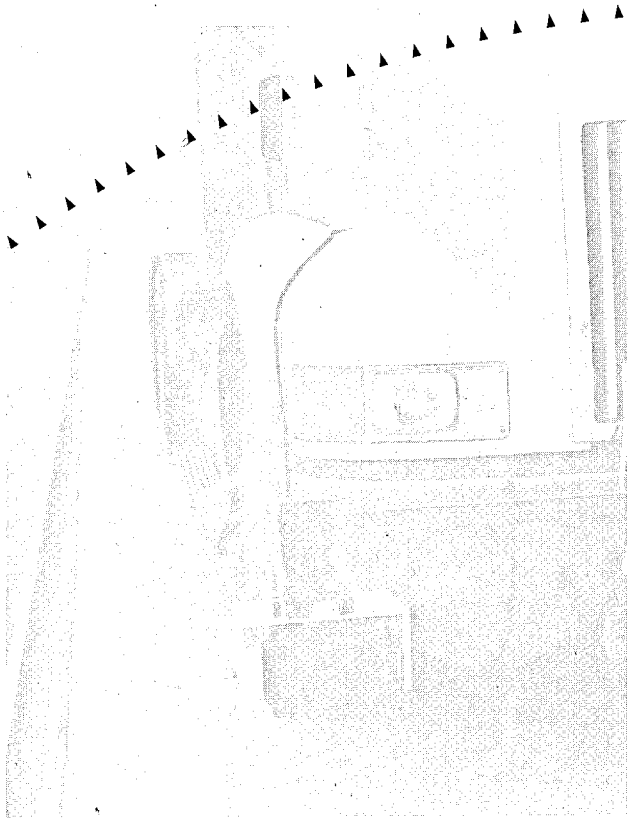
Final Report 1998-30



The Virtual Bumper: A Control Based Collision Avoidance System For Highway Vehicles

Minnesota Guidestar

Technical Report Documentation Page

1. Report No.	2.	3. Recipient's Accession No.
MN/RC - 1998-30		

4. Title and Subtitle	5. Report Date
The Virtual Bumper: A Control Based Collision Avoidance System For Highway Vehicles	October 1997
	6.

7. Author(s)	8. Performing Organization Report No.
William Schiller Max Donath	

9. Performing Organization Name and Address	10. Project/Task/Work Unit No.
University of Minnesota Department of Mechanical Engineering 111 Church Street, S. E. Minneapolis, MN 55455	11. Contract (C) or Grant (G) No. (C) 74708 TOC # 2

12. Sponsoring Organization Name and Address	13. Type of Report and Period Covered
Minnesota Department of Transportation 395 John Ireland Boulevard Mail Stop 330 St. Paul, Minnesota 55155	Final Report 1997
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract (Limit: 200 words)

This report summarizes research on a new collision avoidance strategy, the 'virtual bumper.' The research involves development and simulation testing of the virtual bumper, a two-dimensional control strategy that provides steering, throttle, and braking actuation to maneuver a vehicle in a dynamic environment with the goal of avoiding obstacles and other vehicles. The concept applies to both normal and emergency driving conditions. Under all circumstances, the virtual bumper incorporates vehicle dynamic limits to ensure that the control commands are within safe levels. The virtual bumper will attempt to avoid a collision and will, at least, minimize the magnitude of an unavoidable collision.

To test the functionality of the virtual bumper, researchers evaluated several driving scenarios. The scenarios consider both normal driving situations and emergency driving conditions. The normal driving scenarios demonstrated that the control algorithm operates the vehicle similar to the way a human would. This is important because a comfortable and predictable (i.e., intuitive) system response is required for achieving driver acceptance. The emergency scenarios demonstrated that the strategy is capable of reacting appropriately while maintaining safe acceleration/deceleration levels for the vehicle. This evaluation showed that the virtual bumper can provide safe vehicle control for a broad range of driving situations.

17. Document Analysis/Descriptors		18. Availability Statement
collision avoidance collision mitigation virtual bumper	integrated steering throttle and braking radar	No restrictions. Document available from: National Technical Information Services, Springfield, Virginia 22161

19. Security Class (this report)	20. Security Class (this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	197	

THE VIRTUAL BUMPER
A Control Based Collision Avoidance System for Highway Vehicles

Final Report

Prepared by:

Bill Schiller and Max Donath

Department of Mechanical Engineering
The University of Minnesota
111 Church Street SE
Minneapolis, MN 55455

October 1997

Published by

Minnesota Department of Transportation
Office of Research Services
First Floor, MS 330
395 John Ireland Boulevard
St. Paul, MN 55155

Table of Contents

List of Figures

List of Tables

Acknowledgements

The authors would like to acknowledge the many present and former members of the Robotics and Vehicle Guidance Laboratories at the University of Minnesota who provided assistance with this project. These include: Ted Morris who provided useful and time saving advice on software and 3D modeling, and Lee Alexander, Roz Dolid, Sundeep Bajikar, Micah Garlich-Miller, Heon-Min Lim, Saroj Saimek and Alec Gorjestani, who were all generous in sharing their time and knowledge.

Three individuals have had a significant impact on this work. Crag Shankwitz' advice helped focus the work on the issues which are most important. Vassilios Morellas insisted that our problem be viewed from a variety of directions thereby contributing to the final outcome. Jon Minner's insight into the key design issues had a significant effect on the results and thereby reduced the time needed to reach the project's objectives. Without the efforts and contributions of these three individuals, this work would not be complete.

Executive Summary

We have developed a new collision avoidance strategy, the 'virtual bumper' and tested it in simulation for vehicles operating on highways. The virtual bumper is a 2-dimensional control strategy that provides both steering and throttle/braking actuation to maneuver a vehicle in a dynamic environment in order to avoid obstacles and other vehicles. This algorithm has its roots in impedance control and is integrated with a heuristic-based system. We developed this concept to be capable of responding to both normal and emergency driving conditions. Under all circumstances, the vehicle dynamic limits are incorporated in order to ensure that the control commands are within safe levels. The virtual bumper will *attempt* to avoid a collision and will at least, minimize the magnitude of an unavoidable collision (collision mitigation). This approach like all others cannot guarantee that a collision will be avoided.

The virtual bumper is implemented using two separate impedance control loops. One loop is the longitudinal impedance control loop and controls the velocity of the host. This loop acts to maintain the desired headway to target vehicles ahead, as well as to provide emergency braking. The other loop is the lateral impedance control loop with a set of heuristics integrated into its feedback loop. This loop is responsible for maintaining lateral spacing between the host and adjacent target vehicles as well as performing lane change maneuvers. The heuristics provide a mechanism for deciding which is the desired lane for travel and controls the lane change maneuver for both normal and emergency driving conditions. When these loops execute in parallel, the host is provided with control that steers and accelerates/brakes the vehicle to avoid collisions (or at least mitigate the effect of a collision)

Most of the work done to date on control strategies for highway vehicles has been developed in a traditional simulation environment. In order to expedite development, the virtual bumper is developed in a real-time simulation environment. In this simulation environment, the control software executes on a control computer running a real-time operating system, and controls the modeled dynamic behavior of the vehicle under test. Models, which characterize the actual behavior of each sensor, interact with objects represented by solid models in the graphically simulated environment. This type of simulation allows us to evaluate much more than the virtual bumper algorithm. For example, with this approach we are able to evaluate sensor latency issues, controller bandwidth requirements, process timing issues, etc. In addition, when the simulation testing is completed, this software is debugged and ready for implementation on the real vehicle. This form of simulation allows us to move rapidly towards real world implementation.

For this first investigation of the virtual bumper, objects in the environment are sensed using an array of radar range sensors. We introduce a methodology for evaluating the sensor placement on a vehicle driving in a highway environment. This approach involves a graphical tool which characterizes the ability of a given sensor configuration to meet our sensing requirements. We then use an iteration process to determine the final sensor configuration -- an array of fourteen radar units.

To test the functionality of the virtual bumper, several driving scenarios are evaluated. The scenarios consider both normal driving situations as well as emergency driving conditions. The normal driving scenarios demonstrated that the control algorithm operates the vehicle similar to the way a human would. This is important because a comfortable and a predictable (i.e. intuitive) system response is required for achieving driver acceptance. The emergency scenarios demonstrated that the strategy is capable of reacting appropriately while maintaining safe acceleration/deceleration levels for the vehicle. This evaluation has shown that the virtual bumper can provide safe vehicle control for a broad range of driving situations.

Chapter 1

Introduction

1.1 Motivation

Over the past thirty years a great deal of research has addressed the safety of driving. This research has resulted in consumer oriented products, such as anti-lock brakes and traction control, which increase safety by increasing the performance of a vehicle. Other technologies developed through this research have dealt with reducing the risk of injury for vehicle occupants in the event of an accident (i.e. driver air bags, impact absorbing chassis designs, etc).

Although there has also been a great deal of ground braking work done on preventing accidents, most of these concepts have not yet become available in the marketplace. The work described here falls under the area of Advanced Vehicle Control Systems (AVCS), a component of what is today called Intelligent Transportation Systems (ITS). The AVCS includes systems which apply computing and sensing technology to assist the driver in the driving task. Some applications simply warn the driver of potential hazards. Other systems provide the driver with assistance in controlling the vehicle and may also be capable of full automatic control of vehicle motion. With recent advancements in AVCS, as well as the significant improvement in computer systems, this technology is approaching a point at which consumer products will be widely available. The work described here focuses on increasing safety through the development of a driver assistive device which falls under the umbrella of AVCS.

Much work has already been done in the area of driver assistive devices to control inter-vehicle spacing in the direction of travel. Such systems are commonly referred to as headway control or adaptive cruise control (ACC) systems. However, in some emergency driving situations, it has been shown that steering combined with braking is required to avoid collisions [Adams94]. Furthermore, vehicles with large inertias, such as trucks, typically have longer stopping distances than passenger cars [Reed87], which makes steering even more critical in effective collision avoidance maneuvers. For these reasons, we have developed a driver assistive approach, that we call the virtual bumper, which helps to steer and accelerate/brake the vehicle in both normal and emergency driving situations in the event that the driver can not react or sense quickly enough.

It is our contention that a system which can steer and adjust the velocity of the vehicle will effectively reduce the risk of collision. For instance, according to [Najm95], there were 6,093,000 traffic related accidents in the United States in 1993 alone. By evaluating these crash statistics considering the crash types identified in [Najm96], we have identified that a system such as the one described in this document would have either prevented or reduced the damage of approximately fifty

percent of these accidents[1]. Clearly there is a great loss both socially and economically associated with these accidents. Any systems which help reduce traffic accidents will have a positive impact on our society as a whole.

1.2 The Virtual Bumper

In this document, we develop the virtual bumper, a collision avoidance strategy, first introduced by Hennessey et al [Hennessey95]. The virtual bumper is a vehicle control strategy which combines longitudinal and lateral collision avoidance capabilities to drive a vehicle in normal and emergency situations. Our rationale is that braking or throttle and/or transmission shifting is insufficient in many cases. This is particularly true for vehicles with large inertias. This approach makes no assumptions about the mass or dynamic capabilities of the target vehicles in the host vehicle's environment when determining trajectories. This approach also considers dynamic control limits for the host vehicle. It does not guarantee that a collision will be avoided in all driving situations. Instead our approach is to control the vehicle in such a manner that in the event of an unavoidable collision, the damage is minimized (crash mitigation).

The virtual bumper strategy combines impedance control with a heuristic based system. An impedance control approach alone for collision avoidance falls into the potential fields approaches described in the next section and also shares the shortcomings of this category. However, adding the heuristic based system into the impedance control strategy allows the system to make decisions and therefore drive the vehicle more as a human would. The impedance control also compliments the expert system by providing a more general solution which makes the system capable of handling situations that may not have been considered explicitly during the design process.

The main concept behind the virtual bumper is the notion of a "personal space" around the vehicle. When an obstacle or other vehicle intrudes into this space, a "virtual" spring and/or damper is compressed which results in a force or torque which is applied through the steering, throttle and braking subsystems to effect a change in the host vehicle's motion. Since there is no physical contact and since these effects are computed based on sensed incursions, one can in effect continuously adapt the shape of the virtual bumper and the nature of the resulting forces on the vehicle.

The virtual bumper needs to know the location and trajectory of the target vehicles in the host's environment. This information can be determined using a variety of sensing strategies. In the implementation described here. this information is determined using radar sensors. The

[1] The percentage of accidents affected by our approach could actually be higher but our work to date focuses on limited access highway driving; the intersection scenario has not yet been addressed but is another application for future consideration.

approach here is to mount an array of radar sensors around the perimeter of the host vehicle. Then as the vehicle travels down the roadway, a dynamically changing map of the target vehicles in its environment is determined. We will later discuss the placement and signal processing for the radar sensors. The approach described here is independent of the nature of the sensor used. In fact, a GPS receiver and yaw rate gyro and a wireless local area network which allows vehicles within a reasonable operational range to communicate their respective positions and heading angles may essentially serve the same purpose. The only problem with not using a radar like device is that no information would be provided to the host vehicle about obstacles without active devices transmitting their position (e.g. abandoned vehicles on the shoulder, vehicles with no operative units, animals (or humans) on the road, etc.). Using a GPS based approach would significantly reduce the number of radar units needed. Furthermore, such receiver/transmitter capability would allow all vehicles to be aware of the dynamic behavior of every surrounding vehicle that transmits information. Roadway information (i.e. lane boundaries or center lines) is also needed for the virtual bumper but it is not critical how this information is gathered. For this work, the roadway information can be assumed to be determined using DGPS and a map based system as outlined in [Morellas97] [Bajikar97].

For a variety of reasons, including safety, cost, the unavailability of large numbers of radar units, the desire to minimize expensive and time consuming experiments, this first implementation of the virtual bumper was completed using simulation. The simulation environment used here is unique in that it is a real-time simulation using embedded processors similar to the final implementation platform for the vehicle (see chapter 5). Unlike some simulations, the environment is sensed using a realistic high fidelity sensor model. The sensor that we modeled is based on a commercially available radar sensor capable of measuring range and range rate to sensed targets. Furthermore, this graphically generated environment allows us to create various driving scenarios involving multiple vehicles for full debugging and testing of the control strategy. At this point, the software that was developed using experimentally validated lateral dynamic models of our host truck (Navistar 9400) has been tested, debugged and is ready for implementation and evaluation on the real vehicle.

Our ultimate goal is for the virtual bumper to operate as a driver assistive device in a highway environment. The virtual bumper will operate in the background, constantly determining the correct, safe driving commands while the driver is in full control of the vehicle. If and when the driver is reacting inappropriately (e.g. drowsy driver, inadequate response to encroaching vehicles), the system will take over vehicle control and perform a safe maneuver (e.g. pull vehicle onto the shoulder). There are many human factors issues associated with this implementation. In this document, however, we will concentrate only on the vehicle control and sensor portion of the system.

We will not address any of the human interfacing issues here. Although the truck's hardware has been designed to interact with the driver directly, the system design developed here will assume that the driver is not reacting adequately and will control the vehicle autonomously as if the driver was not in the loop. Once it has been verified that the vehicle can indeed be autonomously and safely controlled, the next phase will be to integrate a seamless driver interface. We have already begun the development of such a design. This control strategy can be extended to vehicle operation in both urban roads and rural highway driving environments. But for purposes of initial development, the work here is limited to a two-lane rural interstate highway driving environment.

1.3 Related Work

The organization of our review of related research is categorized by topic. In our work, the focus is on the longitudinal and lateral control of a vehicle in order to avoid collision. Therefore, our literature review is organized based on three possible types of control which may be implemented:

Longitudinal Control
Lateral Guidance
Combined Longitudinal Control/Lateral Guidance

Our approach falls into the category of combined longitudinal control/lateral guidance. For this reason the work in this area has the most relevance to our approach and these references will be reviewed more closely. The work in the longitudinal control area and lateral guidance area are also relevant and are included for a more complete review of the state of technology.

Research has been performed in the AVCS area for more than 30 years and there is a great deal of literature published on this topic. Due to the large number of the publications in this field, it is impractical to do a complete literature search. Instead, we focus on the work performed in the last five years. The emphasis will be on those concepts which represent a significant contribution to the field. For a more complete review of AVCS over the past thirty years, the reader is referred to [Shladover95].

In the following sections and throughout the rest of this document we use the following terminology for consistency with other authors and for clarity. The vehicle which is equipped with computers and sensing technology used for vehicle control or for issuing warnings to the driver is called the "host vehicle". The vehicles around the host vehicle which are obstacles in its environment are referred to as "target vehicles".

1.3.1 Longitudinal Control

Longitudinal control systems are designed to relieve the driver of the task of maintaining desired spacing to the preceding vehicle. Two primary benefits of such a system are increased safety and more efficient traffic flow in today's already crowded urban freeways. These systems typically contain some type of range sensor mounted on the front of the host vehicle which measures the range to the preceding target vehicle. The sensor measurements are then used to adjust the host velocity to maintain the desired vehicle spacing. Vehicle spacing based on a desired headway time (spacing = (headway time)x(host velocity)) is the the most common approach.

The ACC system developed by Siemans Automotive [Howss94] uses two range sensors mounted on the front of a BMW. One sensor is a narrow beam infrared sensor and the other is a three beam radar sensor with a wider view angle. The narrow beam sensor provides information on targets directly in front of the host while the wider beam sensor provides sensing for curved roads as well as neighboring lane and pedestrian sensing. This system uses a fuzzy logic control strategy with inputs of range to target, closing rate and host velocity. This approach controls the velocity by adjusting the throttle indirectly through the cruise control interface and is also capable of electronicly shifting gears.

ACC systems have also used vehicle/roadway communication networks. In an approach developed at Volvo [Palmquist93], electronic signs communicate posted speeds and other traffic law information (i.e. stop signs). Vehicles then travel at the posted speeds with drivers also receiving recommended routing information to allow for smoother traffic flow. Each vehicle driving on their test track maintains constant headway to preceding vehicles. The headway measured using a Leica optical range sensor. This system controls the velocity through throttle and brake actuation.

Considerable work has also been done on ACC systems in Europe in conjunction with the PROMETHEUS program (PROgraM for European Traffic with Highest Efficiency and Unprecedented Safety). Sala and Pressi [Sal94] reviewed the different strategies considered for ACC systems. The headway approaches were divided into autonomous and cooperative ACC. Autonomous systems contain all of the necessary sensing on board the vehicle while cooperative systems have some sensing capabilities on board but also communicate with other vehicles in traffic. The algorithms for autonomous systems are based on constant headway or a safe following distance. Algorithms for cooperative systems also consider target vehicles' dynamics and determine safe following distances based on braking capabilities of lead and following vehicle. Cooperative systems offer the distinct advantage of tighter vehicle spacing which leads to increased traffic densities and flow. However, it is not expected that all vehicles will have such inter-vehicle communication abilities in the near future. Therefore, the autonomous system approach provides a more realistic solution

in the near-term since this approach works regardless of the capabilities of other vehicles on the highway. Furthermore, autonomous systems are also more suited for operating in environments with pedestrians and bicyclists.

Researchers at Nissan have developed a unique approach for their headway control system [Satoh94]. Rather than develop a control strategy to based on a desired parameter (i.e. constant headway), they control the vehicle using a driver model. In this approach, a vehicle is equipped with a laser range sensor for measuring range and closing rate to a lead vehicle. Then seven "expert drivers" drive the vehicle in different driving scenarios and the throttle position is recorded along with the range and closing rate measurements. This information is then used to fit equations which model how these drivers control the vehicle. The driver model is then used to control the throttle position based on the sensor measurements. Experiments were performed and test drivers reported favorably on system response to various situation.

In the US, significant research has been performed on longitudinal control systems. The two primary contributors are the researchers at the California Partners for Advanced Transit and Highway (PATH) at Berkeley and the researchers at the University of Michigan Transportation Research Institute (UMTRI).

The work at PATH on longitudinal control systems has been focused on the area of platooning [Choi95]. Platooning is a strategy where the vehicle spacing is controlled for several vehicles in series to allow for greater traffic density. At PATH, they use a sliding mode controller to maintain a constant vehicle spacing across a range of velocities. In their implementation, four Lincoln Towncars are used with radar sensors mounted on the front grill. They have achieved stable four car platoons with a 4 meter vehicle spacing at velocities of up to 30 m/s (67 mph).

Among the most significant contributions to ACC research to date has been done by Fancher et al [Fancher93][Fancher94a][Fancher94c] at UMTRI. The approach here maintains a constant headway time through actuation of the throttle. The control strategy is based on the relationship of the measured range vs. the closing rate (range rate). Using a graph of this relationship, they define zones which are assigned different modes of operation for the headway controller. In their approach, the measured headway approaches the desired headway, following a smooth first order response with a time constant defined by the control law. This algorithm has been implemented on a Saab 9000 Turbo using a Leica optical sensor for range and range rate measurements. Operational tests have been performed and driver response to the system under a variety of circumstances has been documented [Fancher94c].

Since the work that we describe here is developed for a Navistar semi-tractor it is worth also noting systems that have been developed for larger vehicles. Two ACC systems have

been implemented on commercial vehicles. One system by Schwerberger [Schwerberger94], was developed for a large bus. It fuses measurements from infrared radar and a CCD camera to determine target positions. Eaton Vorad developed an ACC system for a Kenworth T600 model truck [Wissing95][Chakraborty95]. The targets in front of the truck are sensed using a VORAD T200 warning system. The velocity is controlled through actuation of the throttle as well as down-shifting and using the retarder. Eaton's control strategy is based on a desired headway time.

1.3.2 Lateral Guidance

Lateral guidance systems consist of a road sensing system and a lateral controller. The road sensing system determines the location and orientation of the road with respect to the host vehicle. A lateral controller typically uses the road sensing system in its feedback loop to maintain the host vehicle at a desired lateral position on the road. The most common approach is to set the commanded lateral position to be the center of the lane (road following). Some lateral guidance systems perform lane changes or collision avoidance by adjusting the commanded position away from the center of the lane.

The focus in this review is not on the design of lateral controllers. Rather, we focus on what sensing technology is being used for road sensing systems and how well they perform (typically measured by the maximum velocity of the vehicle under control of the lateral guidance system). For this review, roadway sensing systems are grouped into three categories: 1) Machine Vision Systems, 2) Magnetic Referencing Systems and 3) DGPS and map based systems.

Machine Vision Systems

Most of the research in road sensing systems has been done in the area of machine vision. Using vision systems for lane following was pioneered by Dickmanns et al [Dickmanns86]. Their approach uses basic vision processing algorithms to extract features from roadway images. Then, based on models of the road and vehicle, lane boundaries are determined. The sensed road is then used for performing lane following. Their vision system has been extended to include obstacle detection and tracking; their system has successfully performed lane following for a 5-ton van at speeds up to 100 kph [Dickmanns90].

Pomerleau at Carnegie Mellon University was the first to develop an Artificial Neural Network (ANN) based approach for driving a vehicle based on vision information [Pomerleau91]. In this approach, the ANN is first trained while a person drives and then is used for autonomously driving the vehicle down that road. The inputs to the ANN are the pixels values of a low resolution image (30x32 grid) of the road and the output is the desired steering angle. With this approach,

the system can only drive on road types for which the network has been previously trained (i.e. dirt road, two-lane). This approach has also been extended to be capable of performing lane changes [Jochem95]. Due to limitations in the ANN approach, Pomerleau has developed a system called RALPH (Rapidly Adapting Lateral Position Handler) that is a more conventional approach which looks for features in the roadway images that are parallel to the direction of travel [Pomerleau95]. These parallel features may be lane markers, the shoulder or even the oil slick in the lane. The algorithm determines what type of feature it is and calculates the center of the lane. The road position is then used to perform lane following and was recently tested on a 2500 mile trip across the US and performed successful lane following 98 percent of the time.

Vision based systems have also been developed which provide warnings to the driver in the event of roadway departure. One such system was developed by Chapius et al [Chapius95]. In this application, the road position and the vehicle trajectory are monitored. Using this information, the time until a lane boundary is crossed is determined. The time is then used to determine if the driver should be warned of possible roadway departure.

Recently vision based lane following systems have been developed which keep the driver in control of the vehicle. Two such systems were developed by Renault [Lenarchand94] and Daimler-Benz [Franke94]. These systems operate in the background while the driver operates the vehicle. If the driver deviates significantly from the lane center, a small motor on the steering wheel shaft applies a corrective torque to steer the vehicle towards the center of the lane. The system developed by Renault also vibrates the steering wheel when the vehicle leaves the lane if the driver has not used the turn signal.

All of the vision systems developed to date suffer from the same limitations. Systems have been developed which work very well on black roads with white lines. However, the performance of these systems degrade under poor weather conditions where visibilility is limited.

Magnetic Referencing Systems

The researchers at the PATH program at the University of California, Berkeley have worked on a magnetic system for determining road location. In their approach, permanent magnets are embedded along the center of the road. A magnetometer mounted on the vehicle is used to sense the vehicle offset from the center of the road [Peng94]. Much of their work has focused on developing more advanced controllers and they have succeeded in implementing controllers which can perform stable, relatively high lateral g-level (.2 g's) driving maneuvers.

DGPS and map based systems

More recently, work has been done at the University of Minnesota in conjunction with the Minnesota Department of Transportation to use a Differential Global Positioning System (DGPS) as the basis for a road sensing system [Morellas97][Bodor97]. In this work a high precision map of the road is initially made using the DGPS system. Later, the map is used with the real-time DGPS to determine the vehicle's position on the map and perform lane following. This system has successfully been used to keep a Navistar 9400 semi-tractor within the lane at speeds up to 55 mph.

1.3.3 Combined Longitudinal Control/Lateral Guidance

A number of systems have been developed which combine longitudinal control and lateral guidance systems. The work in this area can be sub-divided into combined ACC/lane following systems and longitudinal/lateral collision avoidance systems.

Combined ACC/lane following systems

The combined ACC/lane following systems control the headway but are not capable of lateral collision avoidance. These systems may be capable of performing lane changes but do not control lane changes based on the sensed driving environment. Among these systems, two stand out as the most advanced.

The first system was developed by Reichart and Naab working with BMW [Reichart94]. The approach here is to have the control strategy working in the background while the driver controls the vehicle. A vision system tracks the lane position and determines how a "normal driver" would steer the vehicle. If the driver deviates significantly, a small torque is applied to the steering column by a servo-motor which acts to return the vehicle to the center of the lane. The headway spacing is measured using an infrared laser range sensor. The prototype system informs the driver if he/she is deviating significantly from the desired throttle position through a haptic interface on the throttle. Future work will include ACC capability as well.

A slightly more advanced system was developed by Tribe et al [Tribe95] in conjuction with Lucas and Jaguar in the PROMETHEUS research program. This system also keep the driver in the loop but has more capability. A vision system tracks the lane and allows for automated lane following or driver assistance only. In the driver assistive mode, the system provides low torque correction in following the lane and vibrates the steering wheel when leaving the lane. The headway system uses microwave radar and controls velocity by actuating the throttle and the brakes. This approach also uses a "haptic throttle" to warn the driver of unsafe velocities when the ACC system is not in use.

Longitudinal/lateral collision avoidance

Little work has been done on developing approaches which are capable of simultaneous longitudinal and lateral collision avoidance. Due partially to the cost of implementation, all of the work to date has been done in simulation. The works reviewed here are classified into four categories based on the control strategy: 1) Geometric path planning, 2) Rule-based, 3) ANN and 4) Potential field.

Geometric path planning approaches

Geometric path planning involves approaches which make use of standard optimization techniques for solving the path planning problems. Two fall into this category.

The first approach was proposed by Shiller et al [Shiller95]. In this work, the host vehicle speed and trajectory is compared to that of the target vehicles. Then, assuming constant velocity of the targets, a space of host trajectories is defined geometrically which is free from collision. This collision free space is then limited by the dynamic constraints of the host vehicle. Then the limited collision free space is searched for a path which is the minimal time collision free solution.

The second approach was developed by Fraichard et al [Fraichard92]. This approach proposes a bang-bang type controller which moves the vehicle at the maximum acceleration or deceleration. Based on this control, three possible states for the vehicle are determined at time $t \div 1$ for three control levels over Δt (maximum acceleration, maximum deceleration or no output). Then the possible states for the vehicle are searched over time to determine a time optimal solution.

Both of these approaches assume ideal sensing (knowledge of targets positions and trajectories in all directions from the host) which makes it difficult to implement these systems. Also, these approaches do not consider the issue of driver comfort (e.g. maximum comfortable accelerations).

Rule-based approaches

In the rule-based approach, rules for driving are defined which determine how the vehicle should be controlled. The rules are based on information about the driving scenario (i.e. host velocity, headway, vehicle positions, etc). Expert systems fall into this category.

One simple example of a rule-based approach was outlined by Neihaus and Stengel [Niehaus94]. In this system, rules are defined base on two parameters. The first parameter is the ratio of separation time to the desired separation time and the second parameter is the ratio of the deceleration to avoid collision to the maximum deceleration capability. These values are calculated as if the host vehicle where in each lane and the rules determine whether a lane change is appropriate or if the host vehicle should reduce speed and follow the preceding target vehicle. The outputs of this algorithm are sent to a headway controller and a lane change module.

A more complicated rule based approach was developed at Carnegie Mellon University by Sukhtankar [Sukhtankar97]. In this approach, several reasoning objects for different tasks (i.e. lane change, take exit, etc.) are created. Each reasoning object acts in parallel with its own set of rules and outputs a desired action. The outputs of the reasoning objects are then used to vote for the desired response. The response which is the winner of the vote is selected and sent to the lateral and longitudinal controllers. Since the rules for the different reasoning objects were difficult to tune, a learning algorithm was developed to tune them automatically. The simulation environment described by Sukhtankar uses realistic models for sensors. The sensor models were based on sensors in use on Carnegie Mellon's test vehicle, Navlab.

A method for combining rules into control action; referred to as fuzzy logic has become popular recently. One rule-based approach by Garnier et al [Garnier95] uses fuzzy logic combined with escape lines. In the escape lines approach, a set of smooth paths which the host vehicle can take are defined. Then out of these paths, a desired path is selected which avoids collision with the target vehicles and stays within the limits of the host vehicle's dynamics. The path determined from the escape lines approach and information about the static and dynamic obstacles in the environment is passed to the fuzzy controller. The fuzzy logic controller provides additional correction to the path to avoid collisions. This controller operates on approximately forty rules.

Using a rule-based system such as these seems to make intuitive sense. After all, when we drive we are following an intuitive set of rules or heuristics. Problems, however, arise due to the complexity of rules, their interaction and how they are affected by slightly different driving scenarios. Furthermore, rule based approaches often perform poorly in a situation in which none of the pre-defined rules apply.

Artificial neural networks

ANN approaches have also been tried for controlling a vehicle in a highway environment. One approach was developed Werner and Engels [Werner95]. In this approach, an environment with two target vehicles on a two-lane highway was demonstrated. The relative position and the relative velocity are mapped to an angle from 0 to 360 degrees for each target. Imaginary numbers are then calculated from these angles. These complex numbers for each of the target vehicles is used as input to the ANN. The outputs of the system are the desired steer angle and velocity.

This approach would be difficult to scale to a driving scenario with several target vehicles. Also, it is questionable whether this design would provide safe steering and velocity commands when the system is faced with situations for which the ANN was not trained, the usual problem with ANN's.

Potential fields

The potential fields based collision avoidance approach was first developed for robotic arms by Khatib [Khatib86] and for mobile robots by Krogh [Krogh84]. In this approach, an artificial potential field is generated which has a high potential at the obstacle locations and a low potential at the goal position. Then, by following the gradient of this potential field, a path for the robot is determined which avoids collision. Much work has been done on using the concept for indoor vehicles which in real time learn their own way around obstacles to a destination [Anderson90] [Morellas95]. The path defined by such an approach does not necessarily meet the kinematic constraints of an automobile and much work has been done to find paths which are traversible by car-like robots [Grupen95][Kyriakopoulos95]. Here we highlight only the potential fields applications which address control of an automobile in either city or highway driving environments.

Hassoun and Laugier [Hassoun93] in conjunction with the PROMTHESIS project in Europe, developed a collision warning system based on potential fields. Here, a potential fields strategy is used to calculate what the desired vehicle response should be, given the current states of the vehicle and the surrounding environment. For information about the environment, this simulation used sensor models which were derived from sensors currently in use on a real vehicle. This vehicle control information was provided to the driver as an advisory system for operation of the vehicle.

Kageyama and Nozaki [Kageyama95] described their approach as a risk field. In this approach, a risk field is associated with the lane with the borders of the lane having the highest risk level. The vehicle follows a path along the center of the lane to be in the "valley" of the risk field. This work also mentions similar fields associated with target vehicles and other obstacles but does not go into detail on how the sets of risk fields can be fused to provide smooth vehicle control.

Reichardt and Schick at Daimler-Benz have also developed a potential fields approach to collision avoidance [Reichardt94]. A potential field is determined by adding together potential fields for the road and for each obstacle around the host vehicle. A path is then determined from this potential field which avoids collision while staying on the road. This work described was based in simulation, but other researchers at Daimler-Benz have reported an operational collision avoidance system following this scheme [Ulmer94]; however no experimental data is provided.

