}{2}. \end{cases} \tag{5.18}$$

Having the above properties, our next step is to plan how the vehicle should move based on the velocity and time so that the resulting trajectory will be dynamically feasible and tracked by the lower level controller. Given a fixed path of a predefined a, we can plan the desired acceleration, velocity and travel length along the path as a function of time, t. In our case, we choose

$$s_r(t) = -\frac{AT^2}{4\pi^2}\sin\big(\frac{2\pi}{T}t\big) + \big(\frac{AT}{2\pi} + v_0\big)t \tag{5.19}$$

$$v_r(t) = -\frac{AT}{2\pi}\cos\big(\frac{2\pi}{T}t\big) + \frac{AT}{2\pi} + v_0 \tag{5.20}$$

$$a_r(t) = A\sin\big(\frac{2\pi}{T}t\big) \tag{5.21}$$

where T is the entire arc's traveling time. A is the amplitude of the acceleration and v_0 is the initial velocity. All of them are the design parameters such that $s(T) = 5.244a$. Fig. 5.3 shows an example of the desired profiles of the trajectories for $a = 50$. Note that reasonable choices of the shapes in $a_r(t)$ and $v_r(t)$ have been made for the purpose of consistency in the lap. Finally, we need to represent equations (5.20) and (5.21) as a function of z as well for the use of on-line referencing. To do this, we can sample the travel length and numerically solve the planned velocity and acceleration off-line to construct the static maps of $v(z)$ and $a(z)$. In summary, now we have all the references, $X_r, Y_r, \kappa_r, \psi_r, v_r, a_r$ parameterized as a function of z. According to this, we can definitely consider more detailed plans in the side slip angle, $\beta_r = v_{xr}/v_{yr}$, with a constraint of $v_{xr}^2 + v_{yr}^2 = v_r^2$ for some special cases (i.e. a drifting case). However, since we just plan to have a normal driving situation, we then assume that $\beta_r = 0$ all the times and $\dot{\kappa}_r$ are too small to be negligible.

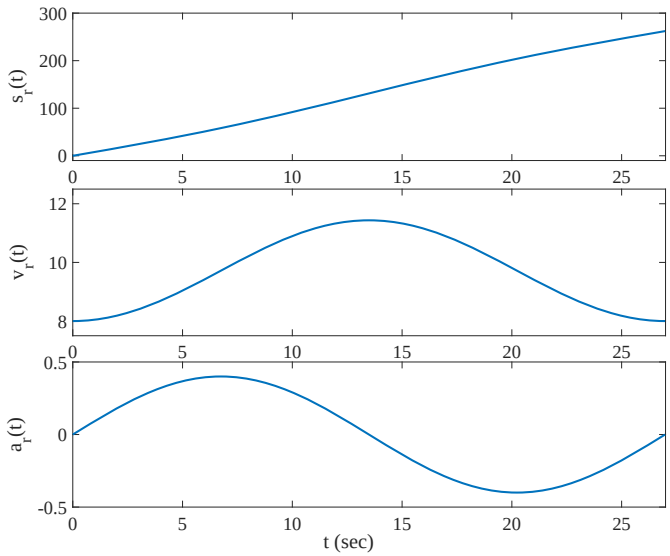


Figure 5.3: The desired profiles of the trajectories for Figure-8.

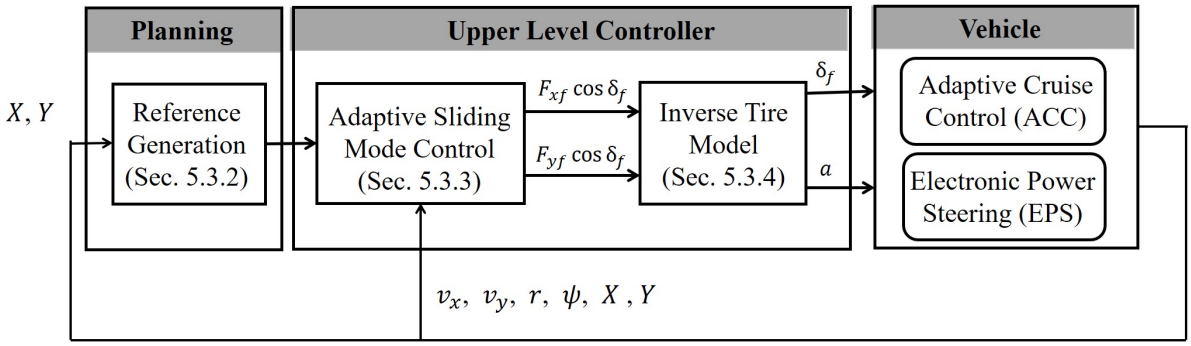


Figure 5.4: Block diagram of the autonomous vehicle feedback control loop.

5.2 Control Design for Path Following

In this section, our task is to develop a nonlinear control architecture for a path following problem. Although the system we introduced in Section 5.1.1 has already included a certain amount of nonlinearity, a high variant of the tire-road condition may still introduced large uncertainties into the model. This will easily degrade the control performance unless we can on-line estimate the road friction coefficient. According to the above discussion, a controller that is able to handle a wide range of situations would be desirable. Therefore, an adaptive sliding mode control becomes a reasonable candidate for our design.

5.2.1 Control System Architecture

Fig. 5.4 presents the structure of the feedback control system which is designed to be hierarchical with two levels: one is the motion planning block and the other one is the upper level controller. In our case, the algorithm in the motion planning will be relatively simple, since the path and the reference trajectories are all parameterized into one variable, z, as shown in Section 5.1.2. Therefore, the only thing we need to do becomes how to find the variable z of the vehicle's projection point on the path. The details explanation will be introduced in Section 5.2.2.

Once we obtain all the references, we can implement an adaptive control law introduced in Chapter 2. Note that since it is not efficient to include the entire nonlinear tire model into the control design, we will first treat the lumped tire forces as the control inputs. Then, implement the inversed tire model to obtain the desired front steering angle and the desired acceleration as our test vehicle has the setup of the control authority in the Electronic Power Steering (EPS) and the Adaptive Cruise Control (ACC) system.

5.2.2 Reference Generation

As already mentioned in the previous section, we need to find the vehicle's projection point, $P = [X_p,\ Y_p]^T$, to get the references (see Fig. 5.1). The idea is simply based on an orthogonal projection of the vehicle to the desired path, $\mathcal{S}$, which can be formulated as the following optimization problem:

$$P^* = \arg\min_{P \in \mathcal{S}} \|C - P\|_F^2. \tag{5.22}$$

where $C = [X,\ Y]^T$ is the current feedback position of the vehicle. Several approximation methods have been proposed to solve this problem in the literature [86]- [87]. In our case, since all the references have been parameterized into one variable, z, the problem become one dimensional and can be easily solved by the Newton's method on-line. Consider the optimization problem of $\min_z f(z)$ where

$$f(z) = \Big[X - \frac{a\cos(z)}{1+\sin^2(z)}\Big]^2 + \Big[Y - \frac{a\sin(z)\cos(z)}{1+\sin^2(z)}\Big]^2. \tag{5.23}$$

The Newton's method is applied to find the roots of $f'(z^*) = 0$. By observing (5.23), we can know that there could be multiple solutions for $f'(z^*) = 0$. To simplify the problem and guarantee a right convergence value, we need to initialize the algorithm based on the previous optimal solution. Define a new function

$$h(z) = -2aY - 2aX\sin^3 z + 6aY\sin^2 z + 6aX\sin z - 2a^2\sin(2z) \tag{5.24}$$

such that $h(z) = 0$ is equivalent to $f'(z) = 0$. The on-line algorithm can be obtained in Algorithm 3.

Algorithm 3 Newton's Method (An Orthogonal Projection of the Vehicle to the Path)

initialize: $z_0^* = \pi/2$
for k = 1,2,... **do**
  initialize: $z_0 = z_{k-1}^*$
  for i = 0,1,2,... **do**
    if $h^2(z_i) \leq \epsilon$ **then**
      $z_k^* = z_i$; **return** wrap(z_k^*) *//wrap z_k^* to the interval of $[\pi/2,\ 5\pi/2]$*
    end if
    $z_{i+1} = z_i - h(z_i)/h'(z_i)$
    if $|z_{i+1} - z_i| \leq \epsilon$ **then**
      $z_k^* = z_{i+1}$; **return** wrap(z_k^*) *//wrap z_k^* to the interval of $[\pi/2,\ 5\pi/2]$*
    end if
  end for
end for

5.2.3 Adaptive Sliding Mode Control Algorithm

In this section, our goal is to design an adaptive controller to track a path with the desired velocity and acceleration trajectories. Consider the dynamics bicycle model described in Section 5.1.1 with "$F_{yf}\cos\delta_f$" and "$F_{xf}\cos\delta_f$" defined as a control input. Given the references of X_r, Y_r, ψ_r, κ_r, v_r and a_r calculated from z^*, we define $e_X = X - X_r$, $e_Y = Y - Y_r$ and $e_\psi = \psi - \psi_r$ and design the stable sliding surfaces, s_Y, s_X and s_ψ as:

$$s_X = \dot{e}_X + \lambda_X e_X = v_x\cos\psi - v_y\sin\psi - \dot{X}_r + \lambda_X e_X \tag{5.25}$$

$$s_Y = \dot{e}_Y + \lambda_Y e_Y = v_x\sin\psi + v_y\cos\psi - \dot{Y}_r + \lambda_Y e_Y \tag{5.26}$$

$$s_\psi = \dot{e}_\psi + \lambda_\psi e_\psi = \dot{\psi} - \dot{\psi}_r + \lambda_\psi e_\psi \tag{5.27}$$

where $\lambda_X > 0$, $\lambda_Y > 0$ and $\lambda_\psi > 0$ are the design parameters to determine the error convergence rates on the sliding surfaces and

$$\dot{X}_r = v_{xr}\cos\psi_r - v_{yr}\sin\psi_r = v_r\cos\psi_r \tag{5.28}$$

$$\dot{Y}_r = v_{xr}\sin\psi_r + v_{yr}\cos\psi_r = v_r\sin\psi_r \tag{5.29}$$

$$\dot{\psi}_r = v_r\kappa_r. \tag{5.30}$$

Then, applying the time derivative to s_X, s_Y and s_ψ, we can obtain the following adaptive sliding mode control law proposed in Chapter 2 as:

$$\dot{s}_\star = \ddot{e}_\star + \lambda_\star\dot{e}_\star = -k_\star s_\star - \hat{\mu}_\star\text{sgn}(s_\star) \tag{5.31}$$

$$\dot{\hat{\mu}}_\star = \begin{cases} \frac{1}{\rho_\star}\left[1 - \frac{2\phi_\star^2}{(|s_\star| + \phi_\star)^2}\right] & \text{if } \hat{\mu}_\star \geq 0 \\ 0 & \text{if } \hat{\mu}_\star < 0 \end{cases}, \quad \text{for } \star \in \{X,\ Y,\ \psi\}. \tag{5.32}$$

Note that $\ddot{e}_Y$, $\ddot{e}_X$ and $\ddot{e}_\psi$ can be derived from equations (5.5), (5.6) and (5.7) as:

$$\ddot{e}_Y = \frac{F_{xf}\cos\delta_f}{m}\sin\psi + \frac{F_{yf}\cos\delta_f}{m}\cos\psi + \frac{F_{yr}}{m}\cos\psi - a_r\sin\psi_r - v_r^2\kappa_r\cos\psi_r \tag{5.33}$$

$$\ddot{e}_X = \frac{F_{xf}\cos\delta_f}{m}\cos\psi - \frac{F_{yf}\cos\delta_f}{m}\sin\psi - \frac{F_{yr}}{m}\sin\psi - a_r\cos\psi_r + v_r^2\kappa_r\sin\psi_r \tag{5.34}$$

$$\ddot{e}_\psi = \frac{F_{yf}\cos\delta_f}{I_z}L_f - \frac{F_{yr}}{I_z}L_r - a_r\kappa_r - v_r\dot{\kappa}_r \tag{5.35}$$

where F_{yr} is calculated from the magic formula (5.8) based on the current state feedback. The adaptive control parameters $k_\star$, $\rho_\star$ and $\phi_\star$ are chosen to be positive as discussed in Chapter 2. Finally, by rearranging (5.31) into a linear system with 2 variables and 3 algebraic equations, $A\mathbf{u} = b$, for

$$\begin{aligned} A &= \begin{bmatrix} \frac{\cos\psi}{m} & -\frac{\sin\psi}{m} \\ \frac{\sin\psi}{m} & \frac{\cos\psi}{m} \\ 0 & \frac{L_f}{I_z} \end{bmatrix}; \qquad \mathbf{u} = \begin{bmatrix} F_{xf}\cos\delta_f \\ F_{yf}\cos\delta_f \end{bmatrix} \\ b &= \begin{bmatrix} \frac{F_{yr}\sin\psi}{m} + a_r\cos\psi_r - v_r^2\sin\psi_r\kappa_r - \lambda_X\dot{e}_X - k_X s_X - \hat{\mu}_X\mathrm{sgn}(s_X) \\ \frac{-F_{yr}\cos\psi}{m} + a_r\sin\psi_r + v_r^2\cos\psi_r\kappa_r - \lambda_Y\dot{e}_Y - k_Y s_Y - \hat{\mu}_Y\mathrm{sgn}(s_Y) \\ \frac{L_r F_{yr}}{I_z} + a_r\kappa_r - \lambda_\psi\dot{e}_\psi - k_\psi s_\psi - \hat{\mu}_\psi\mathrm{sgn}(s_\psi) \end{bmatrix}, \end{aligned} \tag{5.36}$$

we can obtain the control inputs with the left inverse of A as $\mathbf{u} = (A^TA)^{-1}A^Tb$. By observing (5.36), the control inputs is a least square solution that cannot guarantee all the designed sliding variables' dynamics be satisfied. This will result in an ultimate boundedness for the error responses. According to this, we can introduce an additional weighting matrix:

$$W = \begin{bmatrix} W_X & 0 & 0 \\ 0 & W_Y & 0 \\ 0 & 0 & W\psi \end{bmatrix} \succ 0 \tag{5.37}$$

into the controller design to enable the tuning of the relative weight on each sliding surface's dynamics. Therefore, the resultant solution becomes

