**Title**

# Nonlinear Longitudinal Controller Development and Real-time Implementation



**Authors**
Lu, Xiao-Yun
Tan, Han-Shue
Empey, Dan
et al.

**Publication Date**
2000-09-01

**This paper uses Postscript Type 3 fonts. Although reading it on the screen is difficult it will print out just fine.**

CALIFORNIA PATH PROGRAM
INSTITUTE OF TRANSPORTATION STUDIES
UNIVERSITY OF CALIFORNIA, BERKELEY

# Nonlinear Longitudinal Controller Development and Real-time Implementation

**Xiao-Yun Lu, Han-Shue Tan, Dan Empey, Steven E. Shladover, J. Karl Hedrick**

**California PATH Research Report**
**UCB-ITS-PRR-2000-15**

This work was performed as part of the California PATH Program of the University of California, in cooperation with the State of California Business, Transportation, and Housing Agency, Department of Transportation; and the United States Department of Transportation, Federal Highway Administration.



Report for MOU 331

September 2000

ISSN 1055-1425

CALIFORNIA PARTNERS FOR ADVANCED TRANSIT AND HIGHWAYS

# Nonlinear Longitudinal Controller Development and Real-time Implementation (Report on Part of MOU331)

Xiao-Yun Lu
Han-Shue Tan
Dan Empey
Steven E. Shladover
J. Karl Hedrick
PATH, ITS, U. C. Berkeley

July, 2000

**Acknowledgment**

This work was supported by California PATH Program of the University of California and Caltrans. The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the State of California. This report does not constitute a standard, specification, or regulation.

Other PATH AVCS group staff, notably, Wei-Bin Zhang, Paul Kretz, Benedicte Bougler have contributed in hardware support, constructive suggestions, providing documents of previous software and test driving. They are gratefully acknowledged.

## Keywords

*longitudinal control, car platooning, back-stepping control, multi-surface dynamic sliding mode control, integral filters, realtime control code*

## Abstract

This report presents the development of nonlinear longitudinal controllers and their realtime implementation for Automated Highway Systems (AHS) at PATH, U. C. Berkeley supported by the project MOU 331. The demo'97 realtime code was first reconstructed for further development. Based on the new framework, the implementation and comparison of nonlinear controllers in the longitudinal control for cars platooning and other maneuvers have been carried out. Specifically, a dynamic sliding mode controller and a dynamic back-stepping multiple surface controller have been implemented, which are then compared to the previously implemented adaptive sliding mode controller. In the dynamic back-stepping control, a new type of integral filters have been developed, which were used for filtering measured signals and to estimate the derivative of these signals in realtime.

These controllers have been implemented in realtime code and tested at low and high speed. Realtime run shows that each controller has its own characteristics. In general, the optimal dynamic back-stepping sliding surface controller has better performance by using the same set of measured data compared to previously implemented adaptive sliding mode controller. Emphasis has been put on the dynamic back-stepping multiple surface controller development.

# Chapter 1

# Preface

This project is supported by MOU331 started from May 1st 1999 carried out at PATH, U. C. Berkeley. The main purposes of this project are:

(1) To reconstruct the Demo 97 real-time code for further development such as new controller construction, fault detect and management;

(2) To develope new nonlinear controller using optimization in dynamic back-stepping sliding surface control to improve the performance of longitudinal control for car platooning and other maneuvers;

The development of the work is stated chronically as follows.

First, demo code d99_long.c has been generally cleared up, unused variables have been deleted. Then code was restructured. In the restructured code, only the part pertinent to controller remains. Other parts have been removed to isolated sub-modulars. This formate are easier to read and to reuse.

Based on the restructured framework, new controllers have been developed, implemented in C code and tested at both low and high speed. i.e. the dynamic sliding mode controller and the dynamic back-stepping multi-surface controller. Controller design methods for these three controllers are different which naturally leads to different closed-loop dynamics. It seems that the latter using optimal dynamic back-stepping sliding surface method has better performance in the following aspects:

(a) It has smaller transient and steady state tracking error. Thus it is particularly good for performance which requires tight distance control such as automated vehicle merging maneuver.

(b) It has much less settling time, only half of the demo code;

(c) Throttle command has much less chattering;

(d) It is more robust with respect to external disturbances including the geometric situation of the road (up and down), different number of passengers, on/off air conditioners;

This rest of the report are divided in three chapters. Chapter two describe the work on code reconstruction. Chapter three, mainly based a paper to be presented at AVEC 2000 [23], is to present in details the development, test and comparison of nonlinear longitudinal controllers with some new methods used. Chapter four is the prospective work in the future.

# Chapter 2

# Longitudinal Code Reconstruction

The original code has been reconstructed and several new sub-modulars have been built. The new code will be easier to read and to be used for future new controller construction. Main features are :

(1) Fault detect part and maneuver part have been separated into two subroutines: scenario and fault_diag.

(2) A new maneuver *struct* which contains all the maneuver state variables is used to pass these variables between different processes.

(3) All the table look-up mappings have been rewritten which include engine mapping, throttle mapping and gearshift coordination. The rewritten mappings have been tested by using at least two different interpolation methods for each mapping and written in a most reliable way. These mappings have been tested in all the real-time runs since then, which are proved to be reliable.

(4) Because the controller in the original code was adapted to the old engine mappings, some modification in the controller has been made for it to work properly using new engine mappings. The corrected code has been tested at RFS and Crows Landing test tracks, where it behaved similarly to the original code in normal cases.

## 2.1 Submodulars

In the new frame, due to the separation of maneuver and fault diagnosis parts into sub-modulars, the main body is much smaller and thus much easier to understand. In fact, the essential part is the controller: the upper controller to produce desired synthetic acceleration from distance error and the lower controller to produce throttle and brake control command from desired torque.

Now the main function of each sub-modular is briefly described. The communication part only contains read in data and put data to packet. It did not include the communication protocol which is part of the communication driver. It is still in the central part of spd_ctrl ( ).

The functions of each sub-modular are listed bellow. The relationship between these sub-modulars and the central part spd_ctrl ( ) is shown in Fig 2.1.

**trajectory ( )** - To calculate a proper continuous reference speed trajectory for regular platoon control for different test site.

**scenario ( )** - To calculate a proper reference trajectory for specific maneuvers such as splitting, joining, lane changing according to specified maneuver_id (newly separated from main body);

**radar observer ( )** input : Using radar range, radar_range_rate, speed, preceding speed and maneuver data structure to estimate relative_radar_distance, relative_speed.

**magnet_observer ( )** input : speed, preceding speed, maneuver structure, magnet_marker,

This sub-modular is not really a magnet measurement. In fact, 95 percent of data used by this sub-modular are from radar measurement. It is thus abandoned.

**mag_dist_obs ( )**, Newly developed, using speed, preceding speed, magnet counter and preceding magnet counter to estimate the relative moving distance between two vehicles, e.g. current car with respect to the preceding car or with respect to the leader car. To estimate relative distance, a radar may be used for initialization of the relative distance in the first 2 *sec* after starting. It is well-tested and proved to be robust. However, the estimation error increases a bit with the increase of vehicle speed. Maximum accumulated error recorded is $1.8m$ at speed of $56km/h$ with overall running distance of $1500m$. This sub-modular is heavily used in merging maneuver and fault management when radar is in fault.

**fault_detect ( )** to diagnose actuator faults with respect to some prescribed threshold, such as throttle, brake, etc. to ensure they operate properly, and to acknowledge the system by assigning the variable fault_mode a proper value.

**fault_diag ( )** to diagnose measurement faults, such as radar, communication, CAN bus, magnetometer and acknowledge to the system by assigning the variable faul_mode a proper value; Newly separated from the main body.

**Fault_manage ( )** measures to be taken when radar, communication or magnetometer reading are in fault. The basic treatment adopted is to split and stop gradually. This part is currently under construction and will be systematically considered in the future.

**eng_tbl ( )** Nonlinear engine mapping; It has been rewritten.

**alp_tbl ( )** Throttle mapping; It has been rewritten.

**gear_tbl ( )** To determine the corresponding gear ratio at different gear position; It has been rewritten.

**filt_1 ( ), filt_2 ( ), filt_3 ( ), filt_4 ( )** Newly constructed first and second order integral filters for filtering a measured signal which may be polluted by measurement noise and to estimate its derivative, which are implemented with $4th$ order Runge_Kutta method;

**Math. functions**: For sliding mode control, trajectory planning, etc. Newly developed.

**Communication** Communication in longitudinal code is just to read in data from other sources from a data structure and put data to data structure for communication. This part is still kept in the main part because to use a sub-modular to achieve the same purpose will not make it simpler but increase the memory use.