A potential fields scheme for control of a highway vehicle does have one major shortcoming in that it will not necessarily drive a vehicle the way a person would. When humans perform a lane change at highway speeds, the profile of the lane change stays relatively uniform under a variety of circumstances. However, a potential field approach will provide lane change profiles which vary greatly depending on the driving scenarios. This is because the potential field is determined based on measurable target parameters (i.e. headway, closing rate, etc.). As these parameters change

in time, the potential field changes and this results in variations in the profile of the commanded lane change transitions. Also, there are certain driving situations in which drivers make particular decisions. Reichardt and Schick embed these decisions into the shape of the potential field. For instance, they decided that when a target vehicle is approaching from behind with a high closing rate, the host vehicle should not move into the left lane; this is in anticipation that the target vehicle will pass in the left lane. Therefore, a complicated potential shape is created and associated with the target to inhibit the host from moving into the left lane [Reichardt94]. Such complex forces can be generated for various types of scenarios. Ultimately, the issue here is that a decision should be made based on the driving scenario. The potential fields approach alone is not a good method for making driving decisions.

1.4 Report Organization

The remainder of this report is divided into five chapters. The second chapter discusses the theoretical background for the virtual bumper control strategy. The third chapter reviews the issues associated with implementing the virtual bumper. In the fourth chapter, we discuss the approach for placement and evaluation of the range sensor. Our simulation environment and the results from our simulations are discussed in the fifth chapter. Finally, the sixth chapter covers a discussion of the results and our conclusions.

Chapter 2

The Virtual Bumper: Theory and Background

2.1 Introduction

As noted in Chapter One, much work has already been done in the area of driver assistive devices to control inter-vehicle spacing in the direction of travel. Such systems are commonly referred to as headway control or adaptive cruise control (ACC). However, in some emergency driving situations, it has been shown that steering combined with braking is required to successfully avoid collisions [Adams94]. Furthermore, vehicles with large inertias, such as trucks, typically have longer braking distances than passenger cars [Reed87], which makes steering even more critical in effective collision avoidance maneuvers. For these reasons, we have developed a 2-dimensional collision avoidance scheme realized through actuation of the throttle and brakes as well as the steering wheel angle. We have called this approach the virtual bumper due to the characteristics of a bounded space which surrounds the vehicle which when encroached triggers a protective dynamic maneuver of the vehicle. The space and dynamics are defined in software and can thus be dynamically reconfigured.

The virtual bumper concept has its origins in impedance control in which force feedback resulting from physical contact is used to generate the behavior of a mass/spring/damper (i.e. an impedance). Hennessey and Donath [Hennessey93] generalized the impedance control concept beyond its original usage in controlling robotic end effectors in contact with the environment. In this more general setting, the notion of a virtual impedance was developed for use in collision avoidance strategies (e.g. for the elbow) in kinematically redundant arms [Liao97]. While the virtual impedance concept was developed for obstacle avoidance as part of the control strategies for robotic arms, with some modification it can also be applied to collision avoidance control of vehicles.

The virtual bumper was first introduced for collision avoidance of vehicles by Hennessey et al [Hennessey95]. For the virtual bumper concept, we define two spaces around the vehicle: 1) a personal space and 2) a Region of Interest (ROI)(see Figure 2.1). Obstacles that enter the ROI are detected by sensors (e.g. range sensors) which measure their location and their relative velocity to the host (e.g. closing rate). The host vehicle does not necessarily respond to obstacles in the ROI but uses the information about these obstacles in making safe driving decisions. The personal space is a sub-set of the ROI. If there are no obstacles within the personal space, the vehicle follows the desired path. However, if an obstacle encroaches the personal space, a virtual force is computed. The force is then adjusted as necessary to consider all of the obstacles in the ROI. This adjusted virtual force is applied to the host vehicle which "pushes" the host safely away from the encroaching obstacle. The road or more accurately, the lane boundaries, also apply a virtual force to the host

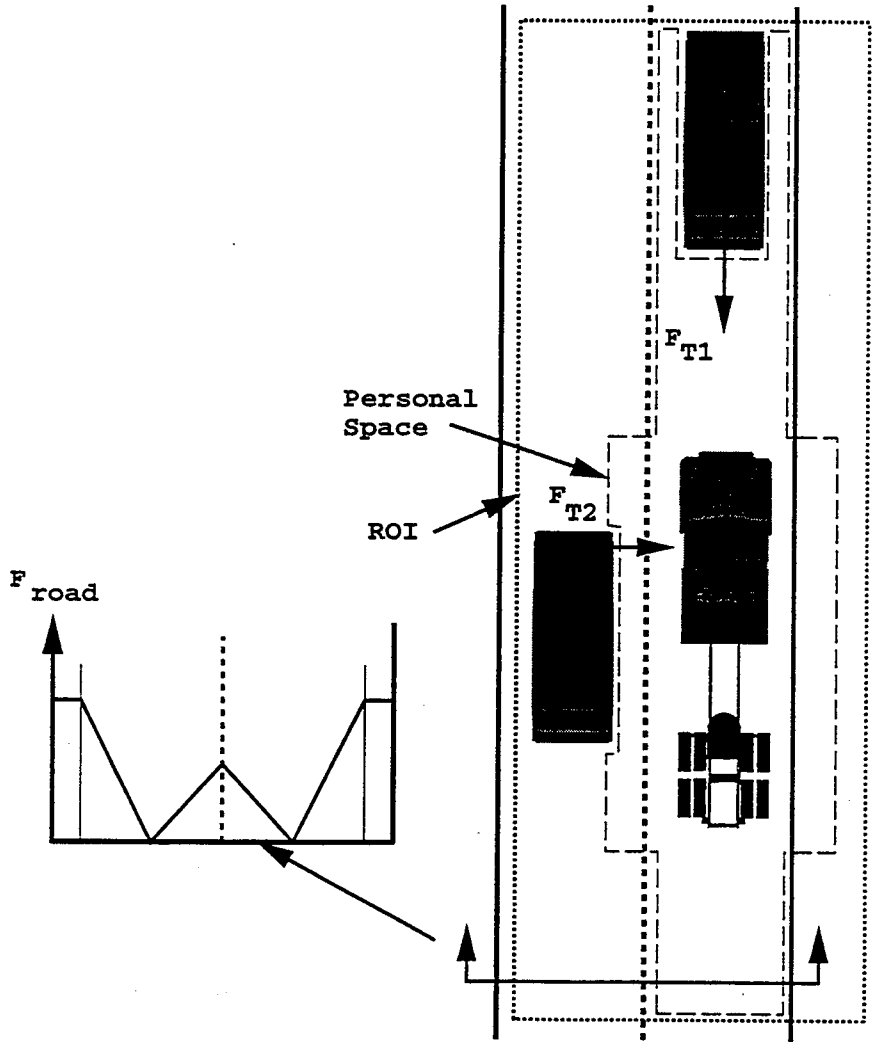


Figure 2.1: Virtual bumper control schematic for highway vehicle.

vehicle that "push" it away from dangerous portions of the road (i.e. shoulders or center-line when there is two way traffic). This requires the ability to sense the vehicle's location relative to the road, e.g., by using highly accurate DGPS. The forces from the obstacles and the road are then applied to an impedance feedback loop. An impedance assigned to the vehicle then determines how the vehicle will respond to the forces imposed on the host.

Developing the virtual bumper from a concept to a working algorithm means addressing four main issues.

1) Size and shape of the personal space and ROI.

2) Range sensor placement and characteristics.

3) Virtual force generation.

4) Selection of host impedance. (Used to transform virtual forces to control commands).

The characteristics and placement of the radar sensors is discussed in Chapter 4. The remainder of the issues are discussed in this chapter.

In the development here, the virtual bumper is composed of two separate impedance control loops combined with a set of heuristics. One of the impedance control loops adjusts the velocity (longitudinal impedance control) and addresses the task of maintaining desired headway. This controller is based on the work by Fancher et al [Fancher94c] [Fancher93] which we reformulated to fit into an impedance feedback control scheme. The other impedance control loop adjusts the vehicle's lateral position in the lane. This second loop allows the system to perform lane changes and maintain desired lateral spacing between vehicles. The heuristics are added to this loop to help decide which should be the desired lane for a maneuver. When the lateral and longitudinal loops are operating together, the host is provided with control which actuates the steering wheel and the accelerator/brakes to maneuver the vehicle on a collision free path.

We will again clarify the terminology that is used in this chapter and throughout this document. The vehicle which is being controlled is referred to as the host vehicle (or the host) and the "obstacle" vehicles are referred to as target vehicles (or the target). Also, there is a coordinate system associated with the road. The coordinate axes are oriented with its x-axis pointing down the road, positive in the longitudinal direction of travel and the y axis is perpendicular to the road, positive when moving from the right to the left lane. The origin of the coordinate system is in the center of the right-most lane and moves down the road with the host's control point (i.e. a moving origing). Later in this chapter, the terms longitudinal and lateral are used interchangeably with x and y, respectively, in discussion and notation.

In this chapter, we present the theory for the algorithm in detail. We will begin the discussion by reviewing the concept of the range vs. range rate phase plane introduced by Fancher et al [Fancher93]. This phase plane provides a good basis for a fundamental understanding of ACC systems and is a good mechanism for classifying different driving scenarios (i.e. normal driving, emergency driving). Next, we will discuss the development of the longitudinal impedance control. Then we will continue the discussion with the application to lateral impedance controller.

2.2 The Range vs. Range Rate Phase Diagram

In this section we discuss the range, range rate diagram introduced by Fancher et al [Fancher94c][Fancher93]. This diagram is a plot of the measured range (R) versus closing rate or range rate ($\dot{R}$) to the preceding vehicle. These plots allow for a fundamental understanding

of headway control systems. The design of controllers can be based on the R,$\dot{R}$ diagram and the diagrams can also be used to evaluate system performance. Furthermore, the phase diagram can be used to graphically represent regions of different driving situations (i.e. high braking required, collision unavoidable, etc.). In this document, the diagram is referenced often in designing and evaluating both the longitudinal and lateral impedance controllers. The R,$\dot{R}$ phase diagram is detailed carefully by Fancher et al in [Fancher94a] and [Fancher94b]. Here, we review these references for the reader's understanding and convenience.

2.2.1 Fundamentals of Headway Control

Before discussing the phase diagram let's quickly discuss some basics of headway control. This will aid in defining some terminology which will be used later.

A headway controller acts to maintain a desired spacing between the host and the preceding target vehicle. Two of the inputs to this controller are the distance to the preceding target (the headway range, R) and the time based rate of change in headway ($\dot{R}$). The $\dot{R}$ is the difference between the target's velocity (V_{target}) and the host's velocity (V_{host}) as shown in the following equation:

$$\dot{R} = V_{target} - V_{host} \tag{2.1}$$

The desired vehicle spacing is often referred to as the desired headway range (R_H) and can usually be set by the driver.

Note the relationship between the measured range and range rate. Consider the situation where the headway controller is approximately tracking the desired headway. If the measured range is too short ($R < R_H$) the controller must slow down the host until $V_{host} < V_{target}$ ($\dot{R} > 0$) to allow the range to increase. Conversely, if ($R < R_H$) the host must decelerate for $V_{host} > V_{target}$ ($\dot{R} > 0$) to reduce the range. From this discussion we see that the goal of the headway controller is to adjust V_{host} to reach the state where $R = R_H$ and $\dot{R} = 0$.

2.2.2 Trajectories in the R,$\dot{R}$ Phase Diagram

The measured range and range rate create points in the R,$\dot{R}$ phase diagram. Over time, these measured points move through the diagram along trajectories. With an understanding of the phase diagram, these trajectories can be used to evaluate the performance of the headway controller.

First, let's discuss the basic relationship between R and $\dot{R}$ and its effect on the phase diagram. Consider some typical trajectories on the R,$\dot{R}$ diagram shown in Figure 2.2 for a headway controller. On the diagram the arrows signify the direction of travel along the trajectories. These arrows show the effect of $\dot{R} > 0$ and $\dot{R} < 0$ on the changes in R. On the left side of the R-axis, $\dot{R} < 0$

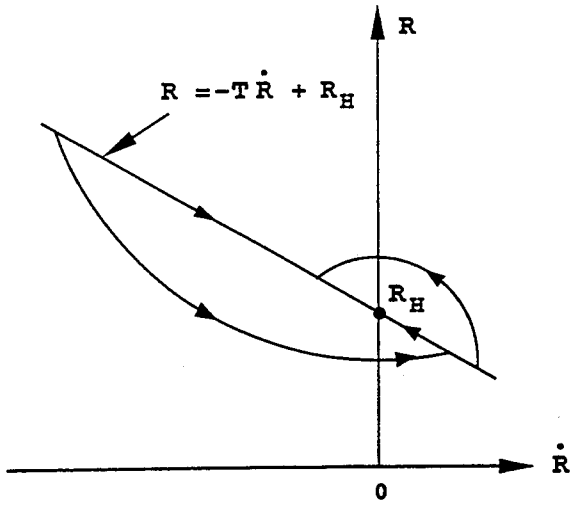


Figure 2.2: Typical trajectories in the R,$\dot{R}$ plane for headway controller.

which means that R is decreasing. The result is that all trajectories on the left side of the R-axis are followed in a downward motion. Conversely, on the right side of the R-axis, $\dot{R} > 0$ so R is increasing. Trajectories in this region are followed in an upward motion. Note that the trajectories in Figure 2.2 end at the R_H,0 point because this is the goal of the headway controller.

Lines in the phase diagram have great significance. A line in these diagrams represents a first order differential equation for R. For instance, examine the equation for the line shown in Figure 2.2 and listed in Equation 2.2.

$$R = -T\dot{R} + R_H \tag{2.2}$$

This line has a slope of $-T$ and intersects the headway controller goal point, R_H,0. This equation also represents a differential equation, in which T is the time constant. From this, we see that if a headway controller adjusts the velocity to follow a first order response in range, the R,$\dot{R}$ pair (or point) will follow a straight line trajectory in the phase diagram.

It is also useful to understand the type of trajectories associated with a constant acceleration/deceleration. Consider the initial point of R_1,$\dot{R}_1$ in Figure 2.3. A constant deceleration level ($\ddot{R}$) can be applied which brings the system to the R_H,0 point. Here, the deceleration level is derived based on the time domain equations for R and $\dot{R}$.

$$R(t) = R_1 + \dot{R}_1 t + \frac{1}{2}\ddot{R}t^2 \tag{2.3}$$

$$\dot{R}(t) = \dot{R}_1 + \ddot{R}t \tag{2.4}$$

Then, since at the goal point $\dot{R} = 0$, Equation 2.4 is set equal to zero and solved for t.

$$t = -\frac{\dot{R}_1}{\ddot{R}} \tag{2.5}$$

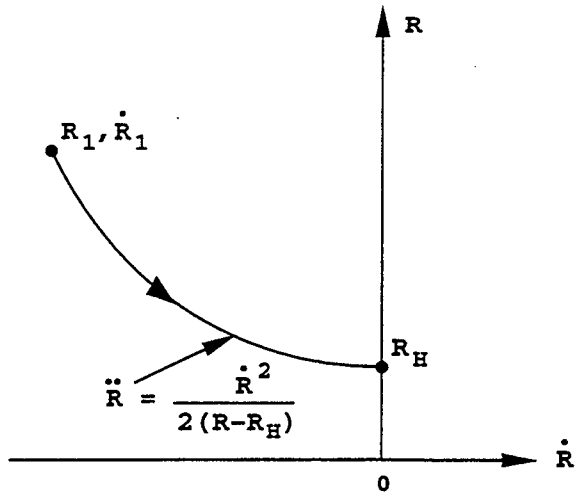


Figure 2.3: Constant deceleration trajectory in the R,$\dot{R}$.

This value for time is then substituted into Equation 2.3 and we solve for the deceleration level, $\ddot{R}$.

$$\ddot{R} = \frac{\dot{R}_1^2}{2(R - R_H)} \tag{2.6}$$

From this equation, we can see that a constant deceleration level is associated with a parabolic trajectory on the phase diagram (see Figure 2.3).

2.2.3 Headway Controller Design Using the R,$\dot{R}$ Phase Diagram

Now that we have explained the R,$\dot{R}$ phase diagram, we will outline its use in designing headway controllers. Here we briefly review a first order controller designed by Fancher et al [Fancher93] using the phase diagram. The purpose of this review is not to detail the control law design. Rather, the purpose is to demonstrate how the phase diagram can be used to specify modes of operation and to see what some typical R,$\dot{R}$ trajectories look like for a headway controller.

In Fancher's design, the control law when the system is under headway control is based on the following equations:

$$V_{command} = V_{target} + \frac{1}{T}(R - R_H) \tag{2.7}$$

where:

$$V_{target} = V_{host} + \dot{R} \tag{2.8}$$

$$R_H = T_H V_{target} \tag{2.9}$$

and:

T_H = Desired headway time selected by driver.

$1/T$= Controller gain.

Then based on the assumption that the velocity controller approximately tracks the commanded velocity, V_{host} is set equal to $V_{command}$. This allows the response for the headway to be reduced to the first order differential equation shown in Equation 2.10 (see [Fancher93]).

$$R = -T\dot{R} + R_H \tag{2.10}$$

From this equation we see that the controller gain determines the time constant of the headway response. Fancher selects the controller gain to allow for a large enough T such that the velocity controller can track the commanded deceleration levels.

With the headway control law defined, we must now define when it should be used. When under headway control, if the time constant, T, is chosen sufficiently large, the R,$\dot{R}$ points follow a straight line in the phase plane defined by Equation 2.10. The slope of this line is $-T$ and it intersects the point R_H. This line defines the dynamic response of the system and also determines when the headway controller is to be used (see Figure 2.4). When a measured point is to the right of this line in the phase plane, the system follows conventional cruise control. For points to the left of this line, the vehicle is under headway control.

There are also other important regions outlined on Figure 2.4. First, notice that there is a maximum range limit associated with the sensors which results in a maximum value for range beyond which no R,$\dot{R}$ points can be measured. In addition to a desired headway there is the safe headway (R_S) which is a defined minimum "safe" headway. The headway controller would still be operating "safely" if it over-shoots R_H[1] but it should not over-shoot R_S. The figure also shows a parabola associated with a maximum deceleration level of the host vehicle (D_{max}). This curve intersects the 0,0 point on the phase plane. Points below and to the left of this curve represent situations in which a collision can not be avoided unless the target vehicle accelerates. In this region, a collision is likely. The figure shows another parabola associated with the maximum deceleration capability of the headway controller (D_{max_H}).[2] This curve intersects the R_S,0 point on the phase plane. Points between this curve and the D_{max} curve represent "dangerous" situations in which the headway controller can not keep from over-shooting R_S and driver intervention is required.

Under this control law, Figures 2.5 and 2.6 show some responses for typical driving. In both cases a target vehicle moving at a constant velocity is approached from behind ($\dot{R} < 0$). When the R,$\dot{R}$ point is above the switching line (see Equation 2.10) the vehicle is in convention cruise control mode so V_{host} is not adjusted. This is signified by the vertical line in the R,$\dot{R}$ trajectory associated with no deceleration ($\dot{R}$ = constant) shown in both figures. In Figure 2.5, the relative velocity is not very large. The vehicle is approached and the host slows down following a line in the phase

[1] In its downward descent.
[2] D_{max} differs from D_{max_H} in Fancher's case since his headway controller does not apply the brakes.

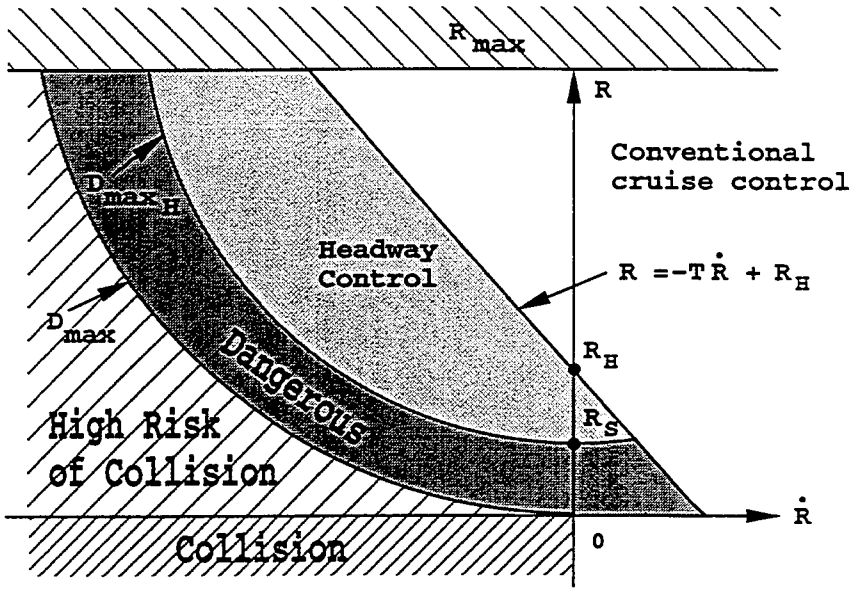


Figure 2.4: Regions in the R,$\dot{R}$ phase diagram.

diagram to R_H,0. The initial relative velocity is larger in the scenario outlined by Figure 2.6. In this case the commanded decelerations are too large to track (Fancher does not use braking) so the response follows a constant deceleration curve associated with coasting in gear. The system follows this curve and overshoots the desired headway. The system continues to decelerate until it reaches the first order response line. The controller then accelerates the vehicle and the trajectory follows the first order response line to the R_H,0 goal point.

From this discussion, we can see that the R,$\dot{R}$ phase diagram is a very useful tool in the development of headway controllers. The phase diagram can be used to determine the mode of operation of headway controllers. For instance, these controllers must switch from a conventional cruise control mode (driving at user set velocity) to a headway control mode. The R,$\dot{R}$ diagram provides an elegant method for defining when to switch by using a defined line. This phase diagram also defines regions which require higher deceleration or in which a collision is unavoidable. These regions can be used to define different modes of controller operation, such as allowing for emergency braking or issuing driver warnings. Furthermore, trajectories in this phase plane can be used to define a desired system response. These responses can be used to develop controllers and the performance of these controllers can be evaluated using the phase diagram. The R,$\dot{R}$ phase diagram has many uses which assist in the understanding of headway controller design.

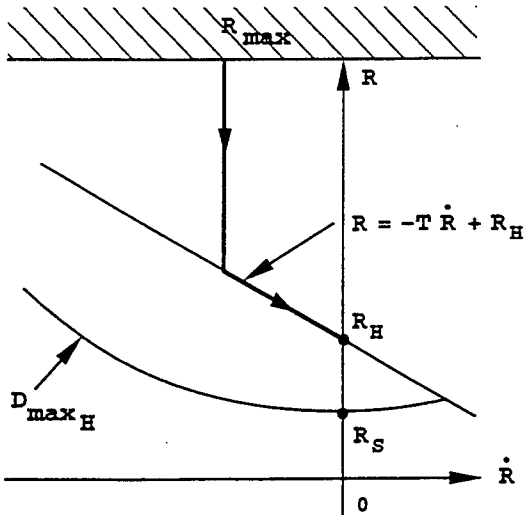


Figure 2.5: Typical R,$\dot{R}$ trajectories for small initial relative velocity.

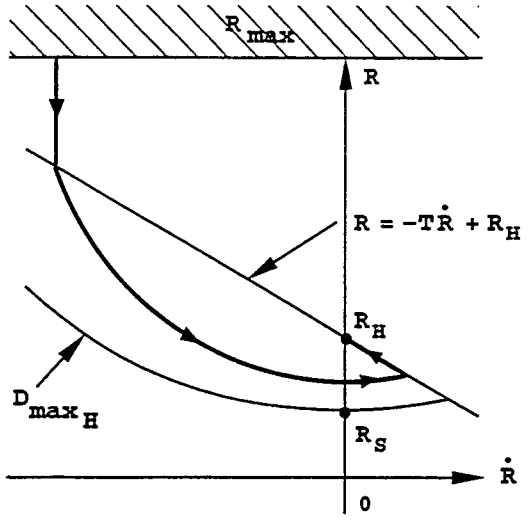


Figure 2.6: Typical R,$\dot{R}$ trajectories for large initial relative velocity.

2.3 Longitudinal Impedance Control

Now using the R,$\dot{R}$ phase diagram, we will discuss the development of the longitudinal impedance controller. The longitudinal impedance control loop is developed to operate as an ACC system. This control loop adjusts the vehicles throttle and brakes to maintain desired spacing with preceding target vehicles.

For the longitudinal impedance controller, a personal space is defined as the region directly in front of the host vehicle (see Figure 2.7). When there are no targets within the personal space, the vehicle travels at the desired velocity. When a target is within the personal space, a virtual force is generated by considering a virtual spring and damper between the host and target vehicle. The force is applied to the vehicle assigned impedance which adjusts the velocity to maintain the desired headway.

For a better understanding of this approach, consider the block diagram for the longitudinal

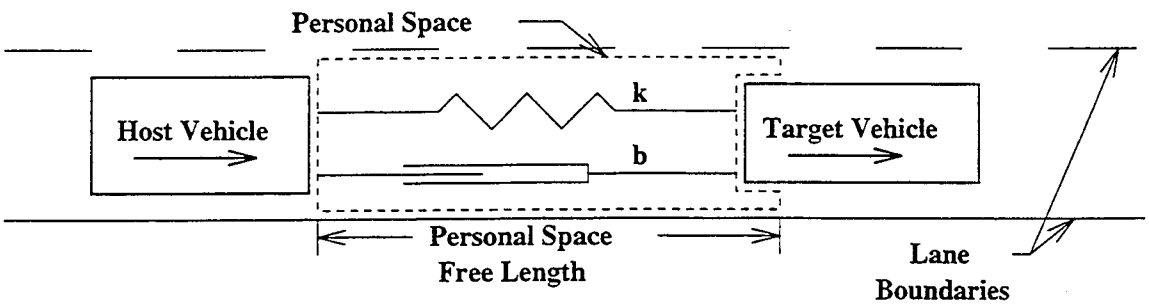
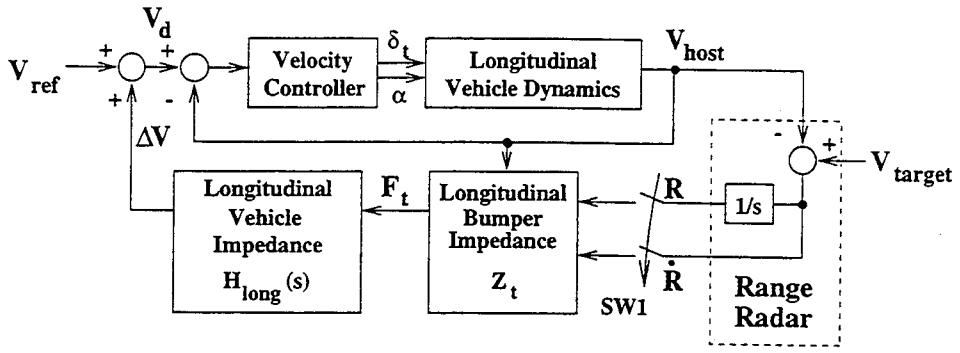


Figure 2.7: One dimensional impedance control for highway vehicle.



V_{ref} = Operator selected nominal velocity.
ΔV = Offset in desired velocity calculated by impedance controller.
V_d = Desired velocity for velocity controller.
δ_t = Commanded throttle position.
α = Commanded brake position.
V_{host} = Host vehicle velocity.
V_{target} = Target vehicle velocity.
s = Laplacian operator.
R = Range to target measured by radar sensor.
$\dot{R}$ = Range rate to target measured by radar sensor.
SW1 = Software switch that is enabled when target is within ROI.
Z_t = Virtual impedance associated with space between target and host vehicles.
F_t = Virtual force calculated for target which is applied to host vehicle.
$H_{long}(s)$ = Longitudinal vehicle impedance.

Figure 2.8: Longitudinal impedance control block diagram.

impedance controller shown in Figure 2.8. This diagram shows two control loops. The inner loop is the velocity control loop. The controller provides both throttle and braking commands to control the host velocity. It is a PI controller and is tuned such that the host velocity will track the acceleration profile of the input signal. Documentation of the velocity controller is provided in Appendix A. The outer loop is the impedance feedback loop and it is only activated when a target is within the host's personal space. In this loop, a virtual force is computed using the range (virtual spring length) and range rate (virtual damper velocity) to the target. The force equation is adjusted based on the host vehicle velocity. The virtual force is then applied to the longitudinal impedance assigned to the host vehicle and results in a velocity offset. The velocity offset is added to the reference velocity to result in a desired velocity. The desired velocity is then used as the input to the velocity controller.

We will now go on to discuss the main design issues outlined in section 2.1. We will discuss the issues in reverse order of their listing, starting with impedance selection, then we will discuss the force computation and finish with defining the personal space. This order of discussion is followed because we must first define the longitudinal impedance of the host to determine what type of

force will produce the desired result. And likewise, understanding the force equations is essential in defining the personal space.

Before continuing, since the longitudinal impedance controller is based in great part on the work of Fancher et al [Fancher94c], we would like to take a moment to note the primary differences. First this approach applies the host's brakes to control the velocity while Fancher's does not. Second, since the brakes are applied, a non-linear control algorithm is implemented when there is a requirement for large decelerations. Fancher's approach is based on linear control equations only. Also, Fancher's work assumed that the velocity controller tracks the desired velocity command issued by the ICC system with negligible error. We do not make this assumption.

Upon further examination, the control strategy implemented here might be more appropriately called admittance control. Admittances accept effort and yield flow, while impedances accept flow and yield effort. From Figure 2.8, we see that the target impedance is labeled correctly as it accepts a flow (R and $\dot{R}$) and yields an effort (F_t). However, since forces are the input to our impedance controller ($H_{long}(s)$) and flow is the output, "admittance controller" would seem to be the more correct term. It is our observation, however, that "impedance control" is the more frequently used term by other researchers when referring to this type of control strategy. Thus in keeping with convention, we also will continue to refer to it as "impedance control" throughout this document, even though "admittance control" is technically the correct term.

2.3.1 Longitudinal Vehicle Impedance Selection

The longitudinal response that the host vehicle has to target vehicles in its personal space is defined by two things: 1) the longitudinal vehicle inpedance and 2) the longitudinal bumper impedance. The vehicle impedance represents the vehicle dynamics (i.e. mass, acceleration/deceleration capabilities). For a given force applied to the vehicle impedance, an appropriate $\Delta\ V$ is produced. The bumper impedance represents the virtual dynamics associated with the space around the host vehicle. For a given encroachment in the personal space, a virtual force is generated which is applied to the vehicle impedance. This decoupling of the vehicle dynamics and the bumper dynamics takes into account the realizable dynamic response of the host vehicle which is key to the success of this control strategy. In this section, we will discuss the longitudinal vehicle impedance selection. The longitudinal bumper impedance is discussed in the next section.

In the area of impedance control, much work has been done in selection of impedance. Impedances can be selected such that the system will behave as a very stiff system or be very compliant (within the limits of the system and the controller [Jossi95b][Jossi95a]). In this application we want an impedance which will limit the velocity offset to change at a rate which the velocity

controller can track reasonably well. Essentially, it is desired that the impedance be matched to the longitudinal dynamics of the vehicle.

Let's consider the equation of motion for a vehicle. The equation of motion for the longitudinal dynamics consists of several forces acting on a mass [Gillespie92] as shown in Equation 2.11.

$$ma = F_{engine} - F_{brake} - F_{roll} - F_{aero} - F_{grade} \tag{2.11}$$

where:

F_{engine} = Engine torque transfered to a force by the transmission and tires.
F_{brake} = Braking torque transferred to a force by the tires.
F_{roll} = Rolling resistance of the tires.
F_{aero} = Aerodynamic resistance to motion.
F_{grade} = Force induced by grade of road.

A force applied to this system by actuating the throttle or the brake causes approximately a first order response in velocity. The relationship of this force to velocity can be represented by Equation 2.12 where m is the mass of the vehicle and the damping coefficient, b, is a function of the rolling resistance, aerodynamic resistance and road grade and changes with velocity.

$$H_{long}(s) = \frac{1}{ms + b} \tag{2.12}$$

Using this transfer function for the longitudinal impedance would be appropriate, but here we have simplified the impedance by neglecting the damping coefficient. The longitudinal impedance assigned to the host vehicle is shown in Equation 2.13.

$$H_{long}(s) = \frac{1}{ms} \tag{2.13}$$

This impedance produces velocity offsets that can be tracked if the forces are limited properly. Forces must be limited such that an acceleration or deceleration will be within the vehicle's limits. Therefore, force limits are directly tied to the longitudinal impedance and vehicle dynamics.

2.3.2 Computation of the Virtual Forces: The Virtual Bumper Dynamics

This section describes the bumper impedance, i.e. the force generated and "applied" to the vehicle as a result of the "compression" of the "bumper." Now that we have specified a vehicle impedance for the longitudinal system, the next step is to determine what the forces will be that act on this system. The equations for the forces will determine what type of response our vehicle will have as a result of other targets entering its personal space. Therefore, the approach here is to select the type of response we desire, and derive the force equations from this response. In other words, we

are defining the dynamics of the virtual bumper, which will generate the forces and apply them to the vehicle.

In our application, we want to be able to respond to a great variety of situations. The driving scenarios will range from approaching a vehicle which is moving slightly slower than the host (i.e. 2 or 3 mph slower) to approaching a stopped vehicle. When approaching a vehicle that is moving only slightly slower, a force based on linear equations can be derived which provides the desired response. However, when a approaching a stopped vehicle, a linear force equation will not give the desired performance. In this situation, a nonlinear force proves to be a better choice. For these reasons, the longitudinal force is calculated using both linear and nonlinear equations. The linear equation is used when relative velocities are small and the nonlinear force is used when greater braking is required of the system.