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = (A^TW^TWA)^{-1}A^TW^TWb. \tag{5.38}$$

5.2.4 Inverse Tire Model

According to the description in Section 5.2.1, our test vehicle has the control authority in the front steering angle, δ_f, and the longitudinal acceleration, a_x, through EPS and ACC systems. To obtain the input commands to the lower level system, we need to convert the

Table 5.1: Tire parameters used in the simulation and the controller design

	simulation used	controller used		simulation used	controller used
B_f	9.0930	12.0930	B_r	7.5335	14.5669
C_f	1.8068	1.2068	C_r	1.4038	1.2893
D_f	4476.4	5819.3	D_r	3754.5	5117.8
E_f	-0.9585	-0.9585	E_r	-0.3107	0.8420

Table 5.2: Vehicle Parameters

vehicle mass	m	1830.59	kg
vehicle rotational inertia	I_z	3477	kgm^2
distance from COG to front axle	L_f	1.1521	m
distance from COG to rear axle	L_r	1.6929	m
gravity	g	9.80665	m/s^2

desired tire forces into the desired front steering angle, δ_f^*, and acceleration, a_x^*. Therefore, given the solution from (5.38), we want to solve the following algebraic equations:

$$u_1 = F_{xf}\cos\delta_f^* = ma_x^*\cos\delta_f^* \tag{5.39}$$

$$u_2 = F_{yf}\cos\delta_f^* = C_{\alpha f}\left[\delta_f^* - \tan^{-1}(\frac{v_y + L_f r}{v_x})\right] \tag{5.40}$$

where $C_{\alpha f} = 2D_f C_f B_f$ is the cornering stiffness of the front tire. We should notice that a linear tire model is used as a matter of guaranteeing an existence of the solution in equation (5.40). However, ignoring the saturation nonlinearity in the controller design is not a good idea. Although an adaptive control algorithm might cover some problems of the model mismatch, for the safety issue, it is still better to set up a saturation bound in the tire slip angle in the real implementation.

In conclusion, to obtain the desired references for the lower level controller, we can first apply the Newton's method to obtain δ_f^* from (5.40) then calculate a_x^* accordingly. Define an error function as

$$e(\delta_f) = u_2 - F_{yf}\cos\delta_f^* = C_{\alpha f}\left[\delta_f^* - \tan^{-1}(\frac{v_y + L_f r}{v_x})\right]. \tag{5.41}$$

The on-line algorithm is provided in Algorithm 4.

5.3 Simulation Results

This section demonstrates the application of the control system architecture proposed in Section 5.2 for a Figure-8 tracking. The controller is connected in closed-loop with a vehicle model that has different tire models used in the control design. As shown in Fig. 5.5, we set

Algorithm 4 Newton's Method (Inverse Tire Model)

initialize: $\delta_0^* = 0$
for k = 1,2,... **do**
  initialize: $\delta_0 = \delta_{k-1}^*$
  for i = 0,1,2,... **do**
    if $e^2(\delta_i) \leq \epsilon$ **then**
      $\delta_k^* = \delta_i;\ \ a_k^* = u_{1k}/(m\cos\delta_k^*);$ **return** $\delta_k^*,\ a_k^*$
    end if
    $\delta_{1+1} = \delta_i - e(\delta_i)/e'(\delta_i)$
    if $|\delta_{1+1} - \delta_i| \leq \epsilon$ **then**
      $\delta_k^* = \delta_{i+1};\ \ a_k^* = u_{1k}/(m\cos\delta_k^*);$ **return** $\delta_k^*,\ a_k^*$
    end if
  end for
end for

Table 5.3: Control Parameters Used in the Simulation

	X	Y	ψ		X	Y	ψ
$\bar{\mu}_\star$	10	10	20	$\lambda_\star$	1	1	1
$\rho_\star$	0.04	0.04	0.04	$k_\star$	3.2	3.2	2
$\phi_\star$	0.35	0.35	0.08	$W_\star$	1	1	1

up quite a lot amounts of mismatch in both the front and the rear tire models compared to the real system. The main purposes of doing this are to show how effectively the modeling errors can be captured by the adaptation algorithm and what would be the possible trade-off. All the tire parameters used in the simulation and the controller are provided in Table 5.1 and the vehicle parameters are chosen to be the same as the test vehicle (described in Table 5.2). The sampling time of the controller design is set to $dt = 0.01$ sec and the Runge-Kutta method with a variable time step integration is utilized for numerically simulating the vehicle dynamics. To demonstrate the control performance at its dynamic limits in the nonlinear region, we further planned our trajectories with high centrifugal acceleration almost reaching 1G during the cornering (see Fig. 5.6).

Simulations are conducted in MATLAB. Fig.5.7 - Fig. 5.10 illustrate the overall performance. All the tuned parameters in the controller can be found in Table 5.3. Note that the results of the proposed controller are compared with that of the controller which sets the adaptation gains being zero all the times. In other word, the sliding controller is used as the baseline controller. Fig. 5.7 and Fig. 5.8 show the basic tracking results of the position and velocity references. As we can see, the adaptation algorithm proposed in Section 5.2.3 receives a better performance with smaller tracking errors. The responses of the sliding variables and the tracking errors of X, Y and heading angle are shown in Fig. 5.9. As discussed in Chapter 2, we can expect that each of the error response will exhibit similar behavior as

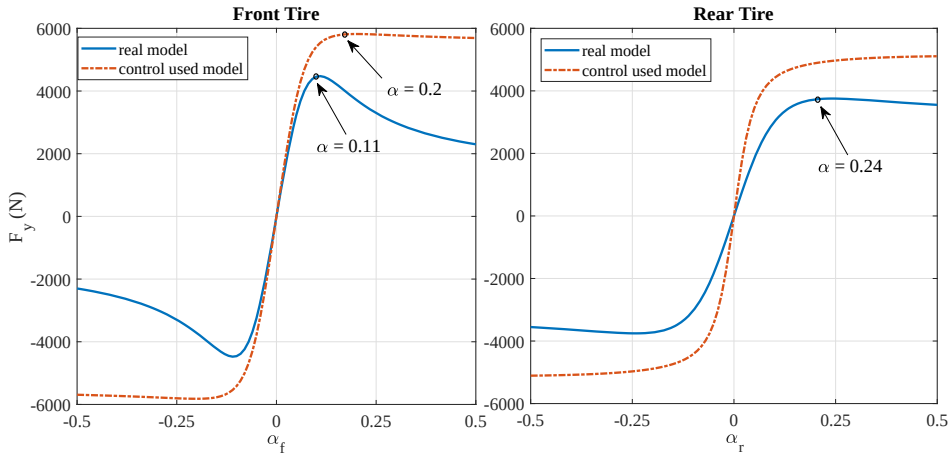


Figure 5.5: Comparison of the tire model error.

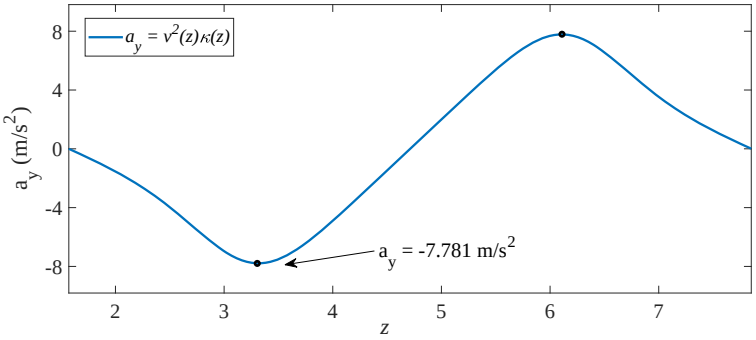


Figure 5.6: Comparison of the tire model error.

its sliding variable but with roughly $k_\star$ scaled. The time evolution of the adaptation gains for each sliding variable are presented in Fig. 5.10. An important point to be noted is that the response of the adaptation gains is a good indicator of the adaptation performance. Having them go up and down means the sliding variables are converging around the boundary layer defined in Theorem 2.3.1. Therefore, we can always tune the control parameters based on the response behavior of the sliding gain. The front and the rear tire side slip angles of both adaptive and baseline controllers are shown in Fig.5.11. According to the plot, it is obvious that the adaptive controller is able to use more tire capacities as it reaches out larger tire side slip angles in both the front and the rear tires. The control commands from the upper level controller of the desired acceleration and the desired front steering angle are shown in Fig. 5.12. We can see that the command provided from the adaptive controller displays an oscillation behavior in the acceleration meaning that the passenger will experience a more aggressive maneuver in driving. So, if we leave the matter of the feasibility for the lower level actuators, we should always be aware of the trade-off between the comfort and the tracking performance while applying the adaptive controller.

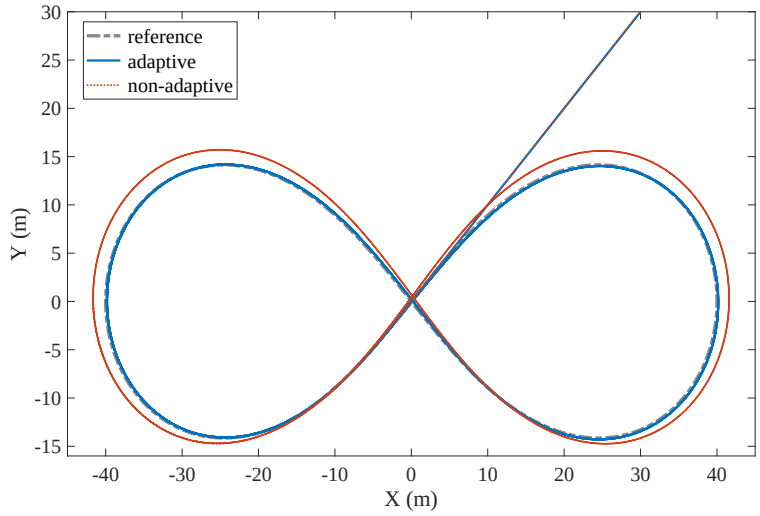


Figure 5.7: Comparison of the Figure-8 tracking performance for the adaptation and non-adaptation controllers.

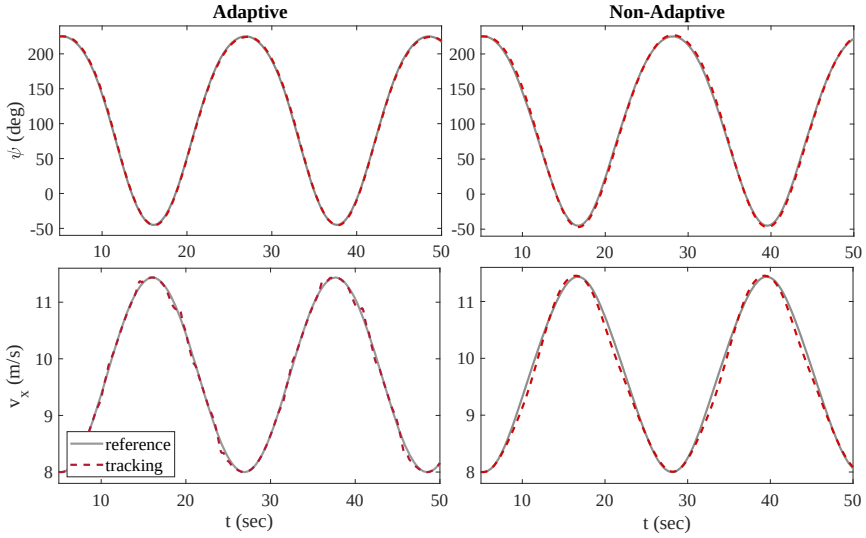
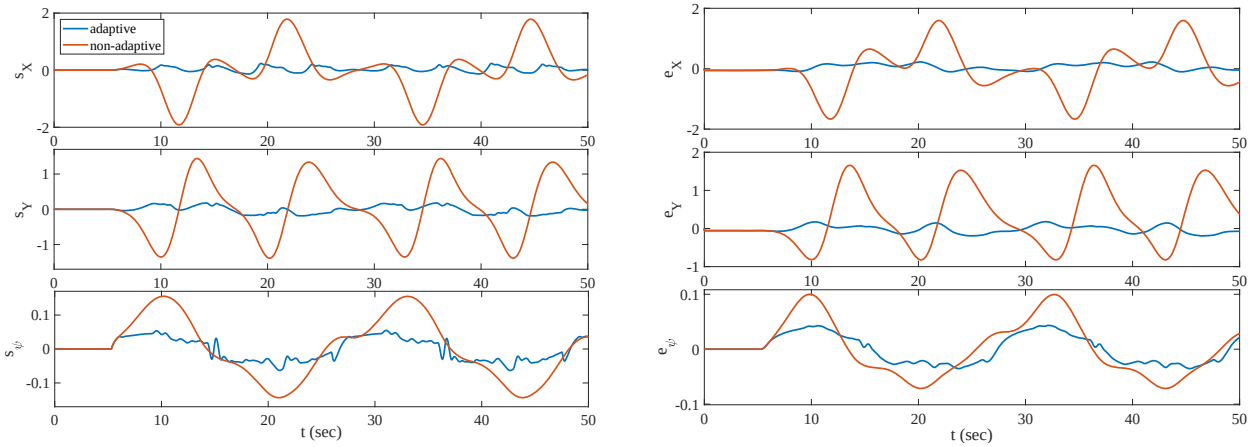


Figure 5.8: Comparison of the velocity and heading angle tracking performance.