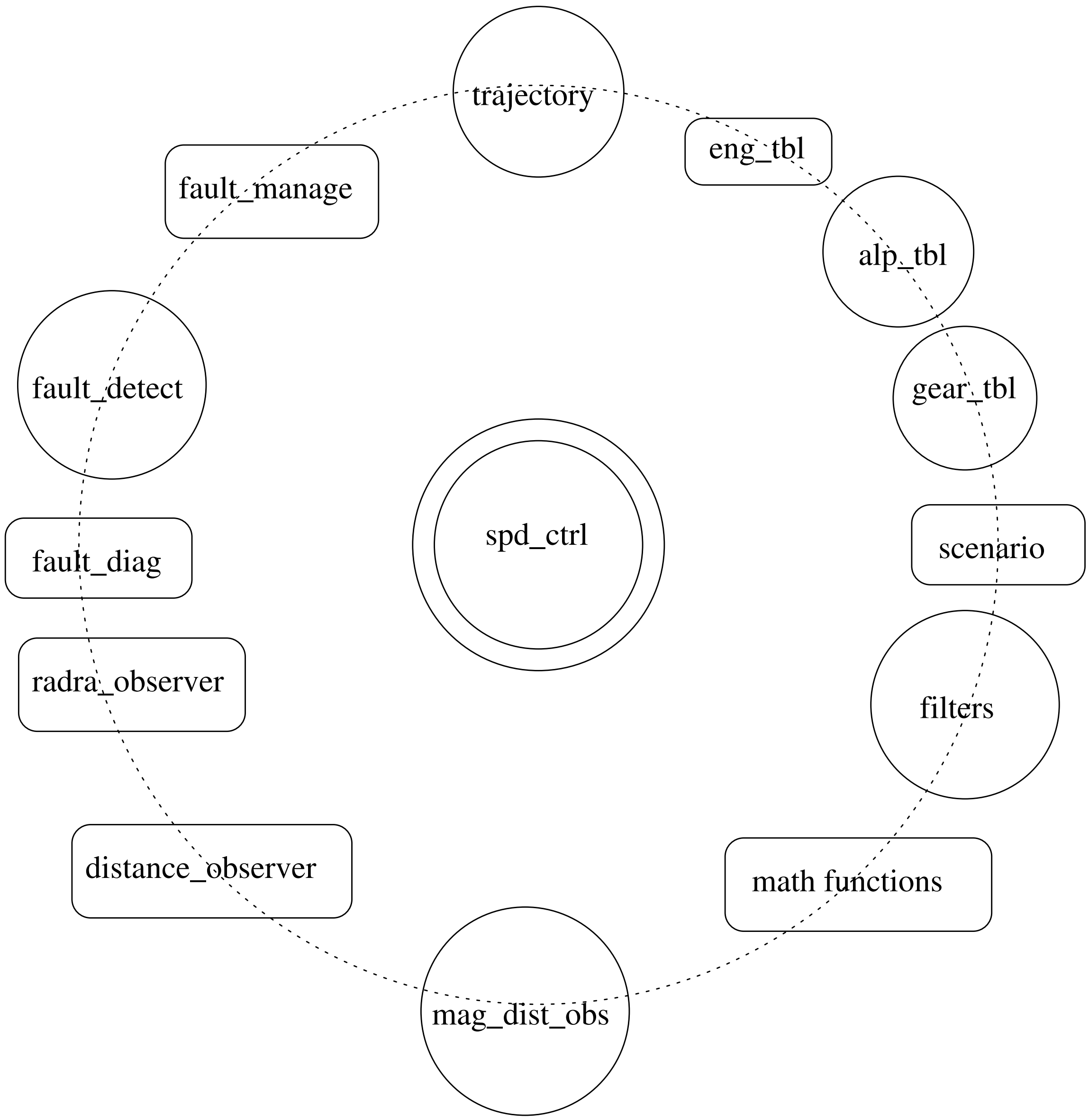


Figure 2.1: Real-time code structure

# Chapter 3

# Nonlinear Control Design and Implementation

This chapter will describe some newly implemented longitudinal controllers. They are compared with the controller used in Demo'97 code. Some other tools are presented such as a type of new integral filters which is used to filter measured signal and to estimate the derivative of the measured signal.

## 3.1 Introduction

Work for longitudinal control of vehicle platooning was initiated in [8, 11, 26, 27]. These works mainly concentrate on the control design of upper level. i.e. from desired distance, actual relative distances and relative speed between vehicles to generate a desired acceleration or torque for each vehicle. To design controller of this level, one needs to know vehicle dynamics. However, to implement practically, one needs to generate lower level command on each vehicle such as throttle control command and brake control command which will require details for engine and brake dynamics. The works in [4, 12, 25] are crucial steps towards this direction. Based on both vehicle dynamics and engine dynamics, sliding mode controller was designed and implemented [5, 6, 15, 13].

In this chapter, the overall following strategy is as follows and is the same for all the control design methods. i.e. each vehicle follows the vehicle in the immediate front. i.e. the relative distance and relative speed with respect to the preceding vehicle is measured and fed into the controller. In this formulation, radar measurement of relative distance between consecutive vehicles can be directly used while the data from the magnet distance observer is only used as accessary or in software redundancy. In this case, the error accumulation problem in distance estimation is avoided, but the string stability becomes a prominent problem. The string stability from a control design viewpoint is successfully discussed in [15, 32].

The following strategy is the same as the previous real-time implementation at PATH, UC Berkeley. It was successfully used in 1997 demo using 8 automated car [5, 15, 13, 33]. Based on the newly structured framework, two new controllers have been designed, implemented in real-time and tested in the track at Richmond Field Station at the speed of $40km/h$ and Crows Landing track at the speed of $73km/h$ for two car platooning. These two controllers are Dynamic Sliding Mode Controller and Optimal Dynamic Back-stepping Multiple Surface Controller. This paper is to present the control design strategies and implementation of these new controllers and to compare the test result. As in the original code, system model including vehicle dynamics and engine dynamics are nonlinear. The nonlinear vehicle dynamics is directly used for control design. The models for engine and brake are represented by several mappings which are not described here, which are referred to [15]. Prominent differences compared to the previous work lies in

(1) Overall control design methods are different;

(2) Data fusion and signal filtering methods are different;

(3) Some new integral filters were used to estimate the derivative of the composite reference state at each backing step in real-time and used to reduce measurement noise in measurement data filtering. For example, to calculate the reference vehicle acceleration from preceding speed and the derivative of engine manifold pressure.

## 3.2 Control Design Methods

This section presents three nonlinear control design methods and integral filters which are based on a new viewpoint. The main difference for these three methods lies in the control design logic and composite reference data. They tend to be more complicated in the following order

$$
\begin{aligned}
\text{sliding mode control} \quad &\rightarrow \quad \text{dynamic sliding surface control} \\
&\rightarrow \quad \text{optimal dynamic back-stepping sliding surface control}
\end{aligned}
$$

### 3.2.1 Integral Filters

Suppose $f(t)$ is a continuously measured signal which may not be differentiable and/or may contain measurement noise. In controller design $\frac{d}{dt}f(t)$ is required in practice. To avoid noise amplification and propagation in this calculation, integral filters are used to estimate the derivative. These integral filters are based on the solution of an inverse Taylor series problem. These second order and third order integral filters are generalized from first order filter used in [33].

**Inverse Taylor Series Problem**

For an arbitrary time instant $s = t_1$, let $t = t_1 - \Delta t$ with $|\Delta t|$ sufficiently small. It is to find the value of $f(s)$ at $s = t$ without resort to differentiation at time instant $t$. Construct a *formal Taylor series* as follows.

$$\begin{aligned} f(t_1) &= f(t + \Delta t) \\ &= f(t) + \dot{f}(t)\Delta t + \frac{1}{2!}\ddot{f}(t)(\Delta t)^2 + O\left((\Delta t)^3\right) \end{aligned}$$

where $f(t)$, $\dot{f}(t)$ and $\ddot{f}(t)$ are unknown and are to be found. Mathematically, this is an **inverse Taylor series problem**. In a usual Taylor series, from known $f(t)$, $\dot{f}(t)$ and $\ddot{f}(t)$, unknown $f(t_1)$ is constructed when $t = t_1 - \Delta t$ with $|\Delta t|$ sufficiently small. An important point here is that, to approximate $f(t_1)$ with $f(t) + \dot{f}(t)\Delta t + \frac{1}{2!}\ddot{f}(t)(\Delta t)^2$ , the error is in the order of $O\left((\Delta t)^3\right)$.

Let $\tau = \Delta t$. The following polynomials with coefficients from a Taylor series

$$\begin{aligned} G_1(s) &= 1 + \tau s \\ G_2(s) &= 1 + \tau s + \frac{1}{2!}\tau^2 s^2 \\ G_3(s) &= 1 + \tau s + \frac{1}{2!}\tau^2 s^2 + \frac{1}{3!}\tau^3 s^3 \end{aligned}$$

are all Hurwitz polynomials. This guarantees global stability of the filter models to be discussed.

**First Order Filter**

Construct the following filter from first order Taylor expansion

$$\tau \dot{z} = -z + f(t), \ \tau > 0 \tag{2.1}$$

Thus it is expected that

$$\dot{z} = \frac{1}{\tau}\left(-z + f(t)\right) \simeq \frac{d}{dt} f(t)$$

**Second order filter model 1**

Similarly, construct the following second filter from the second order Taylor expansion

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= \frac{2}{\tau^2}\left(-\tau z_2 - z_1 + f(t_1)\right) \\ \tau &> 0 \end{aligned} \tag{2.2}$$

Now the following approximation is expected

$$\begin{gathered} z_1 \to f(t_1) \\ z_1 + \tau z_2 \to f(t_1) \\ z_2 \to \tfrac{d}{dt} f(t_1) \end{gathered} \tag{2.3}$$

**Second Order Filter Model 2**

Alternatively, one can construct

$$\begin{aligned} \dot{z}_1 &= \frac{1}{\tau_1}\left(-z_1 + f(t_1)\right) \\ \dot{z}_2 &= \frac{1}{\tau_2}\left(-z_2 + z_1\right) \\ \tau_1, \tau_2 &> 0 \end{aligned} \tag{2.4}$$

The following approximation is expected:

$$\begin{aligned} z_1(t) &\to f(t_1) \\ \frac{1}{\tau_2}\left(-z_2 + z_1\right) &\to \frac{d}{dt} f(t_1) \end{aligned}$$

**Second Order Filter Model 3**

Similarly, with a different input signal to the second equation, it is obtained that

$$\begin{gathered} \dot{z}_1 = \tfrac{1}{\tau_1}\left(-z_1 + f(t_1)\right) \\ \dot{z}_2 = \tfrac{1}{\tau_2}\left[-z_2 + \tfrac{1}{\tau_1}\left(-z_1 + f(t_1)\right)\right] \end{gathered} \tag{2.5}$$

Now, the following approximation is expected

$$\begin{aligned} z_1 &\to f(t_1) \\ z_2 &\to \frac{d}{dt} f(t_1). \end{aligned}$$

The theoretical analysis of these filters are conducted in [18].

### 3.2.2 Sliding Mode Control

It is generally recognized that sliding mode control is robust with respect to external disturbances and are widely used in nonlinear control design [3, 7, 9, 14, 20, 21, 30, 31, 29, 34, 35, 36]. Sliding mode control design strategy may be divided into two independent procedures which are concerned with the choice of sliding surface and the choice of reachability condition to ensure the sliding surface is reached. This method can be applied to the following nonlinear models with matched uncertainties [22].

$$\begin{aligned} \dot{x}^{(1)} &= f^{(1)}(x^{(1)}, x^{(2)}) \\ \dot{x}^{(2)} &= f^{(2)}(x^{(1)}, x^{(2)}) + g^{(2)}(x^{(1)}, x^{(2)})\ u(t) + \Delta(t) \end{aligned}$$

where $(x^{(1)}, x^{(2)}) \in \mathcal{D} \subset \mathbb{R}^n$, $f = (f^{(1)}(.),\ f^{(2)}(.)) : \mathcal{D} \mapsto \mathbb{R}^m$ and $g^{(2)}(.,.) : \mathcal{D} \mapsto \mathbb{R}^m$ and piecewise continuous and

$$\begin{aligned} f(0) &= 0 \\ &\det\left[g^{(2)}(0) \neq 0\right]. \end{aligned}$$

$\Delta(t) : [0, \infty) \mapsto \mathbb{R}^m$, which is uncertainty. It is assumed to be Lebesgue measurable.