Impedance control depends on the strong assumption that the inner-loop controller tracks the commanded position/velocity with a sufficiently high bandwidth [Jossi95a]. In this application, our inner loop velocity controller does not track the commanded velocity exactly. This is because a velocity controller that produces very fast velocity response would not provide a reasonable comfort level for the passengers. This error in tracking of the velocity controller adversely effects the performance of the longitudinal impedance controller; this issue is addressed in Chapter 3. For the derivation of forces in this section, we just assumed that the velocity controller does indeed track the commanded velocity without significant error.

Linear longitudinal forces

The linear force equations can be derived by considering the virtual damper and spring system shown in Figure 2.7. For simplification, let's assume that the target vehicle is moving at a constant velocity. Then, when the target is within the personal space of the host, the relative motion between the vehicles can be modeled as a mass mounted to a rigid surface by a spring and damper (see Figure 2.9). For this system, the force acting on the mass (the host vehicle assigned impedance) is defined by Equation 2.14.

$$F_{long} = -b\dot{x} - kx \tag{2.14}$$

The variable x is the deflection from the free length of the spring. This spring free length is set to be equal to the desired headway spacing. This desired headway spacing is defined to be the distance traveled over a headway time (T_H) which is set by the operator. This distance is $T_H V_{host}$. If we use the desired headway for the spring free length and we describe the distance between the vehicles in terms of the measured range (R) and range rate ($\dot{R}$), then the force equation is described by

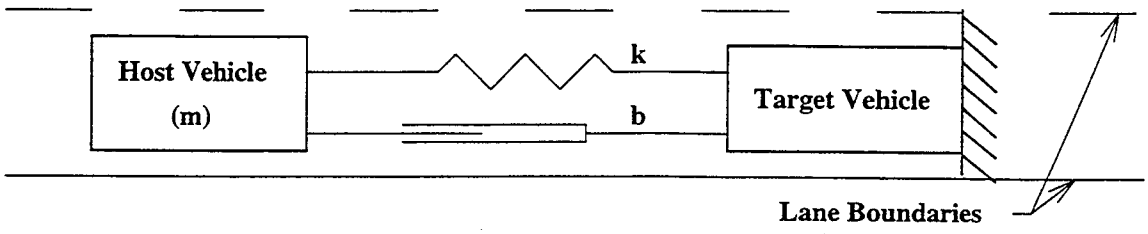


Figure 2.9: Simplified model of one dimensional impedance control.

Equation 2.15.

$$F_{long} = b\dot{R} + k(R - T_H V_{host}) \tag{2.15}$$

It is important to note that the personal space is not defined by the free length of the spring. The range of the personal space is described later in this section.

The characteristic equation for this dynamic system with the force equations defined above is described by Equation 2.16.

$$Ms^2 + bs + k \tag{2.16}$$

This characteristic equation determines the nature of the response for the host vehicle's headway as it approaches a target vehicle. The desired response can be set by selecting the appropriate b and k coefficients.

Let's consider the desired headway response when using a linear force. The linear force is used to respond to driving situations where minimal or no braking is required. In these situations it is desired that:

1) The desired headway is approached smoothly with no overshoot.
2) The commanded response is slow enough to be tracked with minimal or no braking.

The first requirement can be met if the system has approximately a first order response. The second requirement can be met if the time constant of the system is selected large enough that the response can be tracked without significant braking.

Our system response defined by Equation 2.16 is second order but can be made to approximate a first order system. This is done by selecting b and k such that the system is over-damped ($\zeta > 1$). Damping should be set high such that the system response is determined by the dominant pole (p_1). The system will then approximately follow a first order response with a time constant of $\tau = 1/p_1$. In this manner, the b and k coefficients are selected to arrive at the desired first order response. The actual coefficient selection for the forces is discussed in Chapter 3.

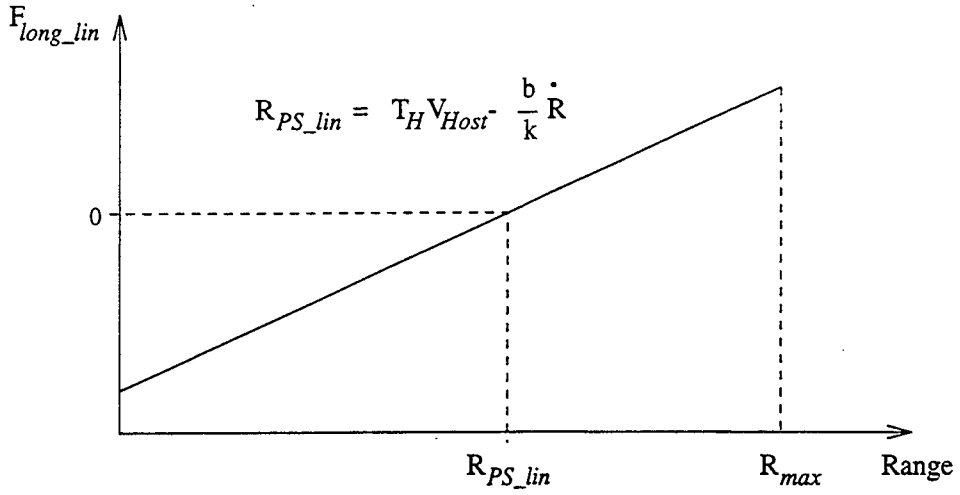


Figure 2.10: Longitudinal linear force vs. range for constant V_{host} and constant V_{target}.

Linear force personal space

The above describes the linear force generation equations and how these forces are transformed into velocity offsets. The next step is to determine at what range the targets will be at when these forces are first applied to the host vehicle. In other words, the range to the edge of the personal space is used to determine when to close switch one (SW1) in the block diagram shown in Figure 2.8.

The personal space is defined simply by looking at the linear equation for longitudinal force (Equation 2.15). Consider the forces generated by this equation as the host approaches a target with a constant relative velocity ($\dot{R} < 0$) as shown in Figure 2.10. As the figure shows, there is a range to the target in which the calculated force is zero. If the range is increased from this point, the force is positive. Conversely, the force is negative for shorter ranges. The positive forces will accelerate the host vehicle while the negative forces will cause it to decelerate. Therefore, the range at which the force is zero defines the point when the forces are applied to the host vehicle. This range is the distance to the edge of our personal space for the linear forces (R_{PS_lin}).

The distance to the edge of our personal space can be calculated using Equation 2.15. If we set the force equal to zero, the value of the range is equal to R_{PS_lin}. Solving the equation for the range leads to Equation 2.17.

$$R_{PS\_lin} = T_H V_{host} - \frac{b}{k}\dot{R} \tag{2.17}$$

From the equation, we can see that the personal space is a function of the relative velocities as well as the host vehicle speed. Intuitively this makes sense because it means that for a given host velocity, the personal space expands as the relative velocities to targets increases. In other words, the velocity is adjusted sooner when approaching a much slower target.

The personal space can also be defined as a line on the R,$\dot{R}$ diagram as shown in Figure 2.11.

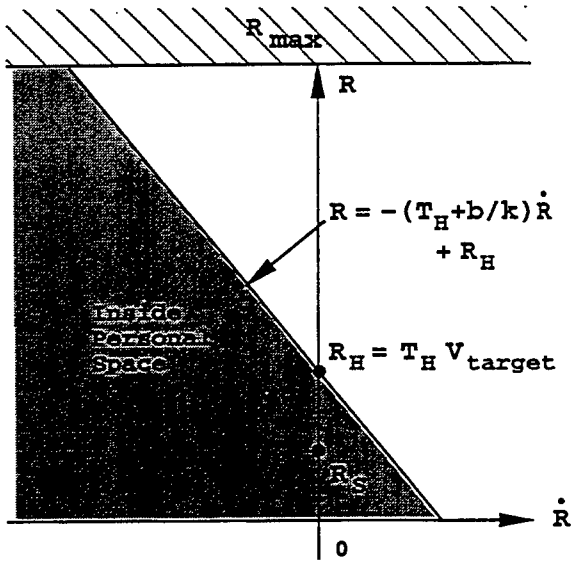


Figure 2.11: Personal space for the longitudinal linear force defined on the R, $\dot{R}$ phase plane.

The line is defined by Equation 2.18 and is derived by substituting $V_{host} = V_{target} - \dot{R}$ into Equation 2.17.

$$R_{PS\_lin} = T_H V_{target} - \left(\frac{b}{k} + T_H\right)\dot{R} = R_H - \left(\frac{b}{k} + T_H\right)\dot{R} \tag{2.18}$$

This line has a slope of $-(T_H + b/k)$ and intersects range axis at the desired headway, $R_H = T_H V_{target}$. The desired headway is based on the target velocity because $V_{host} = V_{target}$ at the end of the tracking maneuver.

Once a target has entered the personal space and the host vehicle has reduced its velocity, a force derived from the target is continually applied to the host until the host velocity returns to its reference velocity (V_{ref}). For example, if a target is moving 10 mph slower than the host, the host decelerates and tracks the target's velocity. If the target now accelerates by 2 mph, a positive linear force is generated. Since the host is traveling slower than V_{ref}, the force is applied to the host vehicle and causes it to accelerate. In this way the host continues to track the target velocity until V_{target} is greater than V_{ref}.

Headway control example using linear forces

To allow for a better understanding of the longitudinal impedance controller, we can examine a typical driving example where linear forces are applied. In this example the target vehicle is traveling at a constant velocity which is slower than the host's desired speed. The host approaches the target and slows down to maintain the desired headway.

The response of the controller is evaluated by plotting the R and $\dot{R}$ responses in the R, $\dot{R}$ plane

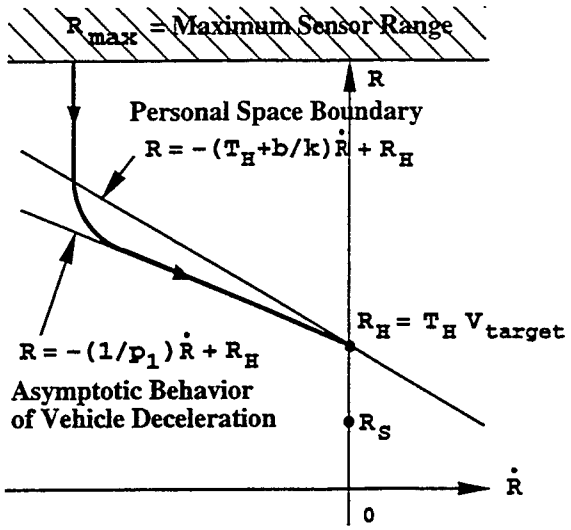


Figure 2.12: R, $\dot{R}$ diagram for typical headway tracking.

as shown in Figure 2.12. When the target is initially sensed, $\dot{R} < 0$ and $R = R_{max}$ where R_{max} is the maximum sensor range. This point in the phase plane is above the line which defines the personal space (Equation 2.18) so the velocity of the host is not adjusted. The target is then approached at a constant closing rate until the R,$\dot{R}$ point falls on or below the personal space line. This is shown as the vertical line segment in Figure 2.12. Once the R,$\dot{R}$ point crosses the personal space boundary line a linear force is applied to the host vehicle. The host velocity is adjusted and the R,$\dot{R}$ points follow approximately a line (a first order response) to the R_H,0 point on the phase diagram. The time constant of the first order response defines the slope of the line. In our case, the systems dominant pole (p_1) determines the time constant as $\tau = 1/p_1$. Therefore, in our response the R,$\dot{R}$ points follow the line with slope $-1/p_1$ to the R_H,0 point. The host vehicle stays at this headway until the target vehicle changes lanes or changes velocity.

Nonlinear longitudinal forces

As stated earlier, the nonlinear forces are applied when greater levels of braking or vehicle deceleration are required. In defining this force equation the first step is to select the desired system response when braking is required.

Consider the driving scenario in which the host vehicle is traveling 55 mph while approaching a target moving 15 mph. The linear force equation would reduce the host velocity approximately following a first order decay type of response. This would mean large deceleration with the brakes fully engaged, followed by smooth adjustment of the throttle to track the target vehicle's velocity. Clearly this is not a driving response which the passengers would find comfortable. Instead, a more

suitable response would be for the system to engage a constant deceleration level. The host vehicle could be decelerated with a constant level of braking and brought to the desired headway without any oscillation. This is the desired response which is used in deriving the nonlinear force equations.

To achieve the desired response, we first determine the constant deceleration level which brings the host vehicle to the desired headway. In section 2.2, these constant deceleration levels are derived in terms of the measured R and $\dot{R}$ and the equation is repeated here in Equation 2.19.

$$D_{host} = \frac{\dot{R}^2}{2(R - R_H)} \tag{2.19}$$

This equation defines a parabolic curve in the R,$\dot{R}$ plane. Given any point on this curve, if the constant deceleration level (defined by Equation 2.19) is applied over time, the vehicle follows a parabolic trajectory in the R,$\dot{R}$ plane to the desired headway (R_H).

The desired headway is based on the velocity of the target vehicle as shown in Equation 2.20.

$$R_H = T_H V_{target} + R_{H_0} \tag{2.20}$$

where:

$$R_{H_0} = \text{Desired headway when } V_{host} = 0.$$

This equation for R_H is used because at the end of the maneuver, the host vehicle will be moving at the same velocity as the target vehicle. Therefore, the desired headway during the maneuver can be thought of as the desired steady state headway. The offset of R_{H_0} is added to prevent the desired headway from being zero when the the target vehicle is stopped ($V_{target} = 0$).

From Equation 2.19, it is clear that the calculated deceleration level goes to infinity as the measured range approaches the desired headway. To prevent this from occurring, R_H is replaced by a scaling headway, $R_{scaling}$, which is a range between R_H and a minimum safe headway R_S. Then the value of $R_{scaling}$ is determined by the deceleration level calculated for the previous time step as shown in Equation 2.21.

$$R_{scaling} = R_S + \left(\frac{R_H - R_S}{D_{max} - D_{PS\_nlin}}\right)(D_{host}(n-1) - D_{PS\_nlin}) \tag{2.21}$$

where:

R_S = Minimum safe headway.

D_{max} = Maximum deceleration capability of vehicle.

D_{PS_nlin} = Deceleration level which defines the personal space for the nonlinear force.

$D_{host}(n-1)$ = Calculated deceleration level at last time step.

The minimum safe headway is calculated using the following equation.

$$R_S = T_S V_{target} + R_{S_0} \tag{2.22}$$

where:

T_S = Minimum safe headway time.

R_{S_0} = Minimum safe headway when host is stopped.

The selection and function of D_{PS_nlin} is discussed below.

Now that we know the desired deceleration level, a force can be calculated by multiplying by the mass of the vehicle. The nonlinear force is calculated per Equation 2.23.

$$F_{long\_nl} = \max \left\{ \begin{array}{c} -\frac{M\dot{R}^2}{2(R-R_{scaling})} \\ -MD_{max} \end{array} \right\} \tag{2.23}$$

Of course in this equation the value of the $\dot{R}$ must be less than zero and the value of R must be larger than $R_{scaling}$. Using $R_{scaling}$ in the force calculation means that the host vehicle will be allowed to overshoot the desired headway but not the safe headway level. The reason for this is to allow for smoother deceleration of the vehicle.

It is important to note that the nonlinear force is only used to decelerate the host vehicle. The moment that the value of $\dot{R}$ is greater than or equal to zero, the linear force is used instead. Then the linear force continues to adjust the host velocity until the desired headway is achieved.

Nonlinear force personal space

Now that we have defined both linear and nonlinear force equations, the next step is to define when to switch between them. The linear and nonlinear forces are used in different regions of the R,$\dot{R}$ plane as shown in Figure 2.13. Regions were the nonlinear force is applied define the nonlinear force personal space.

The nonlinear forces are applied for all measured R,$\dot{R}$ points which are to the left and below the curve defined by the constant deceleration level, D_{PS_nlin} (see Figure 2.13). Using the equation for constant deceleration (Equation 2.19) and this deceleration level, the nonlinear personal space can be solved for in terms of a range (R_{PS_nlin}) as shown in Equation 2.24.

$$R_{PS\_nlin} = R_S + \frac{\dot{R}^2}{2(D_{PS\_nlin})} \tag{2.24}$$

The measured ranges can then be compared against R_{PS_nlin}. If $R < R_{PS\_nlin}$, then the nonlinear force is applied to the system.

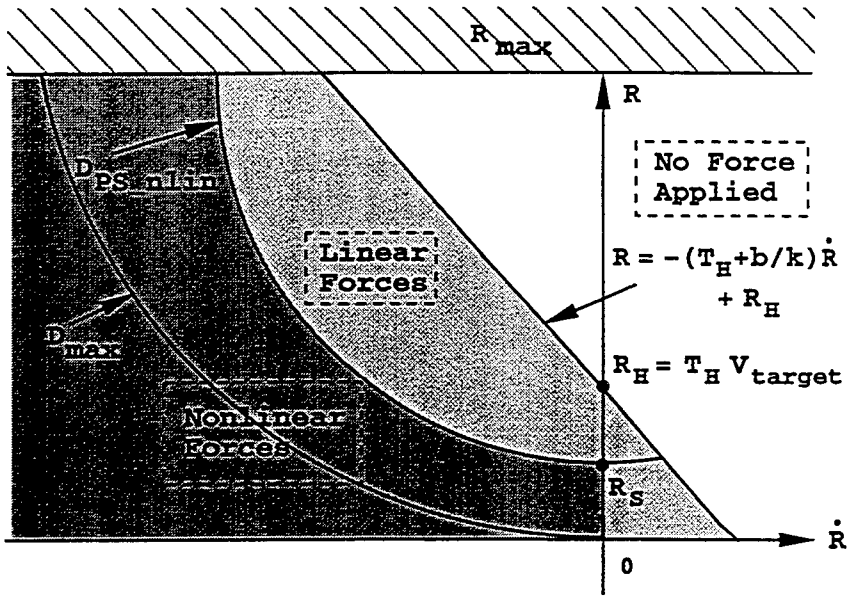


Figure 2.13: Linear and nonlinear longitudinal force zones defined in the R,$\dot{R}$ plane.

In selecting the value of D_{PS_nlin}, one should consider the function of the linear force equations. The linear force equation is used in situations where little or no braking is required. Therefore, the value of (D_{PS_nlin}) should be set to a deceleration level such that the velocity controller can achieve this level with minimal application of the braking system.

Headway control example using nonlinear forces

To allow for a better understanding of the longitudinal impedance controller, let's look at a driving example in which nonlinear forces are applied. In this example the target vehicle is traveling at a constant velocity which is much slower than the host's desired speed. The host approaches the target and slows down to maintain the desired headway.

Again in this example, the response of the controller is evaluated by plotting the R and $\dot{R}$ measures in the R, $\dot{R}$ plane (see Figure 2.14). When the target is initially sensed, $\dot{R} < 0$ and $R = R_{max}$. This point in the phase plane is to the left of the D_{PS_nlin} curve so the nonlinear force is applied immediately. A constant deceleration is applied to the host and the R,$\dot{R}$ points follow a parabolic curve towards the R_H,0 point. Notice that the curve defined by the R,$\dot{R}$ points is not "parallel" with the D_{PS_nlin} curve. This is because the host vehicle is decelerating at a higher rate than D_{PS_nlin}. As the vehicle continues to decelerate, the R,$\dot{R}$ points cross the D_{PS_nlin} curve into the linear force personal space. Then a linear force is applied to continue deceleration, but the system response for the linear forces is typically to slow to drive the system directly to the R_H,0

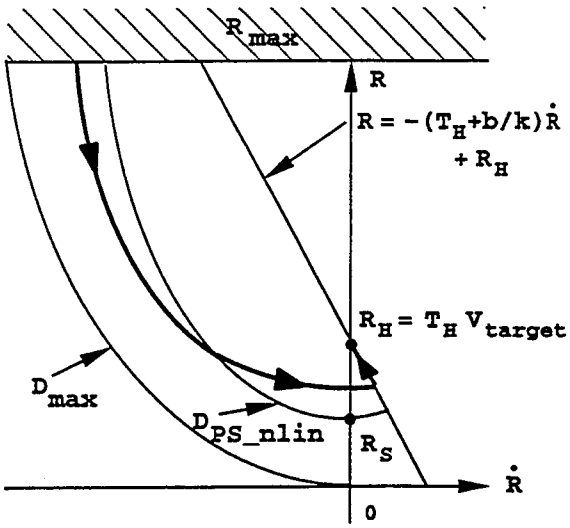


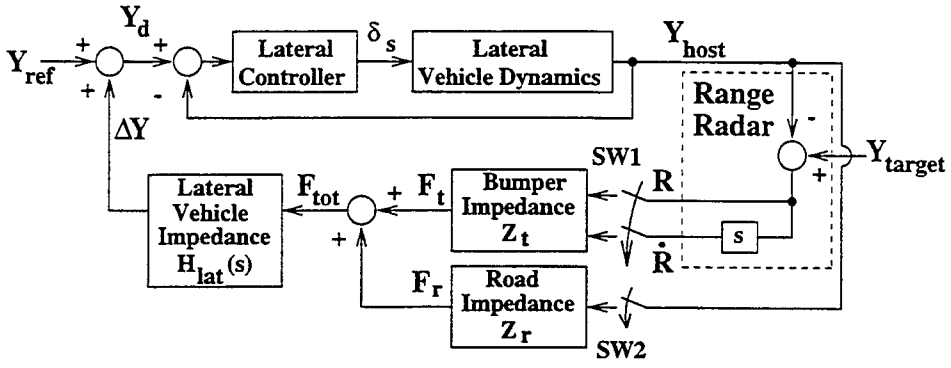
Figure 2.14: R, $\dot{R}$ diagram for headway tracking using nonlinear forces.

point. So the system overshoots the desired headway and drives slower than the target ($\dot{R} > 0$) to allow the headway to increase. As the headway increases, a first order response is followed back to the desired headway. Now the host vehicle stays at this headway until the target vehicle changes lanes or changes velocity.

2.4 Lateral Impedance Control

The next step in the development is to discuss the lateral impedance controller. The lateral impedance controller is implemented following the block diagram structure shown in Figure 2.15. Similar to the longitudinal impedance controller, this controller contains two separate loops. The inner loop is a lateral position controller for the vehicle. This controller is described in appendix B. The outer loop implements the impedance control. For this outer loop, the forces for the targets and the road are each calculated. These forces are then summed and applied to the lateral impedance controller. This generates an offset from the lateral impedance controller. This offset is added to the nominal desired lateral position from the center of the right lane in order to calculate the desired lateral position. The desired position is used as the input to the vehicle's lateral controller.

An impedance control strategy as described above has one major shortcoming: It will not necessarily drive a vehicle in the way a person would. For instance, when humans perform a lane change at highway speeds, the profile of the lane change stays relatively uniform under a variety of circumstances. However, virtual forces in an impedance scheme would typically be based on range and range rate to target vehicles. Based on the forces, the impedance control approach will perform a lane change which will vary for variations in closing rates and vehicle spacing. This is an



Y_{ref} = Operator selected nominal lateral offset from the center of right lane.
ΔY = Lateral offset from the center of the right lane.
Y_d = Desired lateral position for the lateral controller.
δ_s = Commanded steering wheel angle.
Y_{host} = Lateral position of host vehicle in the lane.
Y_{target} = Lateral position of target vehicle in the lane.
s = Laplacian operator.
R = Range to target measured by radar sensor.
$\dot{R}$ = Range rate to target measured by radar sensor.
SW1 = Software switch that is enabled when target is within ROI.
SW2 = Software switch to enable/disable road forces.
Z_t = Virtual impedance associated with space between target and host vehicles.
Z_r = Virtual impedance associated with road.
F_t = Virtual force calculated for target which is applied to host vehicle.
F_r = Virtual force applied by road to host vehicle.
F_{tot} = Total virtual force applied to the host vehicle.
$H_{lat}(S)$ = Lateral vehicle impedance.

Figure 2.15: Lateral impedance control block diagram.

undesirable feature. The problem here is that when people drive, they make decisions to perform or not to perform a lane change. With the impedance-based approach alone, it is difficult to embed these decisions into the force equation. For this reason, the lateral control strategy used here is a combination of impedance control and a heuristic based decision making system.

To integrate the decision making system with the impedance control strategy, the lateral force for the target is broken down into a reflexive and a lane change force. The reflexive force is based on range measures and addresses the task of maintaining lateral inter-vehicle spacing. The lane change force is issued based on a set of heuristics. The decision making system evaluates the traffic configuration and selects the desired lane. If the desired lane is different than the current lane, a lane change force is generated. The force based on the lane change directive and the reflexive force are then summed to result in one virtual target force which is applied to the lateral impedance. It is through this combination of force types that the heuristics are combined with impedance control.

Now that we have a general feel for the structure of the controller, we will go on to discuss the design issues outlined in section 2.1. We will discuss the issues starting with impedance selection and then will discuss the force computation. The personal space is discussed for each type of force in the virtual force generation sections.

2.4.1 Lateral Impedance Selection

The lateral impedance is selected to ensure that the lateral path profile for the host vehicle will be within its safe limits. For instance, given a lateral path profile, the lateral position controller (discussed in Appendix B) will issue steering commands that impose lateral accelerations (with respect to the host, not the road) on the host vehicle. The lateral acceleration with respect to to host should be kept within safe limits to avoid a spin-out or a roll-over condition. Then by understanding the relationship between the given lateral path and the lateral accelerations measured on the host vehicle, safe limits on the lateral path for the vehicle can be determined. The approach here is to select a lateral impedance such that it filters virtual forces into a lateral path which is within these safe limits.

With regard to safety, we are more concerned with the transients in the lateral path than with the absolute lateral position. For this reason, it is more appropriate to define an impedance describing the transformation of the virtual forces into a lateral path velocity (which is the derivative of the original impedance or $H'_{lat}(s)$). We define this impedance as shown in Equation 2.25 such that there is a second order response from the input force to the calculated lateral path velocity.

$$H'_{lat}(s) = \frac{V_{lat}}{F_{lat}} = \frac{b_0}{s^2 + a_0 s + a_1} \tag{2.25}$$

This second order response could be defined such that there is oscillation in the velocity trajectory. However, for the purposes of driver comfort, it is desired that the lateral path velocity is smooth with no oscillation. For this reason, the relationship from force to velocity should be critically damped (Equation 2.26).

$$H'_{lat}(s) = \frac{V_{lat}}{F_{lat}} = \frac{b_0}{(s + c)^2} \tag{2.26}$$

The coefficients for Equation 2.26 can then be selected such that the velocity and acceleration of the lateral path profile are within the safety limits. Note that the output of the lateral impedance block is the absolute offset from the center of the right most lane. Therefore, the lateral impedance for the controller for this system is just the integral of Equation 2.26.

$$H_{lat}(s) = \frac{\Delta Y}{F_{lat}} = \frac{V_{lat}/s}{F_{lat}} = \frac{b_0}{s(s + c)^2} \tag{2.27}$$

To determine the coefficients for the impedance which will keep the lateral path within safe limits, assume a worst case force profile i.e. a step function. The magnitude of the force is some

defined maximum (Equation 2.28).

$$F_{lat}(s) = \frac{F_{max}}{s} \tag{2.28}$$

Using this force function, coefficients for the lateral impedance can be determined in terms of the desired maximum lateral path velocity and accelerations.

First we will determine a maximum lateral velocity in terms of the maximum force and the equation coefficients. Consider the steady state response in velocity to the applied maximum force determined by using the final value theorem (Equation 2.29).

$$V_{lat_{ss}} = \frac{b_0 F_{max}}{c^2} \tag{2.29}$$

For a given step force function, the path has a steady state lateral velocity. Since the system is critically damped, the lateral velocity increases smoothly up to the steady state value with no oscillation or overshoot. This means that the steady state value is also the maximum lateral velocity.

$$V_{lat_{max}} = V_{lat_{ss}} = \frac{b_0 F_{max}}{c^2} \tag{2.30}$$

Determining the maximum lateral path acceleration is a little more involved. The approach is to look at the time response of the lateral path acceleration (Equation 2.31) for the defined maximum force input.

$$A_{lat}(t) = b_0 F_{max} t e^{-ct} \tag{2.31}$$

The maximum acceleration occurs at the point when the derivative of acceleration is zero.

$$A'_{lat}(t) = b_0 F_{max}(e^{-ct} - cte^{-ct}) = 0 \tag{2.32}$$

$$\Rightarrow \qquad t = 1/c$$

From this it is determined that the maximum lateral path acceleration is defined by Equation 2.33.

$$A_{lat_{max}} = A_{lat}(1/c) = \frac{b_o}{c} F_{max} e^{-1} \tag{2.33}$$

The above relationships can be used to solve for the lateral impedance coefficients. These coefficient in terms of the maximum lateral force, velocity and acceleration are as follows:

$$c = \frac{A_{lat_{max}}}{V_{lat_{max}}} e \tag{2.34}$$

$$b_0 = \frac{V_{lat_{max}}}{F_{max}} c^2 \tag{2.35}$$

Using the coefficients determined by these equations means that the lateral velocity and acceleration of the path will stay below the defined maximums ($V_{lat_{max}}$ and $A_{lat_{max}}$, respectively) if the lateral virtual force stays below its defined maximum (F_{max}).

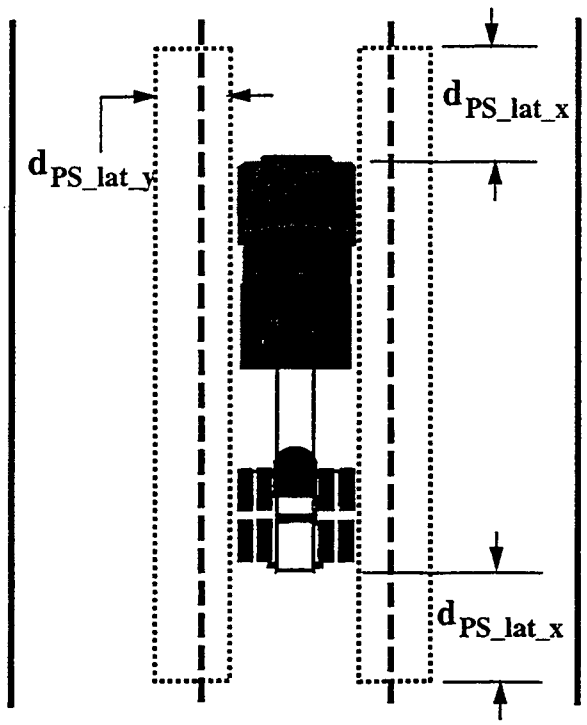


Figure 2.16: Lateral personal space for the host vehicle.

2.4.2 Lateral Reflexive Force Computation: The Lateral Virtual Bumper Dynamics

The reflexive force is associated with the driving task of maintaining lateral inter-vehicle spacing. For this reason, the reflexive force is calculated for vehicles which are in the lateral portion of the personal space (see Figure 2.16). In this section, we define the lateral personal space in detail and then discuss the reflexive force calculation based on a target's position in this personal space.

The lateral personal space is the region immediately to the left and the right sides of the host. The personal space extends to the side of the host a defined distance ($d_{PS\_lat\_y}$). This distance is set such that when a target vehicle of typical width and the host are nominally in the center of adjacent lanes, the target is not within the defined personal space.

The lateral personal space also extends both to the front and to the rear of the vehicle by a distance, $d_{PS\_lat\_x}$ (see Figure 2.16). This distance is extended in order to consider target vehicles which are ahead or behind the host vehicle. $d_{PS\_lat\_x}$ is calculated based on the closing rate or the relative velocity of the sensed targets ($\dot{d}_{target\_x}$) as shown in Equation 2.36.

$$d_{PS\_lat\_x} = \begin{cases} d_{min\_x} - T_{reflex}\dot{d}_{target\_x} & \text{if } \dot{d}_{target\_x} < 0 \\ d_{min\_x} & \text{otherwise} \end{cases} \tag{2.36}$$

where:

T_{reflex} = Relative travel time the side region extends in front and to the rear of host.

d_{min_x} = Minimum distance the side region extends in front and to the rear of host.

$\dot{d}_{target\_x}$ = Longitudinal relative velocity of target.

Extending the personal space in this way gives the lateral inter-vehicle spacing to a target more consideration if the target has a large relative velocity towards the host.

With the personal space defined, we now evaluate the reflexive force calculation. The reflexive force is made up of a lateral component ($F_{lat_{reflex}}$) and a longitudinal component ($F_{lat_{reflex}}$). The lateral component "pushes" the host laterally away from targets which enter the lateral personal space. When a longitudinal component is generated it is added to the other longitudinal forces and applied to the longitudinal feedback loop. This longitudinal force allows for decelerating the host when the lateral spacing to a target becomes too small.

The reflexive force is calculated as linearly increasing from the edge of the personal space to the host. The range rate is not included in this force due to sensor placement issues discussed in Chapter 4. The reflexive force components are as defined by Equations 2.37 and 2.38 and are graphed as shown in Figure 2.17.

$$|F_{lat_{reflex}}| = \begin{cases} 0 & \text{if } d_{target\_y} > d_{PS\_y} \\ F_{lat_{max}} \left(\frac{d_{PS\_y} - d_{target\_y}}{d_{PS\_y} - d_{min\_y}} \right) & \text{if } d_{PS\_y} \geq d_{target\_y} > d_{min\_y} \\ F_{lat_{max}} & \text{if } d_{target\_y} \geq d_{min\_y} \end{cases} \quad (2.37)$$

$$F_{long_{reflex}} = \begin{cases} F_{long_{max}} \left(\frac{.1F_{lat_{max}} + .4F_{lat_{reflex}}}{F_{lat_{max}}} \right) & \text{for } F_{lat_{reflex}} > F_{lat_{max}}/4 \\ 0 & \text{otherwise} \end{cases} \quad (2.38)$$

where:

d_{target_y} = Lateral distance to the target.

d_{PS_y} = Distance the personal space extends from the side of the host vehicle.

d_{min_y} = Minimum lateral distance to target.