5.4 Experimental Verification

5.4.1 Experimental Setup

Real-world experiments were carried out at the Hyundai-Kia Motors California Proving Grounds in California City, CA, USA. The test vehicle is a 5th generation Hyundai Grandeur provided by the Hyundai Motor Company equipped with the actuators and sensors essential for fully autonomous driving (see Fig. 5.13). An Oxford Technical Solutions (OxTS) RT 3000



(a) The response of s_X, s_Y and s_ψ.

(b) The response of e_X, e_Y and e_ψ.

Figure 5.9: Comparison of tracking errors for the adaptation and non-adaptation controllers.

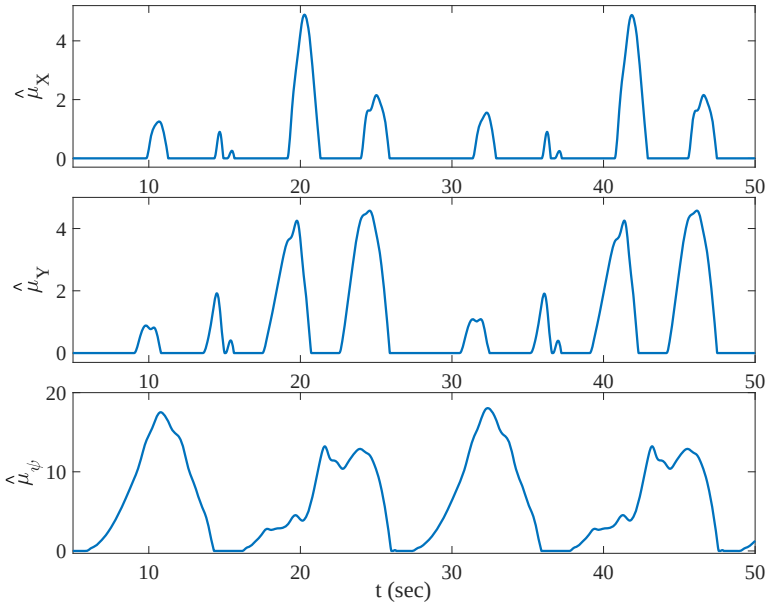


Figure 5.10: The response of the adaptation gain.

sensing system containing an on-board inertial measurement unit (IMU) and a differential global positioning system (dGPS) with real time kinematic (RTK) precision enhancement are used to provide the state information of the vehicle. Details on the accuracy of the sensing system are provided in table 5.4. The setup enables control authority over the front steering angle and longitudinal acceleration. The Electronic Power Steering (EPS) system is utilized for controlling the desired steering angle and the desired acceleration is commanded through the built-in adaptive cruise control (ACC) system on the vehicle. On-line computations are

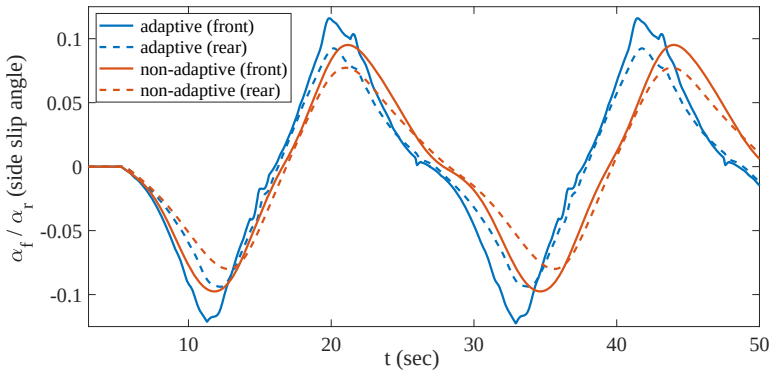


Figure 5.11: Comparison of tire side slip angle for the adaptation and non-adaptation controllers.

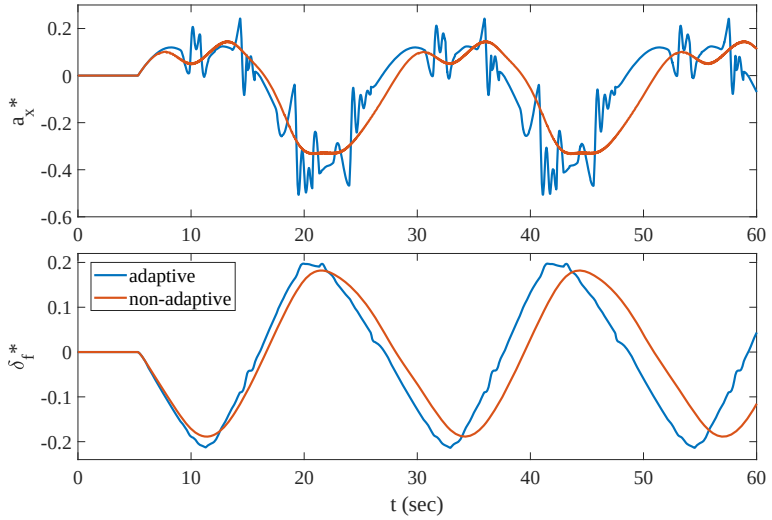


Figure 5.12: Comparison of the control inputs for the adaptation and non-adaptation controllers.

performed on a dSPACE MicroAutoBox II real-time computer (900 MHz IBM PowerPC processor). All the sensors, actuators and computing platform on the vehicle communicate via a CAN bus and the control algorithm is executed at 100 Hz. Note that we add a first order filter before the desired steering signal and the desired acceleration commanded to the actuator to avoid the disengagement of ACC system for emergency. Also, we planned the trajectory less aggressive compared to the scenario that used in the simulation for the safety issue in the real experiment.

Figure 5.13: The control algorithm is implemented in the test vehicle, a 5th generation Hyundai Grandeur.

Table 5.4: Specifications for the dGPS measurement system

sensor type	Oxford RT3000	
position accuracy	2×10^{-2}	m
heading accuracy	1×10^{-1}	deg
angular rate accuracy	0.2	%
velocity accuracy	0.1	%

Table 5.5: Tire parameters used for the experiment.

	nominal	controller used		nominal	controller used
B_f	6.0504	6.0504	B_r	7.5335	10.5469
C_f	1.2071	1.2071	C_r	1.4038	1.2634
D_f	4640.9	6497.3	D_r	3754.5	5256.3
E_f	0.4431	-0.9585	E_r	-0.3107	-0.3418

5.4.2 Experimental Results and Discussion

Table 5.5 shows the nominal tire parameters of the test vehicle. They are identified from the experimental data using a standard nonlinear least-squares regression approach. The final control parameters tuning are given in Table 5.6 and the experimental results are presented in Fig. 5.14 - Fig. 5.16. They are quite consistent with the results shown in the simulation. The adaptive controller has better performance in the position and the heading angle tracking. However, we can see that the baseline controller has a smoother maneuver in both the longitudinal acceleration and the front steering angle.

Table 5.6: Control Parameters Used in the Simulation

	X	Y	ψ		X	Y	ψ
$\bar{\mu}_\star$	5	5	5	$\lambda_\star$	0.8	0.8	0.8
$\rho_\star$	0.28	0.28	0.4	$k_\star$	2	2	2
$\phi_\star$	1.2	1.2	0.08	$W_\star$	1	1	0.5

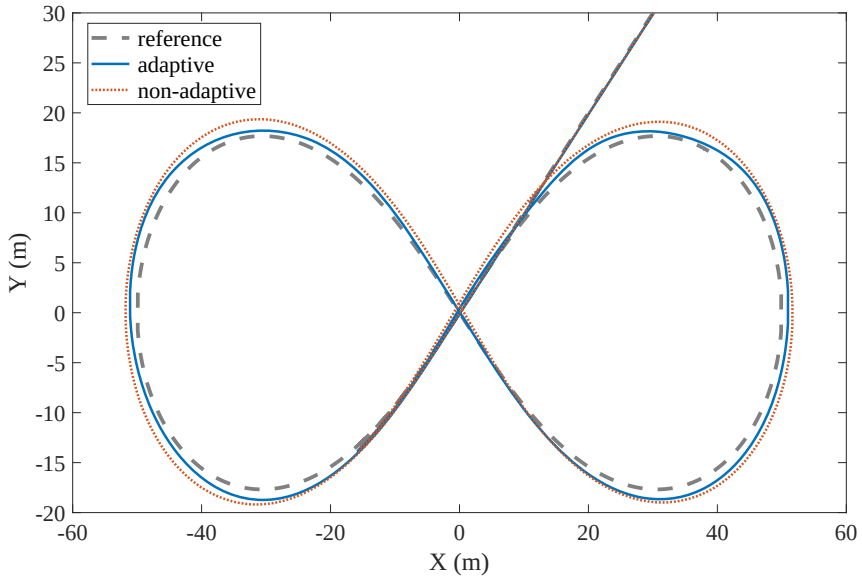


Figure 5.14: Comparison of the path tracking performances for the adaptation and non-adaptation controllers in the experimental run.

5.5 Conclusion

In this chapter, we implement an adaptive sliding mode control algorithm introduced in Chapter 2 for a Figure-8 tracking problem with huge modeling errors in the tire forces. The experiments involve aggressive maneuvers with a high centrifugal force which demonstrate the ability of the control algorithm to adapt the modeling errors in real time. Although the algorithm presented in Chapter 2 is only for a SISO system, it can be easily extended to a MIMO system as shown in Section 5.2. Successful experimental results confirm that the proposed nonlinear control structure can be run in real-time on embedded computing platforms despite the fast sampling time necessary for highly dynamic vehicle maneuvers. As a result, this work is relevant for ensuring that an adaptive sliding mode control algorithm meets sufficient stability levels even with a large modeling mismatch. Hence, it could be an alternative design approach for any time-varying system.

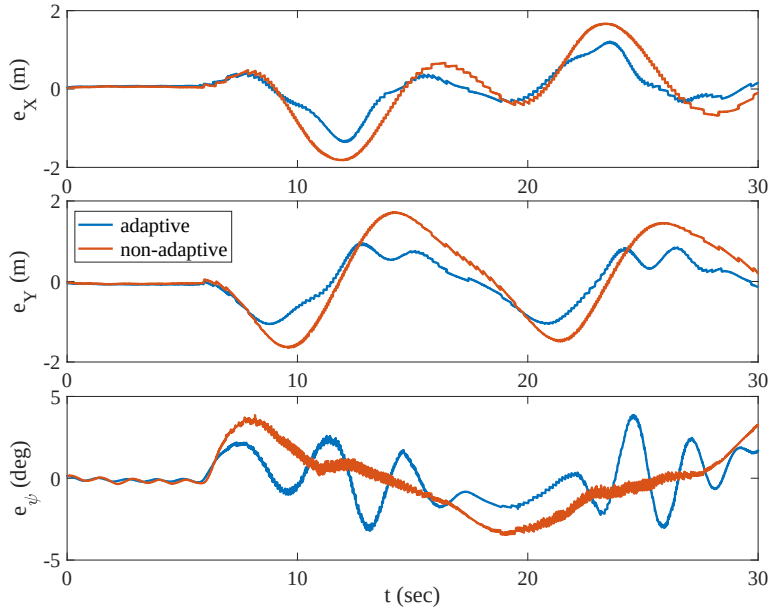


Figure 5.15: Comparison of the tracking errors for the adaptation and non-adaptation controllers in the experimental run.