**Proper choice of sliding surface:**

(a) Choosing $s = s(x)$. This may be considered as a single geometric manifold which may be determined by a set of geometric equations [7, 34, 22]. This is termed a *static sliding surface* because it produces static feedback.

(b) Choosing a sliding surface which is a set of differential equations [9, 20, 21, 29, 30]. This is termed a *dynamic sliding surface* because it may produce dynamic feedback.

**Proper choice of sliding reachability condition:**

It is required that $s\ \dot{s} < 0$, if $s \neq 0$. The sliding reachability condition may produce a discontinuous control signal which is undesirable from the practical point of view. Further work has considered methods to reduce this chattering. There are several effective ways to accomplish this task.

(i) Introduce a layer of thickness $0 < \epsilon << 1$ around the sliding surface such that, when $||s|| > \epsilon$, the controller developed from the sliding mode reachability condition is employed and when $||s|| \leq \epsilon$, an alternative continuous control policy is employed [30, 34].

(ii) Adopt a dynamic sliding mode feedback where effective filtering of the control reduces chattering naturally [3, 20, 21, 29, 30].

(iii) Adopt a continuous reachability condition [7, 15, 21].

There are many such possible reachability conditions which may be broadly defined by:

**Definition** A *general sliding reachability condition* is defined as

$$\begin{aligned} \dot{s} &= -\gamma(s) \\ s &= [s_1, ..., s_m]^\top \end{aligned} \tag{2.6}$$

where $\gamma(s)$ satisfies

(1) $\gamma(s)$ is continuous if $s \neq 0$;

(2) (2.6) is globally asymptotically stable.

Particularly, the following two conditions imply condition (2).
(a)

$$s^\top \gamma(s) > 0, \; s \neq 0$$

(b) $\gamma(s)$ is bounded in a neighborhood of 0.

**Example 1** Consider the following sliding reachability conditions of dimension 1.

(1) A discontinuous sliding reachability condition may be chosen as

$$\begin{aligned} \gamma(s) &= k\ s + k_0\ sign(s) \\ k &> 0, \; k_0 \geq 0. \end{aligned}$$

(2) Its continuous ($\mathcal{C}^0$)round-off is

$$\begin{aligned} \gamma(s) &= k\ s + k_0\ sat(\frac{s}{\varepsilon}) \\ k &> 0, \; k_0 \geq 0 \end{aligned}$$

where $\varepsilon > 0$ determines the thickness of the boundary layer.

However, sliding mode control can only deal with matched uncertainties appearing additively in systems model. Thus for practical application, some other method needs to be incorporated to deal with un-matched uncertainties.

### 3.2.3 Dynamic Sliding Surface Control

The dynamic sliding surface choice is one step forward compared to the previous sliding mode control. The difference lies in the formulation of the error dynamics. The sliding surface is still a first order dynamics similar as above. Sliding reachability condition here is composed of a continuous part and continuous round-off of a switching law. The prominent difference is that a third order integral filter is used to calculate the acceleration and the derivative of desired engine manifold pressure. This effectively avoids the above primitive numerical derivative in real-time implementation. Dynamically, this is equivalent to inserting a third order system and a second order system in the closed-loop. This results naturally a dynamic feedback. Thus essentially, the closed-loop system is of higher dimension compared to the previous design. The extra dimension comes from the integral filters. In this design, sliding gain choice is up to the designer.

### 3.2.4 Optimal Dynamic Back-stepping Multiple Surface Control

This method combines the advantages of different control design methods. Besides, sub-optimization is involved. The main features of this design method can be described as:

(1) Sliding mode method: Using sliding surface and sliding reachability condition as above.

(2) Back-stepping is used in control design logic in formulating error dynamics. In each step, a proper sliding surface is chosen. Thus multiple sliding surfaces naturally result in the end. This is a promising way in dealing with additive unmatched uncertainties in nonlinear models [33].

(3) Integral filters are used to calculate the derivatives of reference signals at each consecutive step. Thus analytic differentiation of reference signal is avoided.

(4) The choice of sliding gain for each sliding surface is in a nonlinear $H_\infty$ sub-optimization approach [2, 16, 17].

This design method is first studied in [15, 33] without sub-optimization. Consider the following SISO nonlinear uncertain system in strict feedback form with uncertainties

$$\begin{aligned}
\dot{x}_1 &= x_2 + \varphi_1(x_1) + \Delta_1(x_1) \\
\dot{x}_2 &= x_3 + \varphi_2(x_1, x_2) + \Delta_2(x_1, x_2) \\
&\cdots\cdots \\
\dot{x}_{n-1} &= x_n + \varphi_{n-1}(x_1, x_2, ..., x_{n-1}) \\
&\quad + \Delta_{n-1}(x_1, x_{n-1}, ..., x_{n-1}) \\
\dot{x}_n &= \varphi(x_1, x_2, ..., x_n) + \psi(x_1, x_2, ..., x_n)u \\
&\quad + \Delta_n(x_1, x_2, ..., x_n)
\end{aligned} \tag{2.7}$$

where

$$\begin{aligned}
|\Delta_i| &\leq \rho_i \left\| x^{(i)} \right\| + l_i \\
x^{(i)} &= (x_1, ..., x_i) \\
\rho_i &\geq 0, \; l_i \geq 0 \\
i &= 1, ..., n.
\end{aligned}$$

Here the only assumption is that

$$\psi(x_1, x_2, ..., x_n) \neq 0$$

in the region of interests. The control tasks are

(a) to make $x_1$ to asymptotically track $x_{1d}$ which may not be differentiable;

(b) to render the closed loop system robustly stable with respect to the uncertainties.

Design a controller with the following procedure with sliding reachability condition $\dot{S} = -kS - k_0 sat(S)$.

**Step 1** Let

$$S_1 = x_1 - x_{1d}$$

Formally differentiate $S_1$ and set

$$\begin{aligned}
\dot{S}_1 &= -\gamma_1(S_1) + S_2 + \Delta_1 \\
S_2 &= x_2 - x_{2d}
\end{aligned}$$

where $x_{2d}$ is to be determined. Besides, $x_{1d}$ may not be differentiable. To avoid this difficulty, use the constructed derivative $x_{1d}\_dot$ from an integral filter to replace the formal derivative $\frac{d}{dt}(x_{1d})$ in $\dot{S}$. It is obtained that

$$\begin{aligned} \dot{x}_1 &= x_1\_dot - \gamma_1(S_1) + S_2 + \Delta_1(x_1) \\ &= x_2 + \varphi_1(x_1) + \Delta_1(x_1) \end{aligned}$$

which leads to

$$x_{2d} = x_{1d}\_dot - \gamma_1(S_1) - \varphi_1(x_1).$$

Thus, to make $x_1$ track $x_{1d}$, it is necessary and sufficient that $x_2$ track $x_{2d}$.

**Step i** $(i > 1)$ Let

$$S_{i+1} = x_{i+1} - x_{(i+1)d}$$

where $x_{(i+1)d}$ is to be determined. Formally differentiate $S_i$ and set

$$\dot{S}_i = -\gamma_i(S_i) + S_{i+1} + \Delta_i$$

Then use the filtered signal $x_{id}\_dot$ to replace $\frac{d}{dt}(x_{id})$ in $\dot{S}_i$, it is obtained that

$$\begin{aligned} \dot{x}_i &= x_{id}\_dot - \gamma_i(S_i) + S_{i+1} + \Delta_i \\ &= x_i + \varphi_i(x_1, x_2, ..., x_i) + \Delta_i \end{aligned}$$

which leads to

$$x_{(i+1)d} = x_{id}\_dot - \gamma_i(S_i) - \varphi_i(x_1, x_2, ..., x_i)$$

This is true for $i = 1, 2, .., n-2$.

**Step n-1** Formally differentiate

$$\begin{aligned} S_n &= x_n - x_{nd} \\ x_{nd} &= x_{n-1d}\_dot - \gamma_{n-1}(S_{n-1}) \\ &\quad -\varphi_{n-1}(x_1, x_2, ..., x_{n-1}). \end{aligned}$$

Set

$$\dot{S}_n = -\gamma_n(S_n) + \Delta_n$$

and replace $\dot{x}_{nd}$ with the filtered signal $x_{nd}\_dot$ constructed from a filter. It is obtained that

$$\begin{aligned} \dot{x}_n &= x_{nd}\_dot - \gamma_n(S_n) + \Delta_n \\ &= \varphi_n(x_1, x_2, ..., x_n) + \psi(x_1, x_2, ..., x_n)u + \Delta_i \end{aligned}$$

from which the controller is constructed as

$$u = \frac{1}{\psi(x_1, x_2, ..., x_n)} \left[x_{nd}\_dot - \gamma_n(S_n) - \varphi_n(x_1, x_2, ..., x_n)\right]$$

In $S-$coordinates, the closed-loop system is in the form

$$\begin{aligned}
\dot{S}_1 &= -\gamma_1(S_1) + S_2 + \Delta_1 \\
&\quad ...... \\
\dot{S}_i &= -\gamma_i(S_i) + S_{i+1} + \Delta_i \\
&\quad ...... \\
\dot{S}_{n-1} &= -\gamma_{n-1}(S_{n-1}) + S_n + \Delta_{n-1} \\
\dot{S}_n &= -\gamma_n(S_n) + \Delta_n
\end{aligned}$$

where the nonlinear part $\gamma(.)$ is determined by sliding reachability condition [20, 21]. Once the form of the sliding reachability conditions are chosen, their gains can be optimized in an $H_\infty$ sub-optimization approach [19].

## 3.3 Controller Construction and Implementation

This section addresses main possible uncertainties and upper level controller constructions using the above three methods and .

### 3.3.1 Real-time Uncertainties

Uncertainties in real-time implementation may be classified follows:

(1) Model mismatch and external disturbances: such as switching on/off of air conditioner, different number of passengers, vehicle aerodynamic force variation and lateral cornering force on the steered wheels, which has a component in the longitudinal direction. When lateral control and longitudinal control are separately applied, this coupling force is not compensated for in the longitudinal control.

(2) System parameter uncertainty: System parameters adopted, which include vehicle parameters, sensor parameters, and actuator parameters, are subject to dynamic variation and different road situation.

(3) Measurement noise to signals: This includes the noise to the speedometer, to radar distance measurement, magnets arrangement error on the ground and the magnetometer measurement error, the error caused by actuators etc.