The force shown in Equation 2.37 is the magnitude only. This lateral force ($F_{lat_{reflex}}$) is defined to be negative when the target is on the left side of the host and positive when the target is on the right side of the host. The longitudinal force is defined as a function of the lateral force. The parameters of 0.1 and 0.4 in Equation 2.38 are determined experimentally and this equation allows for applying greater longitudinal forces (braking the host) when the lateral reflexive force increases (target encroaches farther into the host's lateral personal space).

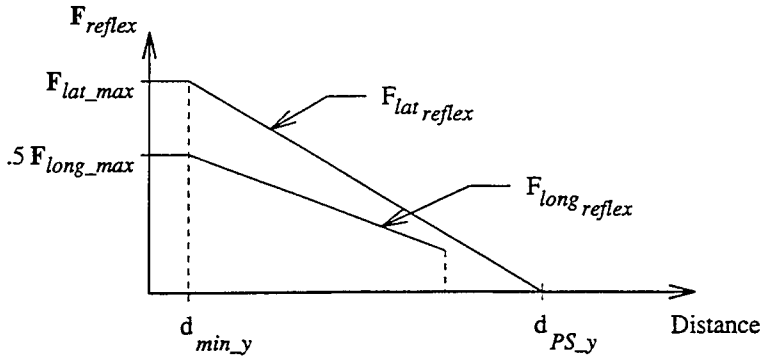


Figure 2.17: Magnitude of the lateral reflexive force equations.

2.4.3 Lane Change Heuristics

The lane change system selects the desired lane and issues forces to perform lane change maneuvers. This system prefers the current lane until there is a target in the host's lane change personal space. When there is a target in this personal space, the desired lane is selected based on the current configuration of the targets. If the desired lane is not the current lane, a lane change force is generated which smoothly transitions the host vehicle to the desired lane. This approach is developed for multiple lane highways and considers both the lane to the right and the left equally when determining the desired lane (i.e. does not prefer passing on the left). In the following discussion, we discuss the desired lane selection method, the lane change maneuver process and then discuss the form of the lane change force.

Selecting the desired lane

The selection of the desired lane can be described in terms of two main steps. First the lanes are evaluated for the existence of a gap (gap existence). If a gap exists, a measure of the size and rate of change in size is calculated in the form of a force (gap force). The lane with the largest gap force is selected as the desired lane. This process is outlined by the flow chart in Figure 2.19.

For determining gap existence, the right and the left lanes are evaluated. The current lane is assumed to have a gap because the host vehicle is already in the lane. A gap exists if there are no vehicles in the side regions of interest for the given lane (see Figure 2.18). In this case we are concerned with any vehicle within the entire region, unlike the reflexive force where the target had to be within the personal space. For this reason, the side regions are considered to be regions of interest (ROI) where knowledge of targets location is required but are not used explicitly in calculation of a force.

If a gap exists in a lane, a gap force (F_{gap}) is calculated for the lane. The gap force is a measure

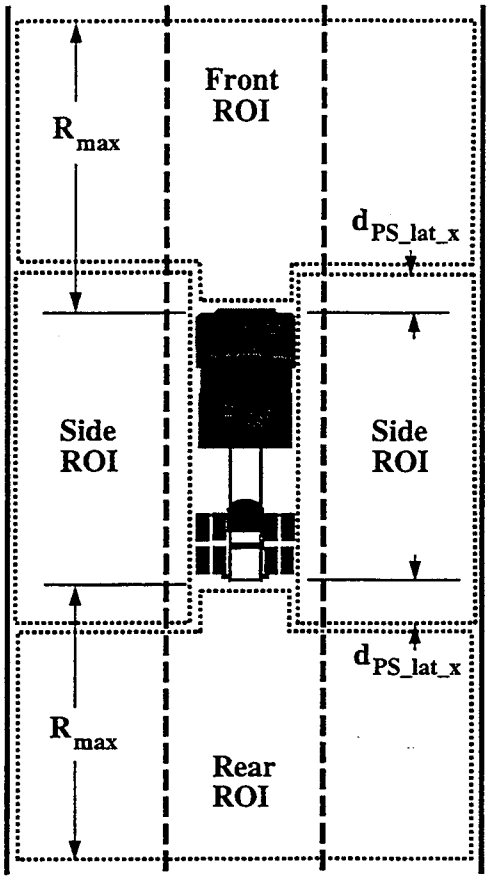


Figure 2.18: Regions of Interest for the lane change system.

of the size and rate of change in size of the existing gap. It is defined by Equation 2.39.

$$F_{gap} = \min \begin{cases} F_{gap\_ahead} \\ F_{gap\_behind} \end{cases} \tag{2.39}$$

where:

F_{gap_ahead} = Gap force associated with target at the front of the gap.

F_{gap_behind} = Gap force associated with target at the rear of the gap.

The F_{gap_ahead} is the longitudinal force which would be applied to the host if it were behind the target in the given lane. The F_{gap_behind} is the longitudinal force which would be applied to the target if it is controlled using the same control strategy and is behind the host vehicle in the same lane. In calculating F_{gap_ahead} and F_{gap_behind} we evaluate the nearest target in the respective lane which is within the maximum sensing range, R_{max}. Then, we define the area in front and to the rear of the host in the right, center and left lanes as the region of interest or ROI (see Figure 2.18). The terminology ROI is used instead of personal space because we are interested in information

about targets in these regions. Measurements of a target in the ROI is used for making a decision while measurements of a target in the PS is used for generating forces.

The F_{gap_ahead} and F_{gap_behind} need some further definition. These forces are determined by the following equations.

$$F_{gap\_ahead} = \begin{cases} F_{long} & \text{if } F_{long} < 0 \\ \dot{R} & \text{if } F_{long} > 0 \\ \dot{R}_{max} & \text{if no target present} \end{cases} \tag{2.40}$$

$$F_{gap\_behind} = \begin{cases} F_{long_t} & \text{if } F_{long_t} < 0 \\ \dot{R} & \text{if } F_{long_t} > 0 \\ \dot{R}_{max} & \text{if no target present} \end{cases} \tag{2.41}$$

where:

$\dot{R}_{max}$ = Maximum measurable range rate.

F_{long_t} = Longitudinal force which would be applied to target under the same control strategy.

From these equations we see that the gap force is set to the longitudinal force if it is negative. These negative longitudinal forces are associated with decelerations. If the longitudinal force is greater than zero, the gap force is associated with the measured $\dot{R}$ which is typically greater than zero for $F_{long} > 0$. If there is no target, the gap force is set to a defined maximum range rate. From this discussion, it is apparent that negative gap forces are associated with undesirable lanes (i.e. host is approaching a target) and positive gap forces are associated with desirable lanes (i.e. target is accelerating away from host or no target in lane).

The gap forces can be thought of as proportional to the risk of a rear-end collision. As the force decreases (larger negative number) the risk of a rear-end collision increases. If the host were in a given lane, F_{gap_ahead} is associated with the risk of rear-ending a target and F_{gap_behind} is associated with the risk of being rear-ended by a target. The gap force for a lane is the minimum of both of these values (see Equation 2.39). In this way, both of these collisions types are weighted equally when selecting the desired lane.

The gap force is calculated using Equation 2.39 for the left and right lane but not for the center lane. The gap force for the center lane is set to $F_{gap_{ahead}}$. This is done because it is difficult to interpret the intentions of a driver of a target vehicle which is approaching the host from behind in an "unsafe" manner. For instance the driver may have a habit of "tail-gating." If the host changes lanes in this case, we may find the vehicle continually changing lanes in heavy traffic. In another situation, a target vehicle may be approaching with a high closing rate with the intention of overtaking the host in the left lane. To avoid being rear-ended, we may want to perform a lane

change but it is not possible to make this decision safely without knowledge of the other drivers intentions. For these reasons, the gap force for the center lane does not consider the targets behind the host.

Using the approach outlined above, the gap forces are determined for each lane which has a gap. As discussed already, F_{gap} is associated with a risk of collision for a lane. The higher the value of F_{gap}, the lower the risk. Therefore, the lane with the largest gap force is selected as the desired lane. The steps involved in the desired lane selection process are summarized in the flow chart shown in Figure 2.19.

Lane change maneuver

Now that the desired lane is selected, it is compared to the current lane. If the current lane is not the desired lane, a lane change maneuver is executed to the desired lane. The steps in the lane change maneuver are outlined by the flow chart shown in Figure 2.20.

The lane change maneuver is performed by a loop which cycles continuously and applies a virtual force until the lane change maneuver has completed. This loop has two basic functions. First it continuously verifies that the desired lane still has a gap and that it is still the best lane. This allows for the host vehicle to abort a lane change if the conditions change and the previously desired lane is no longer the preferred choice. The second function of the loop is to apply the appropriate force to allow for a smooth lane change. Once the host is within the center desired lane by some defined distance, the loop exits and the lane change maneuver is completed.

The first step in the loop of verifying the desired lane is relatively straight-forward and can be understood by examination of the flow chart in Figure 2.20. The lane change force calculation is a little more involved and is discussed in detail.

Lane change force

We have discussed how the desired lane is selected and how the lane change maneuver is performed. Next we need to define the lane change force (F_{LC}) that when applied to the lateral impedance produces the desired lane change profile. The function selected for the lane change force is a pulse function of a magnitude based on the urgency ($F_{urgency}$) of the lane change maneuver (the determination of $F_{urgency}$ is discussed later). The pulse function switches on at the beginning of the lane change maneuver and switches off when the host vehicle is a defined distance from the center of the desired lane. When the pulse is applied to the lateral impedance, the lateral velocity smoothly increases to a steady state velocity and when the pulse switches off, the lateral velocity smoothly decreases to zero. This pulse function applied to our lateral impedance produces a smooth lane

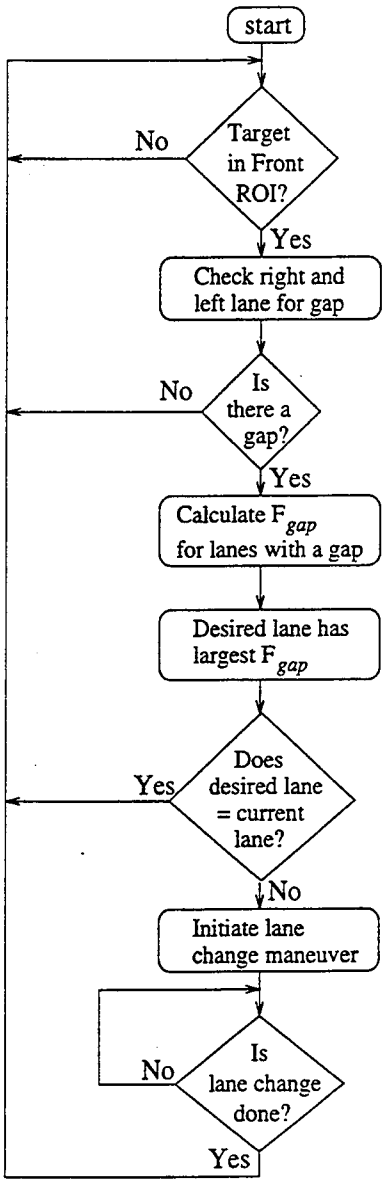


Figure 2.19: Flow chart for selecting the desired lane.

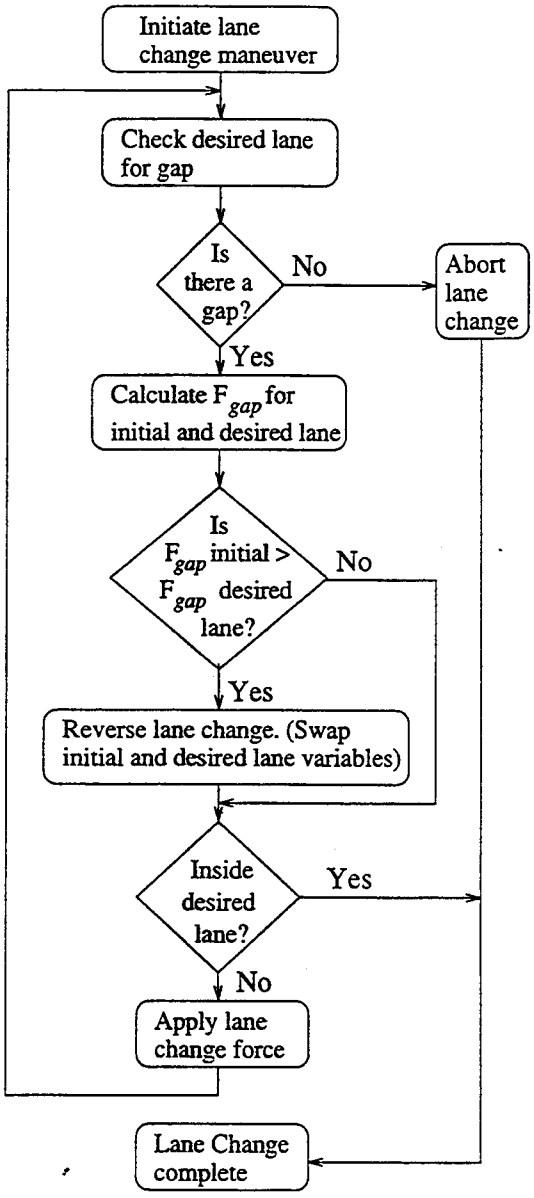


Figure 2.20: Flow chart for performing a lane maneuver.

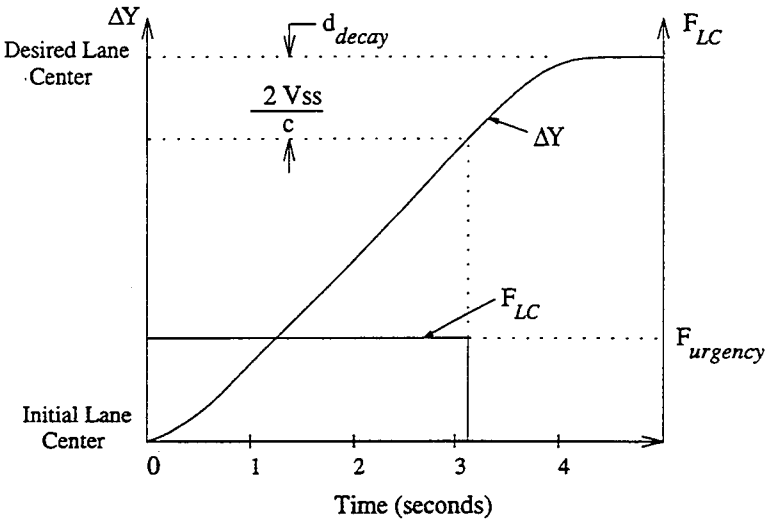


Figure 2.21: Lane change profile for the F_{LC} pulse function.

change profiles as shown by Figure 2.21. The two main design issues for the pulse lane change force are determining when to turn off the force and selecting the appropriate force magnitude ($F_{urgency}$).

First let's discuss when to turn off the lane change force. For the applied force pulse function, after the force function switches to zero, the lateral offset still increases by a certain distance (d_{decay}). Now if the force is turned off when the host is at d_{decay} from the center the desired lane, the lateral offset will smoothly approach and stop at the center of the new lane. Therefore, it makes sense to use d_{decay} as the point to turn the lane change force off.

To determine d_{decay}, the response of the lateral path velocity is evaluated in the time domain. The approach is to first determine the time required for the lateral velocity to decay to zero after the force is turned off. Then this time is used to find the distance traveled in this interval (d_{decay}) by integrating the lateral velocity time response.

Prior to the switch off point for the lateral force, the lateral velocity is at a steady-state ($V_{lat_{ss}}$). The time response of the lateral path velocity (V_{lat}) with no lateral force applied and with an initial velocity of $V_{lat_{ss}}$ is shown in Equation 2.42.

$$V_{lat}(t) = V_{lat_{ss}}(e^{-ct} + cte^{-ct}) \tag{2.42}$$

This system decays exponentially to zero lateral velocity. Such a system will have significantly decayed by five time constants which means the lateral velocity is approximately zero at $t = 5/c$. Now we can determine the distance traveled (d_{decay}) in these five time constants by integrating Equation 2.42 and solving for $t = 5/c$. This leads to the following equation:

$$\Delta Y_{lat}(t) = V_{lat_{ss}}(\frac{2}{c} - \frac{e^{-ct}}{c}(2 + ct)) \tag{2.43}$$

Then solving for $t = 5/c$ leads to:

$$d_{decay} = \Delta Y_{lat}(5/c) = \frac{2}{c} V_{lat_{ss}} \tag{2.44}$$

From Equation 2.44 we see that d_{decay} increases with the lateral velocity. Switching the lane change force off at d_{decay} from the center of the desired lane results in a smooth lane change profile as shown in Figure 2.21.

Next, let's select the magnitude of the lane change force ($F_{urgency}$). The magnitude of the lane change force determines the lateral speed of the lane change maneuver. Consider the maximum lateral velocity derived in section 2.4.1 in Equation 2.30. This can be rewritten to solve for a steady-state velocity in terms of $F_{urgency}$ (see Equation 2.45).

$$V_{lat_{ss}} = \frac{b_0 F_{urgency}}{c^2} \tag{2.45}$$

From this equation it is clear that the lateral velocity will increase with increasing $F_{urgency}$. Therefore, the time for a lane change (T_{LC}) decreases with increasing $F_{urgency}$.

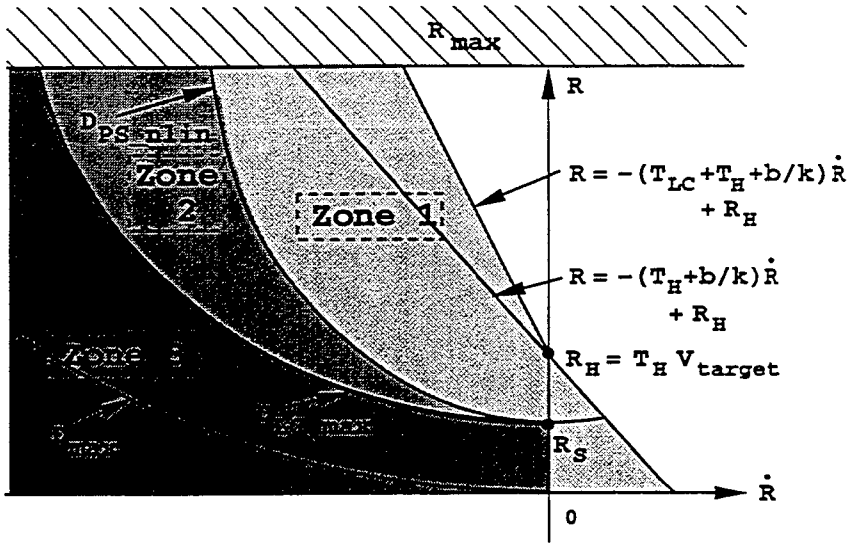
Based on the relationship discussed above, $F_{urgency}$ is set to represent the urgency of a lane change or the danger level of staying in the current lane. Under a wide variety of typical driving conditions, the lane change profile should be constant. But under emergency situations (i.e. stalled target vehicle ahead in lane) the lane change profile should be performed at as high a level that is known to be safe. For these reasons, $F_{urgency}$ is set to a nominal level (F_{LC_nom}) for situations which occur during normal driving and increases to $F_{lat_{max}}$ for emergency situations.

The magnitude of $F_{urgency}$ is set using the R,$\dot{R}$ phase diagram. In this approach, the point defined by range and range rate to the preceding vehicle in the lane is evaluated to see if the R,$\dot{R}$ combination falls in one of three zones in the R,$\dot{R}$ diagram. The three zones are defined which signify typical lane change maneuvers (zone 1), emergency lane change maneuvers (zone 3) and a transition from typical to emergency lane change maneuvers (zone 2). Figure 2.22 shows these zones along with the calculations for $F_{urgency}$. The force stays at a nominal level throughout zone 1. The force increases from $F_{LC_{nom}}$ to $F_{lat_{max}}$ in zone 2 and remains at $F_{lat_{max}}$ throughout zone 3.

The border of zone 1 is defined by the line in Equation 2.46 and defines the personal space for the lane change force.

$$R_{PS_{LC}} = T_H V_{target} - \left(\frac{b}{k} + T_H + T_{LC}\right) \tag{2.46}$$

This line intersects the point R_H,0 just like the line which defines the longitudinal personal space but this line has a steeper slope. The slope is steeper by the time required to perform a lane change (T_{LC}). This means that when the host is approaching a target in a typical driving scenario, the lane change force is applied T_{LC} before the longitudinal controller adjusts the velocity (if a lane is



$$F_{urgency} = \begin{cases} F_{LC_{nom}} & \text{Zone 1} \\ F_{LC_{nom}} + \left(\frac{F_{LC_{max}} - F_{LC_{nom}}}{D_{LC_{max}} - D_{PS\_nlin}}\right)(D_{calc} - D_{PS\_nlin}) & \text{Zone 2} \\ F_{LC_{max}} & \text{Zone 3} \end{cases} \quad (2.47)$$

Figure 2.22: R vs. $\dot{R}$ plane for assignment of $F_{urgency}$.

available for overtaking of course). The result is that a lane change maneuver is completed without any changes in the host velocity.

The magnitudes for $F_{urgency}$ can also be plotted for different measured ranges as shown in Figure 2.23. The ranges at which the force magnitude changes are functions of the relative velocity. Therefore, as a target is approached, $F_{urgency}$ may never get larger than $F_{LC_{nom}}$ even though the range is reducing.

It should be noted that during the lane change maneuver the forces from the road in the area between the current and the desired lane is disabled. This is required for allowing a smooth lane change path. Once the host is within d_{decay} of the center of the new lane, the road forces are enabled.

2.4.4 Virtual Road Force Calculation

There is also a virtual force generated based on the position of the host with respect to the lane. This force has the effect of "pushing" the host away from dangerous areas of the road (i.e. shoulders or the lane divider on a two-way road) towards the center of the lane. The force developed here has both a damping and a stiffness term that act based on the lateral position of the host.

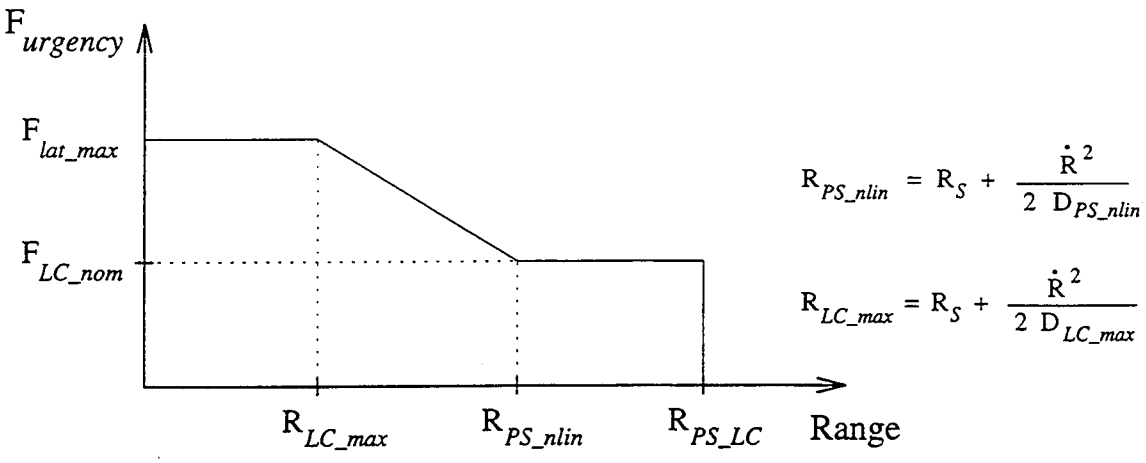


Figure 2.23: Magnitude of $F_{urgency}$ as a function of the measured range.

The stiffness term generates a linear force based on the offset of the host from the center of the lane. This force profile is shown for a two-lane highway in Figure 2.24. The forces of this profile increase to slightly greater than the maximum lateral target force. This is required for the road force to dominate the target force and keep the vehicle on the road. Also since crossing the center-line is not as dangerous as driving on the shoulder, the force increases more gradually (smaller k value) when moving towards the center-line than it does towards shoulder. In this profile the force is only zero at the center of each of the lanes. This was done during the development phase to assist in readily evaluating system performance (i.e. steady-state oscillation). Practically speaking, it is more desireable to have a flat region of zero force or to have a rounded section at the base of each lane. This allows the vehicle to wander slightly in the lane (like humans do). For large scale implementations of this approach the wandering in the lane would be required to reduce wear on the roadways.

There is also a damping term in the road force to reduce oscillation of the vehicle in the lane. This term is tuned using a nominal lane change maneuver. To produce the desired lane change profile, a pulse function is applied to the lateral impedance which is turned off when the host is within d_{decay} of the center of the desired lane (see Figure 2.21). The road force is disabled during the lane change but is enabled again when the lane change force is switched off. Since the offset from the lateral impedance will "coast" to the center of the desired lane without any forces applied, it is desired that the road force is approximately zero during this period. This is accomplished by appropriately selecting a damping coefficient properly.

Consider the the road force shown in Equation 2.48.

$$F_{road} = kY_{error} + bV_{lat} \tag{2.48}$$

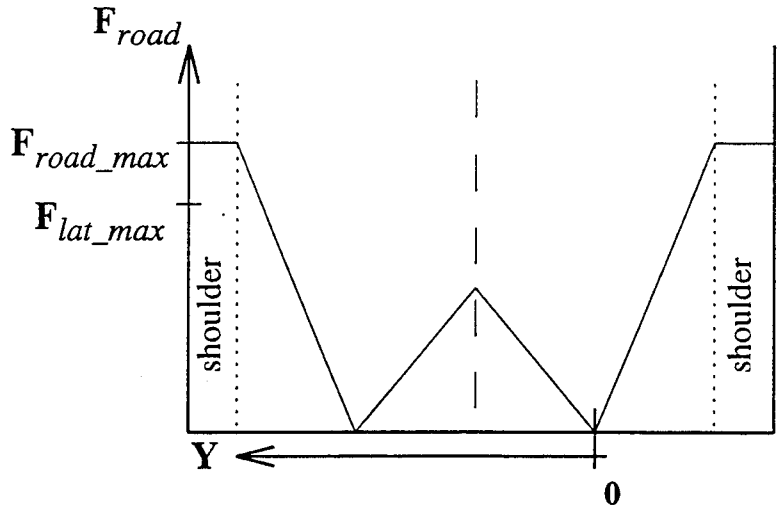


Figure 2.24: Virtual road force profile for a two-lane highway.

where:

Y_{error} = Error of host from center of desired lane.

k = Stiffness coefficient.

b = Damping coefficient.

From the discussion above, the damping coefficient should be selected such that the force is zero when $Y_{error} = d_{decay}$ (d_{decay} is defined in Equation 2.44). Also, toward the end of a lane change maneuver, prior to switching off F_{LC}, $V_{lat} = V_{lat_{ss}}$. Then setting $Y_{error} = d_{decay}$ and $V_{lat} = V_{lat_{ss}}$ leads to the following road force equation.

$$F_{road} = k\left(\frac{2}{c}V_{lat_{ss}}\right) + bV_{lat_{ss}} \tag{2.49}$$

Then solving for the damping coefficient which makes the force zero, leads to the following result.

$$b = -\frac{2k}{c} \tag{2.50}$$

Using this damping coefficient allows the road force to be properly matched with the lane change forces and effectively reduces vehicle oscillation in the lane.

2.5 Summary

In this chapter the theory behind the virtual bumper concept has been developed. The approach is based on two separate impedance loops which individually control the host vehicle longitudinally and laterally. The longitudinal loop is developed which allows for smoothly controlling the headway

spacing between the host and target vehicle. The lateral controller is a combination of an impedance control loop and a heuristic based decision making system. It is capable of shifting in the lanes to avoid obstacles as well as performing lane changes. Addition of the heuristics allows for vehicle control which is similar to how a human would control the vehicle. The two controllers are matched together such that the vehicle is capable of over-taking target vehicles without decelerating in typical driving conditions. Safe control levels are also considered for operation of vehicle in emergency situations.

Chapter 3

Implementation of the Virtual Bumper

3.1 Overview

In the previous chapter we outlined the development of the virtual bumper control strategy for collision avoidance. In this chapter we will discuss the process of implementing the virtual bumper. Within this discussion we detail how controller coefficients are selected to consider safe and smooth operation of the vehicle. For this development, the dynamic capabilities of a Navistar semi-tractor are used. Also, we address some modifications in the control approach required to ensure the practicality of the implementation. Implementation issues related to the sensor placement will not be addressed here but rather we discuss these issues in the chapter on sensor placement (see Chapter 4). As in the previous chapter, we will discuss implementation starting with the longitudinal controller followed by the lateral controller.

3.2 Longitudinal Impedance Control Implementation

In this section, we discuss the implementation process for the longitudinal impedance controller. First, we discuss modifications to the controller which are required for addressing some implementation issues. Then with these modifications in place, we discuss the parameter selection process. For selecting these parameters, we first look at the host vehicle's dynamic limits and then select the impedance and force equation coefficients with these limits in mind. The controller is then tuned to meet the desired response.

3.2.1 Implementation issues

There are two main issues which must be considered for practical implementation of longitudinal impedance control. The first issue is that the velocity controller has a tracking error which degrades the impedance controller performance. We have addressed this issue by modifying the virtual force calculation method. The second issue is that under some typical driving situations, the controller will keep the vehicle velocity below the operator set velocity when no target is present. We have addressed this issue with a modification to the previous control block diagram.

Virtual force modification

The primary implementation issue for the longitudinal impedance controller is addressing the tracking error in the velocity controller. For impedance control to work properly, a high bandwidth

inner control loop is required [Jossi95b][Jossi95a]. When the impedance controller response was derived in section 2.3.2, it was assumed that the velocity controller bandwidth would be sufficiently high so that the tracking error would remain small. However, the velocity controller does not track the commanded velocity without error. This means that the predicted headway response will not be achieved without modifying the impedance controller to compensate for the tracking error. The required compensation has been implemented by modifying the virtual force equations. Here we will first discuss the modification for the linear force and then discuss the nonlinear force modification.

The linear force is used when lower deceleration levels are required to perform headway tracking. With the linear force (see Equation 2.15 repeated here in Equation 3.1) it is desired that the measured headway approaches the desired headway with a smooth first order response.

$$F_{long\_lin} = b\dot{R} + k(R - T_H V_{host}) \tag{3.1}$$

However, the tracking error of the velocity controller degrades the system performance and causes oscillation and over-shoot in the system response.

In order to eliminate the oscillations and overshoot, predictive terms are added to our control law. Here we replace the terms, R, $\dot{R}$ and V_{host} in the force equation with the predicted values described in equations 3.2, 3.3 and 3.4.

$$R_{pred} = R + \dot{R}T - \dot{V}_{host}T^2 \tag{3.2}$$

$$\dot{R}_{pred} = \dot{R} - \dot{V}_{host}T \tag{3.3}$$

$$V_{host_{pred}} = V_{host} + \dot{V}_{host}T \tag{3.4}$$

where:

T = Time constant matched to the velocity controller.

Substituting the predicted values into force Equation 3.1 leads to a modified force equation (F'_{long_lin}) shown in Equation 3.5.

$$F'_{long\_lin} = (bT + (T^2 + T_H T)k)\dot{V}_{host} + (b + kT)\dot{R} + kR_{error} \tag{3.5}$$

With the selected b and k coefficients (see section 3.2.2) the value of T is increased until the oscillation and over-shoot is eliminated from the headway response.

Using the predictive equations for range, range rate and host velocity also changes the personal space. In section 2.3.2, the boundary of the longitudinal personal space is defined as the range at which the linear force (Equation 3.1) is zero. To find the personal space limit when using the predicted values, we set force Equation 3.5 to zero and solve for the range. Assuming that the

initial accelerations of the host vehicle is zero leads to Equation 3.6.

$$R_{PS\_lin} = T_H V_{host} - \left(\frac{b}{k} + T\right)\dot{R} \tag{3.6}$$

From this we see that adding the predictive values for the linear force equation increases the longitudinal personal space by $T\dot{R}$.

To demonstrate the requirement for the predictive terms, we evaluate the system response in the $(R,\dot{R})$ phase plane. In the headway tracking example, the host vehicle approaches a slower target vehicle from behind and the host adjusts its velocity to maintain the desired headway. The initial velocity of the host is 25 m/s (55.9 mph) and the target travels at a constant velocity of 15 m/s (33.6 mph). The virtual force based on the predictive terms (Equation 3.5) is used and we look at the headway response for variations in T from 0 to 2.0 seconds (see Figures 3.1 through 3.4). In the phase diagrams there is both a switching line and a desired response line. The switching line defines the personal space and a virtual force is applied for all $(R,\dot{R})$ points below this line. The desired response line is defined by the selected b and k coefficients and is the approximate trajectory the $(R,\dot{R})$ points should follow. (The selection of the b and k coefficients is discussed in section 3.2.2).