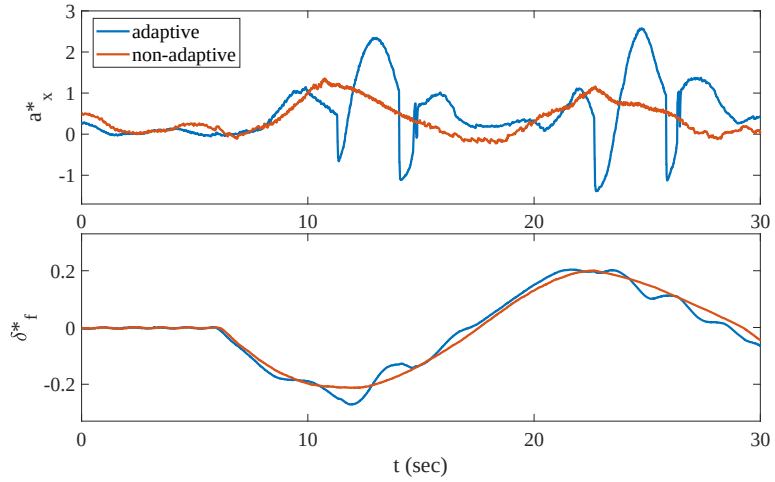


Figure 5.16: Comparison of the control input commands for the adaptation and non-adaptation controllers in the experimental run.

Chapter 6

Autonomous Drifting: Circling and Figure-8 Tracking

6.1 Motivation

"Drifting" is a driving technique that the driver intentionally oversteers the vehicle while maintaining control and driving the vehicle through the entirety of a corner. It can be characterized by a large side slip angle with the rear tires deeply saturated and the front wheels counter-steered. Simpling speaking, high side slip corresponds to 'sideways' motion of the vehicle and counter-steering means that the direction of rotation is opposite that of the steering angle. Having a vehicle traveling "sideway" is intriguing because it is a nonholonomic system from dynamics perspective. However, it is challenging in both predicting and controlling the vehicle under this kind of scenarios compared to normal driving situations as it works on a high nonlinear region of the state space. The studying of drifting has recently gained popularity in academic circle due to the sharp growth in autonomous vehicle market. Several advantages can be gained from fundamentally understanding the vehicle dynamics and control beyond the limits of the tire capacities. It may become useful if the knowledge can be implemented into the decision making and the motion planning during the emergency situations.

Many examples in the literature show that steady-state drifting is an unstable equilibrium point [88, 89] and the maneuver is just purely a regulation problem [90]. However, two fundamental limitations make the control quiet challenging. One is that the equilibrium is needed to pre-calculate according to the model. This increases the difficulty in generating the reference trajectory when it comes to the tracking problems. The other one is the indirect controllability of the rear tire's lateral force. Since we do not have the control authority in the rear wheel steering, the only way to control the lateral tire force is to utilize the longitudinal force via the saturation property. This makes the control system very sensitive to the model uncertainty and the closed-loop system may easily become unstable. To improve the robustness, we then implement the adaptive sliding mode control proposed in Chapter

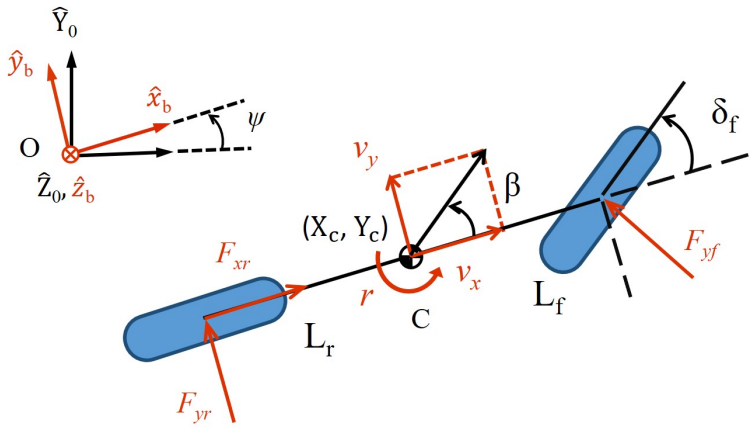


Figure 6.1: Dynamics bicycle model of a rear-wheel-drive vehicle.

2 to the problem of steady-state drifting and further extend the control idea proposed in Chapter 5 to a Figure-8 drifting. For safety purposes, we conduct the experiment by using a 1/10 scale RC vehicle platform built by students at UC Berkeley to verify the control algorithm instead of using a full scale vehicle.

The chapter is organized as follows. We first introduce the vehicle model used for the autonomous drifting. Then, a switched adaptive control structure is introduced based on the idea proposed in [88]. An further extension of the control design from Chapter 5 to Figure-8 drifting is discussed in Section 6.3.2. All simulation and experimental results are presented in the end.

6.2 Modeling and Design Idea

6.2.1 Vehicle Model

According to the discussion, different levels of complexity for the model used in drift control design has been considered in the literature [91, 92]. They can be ranged from a simple two-state bicycle model to a high-fidelity four-wheel model that includes the load transfer and relatively complex tire models. To balance between simplicity of the control design and sufficient model fidelity, in this chapter, we choose a three-state rear-wheel-drive bicycle model depicted in Fig. 6.1. Given an inertial and a body-fixed coordinate systems defined as $\{\hat{X}_0, \hat{Y}_0.\hat{Z}_0\}$ and $\{\hat{x}_b, \hat{y}_b, \hat{z}_b\}$, the model can be described in the following nonlinear differential equations:

$$\dot{v}_x = \frac{1}{m}F_{xr} + r\beta v_x \tag{6.1}$$

$$\dot{\beta} = \frac{1}{mv_x}(F_{yf}\cos\delta_f + F_{yr}) - r \tag{6.2}$$

$$\dot{r} = \frac{1}{I_z}(L_f F_{yf}\cos\delta_f - L_r F_{yr}). \tag{6.3}$$

where m is the mass of the vehicle, I_z is the equivalent yaw moment of inertia, and L_f and L_r are the distances from the vehicle's center of gravity (CoG) to the front and rear axles, respectively. v_x, β and r are the states which denote the longitudinal velocity, the side slip angle and the yaw rate of the vehicle, respectively. The side slip is defined as $\beta \approx v_y/v_x$ denoting the angle difference between the heading and moving directions of the vehicle. δ_f and F_{xr} are the control inputs denote the front steering angle and the longitudinal force provided from the rear tire, respectively. F_{yf} and F_{yr} are the lateral forces on the front and rear tires respectively. A Nonlinear magic formula is used to consider the tire's linear behavior at small slip angles and its nonlinear behavior at large slip angles:

$$F_{y\star} = 2\eta_\star D_\star \sin\left(C_\star \tan^{-1}\left(\left(1-E_\star\right)B_\star\alpha_\star + E_\star\tan^{-1}\left(B_\star\alpha_\star\right)\right)\right). \tag{6.4}$$

Note that the symbol $\star$ is either f or r representing the front or rear tire. $B_\star$, $C_\star$, $D_\star$, and $E_\star$ are the corresponding empirical coefficients that characterize the "pure" lateral tire model. α_r and α_r are the front and rear tire side slip angles defined as

$$\alpha_f = \delta_f - \tan^{-1}(\beta + \frac{L_f r}{v_x}) \tag{6.5}$$

$$\alpha_r = -\tan^{-1}(\beta - \frac{L_r r}{v_x}). \tag{6.6}$$

$\eta_\star$ is a derating factor that accounts for the reduction of the lateral tire force when the longitudinal force is applied. Here, we have $\eta_f = 1$ and

$$\eta_r = \frac{\sqrt{(2D_r)^2 - F_{xr}^2}}{2D_r} \tag{6.7}$$

because a rear-wheel-drive vehicle is considered for the control design.

6.2.2 Design Idea Generation

Inspired by the control structure introduced in [88], the design method proposed in this work utilizes the dynamics surface control technique [93] with a small modification. The main difference is that, in our methodology, it is not required to calculate the equilibrium point in advance to design the control law for steady cornering and path tracking. A detail explanation will be provided in the remainder of this chapter.

As mentioned in Section 6.1, drifting is characterized by a large side slip angle with the rear tires being saturated while doing the cornering. The maneuver is initiated by the vehicle losing the traction in the rear tires or all tires. Then, to hold the vehicle traveling sideway without spinning out, the driver needs to balance between the front steering angle and the throttle to alter the attitude of the vehicle. Based on the above statements, the main task of designing a drift controller is first to make the vehicle spin then to maintain a desired yaw rate and velocity in the direction of travel.

To start the maneuver, an oversteer condition should be induced while entering the turn. This technique is very basic that can be achieved by either pulling the hand brake or pressing the throttle to induce rear traction loss. Once oversteer has been provoked, the controller should recover the vehicle from spinning out by applying a right amount of yaw moment. This is achievable by pointing the front steering angle along the traveling direction if the front tires are not saturated. However, when the front tires already reach their physical limits of the friction circle, it is impossible to generate an enough lateral force to resist the spinning moment. Therefore, to generate a right amount of the yaw moment and to maintain the handling ability of the front steering wheel, the only way is to reduce the lateral tire force from the rear tire. This would be easy if the rear wheel steering is enabled. But, since a four-wheel steering vehicle is a not-quite-ready technology in the modern market, the only way to reduce the later tire force is to increase the longitudinal tire force of the rear tires. This can be inferred by knowing the fact that the total vector sum of the longitudinal and the lateral tire forces generated cannot exceed the tire normal load. By applying this idea in the control design, we can sustain the desired large side slip angle of the vehicle. However, one thing should be notice is that if the control of the throttle is used to generate the desired yaw moment, the vehicle will lose the control authority in the velocity and become underactuated. This so-call the stability-controllability trade off and is the fundamental limitation of the drift control design. With this in mind, the proposed controller then has a structure of two switching conditions.

6.3 Control Design

6.3.1 Control Structure - Steady State Circling

Continuing from the above discussion, the primary objective of the control design would be to track a desired large side slip angle and the secondary objective will be managing the desired velocity. The control structure is built based on the idea presented in the previous section. Start from observing the side slip dynamics of the bicycle model in (6.2):

$$\dot{\beta} = \frac{1}{mv_x}(F_{yf}\cos\delta_f + F_{yr}) - r. \tag{6.8}$$

To control the side slip angle, it is nature to treat the yaw rate as a synthetic input for the reason that it appears linearly in (6.8). Then, follow the design procedure of the dynamics

surface control [93] and combine the adaptive sliding mode control algorithm proposed in Chapter 2. Given any desired pair of the side slip angle and the longitudinal velocity, $(\beta_r,\ v_{xr})$, we define two sliding surfaces as

$$s_\beta = \beta - \beta_r \tag{6.9}$$
$$s_{v_x} = v_x - v_{xr} \tag{6.10}$$

and represent the problem into the following error dynamics:

$$\dot{s}_\beta = \frac{1}{mv_x}(F_{yf}\cos\delta_f + F_{yr}) - r \tag{6.11}$$
$$\dot{s}_{v_x} = \frac{1}{m}F_{xr} + r\beta v_x. \tag{6.12}$$

by assuming $\dot{\beta}_r = 0$ and $\dot{v}_{xr} = 0$ are the equilibrium. Overall, there are two sliding surfaces to be stabilized. Start from equation (6.11). Define the synthetic input of r_d such a way that

$$\tau\dot{r}_d + r_d = \frac{1}{mv_x}(F_{yf}\cos\delta_f + F_{yr}) + k_\beta s_\beta + \hat{\mu}_\beta \text{sgn}(s_\beta) \tag{6.13}$$

for $k_\beta > 0$, $\tau > 0$ and $\hat{\mu}_\beta$ followed with the adaptation law (2.8). Then, we can construct another dynamics surface of $s_r = r - r_d$ and obtain another error dynamics by taking the time derivative:

$$\dot{s}_r = \frac{1}{I_z}(L_f F_{yf}\cos\delta_f - L_r F_{yr}) - \frac{1}{\tau}\Big(\frac{1}{mv_x}(F_{yf}\cos\delta_f + F_{yr}) + k_\beta s_\beta + \hat{\mu}_\beta \text{sgn}(s_\beta) - r_d\Big). \tag{6.14}$$

With the result above, the control law then can be established based on the adaptive control algorithm:

$$\dot{s}_r = -k_r s_r - \hat{\mu}_r \text{sgn}(s_r) \tag{6.15}$$

for $k_r > 0$ and $\hat{\mu}_r$ followed with the adaptation law (2.8). Substitute (6.14) into (6.15) and collect like terms. We can write down the control law in the form of

$$h_1 F_{yf}\cos\delta_f + h_2 F_{yr} = -k_r s_r - \hat{\mu}_r \text{sgn}(s_r) + \frac{1}{\tau}\Big(k_\beta s_\beta + \hat{\mu}_\beta \text{sgn}(s_\beta) - r_d\Big) \tag{6.16}$$

with $h_1 = L_f/I_z - 1/mv_x\tau$, $h_2 = -L_r/I_z - 1/mv_x\tau$ and F_{yr} calculated from (6.4) according to the current feedback states and the rear longitudinal tire force. Then, the desired front steering command can be obtained by solving

$$F_{yf}(\delta_f^*)\cos\delta_f^* = \frac{1}{h_1}\Bigg[- h_2 F_{yr} - k_r s_r - \hat{\mu}_r \text{sgn}(s_r) + \frac{1}{\tau}\Big(k_\beta s_\beta + \hat{\mu}_\beta \text{sgn}(s\beta) - r_d\Big)\Bigg] = u_1$$

Algorithm 5 Newton's Method (Inverse Nonlinear Tire Model)