To improve performance, all the three types of uncertainties need to be compensated in practical control implementation. The simplest case is to assume that the uncertainties of type (1) and (2) are approximated by additively uncertainties in the model, which can be classified as (a) matched and (b) un-matched.

From a dynamics point of view, additive uncertainties are to be compensated in control design although one cannot always achieve this. The measurement noise fed into the controller, however, need to be treated with proper filters. It cannot to be compensated by the controller itself in general.

### 3.3.2 Sliding Surface Controller

The sliding surface is chosen as follows for each pair of consecutive cars

$$S_i = \dot{\varepsilon}_i + a_1 \varepsilon_i + q_1 \left(\dot{x}_i - \dot{x}_{lead}\right) + q_2 \left(x_i - x_{lead} + \sum_{j=1}^{i-1} D_j\right)$$

where $a_1, q_1, q_2 > 0$ and

$$\varepsilon_i = (x_i - x_{i-1}) - D_i$$

$D_i-$ desired spacing between $i-$th and $(i+1)-$th car

$x_i-$ the position of $i-$th car with respect to the leader car

$\varepsilon_i-$ distance error to be regulated

Then the synthetic acceleration can be solved from the sliding reachability condition by using this sliding surface.

It is noted that, in the calculate of $\dot{S}$, one has to face the problem of differentiating $\dot{\varepsilon}_i$ which is essentially the relative speed between car $i$ and car $i-1$. This is implemented simply by using a moving average filter and a numerical derivative

$$\frac{\Delta\, \dot{\varepsilon}_i}{\Delta t}$$

where $\Delta t$ is the sampling time period in work of [5]. It is noted that, using this method, only matched uncertainty $\Delta(t)$, i.e. appearing in the speed channel, can be expelled by properly choosing sliding reachability conditions. However, the uncertainty appearing in the distance channel cannot be expelled in this way. The measurement noise is expected to be filtered out by the moving average filter, but it brings a prominent delay to the controller at the same time. Another feature is that sliding reachability condition used in [5] is continuous which is actually an exponential reaching law.

### 3.3.3 Dynamic Sliding Surface Controller

Sliding surface construction is similar as above but the sliding reachability condition is chosen discontinuous. Besides, the following third order filter is used to estimate the derivative of the speed error.

$$\dot{z}_1 = \frac{1}{\tau_1}\left(-z_1 + \dot{\varepsilon}_i\right)$$

$$\begin{aligned}\dot{z}_2 &= z_3 \\ \dot{z}_3 &= \frac{2}{\tau_2^2}\left(-\tau_2 z_3 - z_2 + \frac{1}{\tau_1}(-z_1 + \dot{\varepsilon}_i)\right)\end{aligned}$$

where $\tau_1, \tau_2 > 0$ are integral filter gains. Now using the following estimation

$$\frac{d}{dt}\left(\dot{\varepsilon}_i\right) \approx z_2 + \tau_2 z_3$$

Similar filter has been used to estimate the derivative of manifold pressure.

Test result shows that performance improvement achieved in throttle command compared to the previous work is that chattering has been greatly reduced. However, integral filter can only improve in measurement noise reduction as expected. It does help in dealing with unmatched uncertainties. That is why there is not much improvement in distance tracking error compared to previous work [5].

### 3.3.4 Optimal Dynamic Back-stepping Sliding Surface Control

Here the design logic is different from the previous two methods. It serves to deal with unmatched uncertainties. There are two sliding surfaces for each pair of cars.

$$S_{i1} = (x_i - x_{i-1}) - D_i$$

The first sliding reachability condition is chosen as

$$\begin{aligned}\dot{S}_{i1} &= -\gamma_1(S_{i1}) + S_{i2}. \\ \gamma_1(S_{i1}) &= -k_{i1}S_{i1} - k_{i10}\ sat(\frac{S_{i1}}{\varepsilon})\end{aligned}$$

where $\varepsilon$ satisfies $0 < \varepsilon << 1$. It controls the thickness of the boundary layer. The second sliding surface is chosen as

$$S_{i2} = \dot{x}_i - x_{2d}$$

where $x_{2d}$ is

$$x_{2d} = x_{1d}\_dot - \gamma_1(S_1)$$

Choose a reachability condition

$$\dot{S}_{i2} = -\gamma(S_{i2}) = -k_{i2}S_{i2} - k_{i20}\ sat(\frac{S_{i2}}{\varepsilon})$$

In the calculation of synthetic acceleration, the derivative of the composite reference state $x_{2d}$ is estimated by the following filter

$$
\begin{aligned}
\dot{z}_1 &= \frac{1}{\tau_1}\left(-z_1 + x_{2d}\right) \\
\dot{z}_2 &= z_3 \\
\dot{z}_3 &= \frac{2}{\tau_2^2}\left(-\tau_2 z_3 - z_2 + \frac{1}{\tau_1}(-z_1 + x_{2d})\right)
\end{aligned}
$$

where $\tau_1, \tau_2 > 0$. This leads to the estimation

$$
\frac{d}{dt}\left(x_{2d}\right) \approx z_2 + \tau_2 z_3
$$

Similar filter has been used in the calculation of the derivative of manifold pressure.

The prominent difference here is that

(1) both the uncertainties in the distance channel and that in the speed channel are expelled by proper choice of sliding reachability conditions;

(2) the switching gains $k_{i10}$ and $k_{i20}$ are chosen according to an estimated magnitude of the additive uncertainties;

(3) the gains $k_{i10}$ and $k_{i20}$ are chosen such that the closed-loop system is internally stable with the matched uncertainties attenuated [19].

### 3.3.5 Real-time Implementation

Lower level control commands are generated similarly for all the three controllers above. Data measurement time period is 20 milliseconds in real-time code. This is also the control command real-time step. Proceeding car speed, current car speed, and the distance between the current car and the proceeding car are the basic signals required to construct the controller. Proceeding speed is obtained from the proceeding car and from the lead car through communication; Current car speed is a fusion of the four measured wheel speed by speedometer; The distance to the preceding car is obtained by the filtering and fusing of radar range measurement.

In both simulation and real-time running, it appeared that second and third order filters are much better than first order filter in the sense that they are stronger in noises reduction and producing better approximation. Compared to moving average filter plus numerical derivative, one more advantage is that they cause much less time delay. In real-time code, the integral filters are implemented with the 3rd order Runge-Kutta method.

## 3.4 High Speed Test at Crows Landing

Crows Landing high speed test of the following three longitudinal controllers from Sept. 28-30 :

(1) restructured demo code (based on d99_long.c)

(2) dynamic sliding mode controller (using integral filters))

(3) optimal dynamic back-stepping sliding surface controller (using integral filters)

These three controllers were all working although performance were different.

**1. Restructured Demo Code**

The restructured demo code has a similar performance as the original code (d99_long.c), which is very difficult to improve just by gain tuning. The parameter tuning seems to be optimal with respect to the control design method. Because moving average filter is used throughout the code, including those sed in numerical derivatives. Although they are robust, some disadvantages are observed:

(1) they causes large time delay;

(2) they causes relatively large estimation error which affects the closed-loop dynamics a lot;

(3) noise reduction is not very effective, which causes chattering in throttle command.

The data is shown in Fig. 3.1.

**2. Dynamic Sliding Mode Controller**

Compared to the demo code d99_long.c, performance is not improved as shown in Fig. 3.2. This may be improved further by parameter tuning. However, due to the use to integral filters and slightly different formation of the sliding surface, noise/disturbance has been largely reduced in throttle command.

**3. Dynamic Back-stepping Sliding Surface Controller**

In primary test, the transient performance of the back-stepping controller is not good enough as shown in Fig. 3.3. There were basically two defects:

(1) Settling time is too long (about 60 seconds after controller being activated);

(2) Transient distance error is too large ($-0.6 \sim 0.5$);

(3) There was a large throttle over-shoot around the first gear shifting;

Compared to the demo code, the advantages are:

(1) Throttle does not chatter that much;

(2) Steady state performance is better.

After further fine tuning, main in gain scheduling, overall performance of this controller has been improved further. Compared to the original demo code, main improvement are in the following aspects as shown in Fig. 3.4:

(a) Transient performance is much better: less distance tracking error: $-0.2 \sim 0.3$ compared to that of $-0.5 \sim 0.6$ in demo code; Settling time is shorter - 35 seconds after control started compared to 60 seconds in demo code;

(b) Distance tracking error in steady state is less that $0.03m$ compared to 0.2 in demo code;

(c) Throttle do not chatter that much as the demo code;

(d) It seems more robust with respect to external disturbances.