The $(R,\dot{R})$ plots of Figures 3.1 through 3.4 clearly show the requirement for the predictive term. When $T = 0$ the force equation reduces to that without the predictive terms (see Equation 3.1). With this force, the response oscillates around the switching line as it approaches the desired headway as shown in Figure 3.1. Then the headway continuously oscillates around the desired headway which is shown by the circles centered near the R-axis. Increasing the predictive coefficient to $T = 1.0$ eliminates the oscillation and over-shoot but the response does not follow a straight line to the desired headway (see Figure 3.2). Increasing T further to $T = 1.5$ improves the response but the trajectory in the $(R,\dot{R})$ still deviates slightly from a straight line as it approaches the R-axis (see Figure 3.3). Increasing T to $T = 2.0$ eliminates the slight deviations and results in the desired response as shown in Figure 3.4.

The impedance controller also needs to be compensated for the error that occurs when tracking the commanded velocity when using nonlinear forces. For the nonlinear forces, we are not concerned with eliminating over-shoot or oscillation. This is because the nonlinear forces are never used during the final portion of headway tracking when the desired headway is approached. In compensating for the tracking error, however, we are concerned with insuring a safe response for the nonlinear forces.

The nonlinear forces are based on a desired deceleration level (see section 2.3.2). In the case of a vehicle stopped in front of the host, these deceleration levels must be tracked in order to avoid collision. The tracking error in the velocity controller means that the vehicle is not being decelerated

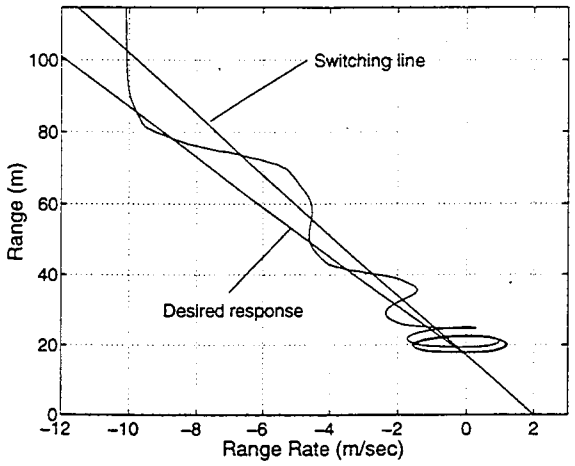


Figure 3.1: R,$\dot{R}$ plot with $b = 2.132$, $k = .232$ and $T = 0$ (no predictive terms).

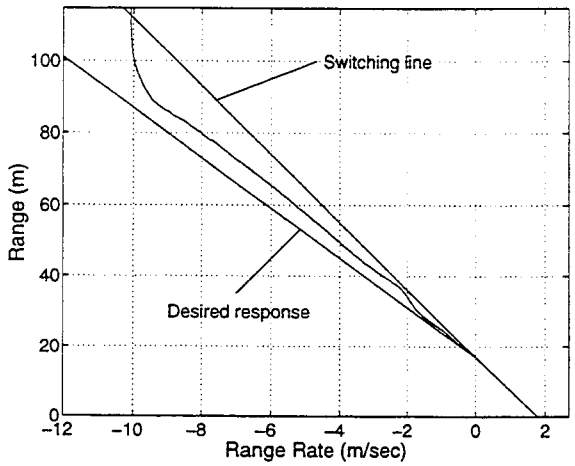


Figure 3.2: R,$\dot{R}$ plot with $b = 2.132$, $k = .232$ and $T = 1$.

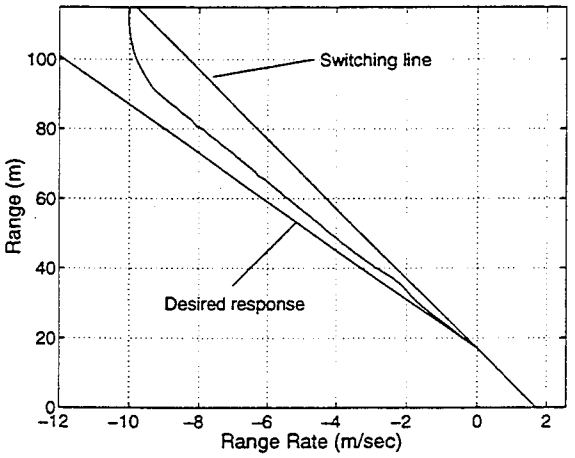


Figure 3.3: R,$\dot{R}$ plot with $b = 2.132$, $k = .232$ and $T = 1.5$.

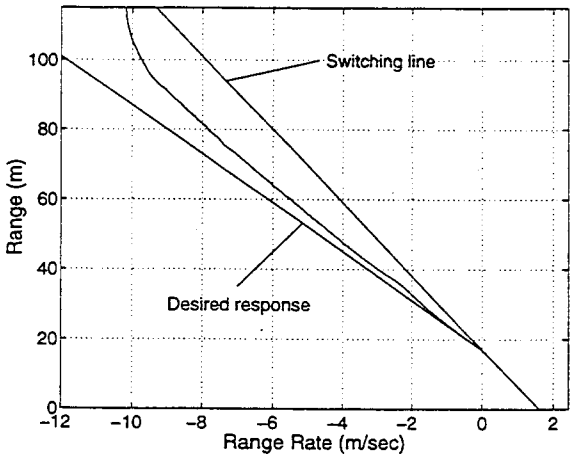


Figure 3.4: R,$\dot{R}$ plot with $b = 2.132$, $k = .232$ and $T = 2$.

fast enough to avoid a collision. Therefore, the commanded deceleration levels must be increased to match with the tracking error of the velocity controller.

The nonlinear forces are determined using Equation 3.7 (see section 2.3.2 for a detailed explanation of this equation).

$$F_{long\_nl} = \frac{M\dot{R}^2}{2(R - R_{scaling})} \tag{3.7}$$

To compensate for the error in tracking velocity the range, R, is replaced with a predicted range, R'_{pred}, as shown in Equation 3.8.

$$R'_{pred} = R + \dot{R}T \tag{3.8}$$

For the nonlinear force term, the host acceleration ($\dot{V}_{host}$) is not considered in the predictive term.

That is because the $\dot{V}_{host}$ acts to reduce the forces in order to reduce oscillation and over-shoot. However, for the nonlinear forces, we do not need to compensate for over-shoot and oscillation so this term is not needed. Then substituting Equation 3.8 into Equation 3.7 results in the following force equation:

$$F'_{long\_nl} = \frac{M\dot{R}^2}{2(R + \dot{R}T - R_{scaling})} \tag{3.9}$$

Using the same T value tuned for the linear force equations results in proper compensation of the nonlinear force equations which allows for safe vehicle control.

Again to demonstrate the requirement of the predictive term, we evaluate the system response in the $(R,\dot{R})$ phase plane. In the situation tested, the host (with an initial velocity of 25 m/s = 55.9 mph)approaches a stopped car in the lane and must decelerate to avoid collision. The nonlinear force applied uses the predictive term (Equation 3.9) and the response is tested for $T = 0$ and $T = 2$ as shown in Figure 3.5. In this figure, in addition to the switching line described earlier, there is a D_{PS_nl} curve and a D_{max} curve. The D_{PS_nl} is a deceleration level which describes a curve in the $(R,\dot{R})$ diagram. For all points below this curve, the nonlinear force is applied. D_{max} is the maximum deceleration capability of the vehicle and its curve defines safe points in the $(R,\dot{R})$ plane. The values of D_{PS_nl} and D_{max} are defined in section 3.2.2.

Figure 3.5 demonstrates the requirement for the predictive term in the force equation. When $T = 0$, the deceleration levels are too small and the host collides with the target traveling approximately $5m/s$ (11.2 mph). However, when $T = 2.0$ seconds, the deceleration levels are high enough for safe operation. Nonlinear forces are used until the host is traveling approximately 2 m/s (4.5 mph) and then a linear force is used to bring the vehicle to a complete stop.

Block diagram modification

Now let's consider how the system responds in some typical driving scenarios. The longitudinal impedance controller discussed has been developed for scenarios in which the host approaches another vehicle from behind and tracks its velocity. However, in everyday driving, the host vehicle may be tracking a target's velocity and then, due to a target or the host vehicle changing lanes, there will no longer be a target in front of the host. The current controller design described in Chapter 2 does not accelerate the vehicle back to V_{ref}.

To allow for accelerating the host vehicle back to the reference velocity, the longitudinal impedance controller is modified to the block diagram structure shown in Figure 3.6. This modification introduces a road force with an additional switch. The road force is an acceleration profile for the host velocity that accelerates the host vehicle to V_{ref}. The two switches are activated based on a target being located in the personal space. If there is a target in the personal space,

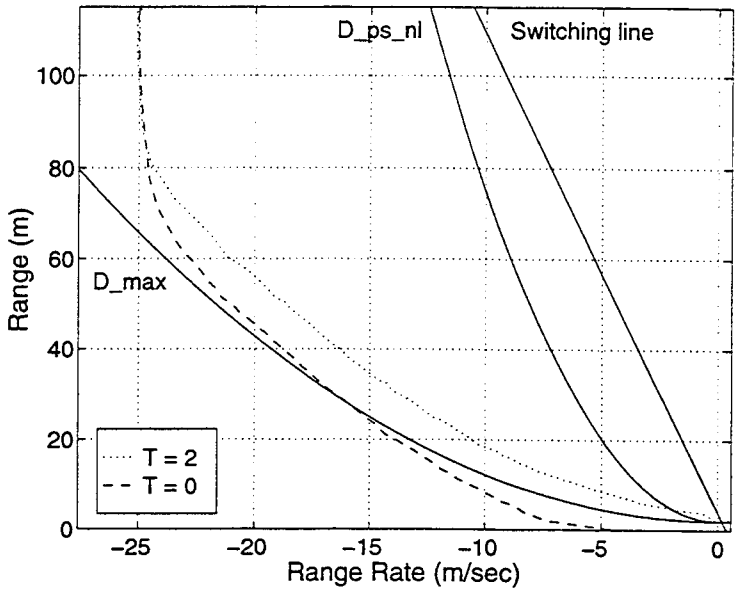


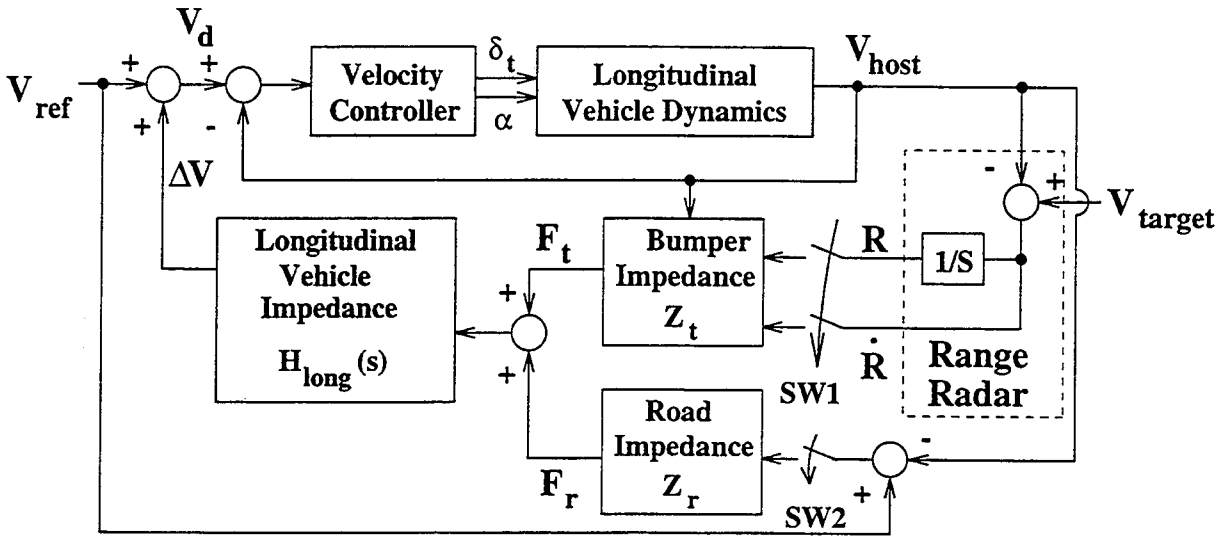
Figure 3.5: $(R,\dot{R})$ plot showing effect of nonlinear force with and without the predictive compensation during emergency braking.

SW1 is closed and SW2 is open. If there is no target in the personal space and ΔV is less than zero (which means $V_d < V_{ref}$), SW1 is open and SW2 is closed. In all other scenarios, both switches are open. This modification allows the controller to respond properly to changes in the host vehicle's personal space.

3.2.2 Parameter selection

For selection of the parameters for the impedance controller, we first evaluate the dynamic limits of the vehicle. For the longitudinal direction, the vehicle dynamic limits are the acceleration and braking capabilities. With this design, since acceleration commands are not issued to avoid collision, the acceleration limit is not a concern. However, since the brakes are applied for collision avoidance, determining the braking limits for the host is critical.

For a given level of brake application the host vehicle decelerates at approximately a constant level of deceleration according to Gillespie [Gillespie92]. The question is then, what is the deceleration level (or g-level) of the semi-tractor under full brake application. Several field tests have been performed to evaluate just that question [Reed87][Radlinski86][Navin86][Kempf92]. With conventional braking systems, deceleration levels varied greatly for the tractors with and without trailers and with various loading. This is due the fact that for optimal braking performance each wheel should be braking at a level based on its loading. However, the truck used in this design, a Navistar tractor, is equipped with ABS braking and is capable of adjusting braking levels



V_{ref}	= Operator selected nominal velocity.
ΔV	= Offset in desired velocity calculated by impedance controller.
V_d	= Desired velocity for velocity controller.
δ_t	= Commanded throttle position.
α	= Commanded brake position.
V_{host}	= Host vehicle velocity.
V_{target}	= Target vehicle velocity.
R	= Range to target measured by radar sensor.
R_d	= Range rate to target measured by radar sensor.
SW1	= Software switch that is enabled when target is within ROI.
SW2	= Software switch that is enabled to accelerate host to V_{ref}.
Z_t	= Virtual impedance associated with space between target and host vehicles.
Z_r	= Virtual impedance associated with road.
F_t	= Virtual force calculated for target which is applied to host vehicle.
F_r	= Virtual force applied by road to host vehicle.
F_{tot}	= Total virtual force applied to the host vehicle.
$H_{long}(S)$	= Longitudinal impedance controller.

Figure 3.6: Longitudinal impedance control block diagram with road force.

for each wheel. Kempf at the Transportation Research Center (TRC) tested a Navistar tractor, similarly equipped with ABS brakes, along with several other semi-tractors and reported the results in [Kempf92]. Stopping distance tests were performed from 60 mph for the Navistar tractor with and without a trailer (the trailer has no braking capability) on three types of road surfaces. The deceleration capabilities ranged from .64 g's to .45 g's. All of the lower g-level braking capabilities were with the trailer attached and it is believed that higher level deceleration would be achieved with a typical trailer with brakes. The lowest g-level in all the tests for the tractor alone was .56 g's. Therefore, for our evaluation a maximum g-level capability is conservatively set at .5 g's for the braking system.

Longitudinal impedance coefficients

Now that we have defined the performance limits of the semi-tractor, the next step is to select the coefficients for the impedance and force equations such that the controller considers these limits. From the longitudinal impedance form, shown in Equation 2.13, the only coefficient is the mass. For this implementation, the mass is set to the mass of the semi-tractor (mass is defined in Appendix A.2). This now leaves only the coefficient definition for the linear and nonlinear longitudinal forces.

Linear force parameters

The linear force is applied to the longitudinal impedance in driving situations where minimal or no braking is required. In these situations it is desired that:

1) The desired headway is approached smoothly with no overshoot.
2) The commanded response is slow enough to be tracked with minimal or no braking.

The first requirement can be met if the system has approximately a first order response. The second requirement can be met if the time constant of the system is selected large enough that the response can be tracked without significant braking.

Our headway response developed in section 2.3.2 is second order and is based on the following characteristic equation:

$$s^2 + 2\zeta\omega_n s + \omega_n^2 \tag{3.10}$$

where:

$$\frac{b}{m} = 2\zeta\omega_n \tag{3.11}$$

$$\frac{k}{m} = \omega_n^2 \tag{3.12}$$

ζ	ω_n	p_1	p_2	b	k	T	b/k	$b/k+T$
—	(rad/sec)	—	—	$(Nsec/m)$	(N/m)	(sec)	(sec)	(sec)
1.3	0.304	0.143	0.648	0.790	0.092	2.7	8.59	11.29
1.7	0.439	0.143	1.350	1.493	0.193	2.4	7.74	10.13
2.0	0.533	0.143	1.990	2.132	0.284	2.0	7.51	9.51
3.0	0.833	0.143	4.855	5.000	0.694	1.5	7.20	8.70

Table 3.1: Controller coefficients for variations in ζ with a dominant pole of $p_1 = .143$ seconds.

The response, however, can be made to approximate a first order response. This is done by setting the damping (b) and stiffness (k) coefficients such that the system is over-damped (the m coefficient is already set to the mass of the vehicle). With an over-damped system , the characteristic equation is written in terms of two real, negative poles (see Equation 3.13) and the system follows a first order response determined by the dominant pole (p_1).

$$(s+p_1)(s+p_2) \tag{3.13}$$

where:

$$p_1 = -\omega_n\left(\zeta - \sqrt{\zeta^2-1}\right) \tag{3.14}$$

$$p_2 = -\omega_n\left(\zeta + \sqrt{\zeta^2-1}\right) \tag{3.15}$$

The issue is then how over-damped should the system be (how large should we set ζ). In the following discussion we describe how control parameters are set and demonstrate the effect of ζ on the system response.

The first step in setting the controller coefficients is selecting the time constant (τ) of the desired response. Here the time constant is set to $\tau = 7.0$ seconds. This is a smaller time constant than that used by Fancher [Fancher94c] in designing his headway controller (he selected $\tau = 12$ seconds for his desired first order response). The smaller time constant translates into a faster headway response. The faster response is achievable because our controller applies braking where as Fancher's controller only actuates the throttle. The time constant is used to set the dominant pole to $p_1 = 1/\tau = 1/7 = .143$.

The second step is to select a value of ζ. Since we haven't determined an appropriate value, we demonstrate the effect of ζ by selecting four different values as shown in Table 3.1. The dominant pole and ζ are then used to solve for ω_n, p_2, b and k.

With the controller coefficients (b and k) determined, the predictive coefficient (T) is set. The approach is to use the predictive force (Equation 3.5) with the personal space defined by Equation 3.6 when implementing the controller. Then, the value of the predictive term is set by increasing T

from the zero until the headway response is smooth with no overshoot. The tuned values for T for the different values of ζ are shown in Table 3.1.

Before we discuss the effects of the various ζ values, let's take a moment to better define the personal space. The personal space is defined by the following equation:

$$R_{PS} = R_H - \left(\frac{b}{k} + T_H + T\right)\dot{R} \tag{3.16}$$

where:

$$R_H = T_H V_{target} + R_{H_0} \tag{3.17}$$

and:

T_H = Desired headway time.

R_{H_0}= Desired headway at zero velocity.

To define the personal space, we must select the appropriate values for T_H and R_{H_0}. Fancher has selected a headway time for his headway controller of 2.0 seconds [Fancher94c]. In our system, a smaller headway time can be allowed because of brake application. Also, remember from section 2.4.3 that the personal space for the lane change maneuver is extended from the longitudinal personal space which is partially defined by T_H. This means a smaller T_H reduces the personal space for the lane change maneuver. For these reasons a headway time of one second is used. The parameter R_{H_0} is set to 2.0 meters to specify the desired headway when the vehicle is stopped in traffic.

With the personal space defined, we can now look at the effect of ζ on the system response. The driving situation that is evaluated is a target vehicle which is moving slower than the host vehicle. The host approaches the target and then tracks the target's velocity to maintain the desired headway. the initial velocity for the host is $V_{host} = 25\ m/s$ (55.9 mph) and the target has a constant velocity throughout the simulation of 18 m/s (40.3 mph). The desired final headway is 20 meters. We evaluate the effect of ζ by plotting the response in the $(R,\dot{R})$ plane as well as plotting the time response of some key parameters.

The $(R,\dot{R})$ phase plots for increasing ζ are shown in Figures 3.7 through 3.10. On each of these diagrams there is a switching line and a desired response line. The switching line defines the personal space and a force is applied for all points below this line. This switching line is determined by Equation 3.16. The desired response line is the line we expect the measured $(R,\dot{R})$ points to follow based on the selected b and k coefficients. This line intersects the R-axis at the desired headway, R_H and has a slope $-1/p_1$. The slope of the line is associated with the time constant of the response.

For each of the $(R,\dot{R})$ plots, the response starts at the $(R_{max},-7)$ point in the plane. All of the plots have a region where the trajectory follows a parabolic curve when the force is first applied. Then after some deceleration, the trajectory follows a straight line to the desired headway at $(R_H,0)$. The straight line portion is slightly off of the desired response line due to the change in dynamics caused by the predictive term, T. the parabolic section signifies a second order response which is caused be the second pole, p_2. As ζ increases, the effect of the second pole becomes smaller and the parabolic section of the trajectory is reduced. This means that as ζ increases a larger portion of the response follows a first order response signified by the line in this plane. Also notice that as ζ increases, the slope of the switching line reduces in magnitude. This occurs because smaller predictive coefficients (T) are needed for increasing ζ (see Table 3.1). the end result is that the personal space is reduced as ζ increases. From the $(R,\dot{R})$ plots it appears that the larger the value for ζ, the more desireable the response.

The trade-off of increasing ζ is observed from the time domain plots shown in Figure 3.11 through 3.14. In these plots we show the time response for R, V_{host}, host acceleration level $(\dot{V}_{host})$, braking level (0.0 is off, 1.0 is completely engaged) and F_{long_in}. These plots show that the system response varies little for the R, V_{host}, $\dot{V}_{host}$ and the braking level. However, as ζ increases, the commanded force changes from being smooth to varying sharply. This increase in variation in force is due to larger controller gain coefficients as ζ increases (see Table 3.1). The increased gains also increase the effect of noise on the system response. Therefore, to allow the system to be robust to noise, a small ζ is desired.

Here we see that there is a trade-off in selecting ζ. A large ζ is desired for a small personal space and a first order response while a small ζ is desired for a noise tolerant design.

Based on the plots described above, $\zeta = 2.0$ is selected for this design. As ζ increases, the personal space reduces and the system approaches a first order response. For this reason a large value of ζ is desired. However, when $\zeta = 3.0$ the force exibits sharp variations. These variations will make the system less stable when sensor noise is introduced. The value of $\zeta = 2.0$ shows significantly less variation in the force value and will be affected less by noise. For these reasons, a value of $\zeta = 2.0$ is chosen.

Nonlinear force parameters

Now that the linear force equation parameters are fully defined, let's define the nonlinear force parameters. The nonlinear force equation is defined in section 2.3.2. Substituting the predictive

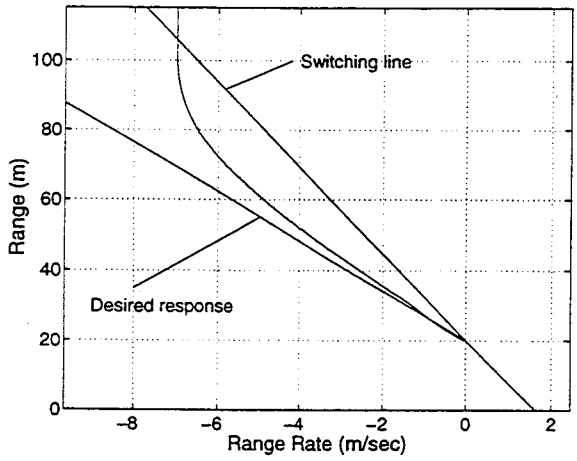


Figure 3.7: R,$\dot{R}$ plot dominant pole of .2 with $\zeta = 1.3$.

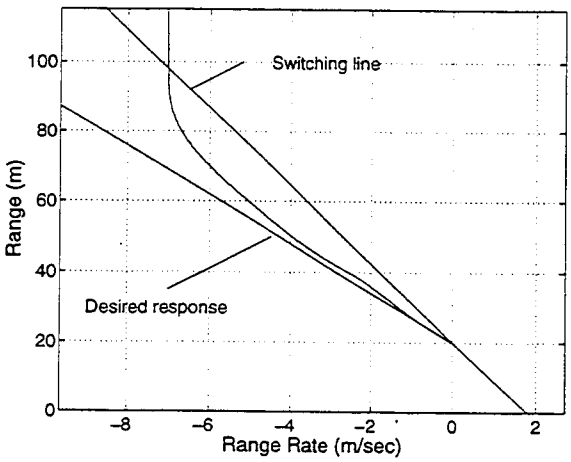


Figure 3.8: R,$\dot{R}$ plot dominant pole of .2 with $\zeta = 1.7$.

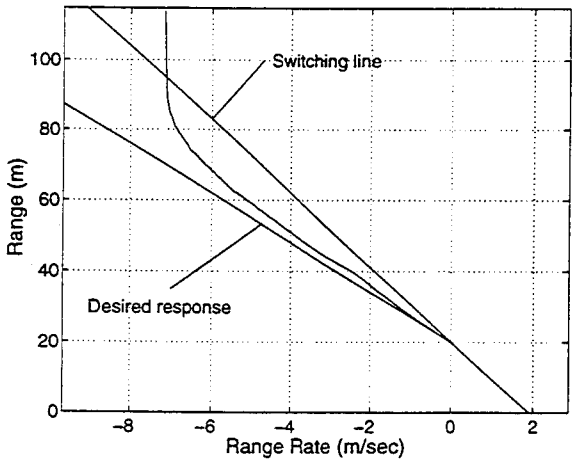


Figure 3.9: R,$\dot{R}$ plot dominant pole of .2 with $\zeta = 2.0$.

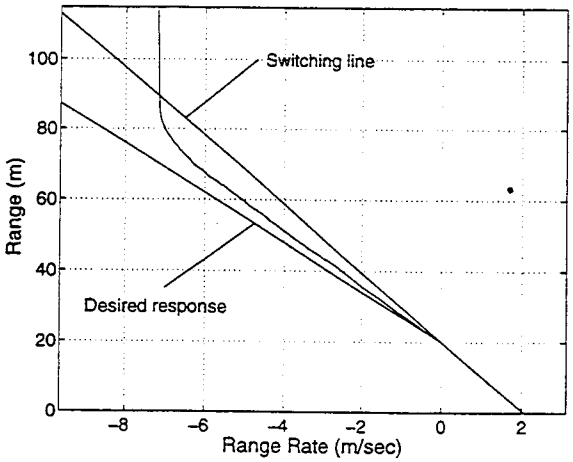


Figure 3.10: R,$\dot{R}$ plot dominant pole of .2 with $\zeta = 3.0$.

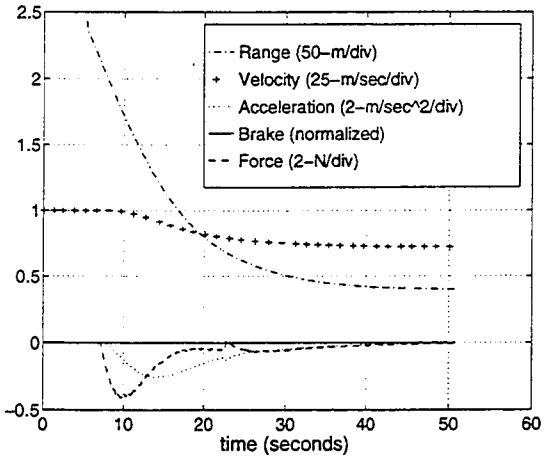


Figure 3.11: Time domain plot with dominant pole of .2 with $\zeta = 1.3$.

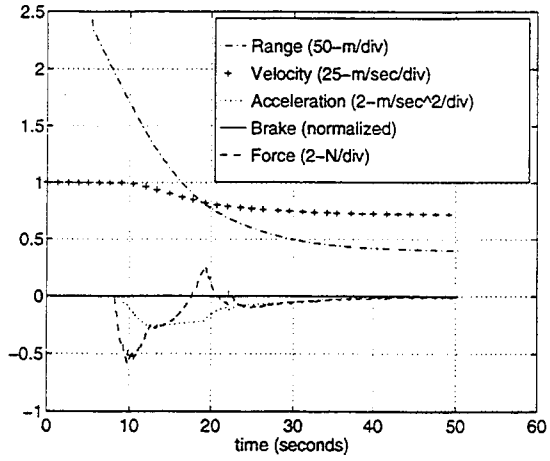


Figure 3.12: Time domain plot with dominant pole of .2 with $\zeta = 1.7$.

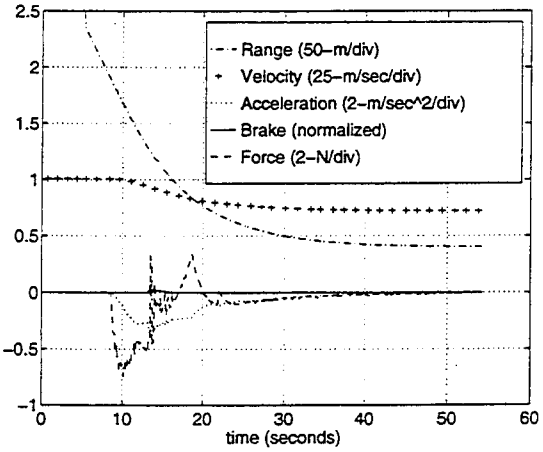


Figure 3.13: Time domain plot with dominant pole of .2 with $\zeta = 2.0$.

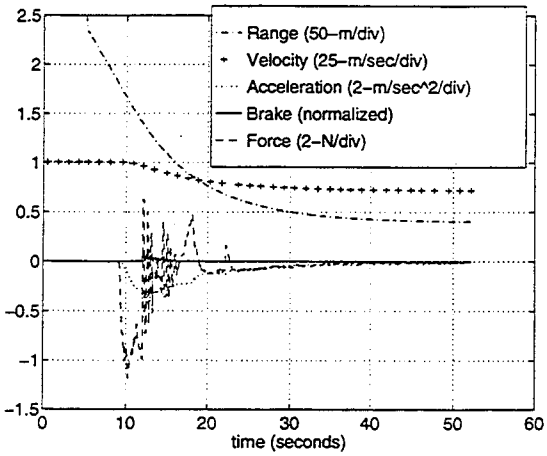


Figure 3.14: Time domain plot with dominant pole of .2 with $\zeta = 3$.

term as described in section 3.2.1 leads to Equation 3.18.

$$F_{long\_nl} = \max \left\{ \begin{array}{c} -\frac{M\dot{R}^2}{2(R+RT-R_{scaling})} \\ -MD_{max} \end{array} \right\} \tag{3.18}$$

where:

$R_{scaling}$ = Range that scales from R_S to R_H.

D_{max} = Maximum deceleration level of host.

The parameters which must be determined for this equation are R_S, R_H and D_{max}. We have already defined R_H and D_{max} so now we only need to select the parameters for R_S.

The minimum safe headway is defined by the following equation:

$$R_S = T_S V_{target} + R_{S_0} \tag{3.19}$$

where:

T_S = Safe headway time.

R_{S_0} = Safe headway at zero velocity.

The term "safe" in safe headway is perhaps not accurate as this approach does not make any assumptions about target vehicle performance capabilities and the calculated safe headway may not in fact be "safe." For our development here, a value of $T_S = T_H/2 = .5$ seconds is used. The parameters R_{S_0} is set to $R_{H_0}/2 = 1.0\ m/s$ to specify the the desired safe headway when the host vehicle stops in traffic.

With the force parameters defined, we next select parameters which determine the nonlinear personal space. The nonlinear personal space is a defined range ahead of the host vehicle. If a target is within this range, the nonlinear force is applied to the host to allow for higher deceleration levels. This range is defined by a curve of constant deceleration in the (R,$\dot{R}$) diagram (see section 2.3.2) described by Equation 3.20.

$$R_{PS\_nlin} = R_S + \frac{\dot{R}^2}{2(D_{PS\_nlin})} \tag{3.20}$$

Since the parameters for R_S have already been selected, we only need to select the deceleration level at which nonlinear forces are applied (D_{PS_nlin}).

The value of D_{PS_nlin} is set considering the dynamic model of the truck developed in Appendix A.2. For this model, when the truck is traveling at 55 mph and the throttle is fully disengaged with no braking, the deceleration level is approximately .04 g's. The truck has a maximum deceleration capability of 0.5 g's. Based on this range and that the linear forces can cause some moderate levels of braking, the value of D_{nl_switch} is set to .07 g's.

Figures 3.15 and 3.16 show the (R,$\dot{R}$) and time domain plots for a headway tracking example using nonlinear forces. For this situation, $V_{host} = 25\ m/s$ (55.9 mph) initially and the target velocity is constant at 8 m/s (17.9 mph). From the (R,$\dot{R}$) phase plane, we see that when the target vehicle is first detected, the sensed point (R_{max},$-17\ m/s$) is in the nonlinear force region of the phase plane (point is to the left of the D_{PS_nlin} curve). The host is decelerated using nonlinear forces until the response trajectory crosses the D_{PS_nlin} curve. Then the linear force is applied which brings the system to the (R_H,0). Figure 3.16 shows the time response for R, V_{host}, $\dot{V}_{host}$, brake level and F_{long} for this example. Note that the braking level increases smoothly up to 70% of fully engaged and then smoothly decreases to fully disengaged. This causes the acceleration level

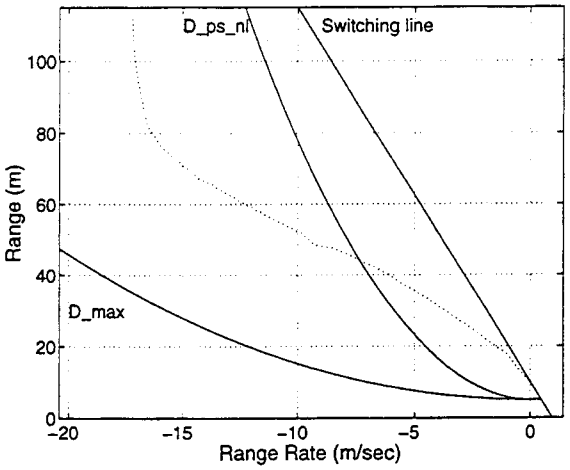


Figure 3.15: $(R,\dot{R})$ plot for headway tracking when nonlinear forces are used. V_{host} = 25.0 m/s initially and V_{target} = 8.0 m/s for the maneuver.