Define $\varphi = \tan^{-1}(\beta + L_f r/v_x)$
initialize: $\alpha_0 = 0$
for i = 0,1,2,...,11 **do**
  if $e^2(\alpha_i) \leq \epsilon$ **then**
    $\alpha^* = \alpha_i$;
    $\delta_f^* = \alpha^* + \varphi$ **return** δ_f^*
  end if
  $\alpha_{i+1} = \alpha_i - e(\alpha_i)/e'(\alpha_i)$
  if i > 10 **then**
    $\delta_f^* = \alpha_m + \varphi$
    return δ_f^*
  end if
end for

using the inverse tire model (6.4). Define an error function as

$$e(\alpha) = u_1 - F_{yf}(\alpha)\cos(\alpha + \varphi) \quad \text{for} \quad \varphi = \tan^{-1}(\beta + \frac{L_f r}{v_x}). \tag{6.17}$$

The on-line algorithm is applied by the Newton's method as shown in Algorithm 5. If the solution exists, then we know that the control law (6.15) is satisfied without the front tire saturated. Therefore, we have an authority in using F_{xr} as a control input to stabilize the longitudinal error dynamics (6.12).

$$F_{xr}^* = -m\beta v_x r - mk_{v_x} s_{v_x} - m\hat{\mu}_{v_x}\text{sgn}(s_{v_x}) \tag{6.18}$$

for $k_{v_x} > 0$ and $\hat{\mu}_{v_x}$ followed with the adaptation law (2.8). On the other hand, if the solution does not exist, we need to coordinate F_{yr} together with the front steering angle to make the desired control law (6.16) valid. Consider equation (6.16) as a weighted sum of the front and the rear lateral tire forces for a desired yaw moment. We then set the front tire force, F_{yf}, to its maximum value and calculate the required lateral force in the rear tire.

$$F_{yr}^* = \frac{1}{h_2}\Big[- h_1 F_{yf}(\alpha^*)\cos(\alpha^* + \varphi) - k_r s_r - \hat{\mu}_r \text{sgn}(s_r) + \frac{1}{\tau}\Big(k_\beta s_\beta + \hat{\mu}_\beta \text{sgn}(s_\beta) - r_d\Big)\Big]$$

for $\alpha^* = \alpha_{\max}\text{sgn}(u_1)$ where $\alpha_{\max}$ is the tire slip angle at which maximum lateral force is obtained. At first, this control law might seem unrealizable for the reason of not having a direct access to control F_{yr}. However, it is actually achievable through the longitudinal tire force. Having the fact that the total vector sum of the force generated by the tire cannot exceed its normal load, we can control the desired lateral force by enforcing the real tire saturated via F_{xr}. Therefore, the final control commands can be constructed as follows.

$$F_{xr}^* = \sqrt{(2D_r)^2 - (F_{yr}^*)^2}$$
$$\delta_f^* = \alpha^* + \varphi.$$

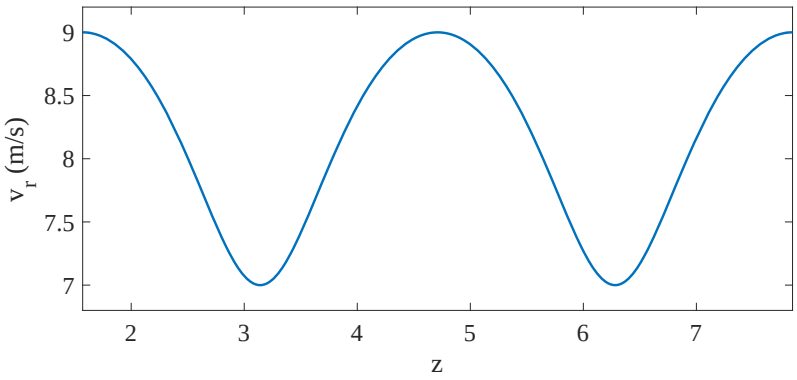


Figure 6.2: An example of a velocity profile for Figure-8 with $a = 23$ m.

In summery, the control structure contains two different modes that are switched based on the front tire force's capacity. If there is quiet amount of cornering force available in the front tire, then we have a spare control in the rear longitudinal force to track the desired velocity. This mode can be treated as an intention to stabilize the system around the drift equilibrium point with no control inputs being saturated. On the other hand, if the front tire has already reached its maximum capacity, we then need to coordinate the longitudinal tire force with the front steering command to track the desired side slip angle first instead of v_{xr}. We can characterize this mode as deepening the vehicle into more oversteer condition to allow the front tire to catch up from losing the control.

6.3.2 Control Structure - Figure-8 Drifting

As shown in the previous section, steady state drifting is just a regulation design problem for an unstable equilibrium point. However, extending the task into a tracking problem poses a challenge for several reasons. The most important one is that the control works around the region that both the front and rear tires are almost saturated. This already makes the control for steady state drifting hard as knowing that the desired longitudinal velocity will be scarified when the front tire is saturated. Therefore, adding an additional tracking reference can only make the problem even worst. Many researchers has proposed different approaches to deal with this problem. One is to add additional control inputs to address the underactuated issue [94]. Another one is to limit the trajectory to two references and assign them to each actuator independently [95]. Finally, the most intuitive one is to treat the trajectory as a sequence of drift equilibriums [96]. Unlike all of these approaches, in this section, we extend the idea of the trajectory planning introduced in Chapter 5 and combine the two control algorithms together into a Figure-8 drifting problem.

There are three main things need to be altered. Start from the trajectory planning. Having a similar approach provided in Section 5.1.2, given a fix path of Figure-8 to track, we need to first plan the velocity profile along the trajectory. However, according to the above discussion, it is impossible to randomly assign a desired trajectory for drifting. It needs to

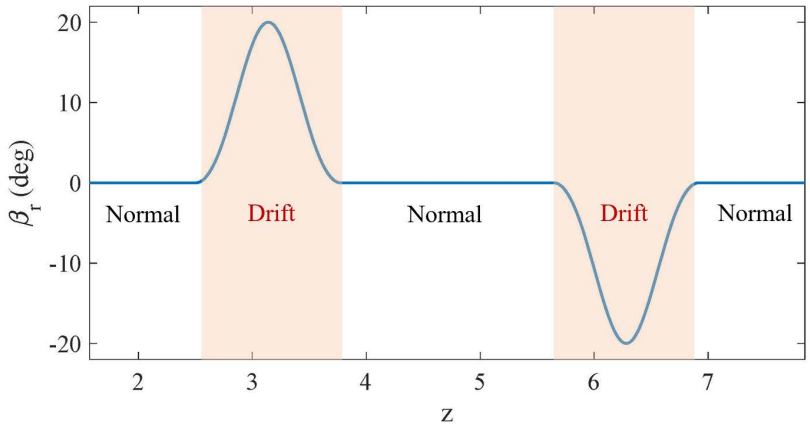


Figure 6.3: An example of a desired side slip angle profile for Figure-8 with $a = 23$.

be set as a sequence of feasible equilibriums for a perfect tracking. To reduce the amount of work, here, we roughly guess a reasonable velocity by knowing that the drift happens around the limit of driving capacity.

$$\|v_r^{\text{guess}}\|_2^2 \approx \varsigma \frac{\mu g}{\max(\kappa_r)} \tag{6.19}$$

where μ is the tire-road friction coefficient, $\max(\kappa_r)$ is the maximum curvature of the turn and $0 \ll \varsigma < 1$ is a derating factor. An example of a desired velocity profile for a Figure-8 is shown in Fig. 6.2 with $a = 23$ m, $\max(\kappa_r) = 0.1304$, $\mu = 0.9$ and $\varsigma = 0.9$. It is specified as a sine shape function varying from 7 to 9 m/s during the entire track based on the guide of $|v_r^{\text{huess}}| \approx 7.8051$ m/s.

Next, we need to change the heading angle references according to the desired side slip angle, β_r, to enforce the drift.

$$\psi_r^{drift}(z) = \psi_r(z) - \beta_r(z). \tag{6.20}$$

where $\psi_r(z)$ is the original heading reference tangent to the Figure-8 track listed in (5.18). An example of a desired side slip angle profile can be found in Fig. 6.3 where the drift is planned to happen during cornering. Finally, with all the references ready, we now can follow the same procedure as described in Section 5.2.3 to design the adaptive control law. Again, this will result in the same linear system with 2 variables and 3 algebraic equations for $A\mathbf{u} = b$ as listed in (5.36) but $\mathbf{u} = [F_{xr},\ F_{yf}\cos\delta_f]^T$ because we have a rear-wheel-drive vehicle. The desired tire forces can be obtained from

$$\mathbf{u} = (A^T W^T W A)^{-1} A^T W^T W b \tag{6.21}$$

as listed in (5.38). One thing we should notice is that since we are generally more care about the X-Y tracking than the heading angle in the normal driving situation, it will make sense to choose W_ψ be relatively small to the other two while the drift maneuver is the

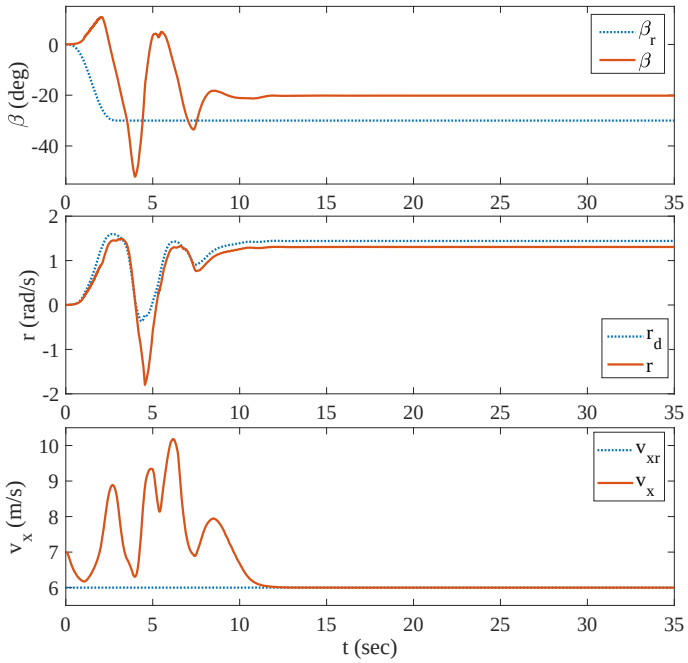


Figure 6.4: The state response of the vehicle for the non-adaptive controller in simulation.

opposite. To distinguish different driving scenarios, we further separate the control logic into two different modes according to the desired side slip angle: one is the normal driving mode ($\beta_r = 0$); the other one is the drift mode ($\beta_r \neq 0$). Each mode will have different control tunings and different weightings on W_X, W_Y and W_ψ. Once we obtain the desired tire forces by solving the linear system (6.21), the final control commands can be obtained via the inverse tire models. Then, follow the design idea proposed in Section 6.3.1 for the drift control. If the desired front tire force is saturated in the drift mode, we will discard the tracking performance in X-Y and rearrange the third algebraic equation in (5.36) to obtain the desired input of the rear tire's lateral force as:

$$F_{yr}^* = \frac{I_z}{L_r}\Big[-\frac{L_f}{I_z} F_{yf}(\alpha^*)\cos(\alpha^* + \varphi) - a_r\kappa_r + \lambda_\psi \dot{e}_\psi + k_\psi s_\psi + \hat{\mu}_\psi \mathrm{sgn}(s_\psi)\Big] \tag{6.22}$$

and map it to a throttle command via the friction circle relation. As a result,

$$F_{xr}^* = \sqrt{(2D_r)^2 - (F_{yr}^*)^2}$$
$$\delta_f^* = \alpha^* + \varphi.$$

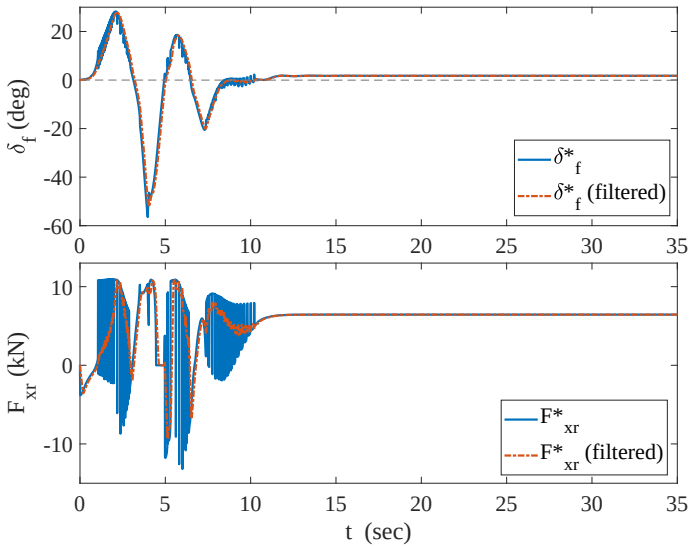


Figure 6.5: The control input response for the non-adaptive controller in simulation.