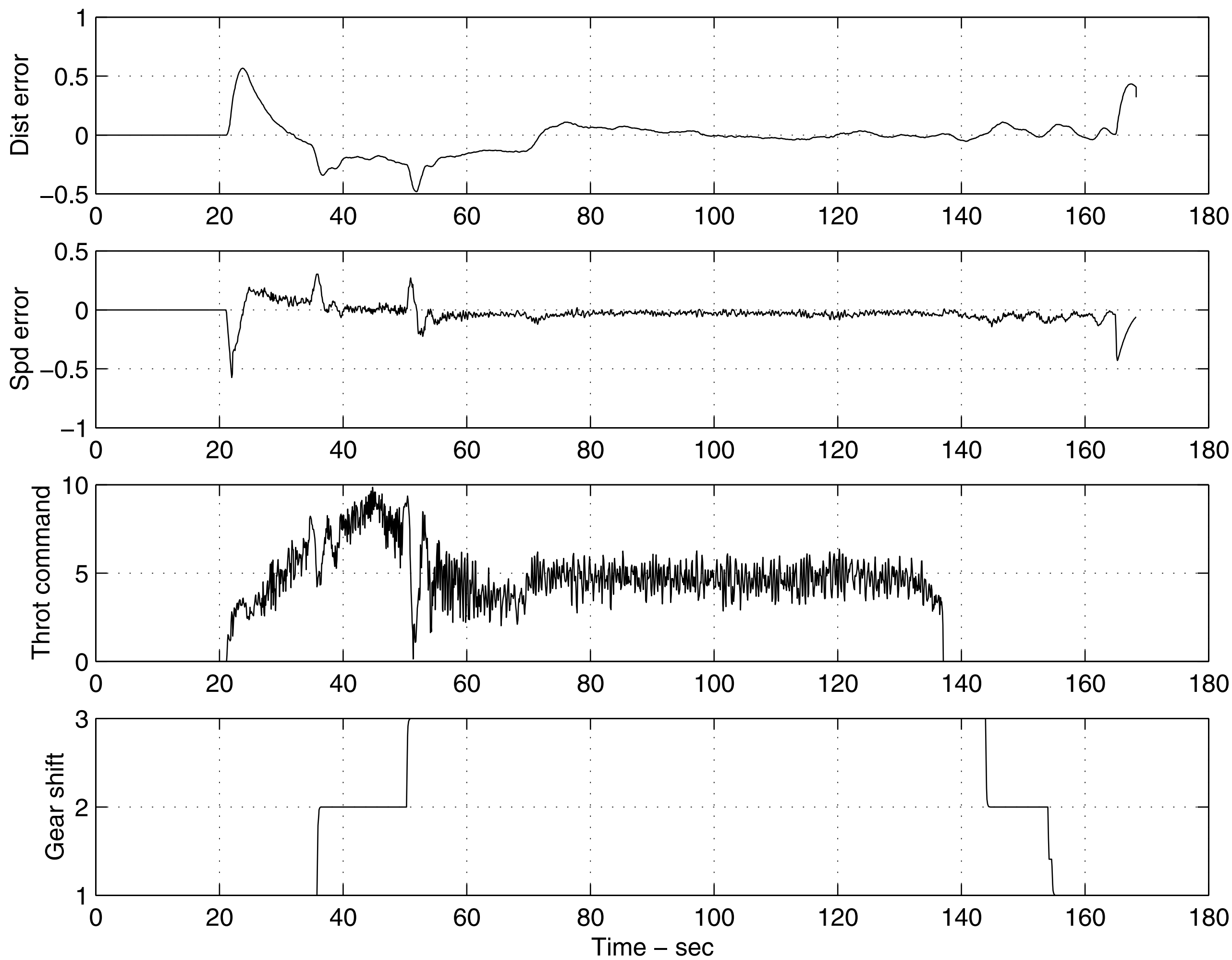


Figure 3.1: Corrected demo code, second car, following distance $4.0m$

Fig. 3.5 - Fig. 3.8 show the test results for the sliding mode controller and the optimal dynamic back-stepping sliding surface controller at low speed, i.e. maximum speed at 13.0 mph and 25.0 mph respectively. One can draw a similar conclusion.

## 3.5 String Stability

String stability is very important for vehicle platooning. It describes how the distance and speed tracking error would behave upstream the platoon. Intuitively, if these errors is reducing upstream, it is said to be string stable; Otherwise, not string stale. Theoretically, string stability is a control design problem and is solved as in [32]. However, different control design methods will result different practical string stability in practice.

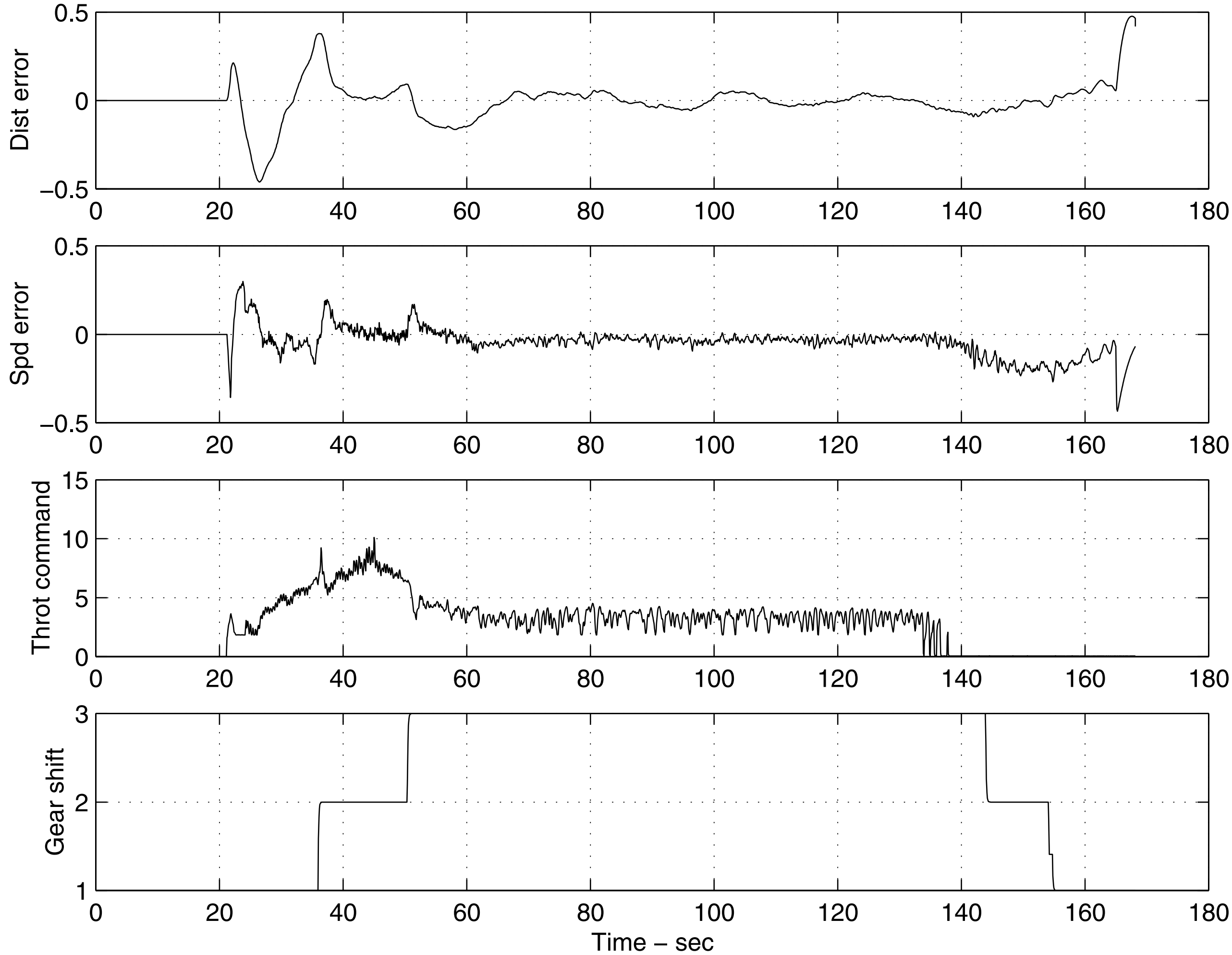


Figure 3.2: Dynamic sliding mode controller, second car, following distance $4.0m$

The string stability of the demo 97 code has been tested for 8 car platooning, which is proved to be good. Although, the dynamic back-stepping multi-surface controller is very good in distance control as required by merging maneuver, its string stability is not very good yet at this stage. For example, distance tracking error for car 2 can be made within $0.05m$. But for car 3, it increased to $0.1m$. Although, this is still smaller than that of the demo code, the increase of the tracking error upstream in the platoon is not a good sign. It could increase further for car 4 (testing needed). This does not mean that back-stepping controller cannot achieve good string stability in platooning. Primary research shows that this may be due to the following factors

(a) Data fusion: In demo code, data used are speed, preceding speed, preceding acceleration, relative distance; In back-stepping controller, preceding acceleration is not used. Instead, an integral filter is used to calculate the derivative of the second sliding surface which contains the

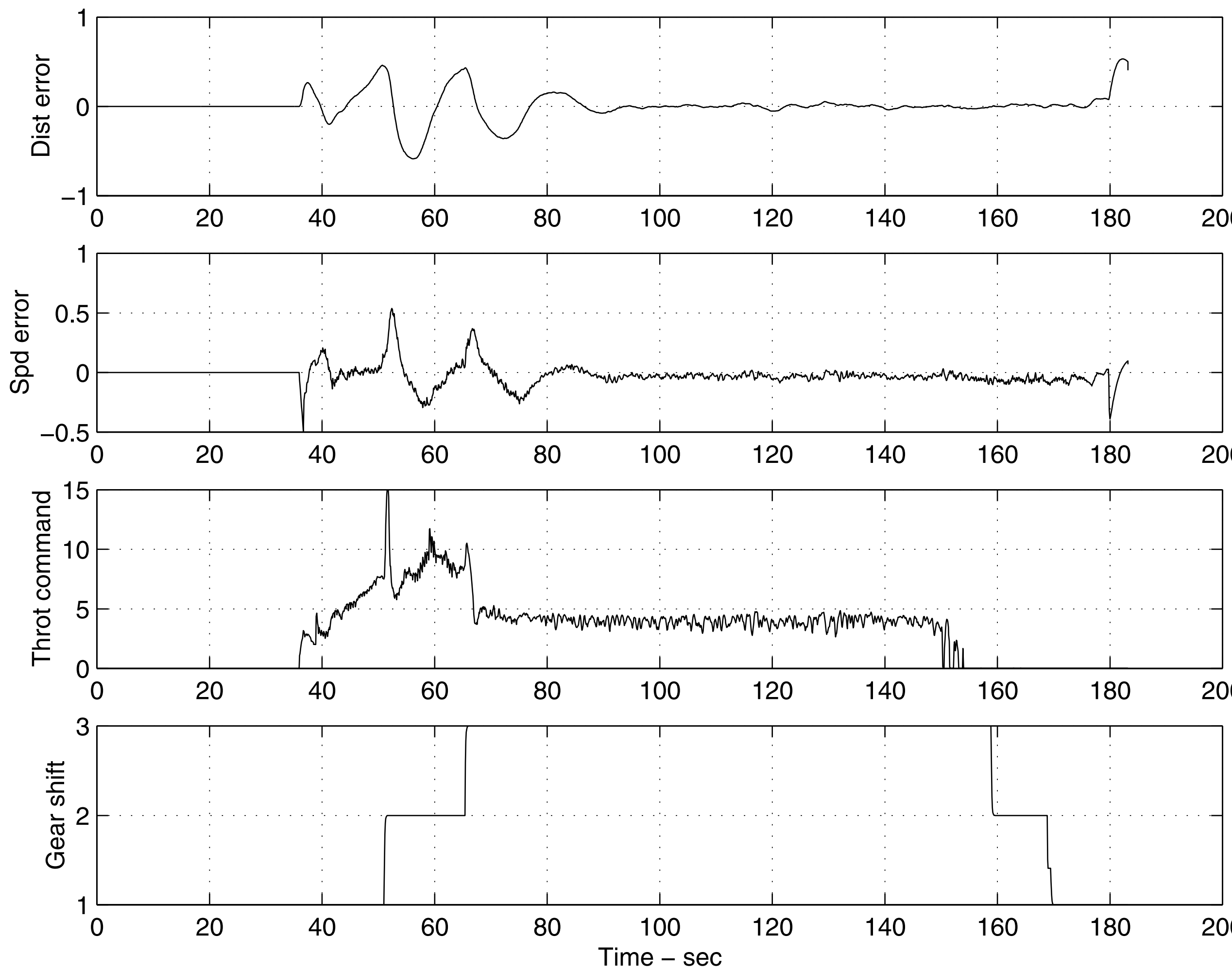


Figure 3.3: Optimal dynamic back-stepping sliding surface controller, second car, following distance $4.0m$

preceding speed.