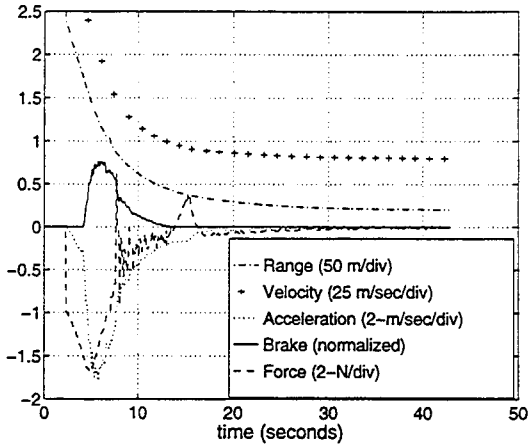


Figure 3.16: Time domain plot for headway tracking when nonlinear forces are used. V_{host} = 25.0 m/s initially and V_{target} = 8.0 m/s for the maneuver.

to increase up to -3.4 m/s and then slowly decay back to zero. This appears to be a reasonably comfortable deceleration profile considering the maneuver the vehicle is performing.

3.3 Lateral Impedance Control Implementation

In this section, we discuss the implementation process for the lateral impedance controller. This process starts with selection of controller parameters. For selecting these parameters, we first look at safe lateral dynamic limits for the host vehicle and then select the impedance and force equation coefficients with these limits in mind. Then, with these coefficients selected, we will discuss various issues which must be addressed for implementation of this design.

3.3.1 Parameter selection

For selection of the parameters for the impedance controller, we first evaluate the dynamic limits of the vehicle. A critical measure of stability for a vehicle is the lateral acceleration measured with respect to the vehicle. High lateral accelerations can cause a vehicle to spin-out or roll-over depending on the type of vehicle. For instance, cars will typically spin-out whereas trucks or commercial vehicle's tend to roll-over. In our study, since we developed the controller for implementation on a Navistar semi-tractor, we are concerned with the roll-over condition.

Different studies have been done on predicting roll-over conditions for commercial vehicles [Ervin86][MacAdam85]. These studies are simulations rather than actual tests due to the cost associated with rolling over a large truck. The roll-over condition is predicted by looking

at a model of the vehicle which is independent of longitudinal velocity. Based on the model and the loading of the truck, a lateral acceleration applied to the vehicle by the tires is determined which causes the tires on the inside (left tires on a left turn, right tires on a right turn) to begin to lift off the ground. This lateral acceleration level defines the roll-over condition ($A_{lat_{roll}}$). These models consider the road to be flat. The roll-over condition is determined to be $A_{lat_{roll}} = .35$ g's by [Ervin86] and $A_{lat_{roll}} = .43$ g's by [MacAdam85]. These levels apply across all velocities because the models are independent of velocity.

For the study here, the roll over condition is set to $A_{lat_{roll}} = 2\ m/s \approx .2$ g's. This conservative number is chosen because the lateral dynamic model (discussed in Appendix B.2) changes with velocity but does not account for cross coupled effects. That is, the lateral dynamic model does not account for longitudinal accelerations. This means that the lateral dynamic model responds the same to steering commands when the host vehicle is traveling 30 mph with no braking as it does when its traveling 30 mph with full braking. This leads to errors in the estimated lateral acceleration level when the host vehicle is applying the brakes. The conservative value for the roll-over condition is selected to allow for this error in the dynamic model.

We should clarify some notation and terminology before continuing further. The acceleration limits discussed above are with respect to the vehicle's coordinate system. Lateral motion with respect to the vehicle's coordinate system is referred to here as "lateral motion" and uses the notation $V_{lat_{xxx}}$ and $A_{lat_{xxx}}$ for lateral velocity and acceleration respectively (here xxx can be replaced with any subscript). Motion controlled by the lateral impedance controller on the other hand is is with respect to the road's coordinate system. Lateral motion with respect to the road is referred to here as "lateral path motion." The lateral path position (Y_{xxx}) is the offset from the center of the right lane. The notation for the lateral path velocity the lateral path acceleration is $\dot{Y}_{xxx}$ and $\ddot{Y}_{xxx}$, respectively.

In this section we will define the parameters for the lateral impedance and force equations considering the lateral acceleration limit of the host vehicle ($A_{lat_{roll}}$). The selected parameters will determine the lateral velocity ($\dot{Y}_d$) and the lateral acceleration ($\ddot{Y}_d$) of the desired lateral path profile the host vehicle will follow The limits do not tie directly to our lateral acceleration limit ($A_{lat_{roll}}$) because $A_{lat_{roll}}$ is with respect to the host, not the path. In this application, the lateral controller (defined in Appendix B) plays a large role in the safety of the commanded paths. When the lateral controller tracks the commanded path, the steering commands will generate lateral accelerations (A_{lat}) for the vehicle. Therefore, the performance of the lateral controller must be considered in the parameter selection process. In this discussion, the approach is to first define the parameters for the lateral impedance and force equations as if there is perfect path tracking. Then we evaluate

the safety of the response of the lateral controller for the desired path by observing the vehicle's lateral acceleration levels.

Lateral impedance coefficients

First, consider the lateral impedance associated with the vehicle. The equation and coefficients for this impedance are defined in section 2.4.1 and restated here for convenience.

$$H_{lat}(s) = \frac{b_0}{s(s+c)^2} \tag{3.21}$$

where:

$$c = \frac{\ddot{Y}_{max}}{\dot{Y}_{max}} e \tag{3.22}$$

$$b_0 = \frac{\dot{Y}_{max}}{F_{lat_{max}}} c^2 \tag{3.23}$$

Using the coefficients determined by these equations means that the lateral velocity and acceleration of the path will stay below the defined maximums ($\dot{Y}_{max}$ and $\ddot{Y}_{max}$, respectively) if the lateral virtual force stays below its defined maximum ($F_{lat_{max}}$). The question is what are the desired $\dot{Y}_{max}$ and $\ddot{Y}_{max}$ values.

The lateral path velocity ($\dot{Y}$) approximately determines the time to perform a lane change. Other authors have chosen lane change profiles for their lateral controllers which takes approximately 4 seconds [Hennessey94][Hatipoglu95]. Here, we choose 4 seconds for the duration of our nominal lane change maneuver as well. Considering that a typical lane width is 3.65 meters (12 feet), this translates into a lateral path velocity of $\dot{Y} = .91\ m/s$. However, to allow for transients at the beginning and end of the lane change, are accelerations and deceleration at the beginning and end of the lane change, we increase the nominal lane change velocity to $\dot{Y} = 1.0\ m/s$. Then for an emergency lane change we allow the lateral path velocity to be twice the nominal value. This leads to a lateral path velocity limit of $\dot{Y}_{max} = 2.0$ m/s. This value for $\dot{Y}_{max}$ is used for a starting point and is adjusted as necessary to stay within the vehicle's safe control limits.

In specifying the lateral path acceleration limit ($\ddot{Y}_{max}$), we must be careful to note the difference between path lateral acceleration and vehicle lateral acceleration. As stated earlier, the vehicle lateral acceleration is with respect to the vehicle. When a vehicle begins tracking a lateral path which is moving to the left, the vehicle lateral acceleration is to the right. The path tracking is achieved through a combination of the lateral acceleration and vehicle rotation. One can observe the lateral acceleration as a driver maneuvers a vehicle around a clover leaf ramp and feels the g-forces pushing herself/himself towards the driver-side door. The initial lateral path acceleration, on the other hand is in the same direction as the commanded path. This makes it confusing to select a value for $\ddot{Y}_{max}$ based on the limits for the lateral vehicle acceleration ($A_{lat_{roll}}$).

$\dot{Y}_{max}$ (m/s)	$\ddot{Y}_{max}$ (m/s^2)	$F_{lat_{max}}$ (N)	b_0 (m/Ns^3)	c (rad/s)
2.0	4.0	1.0	59.11	5.44

Table 3.2: Lateral impedance controller coefficients and desired design limits.

The approach followed here for selecting $\ddot{Y}_{max}$ is to set it equal to $A_{lat_{roll}}$ initially and adjust it as necessary. Using this value, the lateral accelerations observed for all of the driving scenarios discussed in Chapter 5 stayed well below the limit of $A_{lat_{roll}}$. However, with this value of $\ddot{Y}_{max}$, the lateral path generated responds slowly to virtual forces and the controller was unable to respond fast enough for avoiding "side-swipe" collisions. For this reason, $\ddot{Y}_{max}$ was increased from 2.0 m/s to 4.0 m/s. Using this value, the vehicle lateral accelerations observed for all of the driving scenarios stayed below 1.0 m/s ($\approx$ .1 g's) and the system was able to avoid being "side-swiped".

To fully define the impedance coefficients we now need to define a maximum force ($F_{lat_{max}}$). Notice that $1/F_{lat_{max}}$ is in the numerator of the impedance (see equations 3.21 and 3.23). This means that when a force is applied to the system, it will be normalized by the numerator coefficient (Equation 3.23). Realistic values for the virtual force could be used, but since the force will be normalized anyways, we use a maximum force is set to be 1.0 ($F_{lat_{max}} = 1.0\ N$).

Using the above defined values for $\dot{Y}_{max}$, $\ddot{Y}_{max}$ and $F_{lat_{max}}$, results in the following values for the impedance coefficients:

$$c = \frac{\ddot{Y}_{max}}{\dot{Y}_{max}} e = \frac{4.0}{2.0} 2.718 = 5.44\ \frac{rad}{sec}$$

$$b_0 = \frac{\dot{Y}_{max}}{F_{lat_{max}}} c^2 = \frac{2.0}{1.0} (5.44)^2 = 59.11\ \frac{m}{Nsec^3}$$

Using these coefficients for the lateral impedance, the vehicle will be controlled within its safe performance limits. The coefficients and limit values are summarized in Table 3.2. Vehicle response during lane changes for normal and emergency driving situations are shown in Figures 3.18 and 3.19, respectively. The details of these plots are discussed later.

Reflexive force parameters

With the lateral impedance coefficients defined, we now define the virtual force coefficients. Here we look at the reflexive force parameters first. The reflexive lateral force is defined in Chapter 2.4.2 and repeated here in equations 3.24 and 3.25.

$$F_{lat_{reflex}} = \begin{cases} 0 & \text{if } d_{target\_y} > d_{PS\_y} \\ F_{lat_{max}} \left(\frac{d_{PS\_y} - d_{target\_y}}{d_{PS\_y} - d_{min\_y}} \right) & \text{if } d_{PS\_y} \geq d_{target\_y} > d_{min\_y} \\ F_{lat_{max}} & \text{if } d_{target\_y} \leq d_{min\_y} \end{cases} \quad (3.24)$$

d_{PS_y} (*meters*)	d_{min_y} (*meters*)	d_{min_x} (*meters*)	T_{reflex} (*seconds*)
1.40	0.50	2.00	1.00

Table 3.3: Lateral personal space parameters for the reflexive force.

$$F_{long_{reflex}} = \begin{cases} F_{long_{max}} \left(\frac{.1F_{lat_{max}} + .4F_{lat_{reflex}}}{F_{lat_{max}}} \right) & \text{for } F_{lat_{reflex}} > F_{lat_{max}}/4 \\ 0 & \text{otherwise} \end{cases} \tag{3.25}$$

where:

d_{target_y} = Lateral distance to the target.
d_{PS_y} = Distance the personal space extends from the side of the host vehicle.
d_{min_y} = Minimum lateral distance to target.

The maximum force parameters ($F_{long_{max}}$ and $F_{lat_{max}}$) have already been defined so here we only need to set the personal space parameters.

The reflexive personal space extends to the side by d_{PS_y} and forward/rearward by d_{PS_x} The value of d_{PS_y} is set such that when a typical width target vehicle and the host are nominally in the center of adjacent lanes, the target is not within the defined personal space. Using the vehicles in the simulation model results in $d_{PS\_y} = 1.4\ m$. The value d_{min_y} is set to $0.5\ m$ considering minimum sensing ranges for typical sensors. The forward/rearward distance (d_{PS_x}) is set based on Equation 3.26.

$$d_{PS\_x} = \begin{cases} d_{min\_x} - T_{reflex}\dot{d}_{target\_x} & \text{if } \dot{d}_{target\_x} < 0 \\ d_{min\_x} & \text{otherwise} \end{cases} \tag{3.26}$$

where:

T_{reflex} = Relative travel time the side region extends in front and to the rear of host.
d_{min_x} = Minimum distance the side region extends in front and to the rear of host.
$\dot{d}_{target\_x}$ = Longitudinal relative velocity of target.

This distance is set to match up with the desired headway distance. This results in $T_{reflex} = 1.0\ sec.$ and $d_{min\_x} = 2.0\ m$. The personal space parameters for the reflexive force are summarized in Table 3.3.

Lane change force parameters

Next, let's consider the lane change force parameters. The lane change force profile is defined in section 2.4.3. This force profile is a pulse function with a magnitude of $F_{urgency}$ (see Figure 2.21).

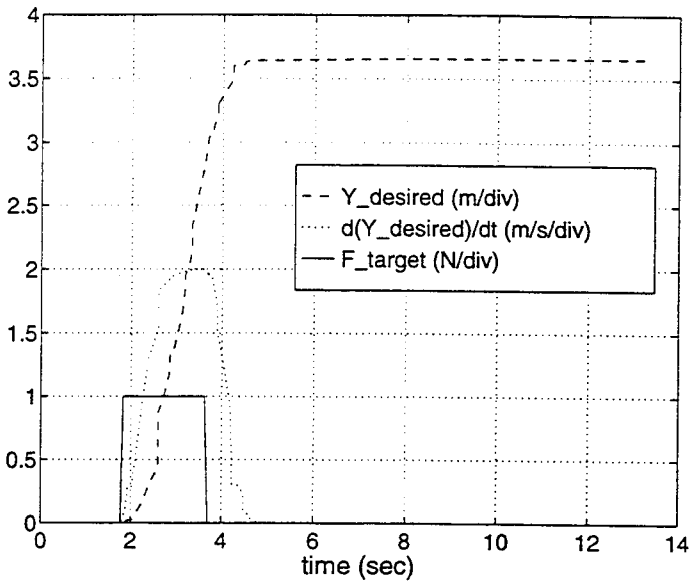


Figure 3.17: Emergency lane change profile when $F_{lat_{max}}$ is used in lane change force.

The magnitude of $F_{urgency}$ ranges from $F_{LC_{nom}}$ up to $F_{lat_{max}}$ (see Figure 2.22). The value of $F_{lat_{max}}$ is associated with an emergency lane change maneuver and has already been defined ($F_{lat_{max}} = 1.0\ N$). The value of $F_{LC_{nom}}$ is associated with typical lane changes and is defined here.

To select $F_{LC_{nom}}$, first consider the lane change profile when $F_{urgency}$ is set to $F_{lat_{max}}$ for emergency lane change maneuver (see Figure 3.17). From the figure we see that the desired lateral path velocity during the lane change maneuver is 2.0 m/s ($\dot{Y}_{max}$). The magnitude of the desired lateral path velocity is directly proportional to the lateral force magnitude (see Equation 2.29). So reducing the lateral force will result in slower lateral path velocities. Earlier we defined that a lateral path velocity of $\dot{Y}_{LC_{nom}} = 1.0\ m/s$ is desired for a nominal lane change maneuver. Since this is 1/2 the maximum lateral path velocity, the force for the nominal lane change maneuver should be half the maximum lateral force (Equation 3.27).

$$F_{LC\_nom} = \frac{F_{lat_{max}}}{2} = 0.5N \tag{3.27}$$

Using this force magnitude for $F_{urgency}$ in the lane change force profile results in the desired nominal lane change maneuver.

Figures 3.18 and 3.19 show a nominal and an emergency lane change maneuver for this control approach. For the nominal lane change maneuver (Figure 3.18), the lane is completed in approximately 4.5 seconds and during the lane change, the vehicle has a lateral path velocity of approximately 1.0 m/s. This meets the desired performance for the lane change. We also need the lateral acceleration of the vehicle (A_{lat}) to be below $A_{lat_{max}} = 2.0\ m/s$. From the plot we see

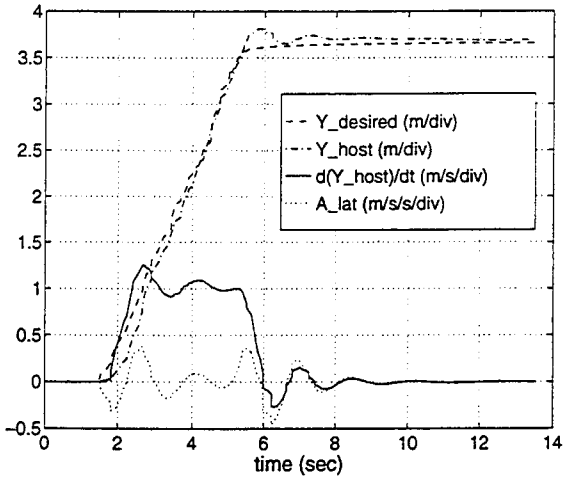


Figure 3.18: Nominal lane change maneuver when traveling at 25 m/s (55.9 mph).

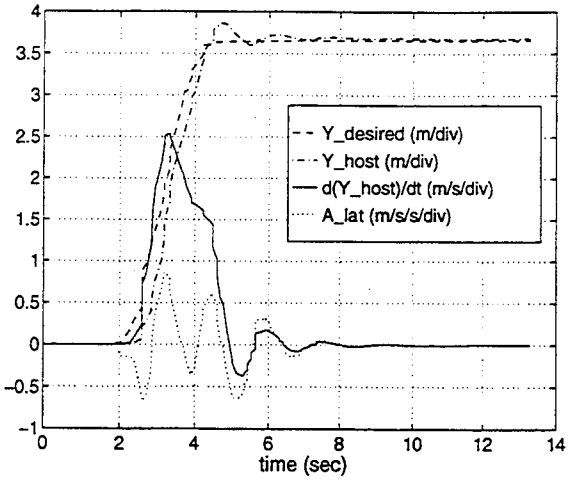


Figure 3.19: Emergency lane change maneuver when traveling at 25 m/s (55.9 mph).

that A_{lat} oscillates during the maneuver but stays below a magnitude of 0.5 m/s so the maneuver is "safe." (Note that the oscillations in the lateral acceleration is due to the lateral controller (see Appendix B) and is not caused by the impedance controller. An improved lateral controller design can reduce/eliminate these oscillations in A_{lat}). An emergency lane change maneuver is shown in Figure 3.19. This lane change maneuver is completed in approximately 2.5 *seconds* and during the lane change, the vehicle has an average lateral path velocity of approximately 2.0 m/s. This meets the desired performance for the emergency lane change maneuver. Again, we see that the lateral acceleration of the vehicle stays below $A_{lat_{max}}$. The lateral acceleration oscillates and stays below a magnitude of 1.0 m/s so the lane change maneuver is "safe." The oscillations in A_{lat} could be reduced or eliminated with an improved lateral controller design.

3.3.2 Implementation issues

There are three main issues which must be considered for practical implementation of the lateral impedance control. These issues are related to: 1) the tracking error of the lateral controller, 2) lateral controller bandwidth and 3) virtual force limits required for safe operation across all velocities.

Lateral controller tracking issues

First, let's consider the issues related to the tracking error in the lateral position controller. Like the longitudinal impedance controller, the tracking error of the inner control loop causes oscillations

in the lateral position. However, in this case, the oscillations are due to a mismatching of the forces associated with the road and the targets. This problem is most pronounced during the lane change maneuver.

The road force (F_{road}) is tuned to match up with the lane change force (F_{LC}) as described in section 2.4.4. The damping and stiffness coefficients of the road force are selected such that the total road force is approximately zero when the lane change force is turned off. F_{LC} is turned off when the commanded position Y_d is within d_{decay} from the center of the desired lane. The road force, however, is based on the lateral position of the host (Y_{host}). This means that F_{road} is approximately zero when Y_{host} is d_{decay} from the center of the lane. The problem is that Y_{host} does not track the commanded position (Y_d) without error. Therefore, when F_{LC} turns off, $Y_{host} \neq Y_d$ which results in a nonzero road force. The nonzero road force typically causes the commanded position to overshoot the center of the desired lane and results in excessive oscillation.

Examples of this effect of tracking error are shown in Figure 3.20 and 3.21 for a nominal and emergency lane change maneuver, respectively. These plots show the commanded position (Y_d) and the actual position (Y_{host}) as well as the road (F_{road}) and target (F_{target}) forces. The target force in this case is the same as the lane change force (F_{LC}). Note that during the lane change ($F_{target} \neq 0$), the road force is not applied. When the lane change force is turned off ($F_{target} = 0$), the road force is applied again. At the point when the road force is applied, it is desired that $F_{road} = 0$. For the nominal lane change maneuver (see Figure 3.20) the tracking error is relatively small. In this case, $F_{road} \approx 0$ when F_{target} switches off. This small tracking error causes some oscillation in Y_d. The tracking error is larger for an emergency lane change maneuver (see Figure 3.21). This leads to an F_{road} which is slightly greater than zero when F_{target} is switched off. The result is that Y_d over-shoots the center of the new lane. These errors are relatively minor but would increase if a lateral controller has a larger tracking error. Furthermore, since the oscillation and over-shoot is in the desired position it will be amplified in the actual position. It is desired that the oscillation due to the tracking error be eliminated.

To resolve this problem of tracking error, the block diagram has been modified as shown in Figure 3.24). This diagram shows the lateral road force calculated based on the desired position (Y_d) instead of the actual position (Y_{host}). This eliminates the tracking error in the signal for the road force. Now, during a lane change maneuver, F_{target} is switched off when Y_d is at d_{decay} from the center of the desired lane. Since F_{road} is based on Y_d it will be approximately zero at this point. This allows the force for the lane change and the road to appropriately matched and results in smooth desired lane change profiles.

Figures 3.22 and 3.23 show nominal and emergency lane change maneuvers using this

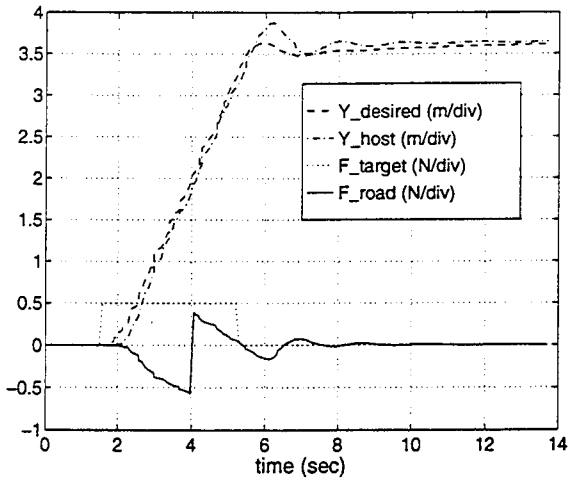


Figure 3.20: Nominal lane change maneuver without compensation for the tracking error of the lateral controller.

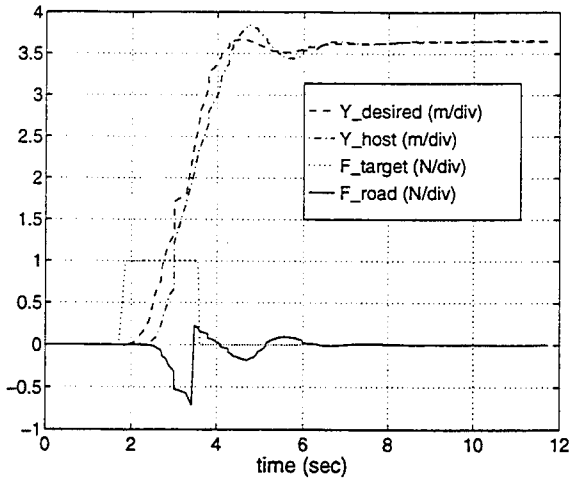


Figure 3.21: Emergency lane change maneuver without compensation for the tracking error of the lateral controller.

modification for the tracking error. These plots show the commanded position (Y_d) and the actual position (Y_{host}) as well as the road (F_{road}) and target (F_{target}) forces. The target force in this case is the same as the lane change force (F_{LC}). Note that during the lane change ($F_{target} \neq 0$), the road force is not applied so we are primarily interested in the value of F_{road} after F_{target} returns to zero. Both the nominal and emergency lane change maneuver plots show that $F_{road} \approx 0$ when F_{target} switches to zero. The result is that the desired position smoothly approaches the center of the new lane with no oscillation or over-shoot. From these figures we see that the modification eliminated the problems associated with tracking error of the lateral controller.

Lateral controller bandwidth issue

Another implementation issue for the lateral impedance controller is tied to the bandwidth of the lateral position controller. If a lateral controller has a low bandwidth, it will respond slowly to changes in the commanded position which will limit the overall capabilities of the impedance controller. For instance, if the controller response is too slow, the system can not respond to a "side-swipe" situation. In this situation, the commanded positions would produce a collision free path but the controller does not track the path in time and the target vehicle will collide with the host. Also, the system response degrades as the bandwidth decreases for the scenario of approaching a stalled car. In this case, the longer it takes the host vehicle to change lanes, the slower the host vehicle speed will be at the end of the lane change maneuver. For these reasons, a lateral controller with a higher bandwidth was developed for use with this impedance controller (Appendix B documents the controller design).

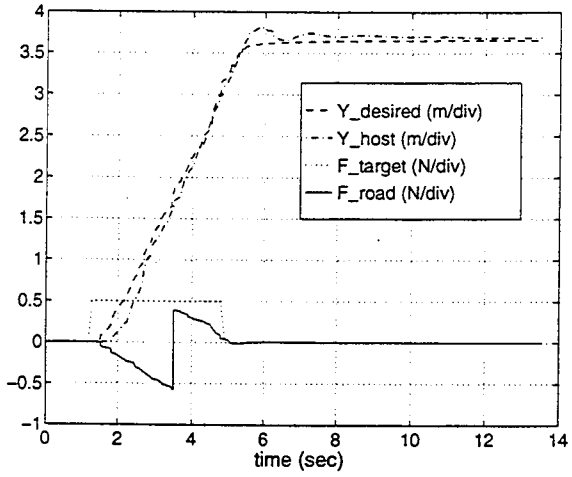


Figure 3.22: Nominal lane change maneuver with compensation for the tracking error of the lateral controller.

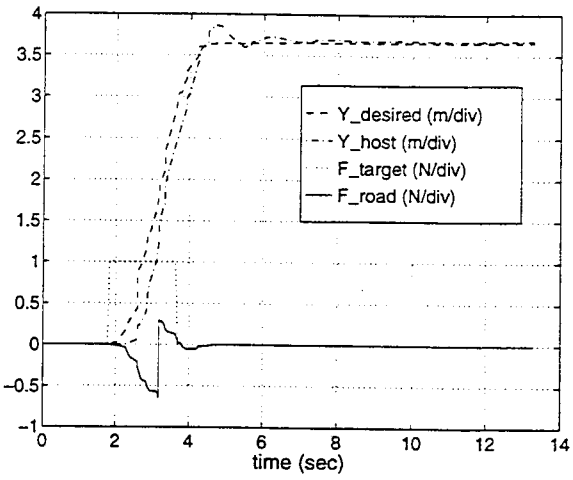


Figure 3.23: Emergency lane change maneuver with compensation for the tracking error of the lateral controller.

Virtual force maximums

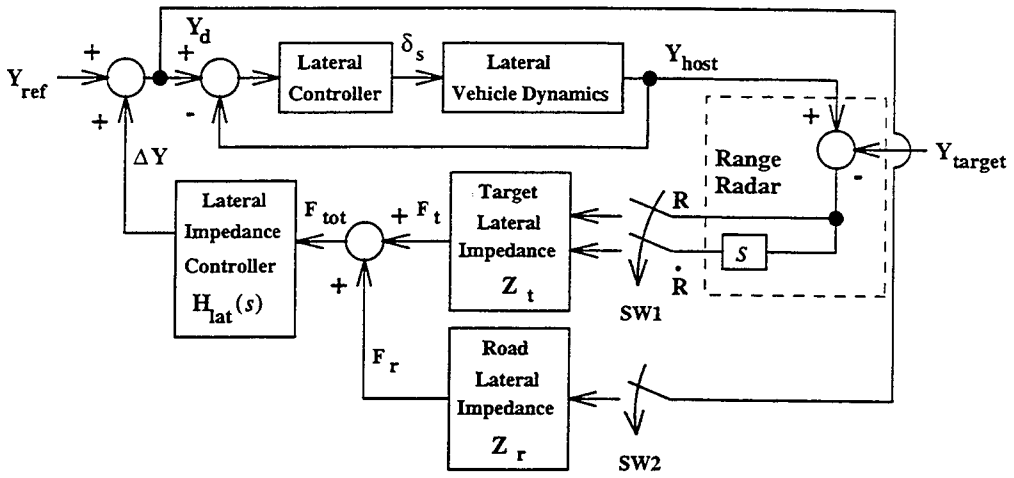
Next, let's consider how we address safe operation of the vehicle across all longitudinal velocities. The approach is to find the lateral path velocity and acceleration that the controller can track at different forward velocities. Then based on these values and the lateral impedance coefficients, a maximum lateral force ($F_{lat_{max}}$) is determined for each velocity. These maximum force values are now used to limit the virtual forces based on the vehicle velocity.

The lateral path velocity and acceleration limits are determined by evaluating the response of the lateral controller (developed in Appendix B) to a lane change profile for a range of velocities. At each given forward velocity, a lane change profile is sent to the lateral controller. The controller must meet two conditions in tracking the lane change profile:

1) Overshoot must be less than 0.25 m.
2) Vehicle lateral acceleration must be less than $A_{lat_{roll}}$.

The profile which is tested first has the lateral path velocity and accelerations set to the maximums ($\dot{Y}_d = 2.0\ m/s$ and $\ddot{Y}_d = 4.0\ m/s^2$). If either of the above conditions are not met, the $\dot{Y}_d$ and $\ddot{Y}_d$ of the lane change profile are reduced until the conditions are both met. The largest values of $\dot{Y}_d$ and $\ddot{Y}_d$ are then the limit values for the given forward velocity. The lateral controller was evaluated in this manner at forward velocities from 1 to 25 m/s (2.2 to 55.9 mph).

In the controller evaluation here, the ratio, $\ddot{Y}_{max}/\dot{Y}_{max}$, is kept constant. This is because it is not possible to change this ratio by adjusting the virtual force alone. If we wanted to adjust this ratio, we would have to change the lateral impedance coefficients with velocity. Therefore, we



Y_{ref} = Operator selected nominal lateral offset from the center of lane.
ΔY = Lateral offset in the lane calculated by impedance controller.
Y_d = Desired lateral position for the lateral controller.
δ_s = Commanded steering wheel angle.
Y_{host} = Lateral position of host vehicle in the lane.
Y_{target} = Lateral position of target vehicle in the lane.
R = Range to target measured by radar sensor.
R_d = Range rate to target measured by radar sensor.
SW1 = Software switch that is enabled when target is within ROI.
SW2 = Software switch to enable/disable road forces.
Z_t = Virtual impedance associated with space between target and host vehicles.
Z_r = Virtual impedance associated with road.
F_t = Virtual force calculated for target which is applied to host vehicle.
F_r = Virtual force applied by road to host vehicle.
$H_{lat}(S)$ = Lateral impedance controller.

Figure 3.24: Lateral impedance control block diagram with modified road force calculation.

chose to keep this ratio constant which allows the maximum lateral force to be calculated by solving Equation 3.23 for the force as shown in Equation 3.28.

$$F_{lat_{max}} = \frac{\dot{Y}_{max}}{b_0} c^2 \tag{3.28}$$

Then using this equation, the maximum lateral force is calculated for each velocity. Table 3.4 summarizes the results for the lateral controller and also shows the maximum lateral forces for each velocity.

From Table 3.4 notice that the lateral acceleration (A_{lat}) stays below our design limits of $A_{lat_{max}} = 2.0\ m/s$ for all of the lane change maneuvers. Note that A_{lat} is not listed for $V_{host} \leq 3.0\ m/s$ because a kinematic model is used instead of a dynamic model for these velocities (see Appendix B). Then, from the table we see that the limiting criteria for setting the maximum

V_{host} (m/s)	$\dot{Y}_{max}$ (m/s)	$\ddot{Y}_{max}$ (m/s^2)	Overshoot (m)	A_{lat} (m/s)	$F_{lat_{max}}$ (N)
1.0	0.3	0.6	0.21	–	0.15
2.0	0.5	1.0	0.22	–	0.25
3.0	1.0	2.0	0.21	–	0.50
4.0	1.2	2.4	0.24	0.73	0.60
5.0	1.4	2.8	0.23	0.84	0.70
7.5	1.7	3.4	0.18	1.04	0.85
10.0	2.0	4.0	0.16	1.08	1.00
15.0	2.0	4.0	0.13	0.73	1.00
20.0	2.0	4.0	0.13	0.51	1.00
25.0	2.0	4.0	0.13	0.50	1.00

Table 3.4: Summary of lateral velocity, acceleration and force limits for a range of longitudinal velocities.

force is the overshoot for a lane change maneuver. The over-shoot begins to exceed our criteria of .25 m maximum for $V_{host} < 10.0\ m/s$ if the lateral force limit isn't reduced properly. Using the force limits in this table for $F_{lat_{max}}$ allows for lane change maneuvers which meet our design requirements for forward velocities from 1.0 to 25.0 m/s.