Table 6.1: Vehicle Parameters (Simulation)

vehicle mass	m	2237	kg
vehicle rotational inertia	I_z	5519.2	kgm^2
distance from COG to front axle	L_f	1.5	m
distance from COG to rear axle	L_r	1.52	m
gravity	g	9.80665	m/s^2

Table 6.2: Tire Parameters Used in the Controller Design (Simulation)

B_f	14.3987	B_r	14.3758
C_f	1.5157	C_r	1.5158
D_f	5154.5	D_r	5096.5
E_f	0.0001	E_r	0.0001

6.4 Simulation Results

In this section, we first validated the proposed control strategy in simulation using MATLAB. The controller is connected in closed-loop with a high fidelity vehicle model containing sophisticated nonlinear tire models. Note that we choose a normal scale vehicle as a plant since we don't have full nonlinear 3D model for the RC car. The nominal model and tire parameters used in the control design are summerized in Table 6.1 and Table 6.2. The control is executed at 100Hz and a moving average filter is added before the steering and longitudinal tire force sent to the plant. This helps smooth out unrealizable chattering commands induced by the switch.

Table 6.3: Control Parameters Used in the Simulation (Steady State Drifting)

	β	r	v_x
$\bar{\mu}_\star$	5	5	5
$\rho_\star$	20	1 5	5
$\phi_\star$	0.05	0.5	0.5

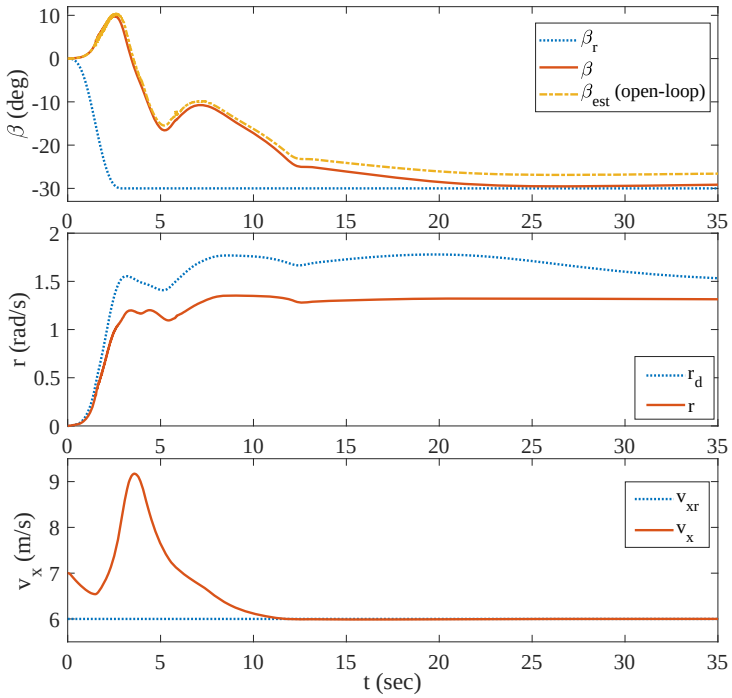


Figure 6.6: The state response of the vehicle for the adaptive controller proposed in Section 6.3.1.

6.4.1 Steady State Drifting

To verify the functionality of the control idea in steady state drifting, we first run the simulation by setting up the adaptation gains equal to zero and the control parameters be $k_r = 4.25$, $k_\beta = 1.7$, $k_{v_x} = 1.7$ and $\tau = 0.4$. The desired side slip angle and the longitudinal velocity are given as $\beta_r = -0.5236$ rad (-30°) and $v_{xr} = 6$ m/s. The vehicle states and control inputs during simulation are shown in Fig. 6.4 and Fig. 6.5. The red lines in Fig. 6.4 show the simulation measurements and the blue dotted lines show the desired references. It can be seen that the control try to manipulates the longitudinal rear tire force with the front steering wheel to achieve the desired side slip angle. However, there is a 10° steady state error in the side slip angle and the front tire is not counter steered (see Fig. 6.5). More aggressive control gains can be tuned to improve the performance. However, this leads the system become unstable in the end.

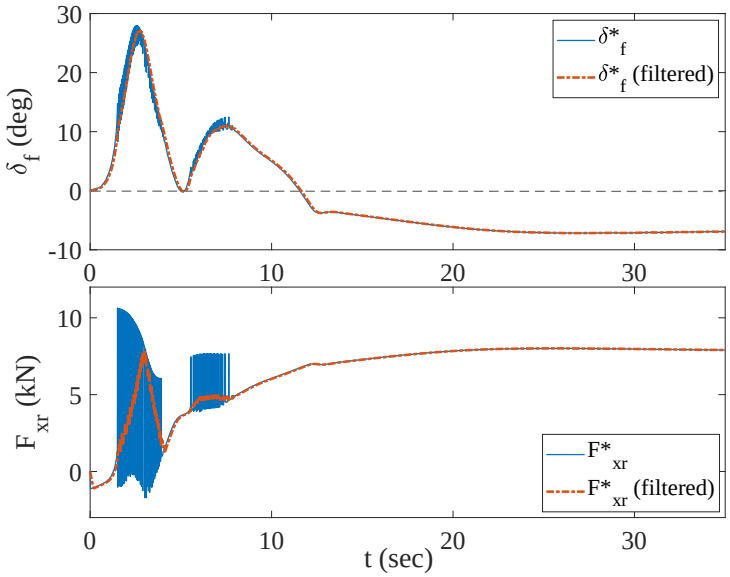


Figure 6.7: The control input response for the adaptive controller proposed in Section 6.3.1.

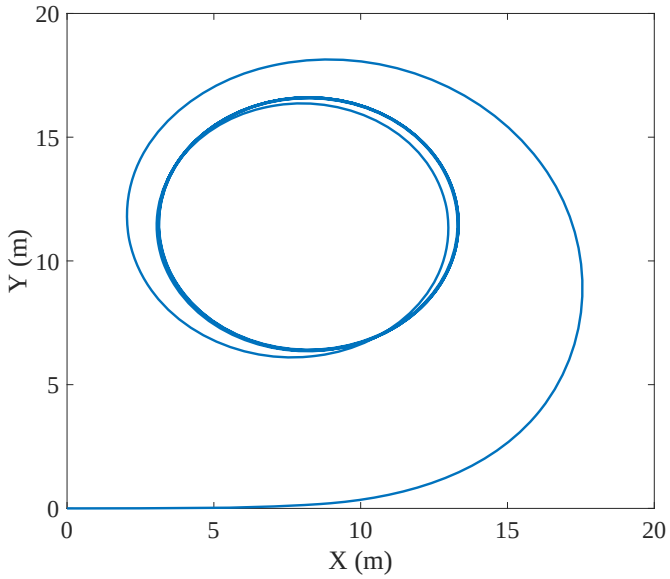


Figure 6.8: The measured path of the vehicle in the steady state drifting.

To improve the robustness and the tracking performance, the control design with an adaptation law proposed in Section 6.3.1 then is applied. Fig. 6.6 - Fig. 6.9 show the simulation results. As expected, the controller incorporates with the adaptation algorithm has better performances that the states converge closely to the desired values and the front steering wheel counter-steered (see Fig. 6.7). Fig. 6.8 shows the measured path of the vehicle for the entire simulation run. We can see that it converges to a circle and hence reaches a

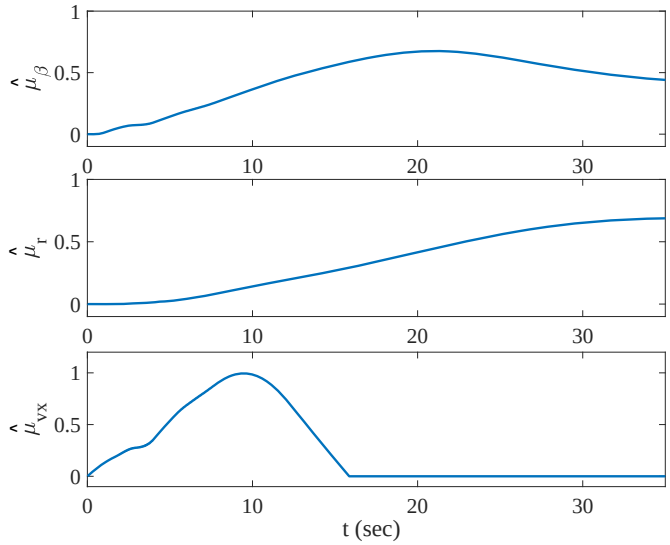


Figure 6.9: The response of the adaptation gains in simulation.

steady state drifting. The convergence of the responses in the adaptation gains shown in Fig. 6.9 denote how effective the modeling error can be adapted. The control parameters are set to $k_r = 1$, $k_\beta = 1$, $k_{v_x} = 0.5$ and $\tau = 0.4$ and the rest in the adaptation law are listed in Table 6.3.

We also implement the side slip angle estimation algorithm proposed in Chapter 4 to verify how well the observer would perform under such an extreme maneuver like drift. The observer is run aside with the state feedback control loop and the estimated side slip angle can be found in Fig. 6.6 denoted by the yellow dashed dotted line. As we can see, the estimate performance is quiet good but has a small discrepancy (3° - 6°) in the extreme case in the steady state.

6.4.2 Figure-8 Drifting

This section demonstrates the application of the control design in Figure-8 drifting with $a = 23$ m. The desired velocity and side slip angle profiles are chosen as shown in Fig. 6.2 and Fig. 6.3. All the references are built by following the same procedure introduced in Section 5.2.2. The simulation results are shown in Fig. 6.10 - Fig. 6.12. Different control parameters are tuned and chosen under two driving scenarios, the normal drive mode and the drift mode (see Table 6.4). As expected, Fig. 6.10 shows that the vehicle cannot track the path exactly with the drift maneuver during the cornering. It has a trade-off between tracking a high side slip angle and following the X-Y path precisely in the same time. The control performance of the side slip angle and the longitudinal velocity can be found in Fig. 6.11 and Fig. 6.12. It is obvious that the control will take the initiative to track the side slip angle and discard the velocity reference while in the drift mode. The entire maneuver

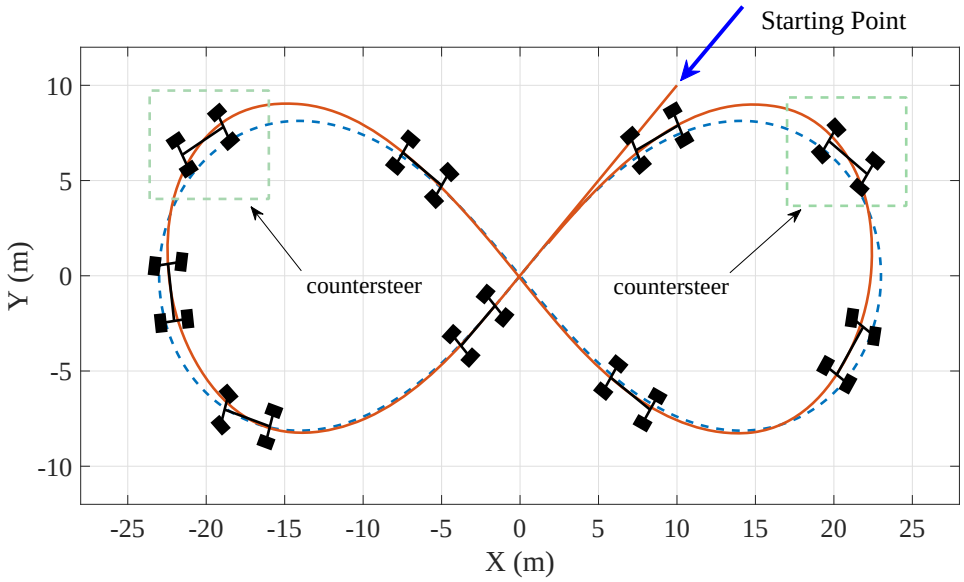


Figure 6.10: The desired and measured paths for Figure-8 drifting.

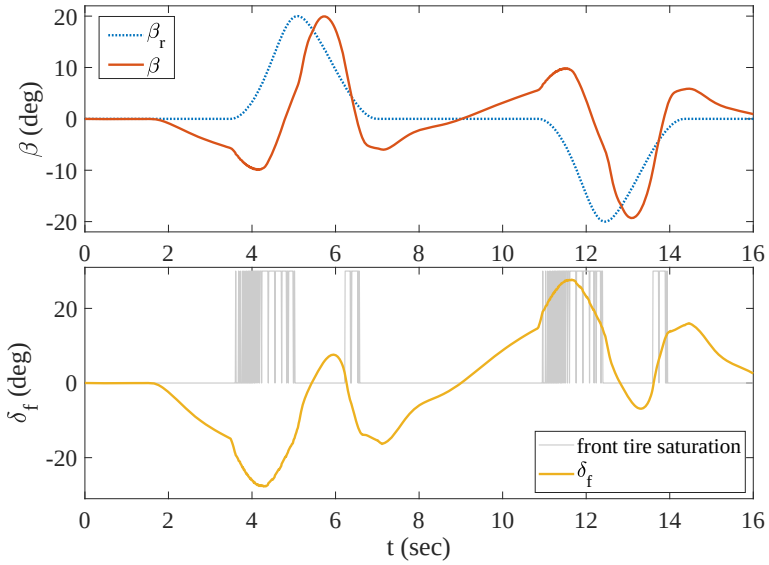


Figure 6.11: The response of the side slip angle and the front steering inputs for Figure-8 drifting.

comes along with a high longitudinal throttle command to induce the oversteer and results in a front wheel counter steering accompanied with huge braking to recover from the drift.