(b) Distance tracking and settling time are made too stringent in back-stepping controller;

(c) Fine tuning is insufficient, which need to be carried out on test site at higher speed.

Primary test also showed that directly using preceding acceleration will definitely increase string stability in platoon. However, this needs further test. Strings stability improvement is currently carried on along with other work on longitudinal controller development.

## 3.6 Concluding Remarks

Preliminary implementation shows that Optimal Dynamic Back-stepping Sliding Surface Controller seems suitable for longitudinal control of car platooning as both steady state performance

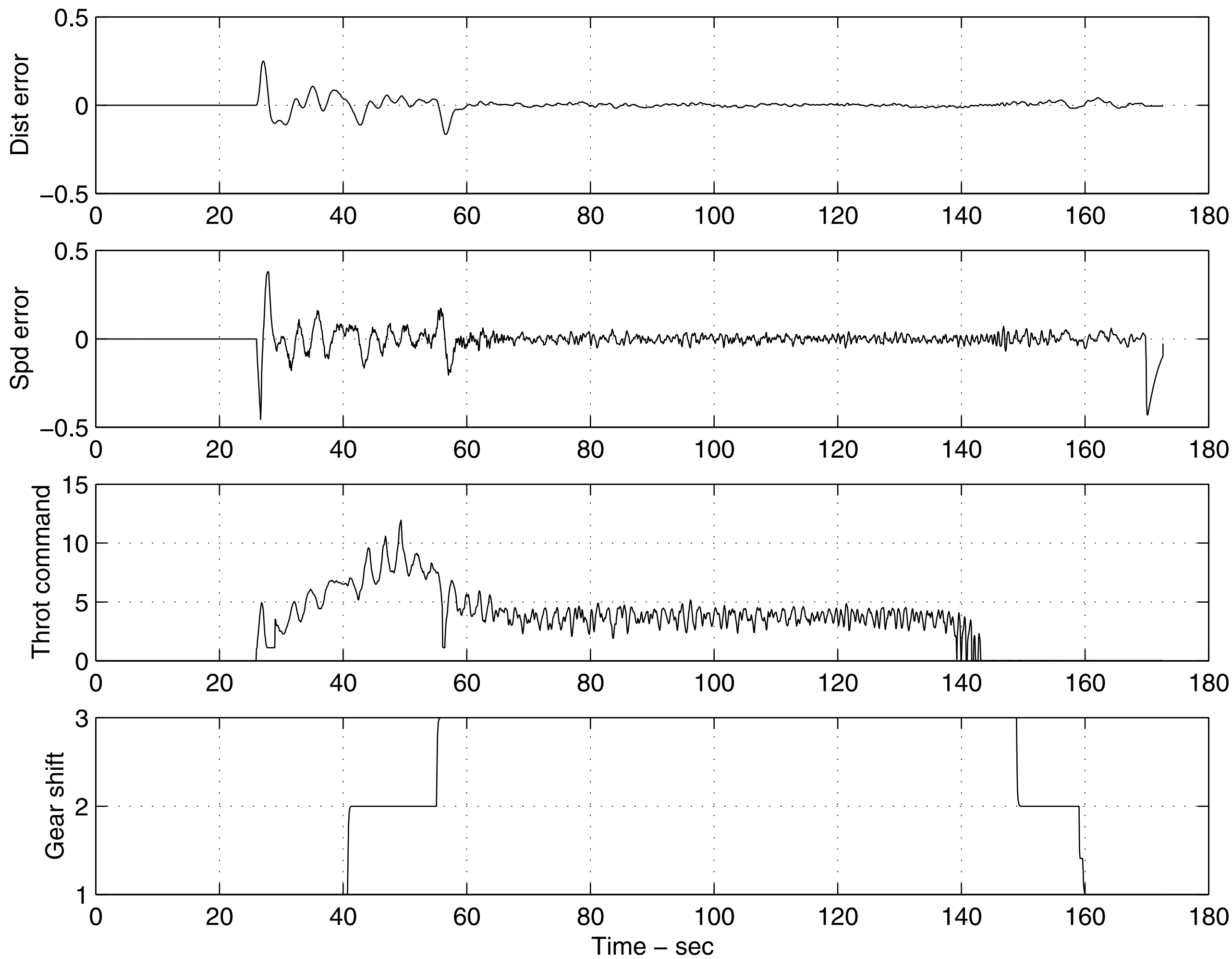


Figure 3.4: Optimal dynamic back-stepping sliding surface controller, one car, virtual distance $4.0m$

and transient performance are better. This may be due to the following reasons.

(1) Back-stepping design logic emphasizes speed control while put distance regulation in a relatively weaker position. This may help to shape the closed-loop dynamics;

(2) Sliding reachability condition effectively expel uncertainties in both distance and speed channels, which result in a robust closed-loop system;

(3) Integral filters effectively reduces measurement noise and incurs much less time delay compared to the moving average plus numerical derivative. Direct analytic derivative of reference state is avoided. This is crucial for robust performance and real-time implementation;

(4) Optimal sliding gain choice in a nonlinear $H_\infty$ approach further increases the robust stability margin and improves the performance.

These new controllers are not matured yet. They need further fine tuning and modification

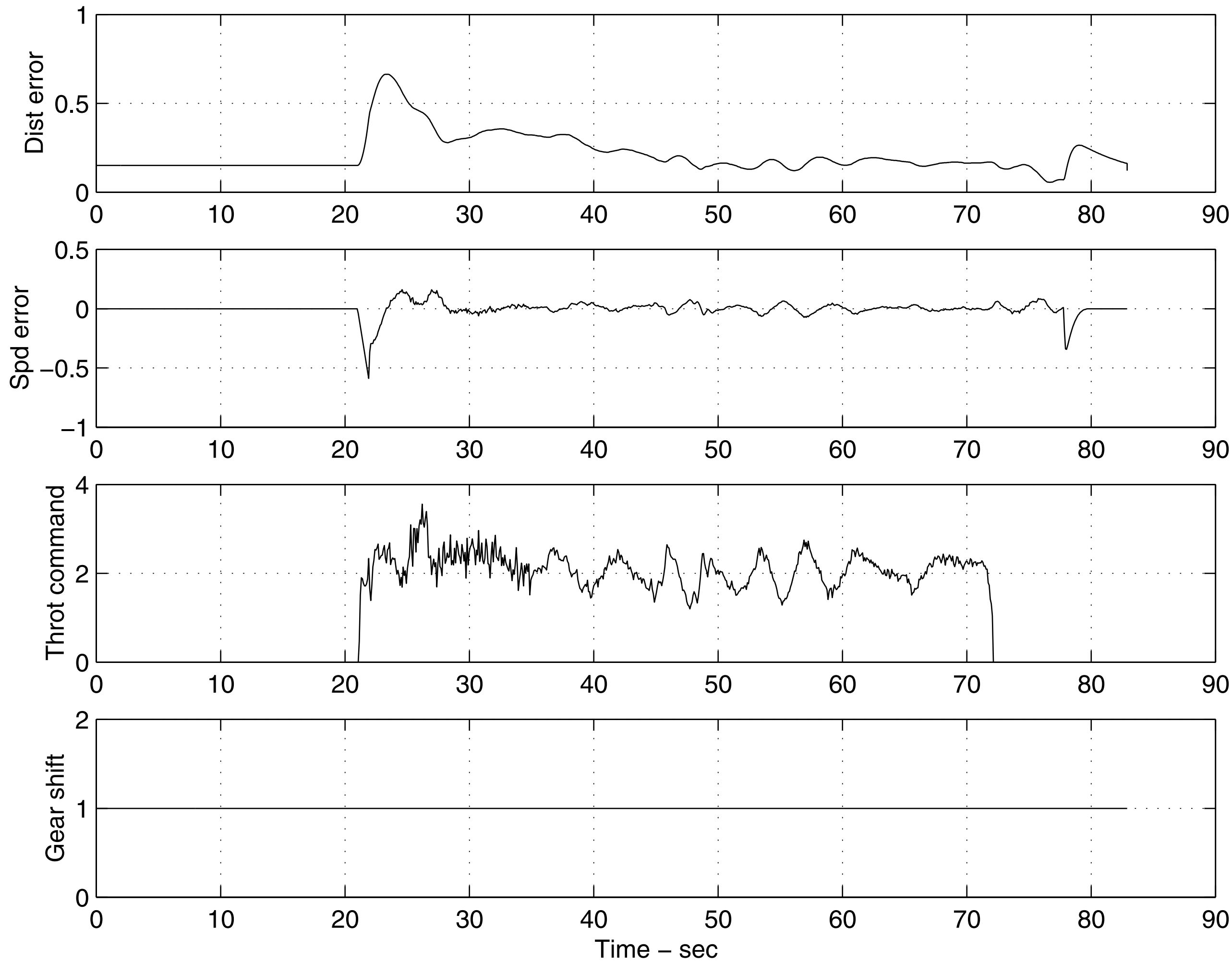


Figure 3.5: Demo code, sliding mode control, Max speed 13mph

to improve performance so that they can be used in the regulation layer for different maneuvers. This will be carried out in future work.