Chapter 4

Range Sensor Placement

4.1 Introduction

The virtual bumper algorithm requires information about the various obstacles (or targets) located within its environment. It does not matter how the information about the environment is determined. One approach for determining the target vehicles' position that has been pursued by some researchers is that all vehicles in the driving environment broadcast their positions across a local area network (LAN). This is a cooperative driving environment and requires all vehicles on the highway to be equipped with the appropriate sensors. This approach is a valid solution but it will take several years before all vehicles have such equipment to make this approach widely applicable. For this reason, we are interested in sensing the environment using host vehicle mounted sensors which operate independent of the capabilities of other vehicles. The two main sensor technologies commonly used for this type of application are vision systems and range radar systems. For the implementation discussed here, the environment information will be measured using radar sensors.

4.2 Why Use Radar?

Since we have decided to sense the driving environment using vehicle mounted sensors, two sensor technologies were considered: vision systems and range radar systems. Here we address why we have chosen to base our environment sensing approach on radar sensors.

One of the main reasons for using radar sensing over vision is related to the primary function of the virtual bumper. The virtual bumper is designed to be primarily an assistive device. It works in the background relieving the driver of some of the stress associated with the driving task and provides additional information to the driver. Radar sensors fit in well with this scheme as they provide information the driver may not necessarily sense (in blind spots for example). In addition, radar sensors ought to be able to provide accurate information in rain, fog, sleet and snow. This means that radar sensors can be used to provide additional information to the driver, as well as to the virtual bumper. Vision systems can not provide information regarding conditions around the vehicle when the visibility is poor.

There are other key benefits of radar sensors over vision systems. Radar sensors typically provide more accurate range measurements. Furthermore, these sensors typically provide closing rate (or range rate) measurements as well. Closing rate information is a crucial measurement in determining safe collision avoidance maneuvers.

There are also short comings for radar sensors. The primary drawback is the inability to provide

two dimensional information about a target. The radar sensors will provide the range and range rate to a target but will provide no information regarding the position of the target lateral to the centerline of the sensors cone angle. This results in an error when using radar sensors to measure a targets position in two dimensional space. To address this weakness of the radar sensors, we have proposed using an array of radar sensors which are mounted on the perimeter of the host vehicle. The sensors can be arranged such that their cone angles overlap. With this approach, several sensor readings can be combined in order to improve the accuracy in locating a target in the host's environment.

Using an array of radar sensors around the vehicle is a new concept. Several authors have used radar for sensing the environment but there applications are focused only on headway sensing and do not address obstacle detection to the sides or behind the vehicle [Choi95][Wissing95][Kawashima95][Kamiya96][Schumacher95]. Vision systems, on the other hand, have been developed which have several sensors mounted around the vehicle [Dickmanns94][Reichardt95][Noll95][Usami95]. However, none of these authors address evaluation of the vision sensor placement on more than an intuitive level. Here we use an array of radar sensors for sensing the environment and introduce an approach for determining the desired sensor configuration.

4.3 Problem Definition

When using several radar sensors to map the host vehicle's environment, there are an infinite set of sensor configurations to choose from. The variables are the sensor characteristics (i.e. cone angle, range), the number of sensors and the sensor mounting points and orientation. The problem here is to determine an appropriate sensor configuration which to meets the target sensing requirements for the virtual bumper algorithm when operating in a limited access highway environment such as an interstate highway.

Sensor placement for interstate driving is primarily concerned with detection of targets traveling approximately along the same vector of travel as the host vehicle. This means that the sensor placement developed here would not necessarily work well for city driving environments where detection of pedestrians or cross traffic at intersections is required. Also, this sensor placement will be limited in ability to sense any targets moving perpendicular to the host, such as a deer crossing a road. The sensor placement is also based on a straight section of road. Curved road segments complicate the requirements of sensor placement and should be considered further for future evaluation of sensor placement. The particular sensing requirements of the interstate driving environment are discussed further in section 4.4.

For evaluating the sensor configuration, we follow these guidelines:

1) Model sensor using a simple 2D model with no side lobes.
2) Do not consider the effects of sensor cross-talk.
3) Output of sensor is range/range rate (no access to radar signal).
4) Base model specifications on commercially available sensors.

These guidelines are followed to simplify the evaluation and to facilitate transferring from simulation to implementation where commercially available sensors will be used.

4.4 Sensor Placement Requirements

The sensing requirements for the virtual bumper depend on the driving environment for which it is developed. For this application, the virtual bumper has been developed for an interstate driving environment. The sensing requirements for interstate driving are sub-divided here into sensing requirements for each virtual target force type (discussed in chapter 2). The virtual target force types are the longitudinal force (F_{long}), the lateral reflexive force (F_{reflex}) and the lane change force (F_{LC}).

The longitudinal target force allows the host vehicle to perform headway tracking to preceding vehicles. The requirement for this force is that the range to targets ahead in the current lane and neighboring lane is measured with minimal error. Range to targets in neighboring lanes is also required for headway tracking during lane changes. The error in these range measurements results in a corresponding error in tracking. That is, a 2 meter error in measured headway results in a 2 meter error in the actual headway. From this we see that the error is more critical as the range to the target decreases. Any range sensor has some error associated with its measurement. Our goal is to ensure that minimal measurement error is introduced due to the placement of the sensors.

The lateral reflexive force is applied to the host to maintain minimum spacing between the host and target vehicles in the adjacent lane. This force is applied for all targets within the lateral personal space (see section 2.4.2). Therefore, it is required that the lateral distance (d_y) is measured accurately for all targets within this space.

The lane change force is generated using a set of heuristics which determines the desired lane of interest (see section 2.4.3). To determine the desired lane, the system requires as input the lane in which each target is located. The significance of this requirement is that the exact lateral position of targets ahead and behind of the host is not needed. Instead, only a rough measure of a target's lateral position (i.e. its lane) is required. If exact lateral position measurements were required, range sensors such as radar would probably prove to be inadequate for this application.

In addition to the target sensing requirements, there is also a sensor placement requirement related to cost. The radar sensors which meet the requirements of the problem (i.e. maximum range, bandwidth, etc.) are still quite expensive. Therefore, a requirement for the sensor placement configuration is to use as few range sensors as possible and still satisfy the requirements listed above.

4.5 Range Sensor Model

For the evaluation of different sensor placements, a reasonably accurate model range sensor is needed. Here, we define a simple 2D sensor model and define some sensor model parameters for commercially available units.

Associated with a range sensor is a volume in 3D space. The volume can be defined by a cone with its tip at the sensor origin with a length that is the maximum range of the sensor. The angle of the cone is the "cone angle" of the sensor. The range sensor detects anything that is in the volume defined by the cone. For targets within the cone, a measure of range (and sometimes range rate) is provided but no information of a target's position perpendicular to the sensor centerline is measured. For the development here, the range sensor is modeled in 2D space as a triangle (see Figure 4.1). The 2D model is determined by the projection of the cone onto the 2D plane. This 2D model is defined for a particular sensor by its maximum range and cone angle.

In the sensor placement developed here, two sensor models are used. One sensor is a long range millimeter-wave radar sensor developed by Eaton-Vorad [Eat96] This sensor has been developed for use in automotive radar applications and provides range and range rate measurements to target vehicles within its detection area. This radar sensor is the first to be commercially available with high range (400 ft) and range rate sensing capabilities. The second sensor is a shorter range sensor based on technology developed by Amerigon [Ame95]. This sensor has a shorter sensing range and a wider cone angle. Both sensor's specifications are defined in Table 4.1.

4.6 Target Point Models

Now we have a model of the range sensor, we will discuss how we process the range readings into representations of the targets. In order to do this we must develop a model for the target. In our approach, we do not track the position of the target over time. Instead, for each set of range measurements, we calculate sensed points for each of the sensors which is detecting the target. Then all of the calculated points are treated as sensed points on the target and are used individually for the measurement error calculations in section 4.7. The method used for determining the points for a sensor is based on a target point model. Here we discuss four target point models: two point

Specification	Sensor	
	Eaton-Vorad	Amerigon
Technology	mm-wave radar	mm-wave radar
Cone Angle	4.0°	30 – 180°
Max. Range	120 meters	15.0 meters
Min. Range	0.3 meters	0.2 meters
Range Error	±3%	0.2 meters
Range Resolution	0.2 meters	0.2 meters
Range Rate ($\dot{R}$)	YES	NO
Max. $\dot{R}$	160 km/hr	—
Min. $\dot{R}$	0.4 km/hr	—
$\dot{R}$ Error	±0.5%	—
$\dot{R}$ Resolution	0.2 m/s	—
Bandwidth	10 Hz	analog
Carrier Frequency	24.125 GHz	56 GHz

Table 4.1: Eaton-Vorad and Amerigon range sensor specifications.

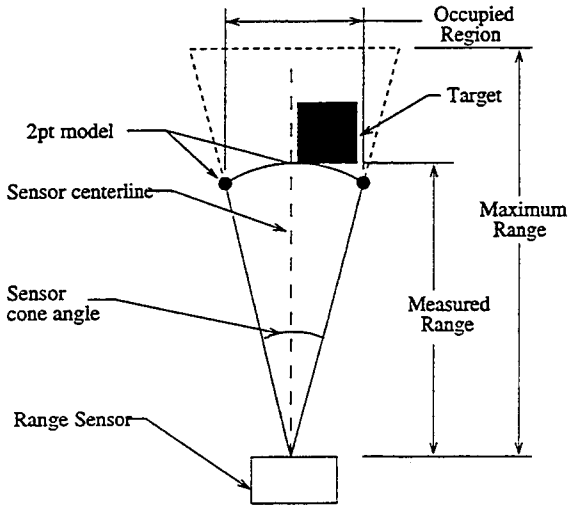


Figure 4.1: Two dimensional model of range sensor.

model (2 pt), two point reduced (2 pt_reduced), one point outside (1 pt_outside) and one point center (1 pt_cntr).

The first target point model is the two point model (2 pt). and is shown in Figure 4.1. From the figure we see that a target is detected and the range is measured. Since the range sensor provides no measurement of the target's position perpendicular to the sensor's centerline, this target can be located anywhere within the cone angle at the measured range. Therefore, a line with endpoints on each side of the cone angle at the measured range can define an occupied region where the target may be anywhere beyond this line defining the occupied region. The end points of this line are the points for the 2 pt model. This is a conservative way of representing the position of the target and leads to error when using a range sensor to determine a target's lateral and longitudinal position relative to the host vehicle. The error increases with increasing range and increasing cone angle.

The next target point model is developed to reduce the lateral positioning error present in the 2 pt model. This model is based on a simple approach of placing range sensors such that their cone angles overlap to increase the accuracy of the lateral position measurement. Consider for example the two sensors shown in Figure 4.2. One of the sensors (sensor 2) is detecting a target while the other sensor (sensor 1) is detecting nothing. The occupied zone associated with the measurement for sensor 2 could be defined as discussed earlier but this does not make use of the information provided by sensor 1. The fact that sensor 1 senses nothing means that the area within its cone angle is not occupied. Therefore, this information can be used to reduce the occupied region associated with sensor 2. In this case, the right edge of sensor 1's cone angle is used (instead of the left edge of sensor 2's cone angle) to define a reduced occupied region (see Figure 4.2). The end points for the reduced occupied region define the two point reduced (2 pt_reduced) target point model.

In the above discussion, if sensor 1 is detecting a target, the occupied region would instead be defined based on sensor 2 alone This is because sensor 1 provides no information to reduce the occupied zone. In this case, sensor 1 would also have a similar occupied zone defined, based on its own cone angle and range measurement.

The edges of the sensor cone angles should be arranged such that they assist in providing meaningful information. Overlapping sensor cone angles allows for better lateral resolution at greater ranges from the host. At these larger ranges, the lateral measurement of interest is the target's lane. Therefore, sensor edges should be aligned with lane boundaries as much as possible to allow for greater lateral resolution near these boundaries. This approach is used in the sensor placements discussed later.

The third target point model is the one point outside (1 pt_outside) model. This model is used when the sensor has a narrow cone angle (i.e. Eaton-Vorad radar model) and the target is along side

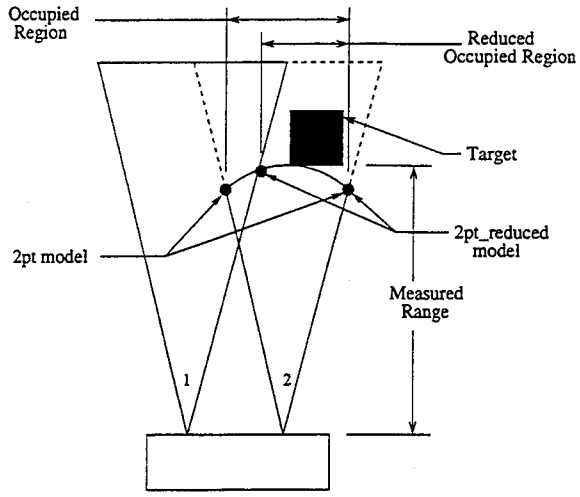


Figure 4.2: Overlapping sensors to reduce the occupied region.

of the host (see Figure 4.3). For this case, the 2 pt and 2 pt_reduced models are too conservative. Typically it is accurate to say that the point from the 2 pt model which is on the outside edge (hence the name, 1 pt_outside) of the cone angel is a point on the actual target. Therefore, we only use this point in defining our 1 pt_outside model.

The last target point model is the one point center (1 pt_cntr) model and is used for wide angle sensors such as the Amerigon sensor (see Figure 4.3). For wide cone angle sensors, the 2 pt and 2 pt_reduced model is too conservative of an approach. The mode just discussed, 1 pt_outside, on the other hand is too risky. Instead, for this model we use only the center point of the line segment defined by the 2 pt model. This model will be shown later to be a nice trade-off between risky and conservative approaches for sensors with wide cone angles.

In the following evaluation, the 2 pt_outside model is used for all the sensors whenever sensor overlap exists. If sensor overlap doesn't apply or if overlapping sensors both sense the target, the 2 pt model is used instead. These models are used in this evaluation unless otherwise stated.

4.7 Evaluation Approach

Determining which sensor placement features are desirable can be a very difficult task. This is due primarily to the fact that there are no methods for measuring the error introduced by the placement of the sensors. To allow for a more quantitative analysis of sensor layout, we have developed plots which show the error due to the sensor configuration. Different plots are developed for each sensing requirement associated with each virtual target force. From these quantitative plots we are able to make a stronger, qualitative decision about preferred sensor configurations. Note that these plots

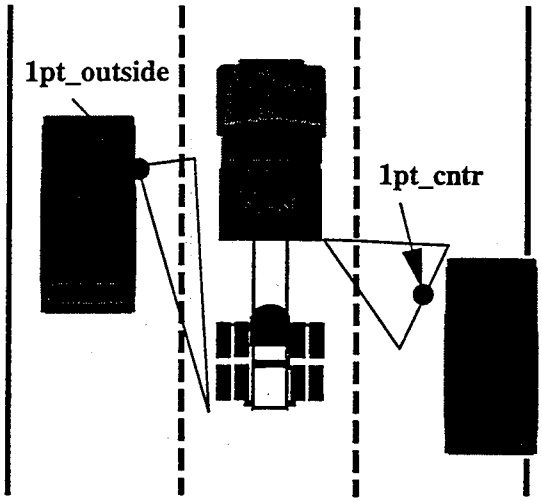


Figure 4.3: The 1 pt_cntr and 1 pt_outside target models for reducing measurement error.

show the error due to the sensor placement only and do not consider the error associated with the individual sensor measurement.

The plots are created for the space surrounding the host. A grid is placed around the host which extends forward to the maximum sensing range and to each side one full lane width (see Figure 4.4). The grid does not extend behind the host because symmetry of the sensor configuration is assumed. (Figure 4.4 also shows the lane boundaries as well as a "compressed" picture of the truck to give the reader a feel for the scaling of this figure.) Then a single target is placed such that it is centered on the first grid point. All of the sensor measurements are recorded and the target is moved to the next grid point and the measurements are again recorded. This continues until all of the grid points have been visited. With all the data collected, the sensor measurements are used to determine a measured target position for each grid point. The measured position is then compared to the actual position to determine an error measurement for each virtual force type. The value at each grid point is then plotted for each force type resulting in three error plots for a given sensor configuration.

For the longitudinal force, a range error plot is determined. The range error plot uses the error of the measured range in the longitudinal direction (R_{error}) as defined by the following equation:

$$R_{error} = R_{actual} - R_{measure_{min}} \tag{4.1}$$

This measure is calculated and plotted for each location that the target visits. If R_{error} is greater than zero, the measure is considered to be conservative because the target is actually farther away than measured. Conversely, if R_{error} is less than zero, the measure is considered to be risky.

Figure 4.5 shows a typical range error plot. On this figure, the area immediately surrounding the vehicle is white. This area represents a minimum spacing from which the target is kept away from

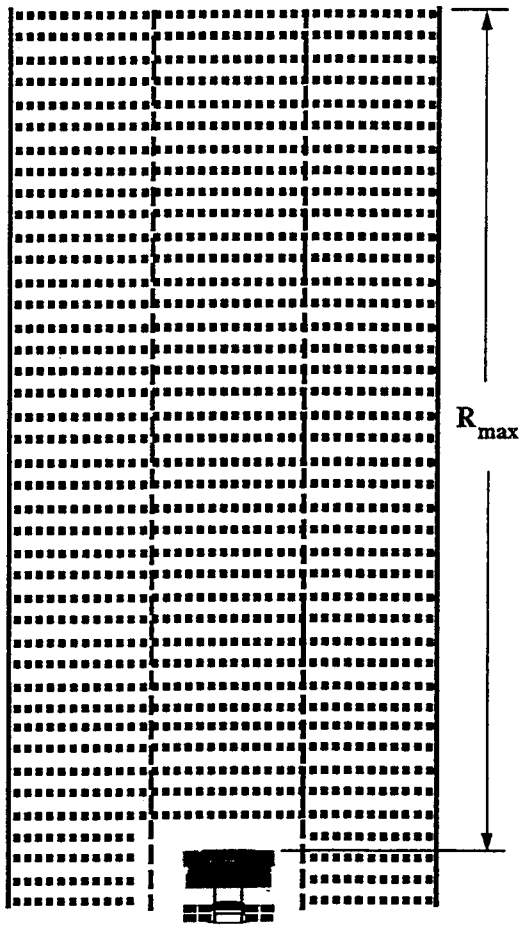


Figure 4.4: Grid of target location points for evaluation of sensor measurements.

the host. Since the target is actually centered at each grid point when the sensor measurements are recorded, the target is almost in contact with the host at these boundary points. There is also a region on both sides of the host that is dark gray that is described as the "Not Detected" region. In this region, the sensor coverage is such that the target is not detected at all. The remainder of the plot is gray scale representing the value of R_{error} for the given grid point. The scale to the right of the plot identifies the actual error value in meters.

For the lateral reflexive force, a lateral error plot is created. The lateral error plot uses the error for the measured lateral distance to the target (Y_{error}) defined by the following equation:

$$Y_{error} = |Y_{actual}| - |Y_{measure}|_{min} \tag{4.2}$$

If Y_{error} is greater than zero, the measure is considered to be conservative because the target is actually farther away than measured. Conversely, if Y_{error} is less than zero, the measure is considered to be risky. As discussed in the requirements section (section 4.4), lateral error is only of concern when a target is in the lateral personal space. The lateral personal space extends in front of

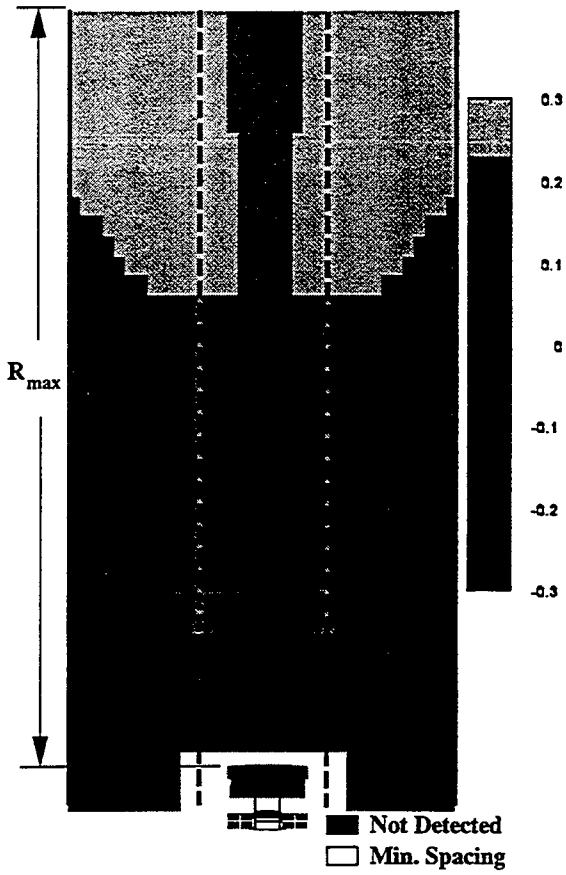


Figure 4.5: Range error plot example.

the host up to $d_{PS\_x} = 27\ m$ for a worst case scenario where $V_{host} = 25\ m/s$ and $V_{target} = 0\ m/s$ (see Equation 3.26, section 3.3.1). Therefore, Y_{error} is only calculated and plotted for the region in the adjacent lanes along side and up to 27 m in front of the host.

Figure 4.6 shows a typical lateral error plot. On this figure, the area immediately surrounding the vehicle and the current lane is white. This area represents the minimum spacing from which the target is kept away from the host as well as the untested region where Y_{error} is not relevant. There is also a region adjacent to the host that is dark gray and described as a "Not Detected" region. In this region, the sensor coverage is such that the target is not detected at all. The remainder of the plot is gray scale representing the value of Y_{error} for the given grid point. The scale to the right of the plot identifies the actual error value in meters.

The error plots associated with the lane change force are a little more complicated. As discussed in the requirements section (section 4.4), the exact lateral position of the target is not needed. Rather, the lane which the target is located in is required. Therefore, we need a plot which shows if the lane assigned to the target based on sensor measurements is the correct lane for the target.

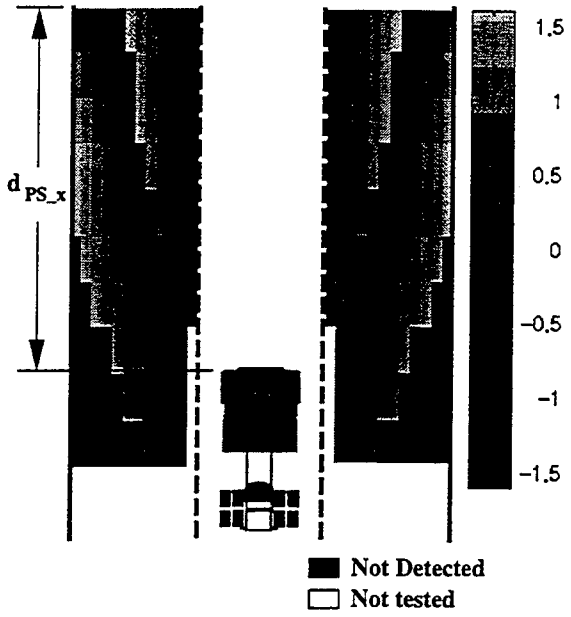


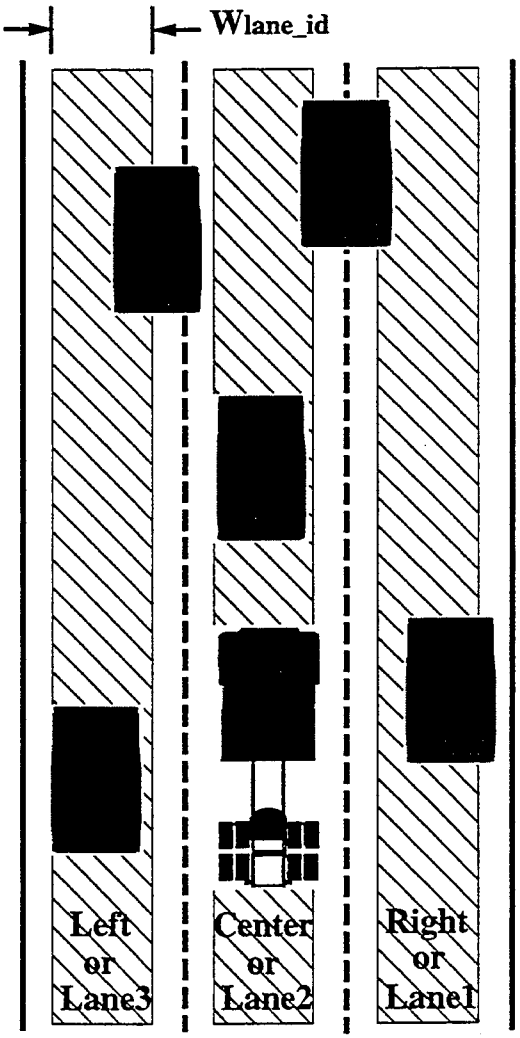
Figure 4.6: Lateral error plot example.

First, let's discuss how the target's lane is determined. For implementation purposes, each of the lanes on the highway is assigned an integer number starting from 1 in the right lane and incrementing for each additional lane (see Figure 4.7). Lane are identified using numbers instead of names (i.e. right-lane, center-lane, etc.) to eliminate ambiguity of lane identification during a host lane change maneuver. In the evaluation here, however, the host is always in lane 2 (center lane). Therefore, for purposes of this discussion, the lanes are referred to as the right, center and left, all with respect to the host being in the center lane. For each of these lanes, there is a lane id region which is W_{lane_id} wide and is centered in the lane. A target is within a lane if it has a point which is within the lane id region. Also, A target can be assigned to more than one lane at a time. Figure 4.7 shows the position for several targets and the corresponding lane assignment.

Now using this approach, both the lane id based on the target's sensed points (L_{id_sensed}) and lane id based on the target's actual position (L_{id_actual}) are determined. These values are compared to calculate a lane id error value ($L_{id_{error}}$) as shown in the following equation:

$$L_{id_{error}} = \begin{cases} \text{Correct Lane} & \text{if } L_{id\_sensed} = L_{id\_actual} \\ \text{Incorrect Lane} & \text{if } L_{id\_sensed} \neq L_{id\_actual} \\ \text{Not Detected} & \text{if target not detected by sensors} \end{cases} \tag{4.3}$$

These error values from Equation 4.3 are assigned a gray level for the $L_{id_{error}}$ plots. For these plots the value of $L_{id_{error}}$ is calculated for each grid point and the corresponding gray value is assigned. Figure 4.8 shows an example $L_{id_{error}}$ plot. For these plots it is desired that the dark gray regions



Target #	Lane ID
1	1
2	1 & 2
3	2
4	3
5	3

Figure 4.7: Various target positions and corresponding lane identification.

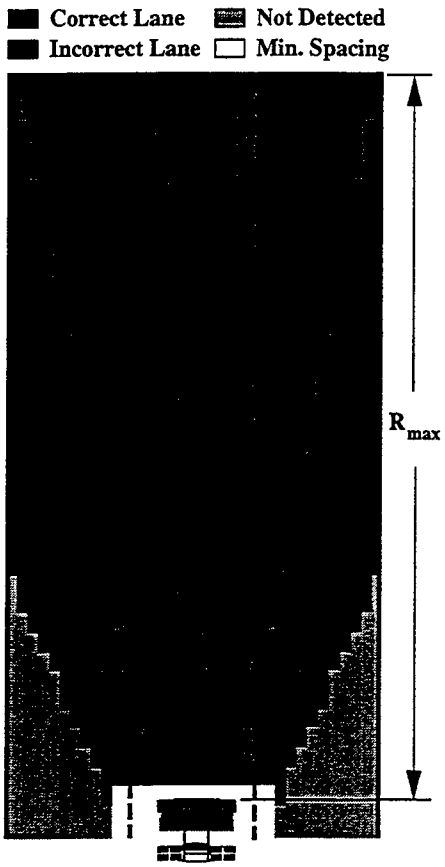


Figure 4.8: $L_{id_{error}}$ plot example.

Vehicle	Width		Length	
	meters	feet	meters	feet
Host	2.49	8.2	9.91	32.5
Target	2.03	6.7	5.59	18.3

Table 4.2: Host and target vehicle dimensions for this sensor placement evaluation.

associated with $L_{id_{error}}$ = Correct Lane cover as much of the area as possible.

The sensor placement evaluation process discussed here is dependent on both the host's and target's size and shape. For this evaluation the host is set to be equal to the size of a Navistar semi-tractor and the target to the size of a mid-sized car. The width and length of these vehicles is documented in Table 4.2.

4.8 Sensor Placement Evaluation

Clearly there are an infinite set of sensor placement configurations from which to choose. Since testing random sensor configurations is impractical and inefficient, we will only evaluate a reduced

set of sensor configurations. These sensor configurations are arrived at through an understanding of our range sensor model and the requirements for the task. The reasoning behind these configurations is discussed below.

The sensor placements are first evaluated for their ability to identify the correct target lane using the $L_{id_{error}}$ plots. The $L_{id_{error}}$ plots are evaluated first because identifying a target's lane is the most difficult requirement for the sensor configuration to meet. Once a sensor configuration is determined which adequately determines target lane position, the configuration is then evaluated for its range error. The sensor placement is modified as necessary to meet the range error requirements. Finally, the configuration is evaluated to see how it meets the lateral error requirements and again is modified as necessary.

Evaluation of lane identification error

In this section, two range sensor configurations are compared for their ability to identify a target's lane. These sensor configurations are developed based on the requirement to use a minimal set of sensors. Configuration 1, shown in Figure 4.9, has three forward facing sensors while configuration 2, shown in Figure 4.11 has four forward facing sensors. Both sensor configurations are symmetrical but only the forward facing sensors are shown for simplicity.

The first configuration has only one sensor pointed directly forward (sensor 2) for measuring headway to preceding vehicles in the lane. The sensors adjacent to sensor 2 serve the purpose of reducing the occupied zone to approximately the width of the center lane. The inside edges of these sensors (sensors 1 and 3) are almost parallel to the boundaries of the host's lane.

Figure 4.10 shows the $L_{id_{error}}$ plot for this sensor configuration using the sensor placement values defined in Table 4.3. The placement is based on the vehicle coordinate axis shown in Figure 4.9 and the orientation angles are measured positive counter-clockwise from the x-axis. From this $L_{id_{error}}$ plot (Figure 4.10) we see that there is a region in the center lane that extends to the maximum range in which the target lane is correctly identified. However, there is also a region that reaches into both adjacent lanes as range increases, in which the lane is incorrectly identified. In this region, the sensor configuration is unable to determine if the target is in an adjacent lane or if it is in the center lane. The implication here is that if a target is approached when it is in the left or right lane, we will be unable to identify the lane of the target initially. As the host approaches the target, the target will enter the correct lane region at approximately half of the maximum range from the host. Then as the host continues to approach the target, the target's lane will be correctly identified as the left or right lane. This error means that the host will begin to perform headway tracking as it approaches the target until it is determined that the target is not in the center lane. When the

target's lane is correctly identified, the host will accelerate back to the desired speed. This type of error in lane identification seriously degrades the performance of the system.

The second sensor configuration uses four forward facing sensors instead of three (see Figure 4.11). For this configuration, two sensors (2 and 3) overlap in the lane directly in front of the host. The purpose behind using two sensors for detecting objects in the center lane is that when there is a target in the adjacent lane, one of these sensors may detect its presence (if the target is far enough away). However, the other overlapping sensor will not sense the target and can be used to reduce the occupied region for the sensor sensing the target so that it does not include the host's lane. The objective here is to eliminate the problem observed with configuration one. There are also two outside sensors (sensors 1 and 4) for this configuration. These outside sensors serve the purpose of reducing the occupied zone to approximately the width of the center lane when a target is in the lane in front of the host. The inside edges of these sensors are almost parallel to the boundaries of the host's lane.

Figure 4.12 shows the $L_{id_{error}}$ plot for sensor configuration 2A. This configuration places the sensors approximately as shown in Figure 4.11 with the exact sensor placement and orientation defined by configuration 2A as listed in Table 4.4. From this plot we see that again there is a region in the lane in front of the host where the target is assigned an incorrect lane id. In this region the target is sensed to be (ambiguously) in both the center and adjacent lanes. This error is caused by the poor lateral resolution of the sensors and will always exist to some extent for this type of sensing approach. We also see from the plot a significant improvement over sensor configuration 1. With configuration 2A, targets in the adjacent lane are identified correctly for ranges up to the maximum sensor range. This means that targets in adjacent lanes immediately have correct lane identification when they first enter into the sensing range. The result is that the host will approach these obstacles and pass without performing headway control or performing unnecessary lane changes. For this reason, configuration 2 is selected as the desired sensor configuration for the forward looking sensors.