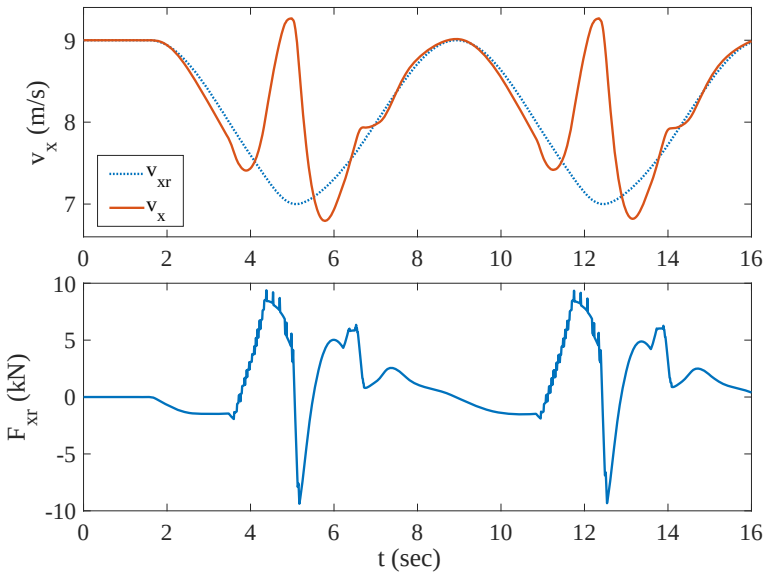


Figure 6.12: The response of the longitudinal velocity and the longitudinal tire force inputs for Figure-8 drifting.

Table 6.4: Control Parameters Used in the Simulation (Figure-8 Drifting)

	X	Y	ψ		X	Y	ψ
$\bar{\mu}_\star$	1	1	1	$\bar{\mu}_\star(0)$	0.01	0.01	0.01
$\rho_\star$	0.8	0.8	0.8	$\phi_\star$	0.2	0.2	0.1
$k_\star$ (normal)	2	2	0.5	$k_\star$ (drfit)	1.1162	1.116	2.407
$W_\star$ (normal)	1	1	0.6	$W_\star$ (drift)	1	1	20
$\lambda_\star$ (normal)	2	2	0.1	$\lambda_\star$ (drift)	0.93	0.93	2.324

6.5 Experimental Verification

6.5.1 Experimental Platform

To do the experiment, we used a 1/10 scale rear-wheel-drive RC vehicle platform named Berkeley Autonomous Race Car (BARC) developed by students in UC Berkeley (see Fig. 6.13). The model identification and state estimation methods are based on previous work [90]. The parameters of the RC vehicle and tires are summarized in Table 6.5 and Table 6.6. The vehicle is controlled through the use of a micro-controller and a suite of sensors where the front wheel is steered by a servo motor and the motor is controlled by a Electronic Speed Control (ESC) unit. An indoor navigation system kit from Marvelmind is used for the localization. It provides ± 2 cm accuracy with an 8 Hz update rate. The code is implemented in Python using ROS (Robot Operating System).

Figure 6.13: Berkeley Autonomous Race Car (BARC).

Table 6.5: Vehicle Parameters (BARC)

vehicle mass	m	2.076	kg
vehicle rotational inertia	I_z	0.02298	kgm^2
distance from COG to front axle	L_f	0.141	m
distance from COG to rear axle	L_r	0.119	m
gravity	g	9.80665	m/s^2

Table 6.6: Tire Parameters Used in the Controller Design (BARC)

B_f	14.522	B_r	7.6747
C_f	1.1993	C_r	1.6916
D_f	2.4	D_r	2.6
E_f	-0.0584	E_r	0.95

6.5.2 Experimental Results - Steady State Drifting

The experiments were conducted in an indoor space. For the time issue, only a steady state drifting is done to validate the control idea. In order to have reasonable target references for the side slip angle and the longitudinal velocity, we began the tests by running a steady state cornering with the velocity and steering inputs increased slowly to make sure that the vehicle will not collide with the wall. The final target references are set to be $\beta_r = 40^\circ$ and $v_{xr} = 1$ m/s and the tuned control parameters are $k_r = 0.08$, $k_\beta = 0.04$ and $k_{vx} = 0.08$. Fig. 6.14 and Fig. 6.15 show the experimental results of the state response and the control input

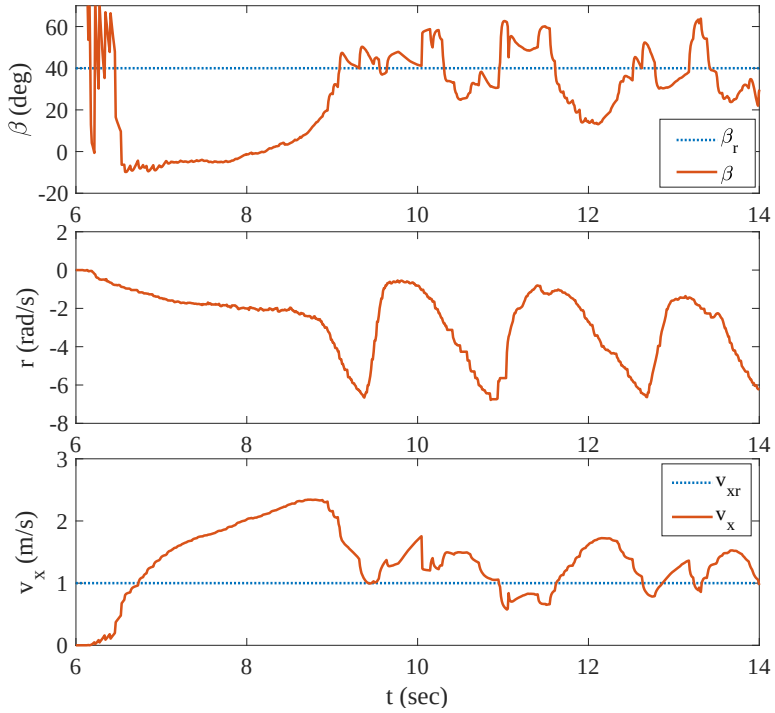


Figure 6.14: The experimental results of the tracking performance for a steady state drifting.

commands. As we can see from the plot of the side slip angle and the yaw rate responses, the controller can successfully initiate the drift but barely sustain the maneuver over 1 sec after the intention of a counter steering. There are several possible reasons that cause the control to be fail in maintaining the drift. One is the slow update rate of the indoor GPS sensor (8 Hz) and the other one is the delay of the actuator on the longitudinal tire force. Since we have the motor controlled via the Electronic Speed Control (ESC) unit, an indirect access of the control to the tire force can easily cause a delay during the switch. As a result, the system become unstable with an oscillation in the state.

6.6 Conclusion

This chapter implements the adaptive sliding control algorithm proposed in Chapter 2 to the problem of autonomous drifting. Both of the steady state circling and Figure-8 tracking maneuvers are considered in the control design. The simulation results show impressive control performances under certain amount of model uncertainties. A proof of concept in the experimental test using a low cost RC vehicle platform further ensures the feasibility of the control logic.

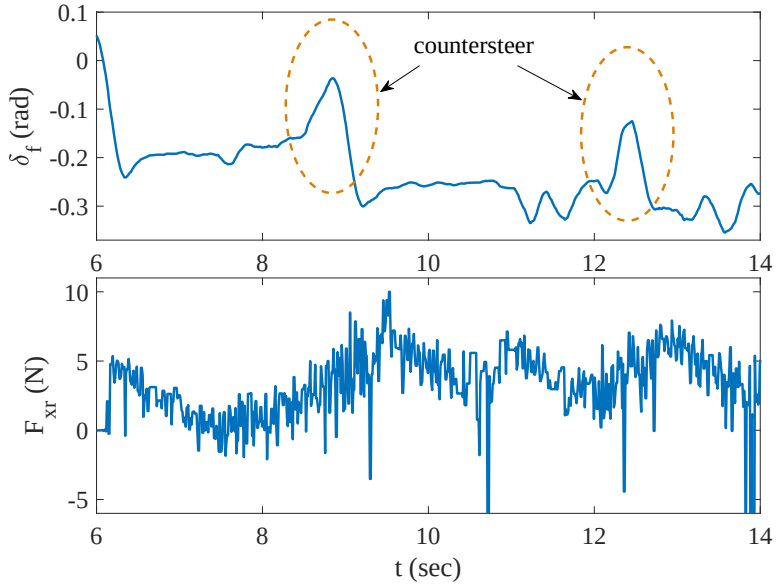


Figure 6.15: The control inputs commands during the experimental run.

Bibliography

[1] F. Plestan, Y. Shtessel, V. Bregeault, and A. Poznyak, "New methodologies for adaptive sliding mode control," *International journal of control*, vol. 83, no. 9, pp. 1907–1919, 2010.

[2] D. C. McFarlane and K. Glover, *Robust controller design using normalized coprime factor plant descriptions*. Springer, 1990, vol. 138.

[3] A. K. Packard, "What's new with mu: Structured uncertainty in multivariable control." 1989.

[4] V. Utkin, J. Guldner, and J. Shi, *Sliding mode control in electro-mechanical systems*. CRC press, 2009.

[5] R. A. Nichols, R. T. Reichert, and W. J. Rugh, "Gain scheduling for h-infinity controllers: A flight control example," *IEEE Transactions on Control systems technology*, vol. 1, no. 2, pp. 69–79, 1993.

[6] J. Gadewadikar, F. Lewis, K. Subbarao, and B. M. Chen, "Structured h-infinity command and control-loop design for unmanned helicopters," *Journal of guidance, control, and dynamics*, vol. 31, no. 4, pp. 1093–1102, 2008.

[7] M. Krstic, I. Kanellakopoulos, P. V. Kokotovic *et al.*, *Nonlinear and adaptive control design*. Wiley New York, 1995, vol. 222.

[8] K. Narendra and L. Valavani, "Stable adaptive controller design–direct control," *IEEE Transactions on Automatic Control*, vol. 23, no. 4, pp. 570–583, 1978.

[9] F. Borrelli, *Constrained optimal control of linear and hybrid systems*. Springer, 2003, vol. 290.

[10] A. Bemporad and M. Morari, "Robust model predictive control: A survey," in *Robustness in identification and control*. Springer, 1999, pp. 207–226.

[11] W. Langson, I. Chryssochoos, S. Raković, and D. Q. Mayne, "Robust model predictive control using tubes," *Automatica*, vol. 40, no. 1, pp. 125–133, 2004.

[12] M. Rubagotti, D. M. Raimondo, A. Ferrara, and L. Magni, "Robust model predictive control with integral sliding mode in continuous-time sampled-data nonlinear systems," *Automatic Control, IEEE Transactions on*, vol. 56, no. 3, pp. 556–570, 2011.

[13] Y.-W. Liao, S. Pan, F. Borrelli, and J. K. Hedrick, "Adaptive sliding mode control without knowledge of uncertainty bounds," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 905–911.

[14] Y.-W. Liao and J. K. Hedrick, "Robust model predictive control with discrete-time integral sliding surface," in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 1641–1646.

[15] ——, "Discrete-time integral sliding model predictive control for unmatched disturbance attenuation," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 2675–2680.

[16] ——, "A discrete-time integral sliding model predictive control for obstacle avoidance of ground vehicles," in *ASME 2015 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers, 2015, pp. V003T44A002–V003T44A002.

[17] C. Edwards and S. Spurgeon, *Sliding mode control: theory and applications*. Crc Press, 1998.

[18] I. Boiko, L. Fridman, and R. Iriarte, "Analysis of chattering in continuous sliding mode control," in *Proceedings of the 2005, American Control Conference, 2005.* IEEE, 2005, pp. 2439–2444.

[19] B. Yao and M. Tomizuka, "Smooth robust adaptive sliding mode control of manipulators with guaranteed transient performance," *Journal of dynamic systems, measurement, and control*, vol. 118, no. 4, pp. 764–775, 1996.

[20] V. I. Utkin, "Sliding modes in optimization and control problems," 1992.

[21] M.-L. Tseng and M.-S. Chen, "Chattering reduction of sliding mode control by low-pass filtering the control signal," *Asian Journal of control*, vol. 12, no. 3, pp. 392–398, 2010.

[22] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. prentice-Hall Englewood Cliffs, NJ, 1991, vol. 199, no. 1.

[23] S. Sastry and M. Bodson, *Adaptive control: stability, convergence and robustness*. Courier Corporation, 2011.

[24] Y.-J. Huang, T.-C. Kuo, and S.-H. Chang, "Adaptive sliding-mode control for nonlinearsystems with uncertain parameters," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 38, no. 2, pp. 534–539, 2008.

[25] M. A. Hussain and P. Y. Ho, "Adaptive sliding mode control with neural network based hybrid models," *Journal of Process Control*, vol. 14, no. 2, pp. 157–176, 2004.

[26] C. E. Hall and Y. B. Shtessel, "Sliding mode disturbance observer-based control for a reusable launch vehicle," *Journal of guidance, control, and dynamics*, vol. 29, no. 6, pp. 1315–1328, 2006.

[27] V. I. Utkin and A. S. Poznyak, "Adaptive sliding mode control," in *Advances in sliding mode control*. Springer, 2013, pp. 21–53.