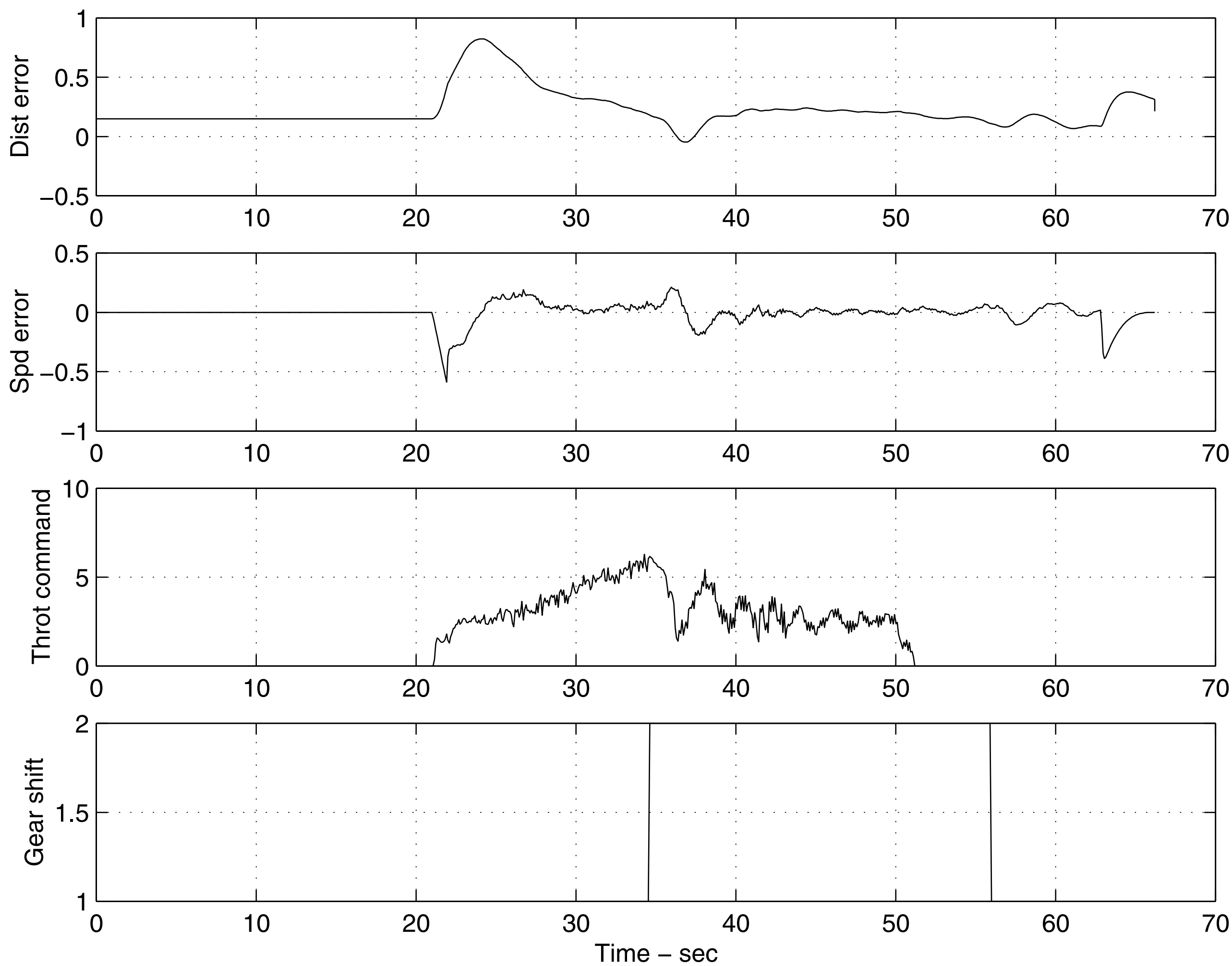


Figure 3.6: Demo code, sliding mode control, Max speed 25mph

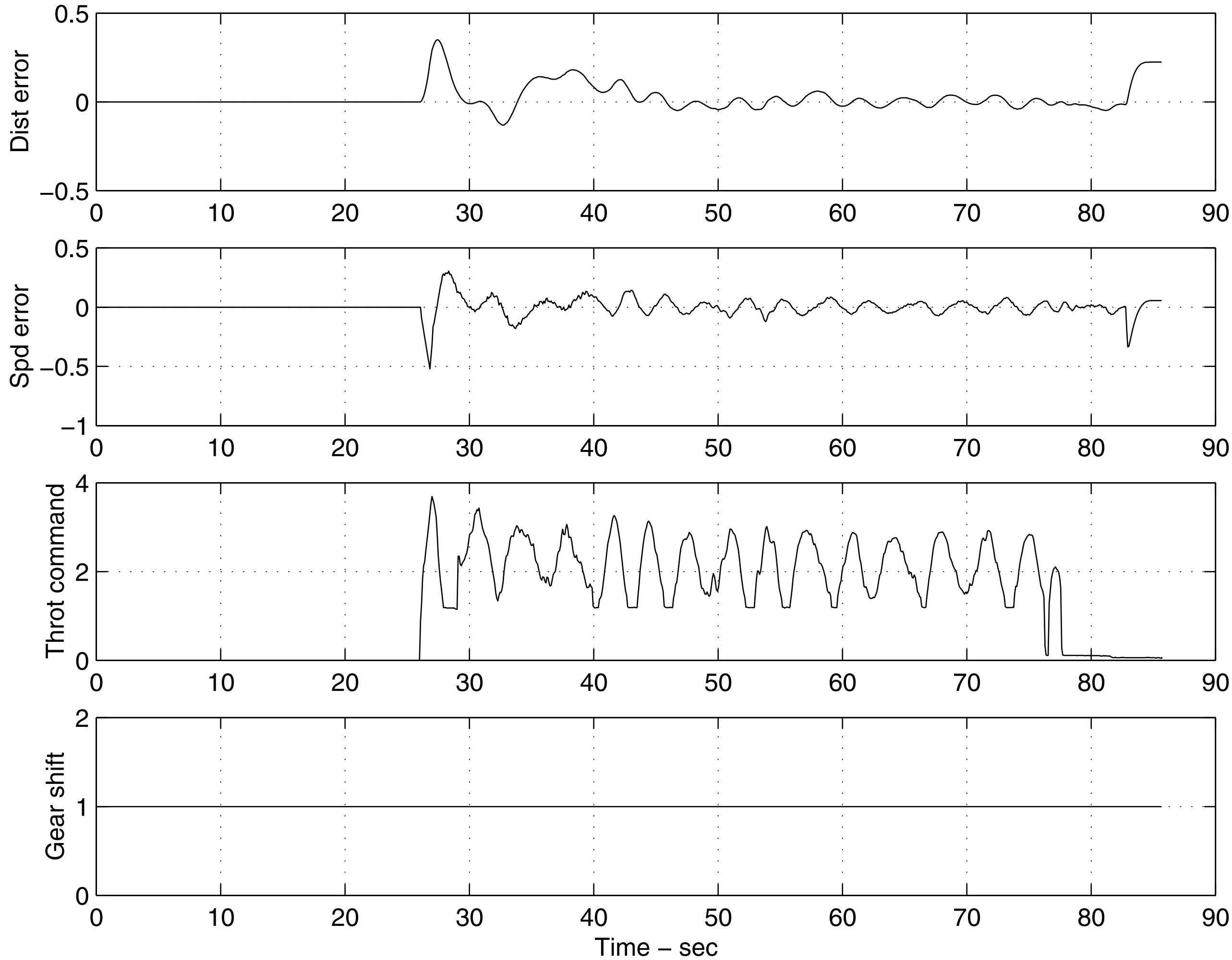


Figure 3.7: Optimal dynamic back-stepping sliding surface control, Max speed 13mph

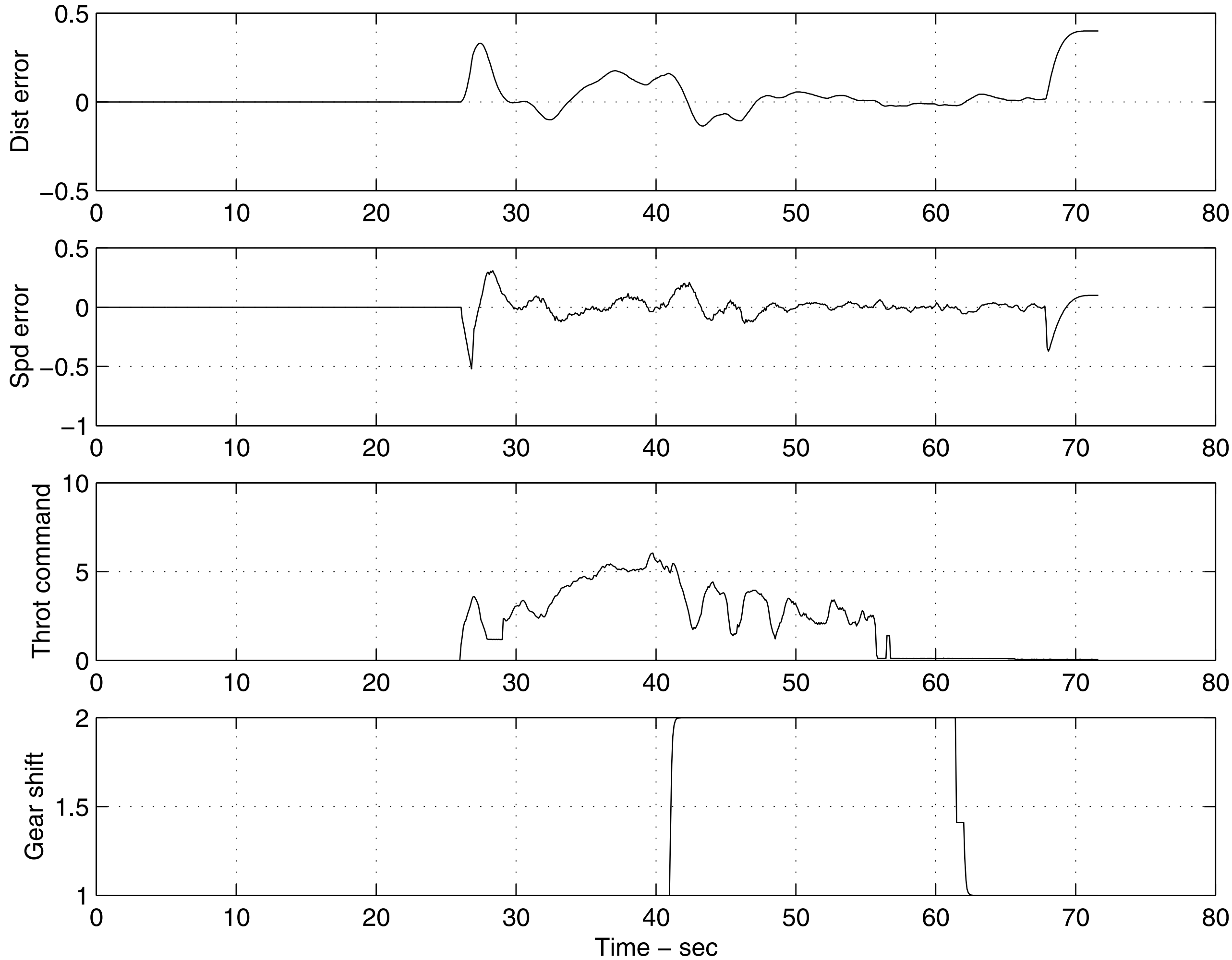


Figure 3.8: Optimal dynamic back-stepping sliding surface control, Max speed 25mph

# Chapter 4

# Prospective of Future Work

Future work for longitudinal control code development will contain, but not restricted to, the following aspects.