[28] B. Draženović, "The invariance conditions in variable structure systems," *Automatica*, vol. 5, no. 3, pp. 287–295, 1969.

[29] H. K. Khalil, "Nonlinear systems," 2002.

[30] I. Barkana, "Defending the beauty of the invariance principle," *International Journal of Control*, vol. 87, no. 1, pp. 186–206, 2014.

[31] F. Blanchini, "Lyapunov methods in robustness. an introduction," *Lecture notes in Automatic Control, Bertinoro (Italy)*, 2009.

[32] B. Wang, X. Yu, and G. Chen, "Zoh discretization effect on single-input sliding mode control systems with matched uncertainties," *Automatica*, vol. 45, no. 1, pp. 118–125, 2009.

[33] M. N. Zeilinger, D. M. Raimondo, A. Domahidi, M. Morari, and C. N. Jones, "On real-time robust model predictive control," *Automatica*, vol. 50, no. 3, pp. 683–694, 2014.

[34] P. Falugi and D. Q. Mayne, "Getting robustness against unstructured uncertainty: A tube-based mpc approach," 2014.

[35] A. Bemporad, F. Borrelli, and M. Morari, "Min-max control of constrained uncertain discrete-time linear systems," *Automatic Control, IEEE Transactions on*, vol. 48, no. 9, pp. 1600–1606, 2003.

[36] A. Richards and J. How, "Robust stable model predictive control with constraint tightening," in *American Control Conference, 2006.* IEEE, 2006, pp. 6–pp.

[37] R. C. Shekhar and J. M. MacIejowski, "Optimal constraint tightening policies for robust variable horizon model predictive control," in *Proceedings of the IEEE Conference on Decision and Control*, 2012, pp. 5170–5175.

[38] S. Rakovic, E. Kerrigan, K. Kouramas, and D. Q. Mayne, "Approximation of the minimal robustly positively invariant set for discrete-time lti systems with persistent state disturbances," 2003.

[39] L. Wu, P. Shi, and X. Su, *Sliding Mode Control of Uncertain Parameter-Switching Hybrid Systems*. John Wiley & Sons, 2014.

[40] B. M. Houda and N. Said, "Discrete predictive sliding mode control of uncertain systems," in *Systems, Signals and Devices (SSD), 2012 9th International Multi-Conference on*. IEEE, 2012, pp. 1–6.

[41] Y. Wang, W. Chen, M. Tomizuka, and B. N. Alsuwaidan, "Model predictive sliding mode control: For constraint satisfaction and robustness," in *ASME 2013 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers, 2013, p. V003T44A005.

[42] V. Utkin and J. Shi, "Integral sliding mode in systems operating under uncertainty conditions," in *Decision and Control, 1996., Proceedings of the 35th IEEE Conference on*, vol. 4. IEEE, 1996, pp. 4591–4596.

[43] K. Abidi, J.-X. Xu, and Y. Xinghuo, "On the discrete-time integral sliding-mode control," *Automatic Control, IEEE Transactions on*, vol. 52, no. 4, pp. 709–715, 2007.

[44] S. Z. Sarpturk, Y. Istefanopulos, and O. Kaynak, "On the stability of discrete-time sliding mode control systems," *IEEE Transactions on Automatic Control*, vol. 32, no. 10, pp. 930–932, 1987.

[45] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*. Nob Hill Publishing, 2017.

[46] M. Fiacchini, T. Alamo, and E. F. Camacho, "On the computation of convex robust control invariant sets for nonlinear systems," *Automatica*, vol. 46, no. 8, pp. 1334–1338, 2010.

[47] S. V. Rakovic, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne, "Invariant approximations of the minimal robust positively invariant set," *IEEE Transactions on Automatic Control*, vol. 50, no. 3, pp. 406–410, 2005.

[48] D. Q. Mayne, M. M. Seron, and S. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[49] M. Zhao and X. Tang, "Robust tube-based mpc with piecewise affine control laws," in *Abstract and Applied Analysis*, vol. 2014. Hindawi Publishing Corporation, 2014.

[50] T. Pilutti, G. Ulsoy, and D. Hrovat, "Vehicle steering intervention through differential braking," *Journal of dynamic systems, measurement, and control*, vol. 120, no. 3, pp. 314–321, 1998.

[51] H. E. Tseng, B. Ashrafi, D. Madau, T. A. Brown, and D. Recker, "The development of vehicle stability control at ford," *IEEE/ASME transactions on mechatronics*, vol. 4, no. 3, pp. 223–234, 1999.

[52] M. Hara, S. Kamio, M. Takao, K. Sakita, and T. Abe, "Traction control system," May 28 1991, uS Patent 5,018,595.

[53] R. Wilson-Jones, R. H. A. H. Tribe, and M. Appleyard, "Driver assistance system for a vehicle," Jun. 9 1998, uS Patent 5,765,116.

[54] S. Lee and J.-B. Song, "Robust mobile robot localization using optical flow sensors and encoders," in *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, vol. 1. IEEE, 2004, pp. 1039–1044.

[55] D. M. Bevly, J. Ryu, and J. C. Gerdes, "Integrating ins sensors with gps measurements for continuous estimation of vehicle sideslip, roll, and tire cornering stiffness," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 4, pp. 483–493, 2006.

[56] D. M. Bevly, J. C. Gerdes, C. Wilson, and G. Zhang, "The use of gps based velocity measurements for improved vehicle state estimation," in *American Control Conference, 2000. Proceedings of the 2000*, vol. 4. IEEE, 2000, pp. 2538–2542.

[57] D. Selmanaj, M. Corno, G. Panzani, and S. Savaresi, "Robust vehicle sideslip estimation based on kinematic considerations," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14 855–14 860, 2017.

[58] J. Farrelly and P. Wellstead, "Estimation of vehicle lateral velocity," in *Control Applications, 1996., Proceedings of the 1996 IEEE International Conference on.* IEEE, 1996, pp. 552–557.

[59] S. Strano and M. Terzo, "Constrained nonlinear filter for vehicle sideslip angle estimation with no a priori knowledge of tyre characteristics," *Control Engineering Practice*, vol. 71, pp. 10–17, 2018.

[60] J. Dakhlallah, S. Glaser, S. Mammar, and Y. Sebsadji, "Tire-road forces estimation using extended kalman filter and sideslip angle evaluation," in *American Control Conference, 2008.* IEEE, 2008, pp. 4597–4602.

[61] Y. Aoki, T. Inoue, and Y. Hori, "Robust design of gain matrix of body slip angle observer for electric vehicles and its experimental demonstration," in *Advanced Motion Control, 2004. AMC'04. The 8th IEEE International Workshop on.* IEEE, 2004, pp. 41–45.

[62] L. Li, G. Jia, X. Ran, J. Song, and K. Wu, "A variable structure extended kalman filter for vehicle sideslip angle estimation on a low friction road," *Vehicle System Dynamics*, vol. 52, no. 2, pp. 280–308, 2014.

[63] L. Shao, C. Jin, C. Lex, and A. Eichberger, "Nonlinear adaptive observer for side slip angle and road friction estimation," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 6258–6265.

[64] S.-H. You, J.-O. Hahn, and H. Lee, "New adaptive approaches to real-time estimation of vehicle sideslip angle," *Control Engineering Practice*, vol. 17, no. 12, pp. 1367–1379, 2009.

[65] C.-S. Liu and H. Peng, "A state and parameter identification sclieme for linearly parameterized systems," 1998.

[66] H. F. Grip, L. Imsland, T. A. Johansen, J. C. Kalkkuhl, and A. Suissa, "Vehicle sideslip estimation," *IEEE control systems*, vol. 29, no. 5, 2009.

[67] J. L. Coyte, B. Li, H. Du, W. Li, D. Stirling, and M. Ros, "Decision tree assisted ekf for vehicle slip angle estimation using inertial motion sensors," in *Neural Networks (IJCNN), 2014 International Joint Conference on*. IEEE, 2014, pp. 940–946.

[68] A. Y. U. H. Peng and H. Tseng, "Experimental verification of lateral speed estimation methods," *a a*, vol. 1.

[69] B. Boada, M. Boada, and V. Diaz, "Vehicle sideslip angle measurement based on sensor data fusion using an integrated anfis and an unscented kalman filter algorithm," *Mechanical Systems and Signal Processing*, vol. 72, pp. 832–845, 2016.

[70] C. M. Kang, S.-H. Lee, and C. C. Chung, "Vehicle lateral motion estimation with its dynamic and kinematic models based interacting multiple model filter," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 2449–2454.

[71] B.-C. Chen and F.-C. Hsieh, "Sideslip angle estimation using extended kalman filter," *Vehicle System Dynamics*, vol. 46, no. S1, pp. 353–364, 2008.

[72] D. Piyabongkarn, R. Rajamani, J. A. Grogg, and J. Y. Lew, "Development and experimental evaluation of a slip angle estimator for vehicle stability control," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 1, pp. 78–88, 2009.

[73] M. De Martino, F. Farroni, N. Pasquino, A. Sakhnevych, and F. Timpone, "Real-time estimation of the vehicle sideslip angle through regression based on principal component analysis and neural networks," in *Systems Engineering Symposium (ISSE), 2017 IEEE International*. IEEE, 2017, pp. 1–6.

[74] A. Y. Ungoren, H. Peng, and H. Tseng, "A study on lateral speed estimation methods," *International Journal of Vehicle Autonomous Systems*, vol. 2, no. 1-2, pp. 126–144, 2004.

[75] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[76] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, pp. 267–288, 1994.

[77] E. M. Eksioglu and A. K. Tanc, "Rls algorithm with convex regularization," *IEEE Signal Processing Letters*, vol. 18, no. 8, pp. 470–473, 2011.

[78] I. D. Landau, "System identification and control design," 1990.

[79] Y. Song and J. W. Grizzle, "The extended kalman filter as a local asymptotic observer for nonlinear discrete-time systems," in *American Control Conference, 1992.* IEEE, 1992, pp. 3365–3369.

[80] B. Anderson, "A simplified viewpoint of hyperstability," *IEEE Transactions on Automatic Control*, vol. 13, no. 3, pp. 292–294, 1968.

[81] H. K. Khalil, "Noninear systems," *Prentice-Hall, New Jersey*, vol. 2, no. 5, pp. 5–1, 1996.

[82] Y. Landau, "Adaptive control: the model reference approach, 1979," *Marcel Dekker, New York). Hac, A. Adaptive control of vehicle suspension. Veh. System Dynamics*, vol. 16, pp. 74–77, 1987.

[83] H. B. Pacejka and E. Bakker, "The magic formula tyre model," *Vehicle system dynamics*, vol. 21, no. S1, pp. 1–18, 1992.

[84] R. C. Yates, *A Handbook on Curves and their Properties.* JW Edwards, 1947.

[85] W. Nelson, "Continuous-curvature paths for autonomous vehicles," in *Proceedings, 1989 International Conference on Robotics and Automation.* IEEE, 1989, pp. 1260–1264.

[86] A. Konyukhov and K. Schweizerhof, "On the solvability of closest point projection procedures in contact analysis: Analysis and solution strategy for surfaces of arbitrary geometry," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 33-40, pp. 3045–3056, 2008.

[87] A. Mojaev and A. Zell, "Tracking control and adaptive local navigation for nonholonomic mobile robot," in *In The 8 th Conference on Intelligent Autonomous System.* Citeseer, 2004.

[88] R. Y. Hindiyeh and J. C. Gerdes, "A controller framework for autonomous drifting: Design, stability, and experimental validation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 136, no. 5, p. 051015, 2014.

[89] J. Edelmann and M. Plöchl, "Handling characteristics and stability of the steady-state powerslide motion of an automobile," *Regular and Chaotic Dynamics*, vol. 14, no. 6, p. 682, 2009.

[90] J. Gonzales, F. Zhang, K. Li, and F. Borrelli, "Autonomous drifting with onboard sensors," in *Proc. 13th Int. Symp. Advanced Vehicle Control (AVEC16)*, 2016, p. 133.

[91] E. Ono, S. Hosoe, H. D. Tuan, and S. Doi, "Bifurcation in vehicle dynamics and robust front wheel steering control," *IEEE transactions on control systems technology*, vol. 6, no. 3, pp. 412–420, 1998.

[92] E. Velenis, D. Katzourakis, E. Frazzoli, P. Tsiotras, and R. Happee, "Stabilization of steady-state drifting for a rwd vehicle," in *Proceedings of AVEC*, 2010.

[93] D. Swaroop, J. Gerdes, P. P. Yip, and J. K. Hedrick, "Dynamic surface control of nonlinear systems," in *Proceedings of the 1997 American Control Conference (Cat. No. 97CH36041)*, vol. 5. IEEE, 1997, pp. 3028–3034.

[94] H. Nakano, J. Kinugawa, and K. Kosuge, "Control of a four-wheel independently driven electric vehicle with a large sideslip angle," in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*. IEEE, 2014, pp. 265–270.

[95] M. Werling, P. Reinisch, and L. Gröll, "Robust power-slide control for a production vehicle," *International Journal of Vehicle Autonomous Systems*, vol. 13, no. 1, pp. 27–42, 2015.

[96] J. Y. Goh and J. C. Gerdes, "Simultaneous stabilization and tracking of basic automobile drifting trajectories," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 597–602.