## 4.1 Fault Detection and Management

The purpose of fault detection and management is to construct a fault tolerant longitudinal control system. This is crucial for safety and maneuverability. Once a fault appears, it can be detected by a fault detection system and isolated and managed by a fault management system. When part of the whole system is in fault, we still wish to achieve degraded performance without loss of safety. This is a very challenging area. Here some primary considerations are presented.

Fault detection in different levels is necessary. Currently, fault detection is restricted only in the following two parts: actuators and measurement. In the future, we can consider fault in controller.

In the current code, for each component, the fault diagnosis result is either fault or no fault, which can be quantized as

$$fault\_index = 0, 1$$

In the future, we define degree of fault for each component. i.e. fault_index will assume a continuous spectrum in the real interval $[0, 1]$. Then fault management can be conducted with respect to different fault degree for a specified component.

Fault management is suggested to be carried out in different levels. Primary suggestion of the levels are as follows.

**Level 1:** Communication, radar (hardware)

**Level 2**: Control software

**Level 3**: Actuator: throttle, brake, speedometer, magnetometer (hardware)

**Level 4**: Some other hardware such as CAN bus

Those are listed according to the likelihood of fault (reliability), their impact on the safety, platoon control strategy and performance of overall system.

## 4.2 Development of Maneuvers

For vehicle platooning, merging maneuver has been newly added. But some other maneuvers such as free lane changing, emergency stopping and platooning in radar fault mode, etc. are to be developed in future work. The experience for merging maneuver design will definitely help in the design of free lane changing maneuver.

## 4.3 Transition Control

For practical application, it is necessary to develop and implement transition control for longitudinal controller of AHS. The transition control should be able to transfer the operation smoothly from manual to automatic control and vice versa. This will require a robust transition control algorithm, some hardware modification and some new hardware installation.

## 4.4 Construct Controllers for Different Purposes

It is recognized that the practical performances of longitudinal controllers resulting from different design methods for a specified maneuver are usually not the same. This difference may be caused by many factors which mainly include, but are not restricted to, the following:

(1) The number of measured data and data fusion method;

(2) The dynamic properties of the closed-loop system for that design method;

(3) Some trade-offs commonly encountered in control design: There are several measures of effectiveness to describe the performance of a controller, some of which are contradictory, such as tracking error and robust stability margin, settling time, tracking error and sensitivity, tracking error, robust stability margin and control effort, and etc..

For example, platoon keeping controller demands highly on string stability and tight intervehicle distance following. For merging maneuver, distance measurements are based on the ground. Control is not only critical in distance and speed, but also critical [24]. Some controller may have better performance than others at high speed but not as good at low speed. Thus controller construction should consider these differences. It would be very difficult to construct a single control law which can achieve all the purposes. However, as stated before, it is possible to combine several controllers in parallel in a realtime code. These controllers can be activated in variable structure manner according to the phase changes of a controller and the shift between different maneuvers. Such a strategy not only increases the reliability through software redundancy, but also improves overall performances of the system.

# Bibliography

[1] Bartolini, G. and Zolezzi, T., 1991, Generalised variable structure control of deterministically uncertain systems: chattering reduction and non matching uncertainties, *Proc. 1st European Control Conference, Grenoble,* p2597-2600

[2] C. I. Byrnes, A. Isidori and J. C. Willems, 1992, Passivity, feedback equivalence and the global stabilization of minimum phase nonlinear systems, *IEEE Trans. Antomatic Control*, **AC-36**, p1228-1240

[3] A. Cavallo and De Maria, G., 1993, Some control problems solved via a sliding manifold approach, *Differential Equations and Dynamical Systems,* **Vol. 1,** No. 4, p295-310

[4] D. Cho and J. K. Hedrick, 1989, Automotive power train modelling for control, *Transaction of ASME, J. Dynamic Systems, Measurement and Control*, **Vol. 111**, p568-576

[5] S. B. Choi, 1994, Vehicle longitudinal control test, PATH Research Report, UCB-ITS-PRR-94-15, UC Berkeley

[6] S. B. Choi and P. Devlin, 1995, Throttle and brake combined control for intelligent vehicle highway systems,*SAE paper*, (951897)

[7] R. A. DeCarlo, Zak, S. H. , and Matthews, G. P., 1988, Variable structure control of nonlinear multivariable systems: A tutorial , *Proc., IEEE*, **Vol. 76, No.3**, p212-232

[8] R. E. Fenton and J. G. Bender, 1969, A study of automatic car following, *IEEE Trans. on Vehicle Tech.*, **Vol. VT-18, No. 3**, Nov.

[9] B. Fernandez R. and J. K. Hedrick, 1987, Control of multivariable non-linear systems by the sliding mode method, *Int. J. Control*, **Vol. 46, no. 3**, p1019-1040

[10] A. F. Filippov, 1964, Differential equations with discontinuous right hand side, *Amer. Math. Soc. Translations,* **42** ser.2, pp199-231

[11] W. L. Garrard, R. J. Caudill, A. L. Kornhauser,D. MacKinnon and S. J. Brown, 1978, State-of-the-art of longitudinal control of automated guided transit vehicles, High Speed *Ground Transportation Journal*, **Vol. 12, No. 2**

[12] J. C. Gerdes, D. B. Maciuca, P. E. Devlin and J. K. Hedrick, 1993, Brake system modeling for IVHS longitudinal control, *Advances in Robust and Nonlinear Control*, **DSC-Vol. 53**, ASME Winter Annual Meeting, p119-126

[13] J. C. Gerdes and J. K. Hedrick, 1996, Vehicle speed and spacing control via coordinated throttle and brake actuation, *Proc. 13th IFAC Triennial World Congress*, San Francisco, USA, p183-188

[14] S. Gopalswamy and Hedrick, J. K., 1993, Tracking nonlinear non-minimum phase systems using sliding control , **Int. J. Contr**., **Vol. 57, No. 5**, p1141-1158

[15] J. K. Hedrick, Nonlinear controller design for automated vehicle applications, *Proc. UKACC Int. Conf. on Contr.'98, 1-4 Sept. 1998, Swansea, U. K.*, p23-31

[16] Hill D. and H. Moylan, 1976, The stability of nonlinear dissipative systems, IEEE Trans. Automatic Control, AC-21, p708-711

[17] A. Isidori, 1994, $H_\infty$ Control via measurement feedback for affine nonlinear systems, *Int. J. of Rubust and Nonlinear Control*, **Vol. 4**, p553-574

[18] X. Y. Lu and J. K. Hedrick, 1999, Some new integral filters in control, submitted

[19] X. Y. Lu and J. K. Hedrick, 1999, $H_\infty$ Sub-optimization in multiple surface dynamic sliding mode control

[20] X. Y. Lu and S. K. Spurgeon, 1997, Robust sliding mode control of uncertain nonlinear systems, *Systems & Control Letters*, **Vol. 32, no. 2,** p75-90

[21] X. Y. Lu & S. K. Spurgeon, 1998, Asymptotic stabilization of multiple input nonlinear systems via sliding modes, *Dynamics and Control*, **Vol. 8**, p231-254

[22] X. Y. Lu & S. K. Spurgeon, 1999, Robustness of static sliding mode control for nonlinear systems, *Int. J. Control*, **Vol. 72, no. 5**, p1343-1353

[23] X. Y. Lu, H.-S. Tan, S. E. Shladover and J. K. Hedrick, 2000, Implementation and comparison of Nonlinear longitudinal controllers for car platooning, to be presented at AVEC 2000, Ann Arbor, Sept.

[24] X. Y. Lu, H.-S. Tan, S. E. Shladover and J. K. Hedrick, 2000, Implementation of longitudinal control algorithm for vehicle merging, to be presented at AVEC 2000, Ann Arbor, Sept.

[25] D. B. Maciuca and J. K. Hedrick, 1995, Advanced nonlinear brake system control for vehicle platooning, *Proc. European Control Conference*, Sept. 5-8, Italy, p2402-2407

[26] S. E. Shladover, 1979, Vehicle follower control for dynamic entrainment of automated guideway transit vehicle, *Trans. ASME*, **Vol. 101**, Dec., p314-320

[27] S. E. Shladover, 1980, Longitudinal control of automotive vehicles in close-vehicle formation platoons, *Transportation Research*, **Vol. 14A, No. 2,** April, p63-81

[28] S. E. Shladover, 1995, Review of the state of development of advanced vehicle control systems, Vehicle Systems Dynamics, Vol. 24, p551-595

[29] Sira-Ramirez, H., 1993, On the dynamical sliding mode control of nonlinear systems , *Int. J. Control*, **Vol. 57, No.5**, p1039-1061

[30] Slotine, J-J. E., Coetsee, J. A., 1986, Adaptive sliding controller synthesis for non-linear systems, *Int. J. Control*, **Vol. 43, No.6**, p1631-1651

[31] J.-J. E. Slotine, J. K. Hedrick and E. A. Misawa, 1987, On sliding observers for nonlinear systems, *Trans. ASME, J. of Dyn. Syst., Measurement and Control*, **Vol. 109**, p245-252

[32] D. Swaroop and J. K. Hedrick, String stability of interconnected systems, *IEEE Trans. Auto. Contr.*, **40, no.3**, p349-357

[33] D. Swaroop, J. C. Gerdes, P. P. Yip and J. K. Hedrick, 1997, Dynamic surface control of nonlinear systems, *Proc. of ACC-97*, p3028-3034

[34] Utkin, V. I., 1992, *Sliding Modes in Control and Optimisation*, Springer-Verlag, Berlin

[35] M. Won and J. K. Hedrick, 1996, Multiple-surface sliding control of a class of uncertain nonlinear systems, *Int. J. Control*, **Vol. 64, no. 4**, p693-706

[36] P. P. Yip and J. K. Hedrick, 1998, Adaptive dynamic surface control: a simplified algorithm for adaptive back-stepping control of nonlinear systems, *Int. J. Control*, **Vol. 71, no. 5**, p959-979