With configuration 2 selected as the basic sensor configuration, we now observe the effect of changing some of the configuration's parameters. Consider the orientation of sensors 1 and 4. These sensors are oriented at 2.5° and −2.5° respectively for configuration 2A. Sensor configuration 2B (see Table 4.4) adjusts this angle away from the x-axis to 4.0° for sensor 1 and −4.0° for sensor 4 and the resulting $L_{id_{error}}$ plot is shown in Figure 4.13. Comparing this figure to Figure 4.12 for configuration 2A, we see that increasing these sensor orientation angles makes the correct lane id region in the center lane wider but it no longer extends to the maximum range. Next, let's observe the effects of sensor configuration 2C which adjusts these orientation angle towards the x-axis to 2.0° for sensor 1 and −2.0° for sensor 4. The resulting $L_{id_{error}}$ plot is shown in Figure 4.14. From

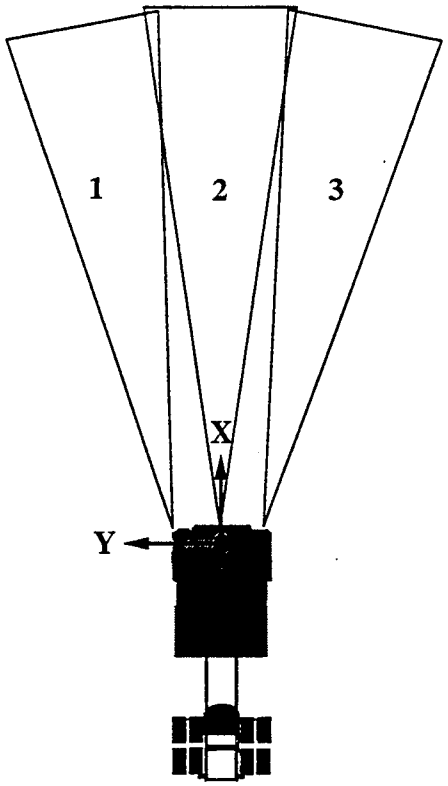


Figure 4.9: Range sensor placement configuration 1 with only the three forward facing sensors shown.

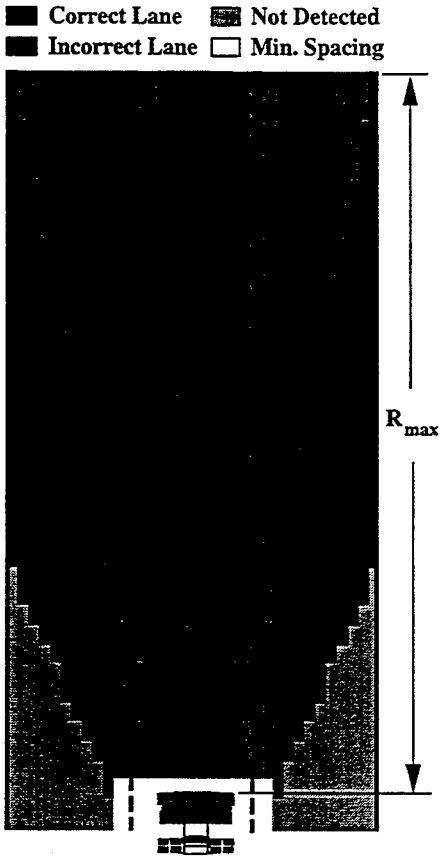


Figure 4.10: $L_{id_{error}}$ plot using sensor configuration 1A.

Sensor		Configuration 1A
1	X-mount (m)	1.27
	Y-mount (m)	1.24
	Θ (deg)	2.5
	Type	E
2	X-mount (m)	1.27
	Y-mount (m)	0.00
	Θ (deg)	0.0
	Type	E
3	X-mount (m)	1.27
	Y-mount (m)	-1.24
	Θ (deg)	-2.5
	Type	E

Sensor Type: (E) Eaton-Vorad, (Am) Amerigon

Table 4.3: Sensor position and orientation for sensor configuration type 1.

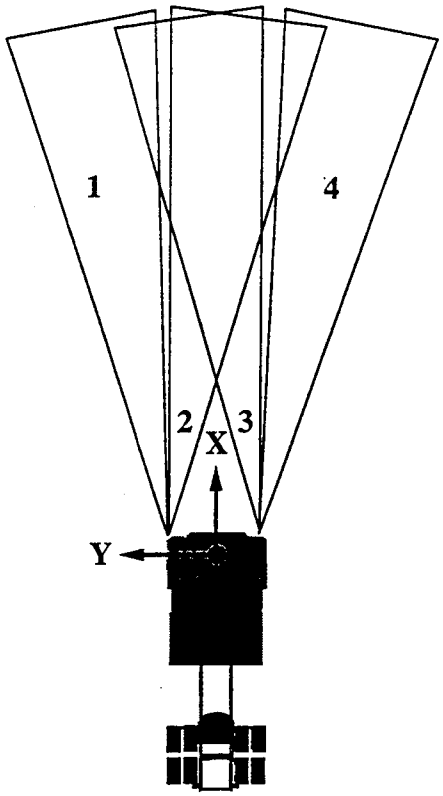


Figure 4.11: Range sensor placement configuration 2.

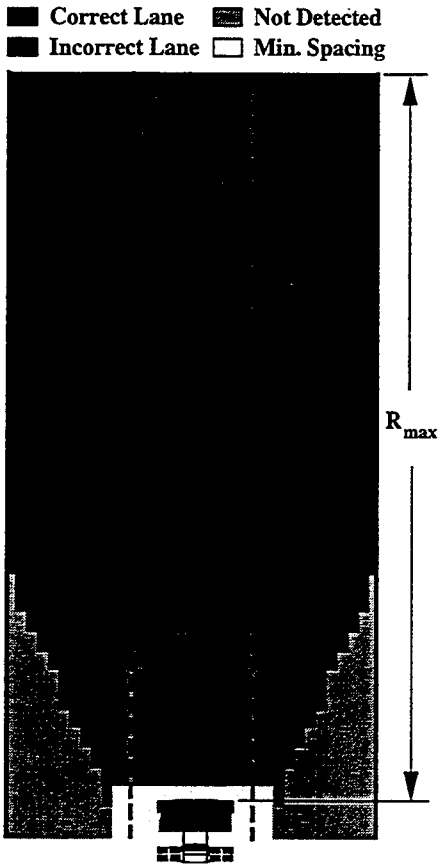


Figure 4.12: Lane identification plot using sensor configuration 2A.

Sensor		Configuration		
		2A	2B	2C
1	X-mount (m)	1.27	1.27	1.27
	Y-mount (m)	1.24	1.24	1.24
	Θ (deg)	2.5	4.0	2.0
	Type	E	E	E
2	X-mount (m)	1.27	1.27	1.27
	Y-mount (m)	1.24	1.24	1.24
	Θ (deg)	-2.0	-2.0	-2.0
	Type	E	E	E
3	X-mount (m)	1.27	1.27	1.27
	Y-mount (m)	-1.24	-1.24	-1.24
	Θ (deg)	2.0	2.0	2.0
	Type	E	E	E
4	X-mount (m)	1.27	1.27	1.27
	Y-mount (m)	-1.24	-1.24	-1.24
	Θ (deg)	-2.5	-4.0	-2.0
	Type	E	E	E

Sensor Type: (E) Eaton-Vorad, (Am) Amerigon

Table 4.4: Sensor position and orientation for sensor configuration type 2.

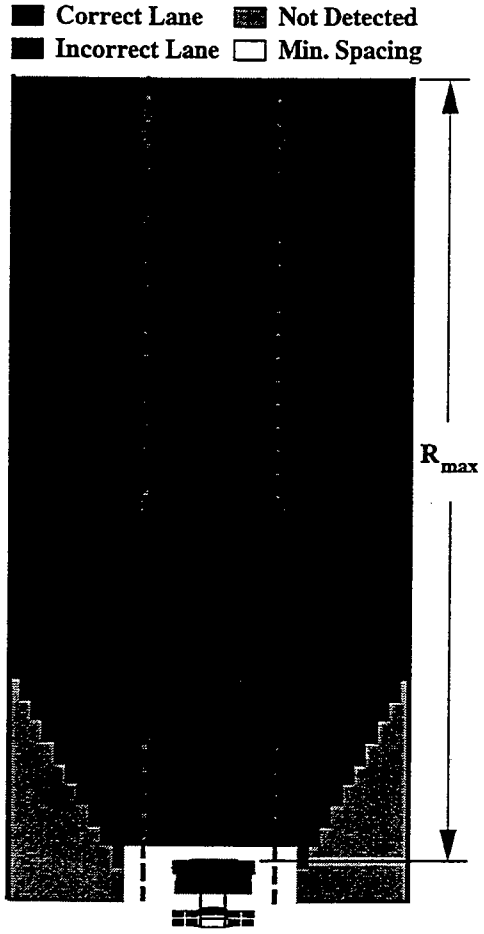


Figure 4.13: Lane identification plot using sensor configuration 2B with sensor orientation per Table 4.4

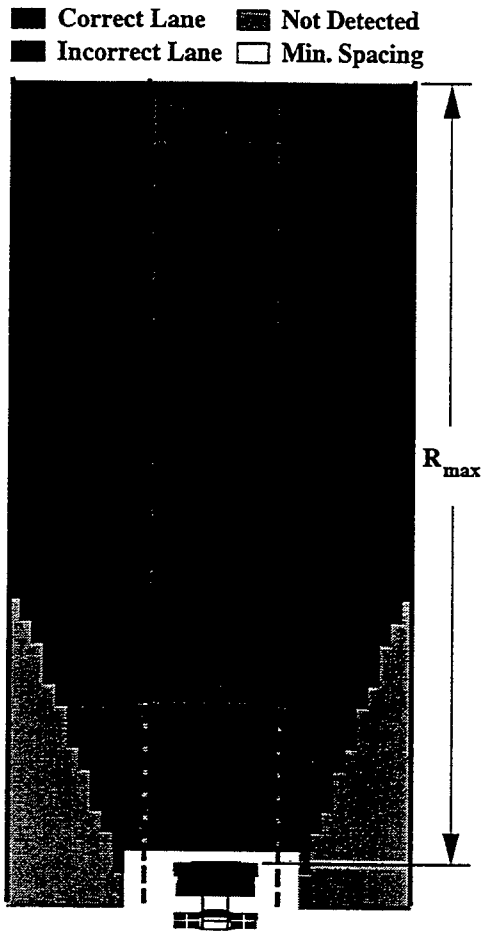


Figure 4.14: Lane identification plot using sensor configuration 2C with sensor orientation per Table 4.4

this figure notice that the correct lane id region in the center lane is now very narrow, but again extends to the maximum range. It is desireable that this correct lane id region in the center lane be as wide as possible and extend out to the maximum range. Therefore, the original configuration 2A has the preferred sensor 1 and 4 orientation.

It is important to note the magnitude of the evaluated sensor angles. The orientation angles varied a total of 2.5° between these three configurations. This small variation in the orientation angles has a significant impact on the results. For an actual implementation, it is not possible to mount and orient sensors without any tolerance coming into play. Therefore, properly identifying targets in the center lane will be difficult at best for a practical implementation.

With configuration 2A selected as our base sensor configuration, there is still one issue to address. From Figure 4.12, notice that there are regions in the adjacent lanes along side the host in which

the target is not detected. Additional sensors must be added for proper target sensing. Figure 4.15 shows sensor configuration 3 which adds four sensors to configuration 2A along the side of the host. Adding these additional sensors reduces the "Not Detected" region in the adjacent lanes significantly. Also, the target's lane is identified correctly for nearly all of the detected areas in the adjacent lanes. There is still an area in each adjacent lane in which the target is not detected. This region is addressed later in sensor placement evaluation for lateral error.

Evaluation of range error

For the longitudinal impedance controller to work properly, the error in measured range to targets ahead of the host must be minimal. The range error (R_{error}) plots represent this measurement error for each grid point around the host (see section 4.7 for a detailed explanation).

Figure 4.17 shows the R_{error} plot for sensor configuration 3A. The R_{error} is represented by the gray level shown in the scale to the right of the plot. From this plot we see that the error is less than ± 0.3 m for the entire region in which the target is detected. This is a reasonable level of error and no sensor modifications are added here to reduce R_{error}. The undetected areas will be addressed when considering the lateral error.

Evaluation of lateral error

We have now met the lane identification and the longitudinal error requirements for the sensor placement. The next step is to evaluate and modify the sensor configuration to meet the lateral measurement requirements.

We begin evaluation by considering the lateral error (Y_{error}) for sensor configuration 3A as shown in Figure 4.18. From the plot, notice that for detected targets, the lateral error is approximately zero along side the host (medium gray) and increases in magnitude as the target moves farther away from the host in the adjacent lane (shown by increasingly lighter shades of gray). From section 4.7 remember that $Y_{error} > 0$ is conservative so this seems like an acceptable situation. However, if the measurements are too conservative, the host will shift in the lane to maintain spacing when a target is already at an adequate spacing. This conservative error is due to the width of the sensor beam at the measured range. To eliminate this error, the sensor reading for sensors 5, 6, 7 and 8 are reduced to a 1 pt_outside target model instead of a 2 pt target model (see section 4.5) up to a range of 5 meters past the host. Figure 4.19 shows the Y_{error} plot using the 1 pt_outside target model for these sensors. Here we notice that there is a much larger portion along side the host in which Y_{error} is approximately zero (medium gray). This modification reduced the error in the side regions.

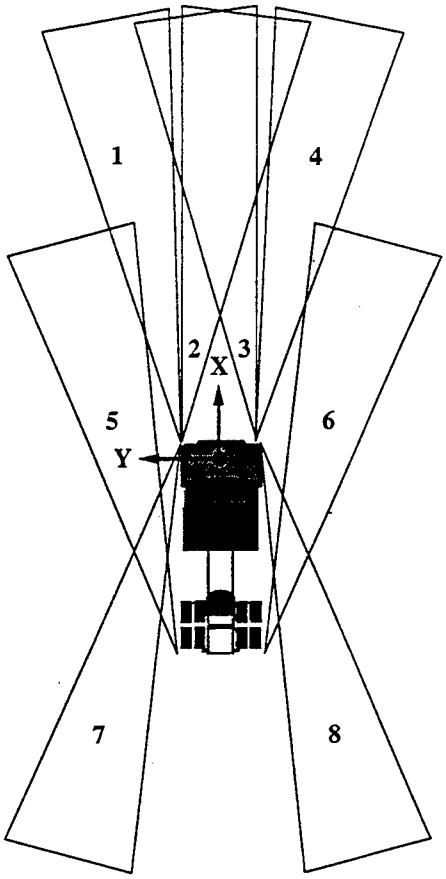


Figure 4.15: Range sensor configuration 3.

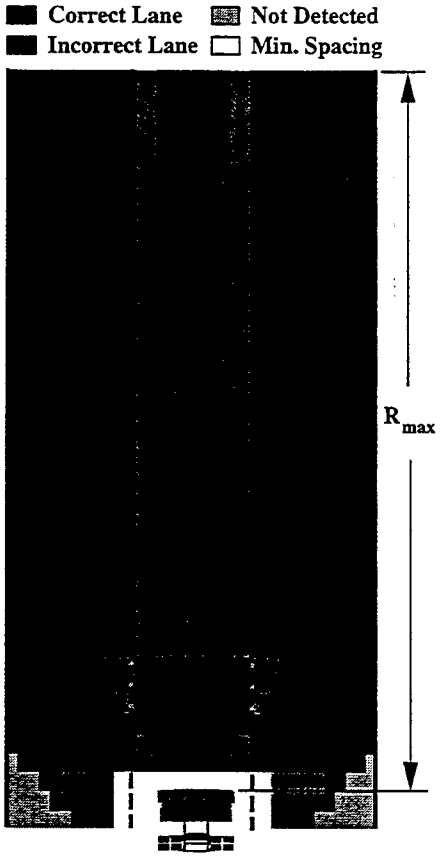


Figure 4.16: Lane identification plot using configuration 3A.

Sensor		Configuration 3A	3B
1	X-mount (m)	1.27	1.27
	Y-mount (m)	1.24	1.24
	Θ (deg)	2.5	2.5
	Type	E	E
2	X-mount (m)	1.27	1.27
	Y-mount (m)	1.24	1.24
	Θ (deg)	-2.0	-2.0
	Type	E	E
3	X-mount (m)	1.27	1.27
	Y-mount (m)	-1.24	-1.24
	Θ (deg)	2.0	2.0
	Type	E	E
4	X-mount (m)	1.27	1.27
	Y-mount (m)	-1.24	-1.24
	Θ (deg)	-2.5	-2.5
	Type	E	E

Sensor		Configuration 3A	3B
5	X-mount (m)	-8.64	-8.64
	Y-mount (m)	1.24	1.24
	Θ (deg)	7.0	6.0
	Type	E	E
6	X-mount (m)	-8.64	-8.64
	Y-mount (m)	-1.24	-1.24
	Θ (deg)	-7.0	-6.0
	Type	E	E
7	X-mount (m)	1.27	1.27
	Y-mount (m)	1.24	1.24
	Θ (deg)	173.0	174.0
	Type	E	E
8	X-mount (m)	1.27	1.27
	Y-mount (m)	-1.24	-1.24
	Θ (deg)	-173.0	-174.0
	Type	E	E

Sensor Type: (E) Eaton-Vorad, (Am) Amerigon

Table 4.5: Sensor position and orientation for sensor configuration type 3.

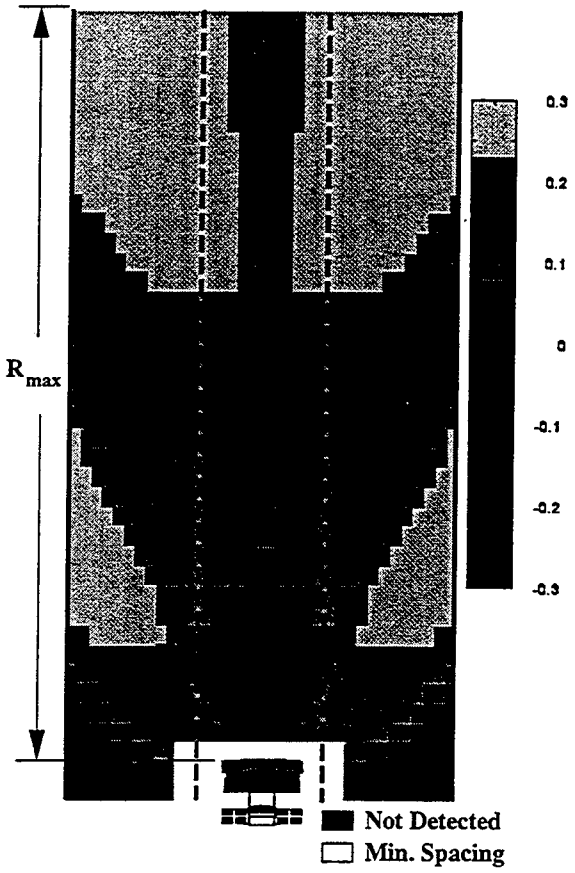


Figure 4.17: R_{error} plot using configuration 3A.

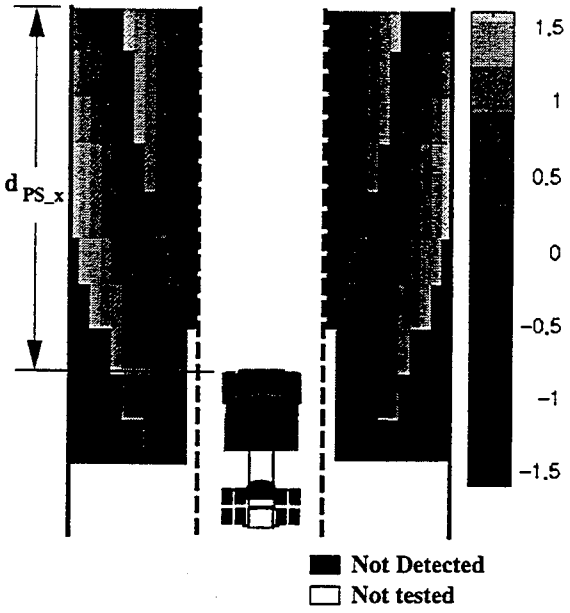


Figure 4.18: Y_{error} plot using configuration 3A.

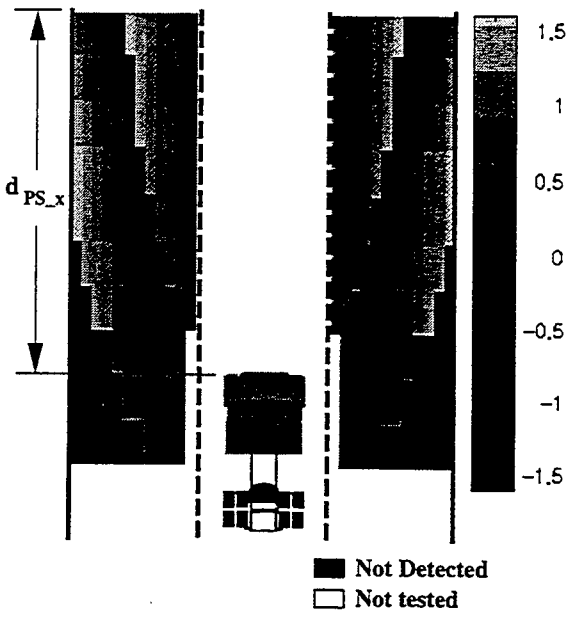


Figure 4.19: Y_{error} plot for configuration 3A using 1 pt_cntr target model.

Next let's address placement of additional sensors to allow for detection of targets along side of the host. Here we add six additional sensors to configuration 3A and identify this sensor placement as configuration 4 (see Figure 4.20).

The six additional sensors are oriented such that they are pointing more or less to the side of the host. For these sensors, only a short sensing range is required and a larger cone angle is desired for better coverage. Therefore, the Amerigon sensor model is used for these sensors. Also the Amerigon sensor is less expensive than the Eaton-Vorad sensor so the Amerigon sensor should be used whenever it can meet the specifications of the problem. The Amerigon sensor, however, does not measure the range rate to the target so the range rate measure is not used in the calculation of the reflexive force as discussed in section 2.4.2.

The selection of the placement and orientation for the six additional sensors of configuration 4 is done by looking at an overhead view of the sensor placement similar to that shown in Figure 4.20. The mounting points for the sensors are selected by adjusting the mounting points until the sensors are reasonably spaced along the side of the truck. The orientation of sensors 9 and 10 is selected such that these sensors coverage is approximately centered on the undetected regions outside of sensors 5 and 6. (The sensors shown in Figure 4.20 are not to scale so the figure does not show this undetected region.) Then, with these parameters defined, we evaluate the orientation of sensors 11, 12, 13 and 14 using the lateral error (Y_{error}) plots.

Before evaluating the effect of sensor orientation angles, let's first determine which target model

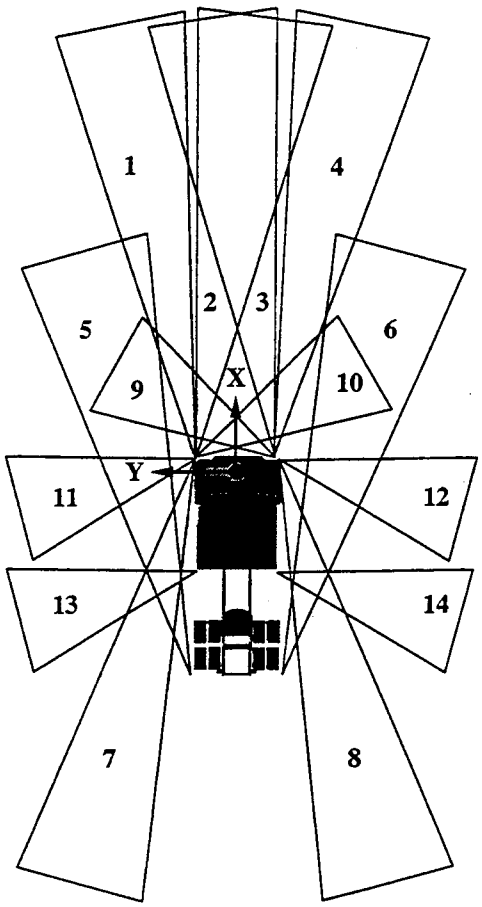


Figure 4.20: Range sensor configuration 4.

Sensor		Configuration 4A	4B	Sensor		Configuration 4A	4B
1	X-mount (m)	1.27	1.27	8	X-mount (m)	1.27	1.27
	Y-mount (m)	1.24	1.24		Y-mount (m)	-1.24	-1.24
	Θ (deg)	2.5	2.5		Θ (deg)	-174.0	-174.0
	Type	E	E		Type	E	E
2	X-mount (m)	1.27	1.27	9	X-mount (m)	1.27	1.27
	Y-mount (m)	1.24	1.24		Y-mount (m)	-1.24	-1.24
	Θ (deg)	-2.0	-2.0		Θ (deg)	70.0	70.0
	Type	E	E		Type	Am	Am
3	X-mount (m)	1.27	1.27	10	X-mount (m)	1.27	1.27
	Y-mount (m)	-1.24	-1.24		Y-mount (m)	1.24	1.24
	Θ (deg)	2.0	2.0		Θ (deg)	-70.0	-70.0
	Type	E	E		Type	Am	Am
4	X-mount (m)	1.27	1.27	11	X-mount (m)	1.27	1.27
	Y-mount (m)	-1.24	-1.24		Y-mount (m)	1.24	1.24
	Θ (deg)	-2.5	-2.5		Θ (deg)	105.0	135.0
	Type	E	E		Type	Am	Am
5	X-mount (m)	-8.64	-8.64	12	X-mount (m)	1.27	1.27
	Y-mount (m)	1.24	1.24		Y-mount (m)	-1.24	-1.24
	Θ (deg)	6.0	6.0		Θ (deg)	-105.0	-135.0
	Type	E	E		Type	Am	Am
6	X-mount (m)	-8.64	-8.64	13	X-mount (m)	-3.05	-3.05
	Y-mount (m)	-1.24	-1.24		Y-mount (m)	1.24	1.24
	Θ (deg)	-6.0	-6.0		Θ (deg)	105.0	135.0
	Type	E	E		Type	Am	Am
7	X-mount (m)	1.27	1.27	14	X-mount (m)	-3.05	-3.05
	Y-mount (m)	1.24	1.24		Y-mount (m)	-1.24	-1.24
	Θ (deg)	174.0	174.0		Θ (deg)	-105.0	-135.0
	Type	E	E		Type	Am	Am

Sensor Type: (E) Eaton-Vorad, (Am) Amerigon

Table 4.6: Sensor position and orientation for sensor configuration type 4.

should be used for the additional sensors. Here we look at the Y_{error} plots using configuration 4A as defined by Figure 4.20 and Table 4.6. Figure 4.21 shows the Y_{error} plot using the 2 pt target model for the Amerigon sensors and Figure 4.22 shows the Y_{error} plot using the 1 pt_cntr target model for the Amerigon sensors. Remember from earlier discussions that the 2 pt target model is a conservative way to represent the sensor measurement. These conservative Y_{error} measurements are shown shown by the lighter gray areas along side the host in Figure 4.21. Using the 1 pt_cntr target model is less conservative. Using this representation of range readings allows Y_{error} to be approximately zero (medium gray) for the region along side the host (see Figure 4.22). Achieving minimal error in this region is required for the host to properly maintain spacing to adjacent targets. Therefore, the 1 pt_cntr sensor model is used for all of the Amerigon sensors (sensors 9 through 14).

Next, we will look at the effect of the orientation of sensors 11 through 14 on the Y_{error}.

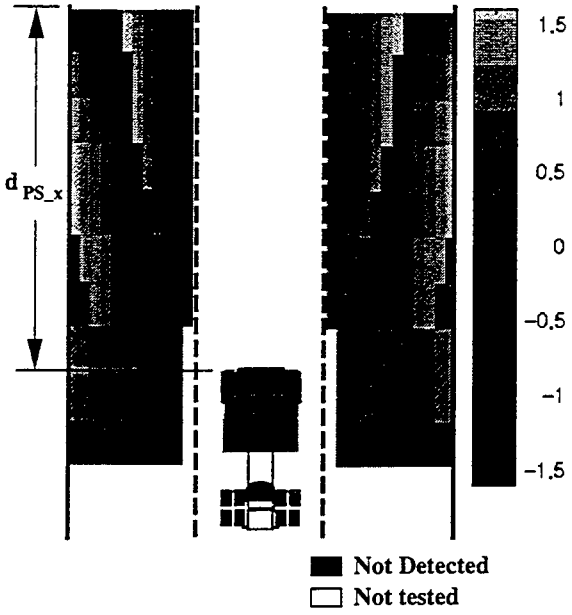


Figure 4.21: Lateral error plot for configuration 4A using 2 pt target model for the Amerigon sensors.

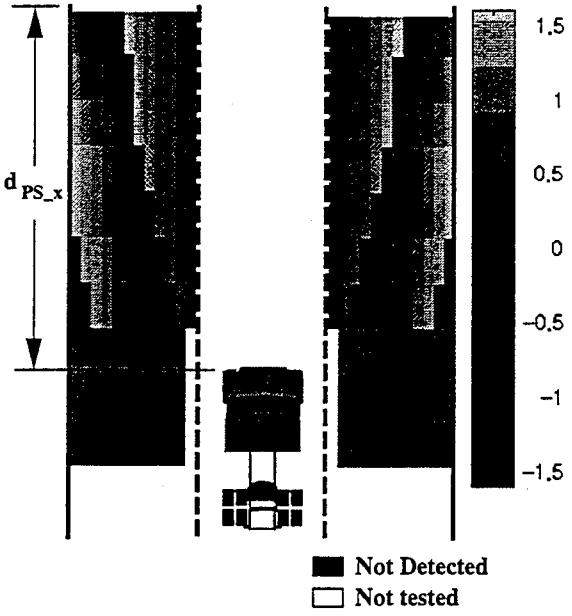


Figure 4.22: Lateral error plot for configuration 4A using 1 pt_cntr target model for the Amerigon sensors.

Figure 4.22 shows the lateral error plot for for configuration 4A (see Table 4.6). For this configuration, the orientation of these sensors set to 105° for sensors 11 and 13 and −105° for sensors 12 and 14 (angles are measured counter-clockwise from the x-axis). Figure 4.23 shows the Y_{error} plot for configuration 4B. For configuration 4B the orientation of these sensors set to 135° for sensors 11 and 13 and −135° for sensors 12 and 14.

From Figure 4.23 we see that there are larger light gray areas (conservative Y_{error}) along side the host for this configuration. This means that this change in sensor orientation made the error measurements more conservative. In fact, any angle for these side facing sensors other than ±90° will add some conservative error to the lateral measurement. However, it is desired that the angle be greater in magnitude than 90° to allow for better coverage of the adjacent lanes using as few sensor as possible. From Figure 4.22 we see that minimal error is introduced in configuration 4A for sensor angles of ±105° for sensors 11 through 14. Therefore, configuration 4A is selected as the desired configuration.

Now, we must check that the additional sensors have not adversely affected the lane identification error or the range error. Figure 4.24 show the $L_{id_{error}}$ plots for configuration 4A. This plot indicates that additional sensors improved the lane identification error for the regions along side the host. However there is a region immediately adjacent to the host which exhibits some error in identifying the correct lane for the target. This is acceptable because if a target is detected in this area, the

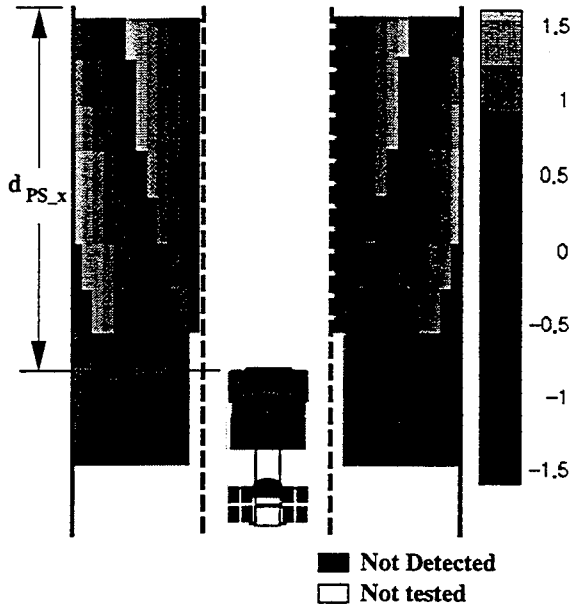


Figure 4.23: Lateral error plot for configuration 4B using 1 pt_cntr target model for the Amerigon sensors.

lane is unavailable for a lane change and the lane identification for the target is not used (see section 2.4.3).

Next, we examine the R_{error} plot for configuration 4A shown in Figure 4.25. The range error in this plot is primarily zero except for a small region directly in front of the vehicle. In this area, $R_{error} > 0$ (lighter gray) which makes these errors conservative. This error is introduced by the Amerigon sensors because of the wider cone angle and their orientation on the host vehicle. Fortunately, the error is conservative and it only shows up in a small region at the front of the host. Therefore, sensor configuration 4A also meets the R_{error} requirements for the system.

Based on the evaluation discussed here, range sensor configuration 4A is selected for implementation with the virtual bumper. This sensor placement meets the requirements for the system outlined in section 4.4 and should provide adequate sensing for the vehicle control algorithm.

4.9 Limitations

The approach of using range sensors for sensing the region around the host has several limitations. Here we outline the limitations which have the most impact a practical implementation of this approach.

The first factor is that the sensors' ability to identify a target's lane is limited by the host and

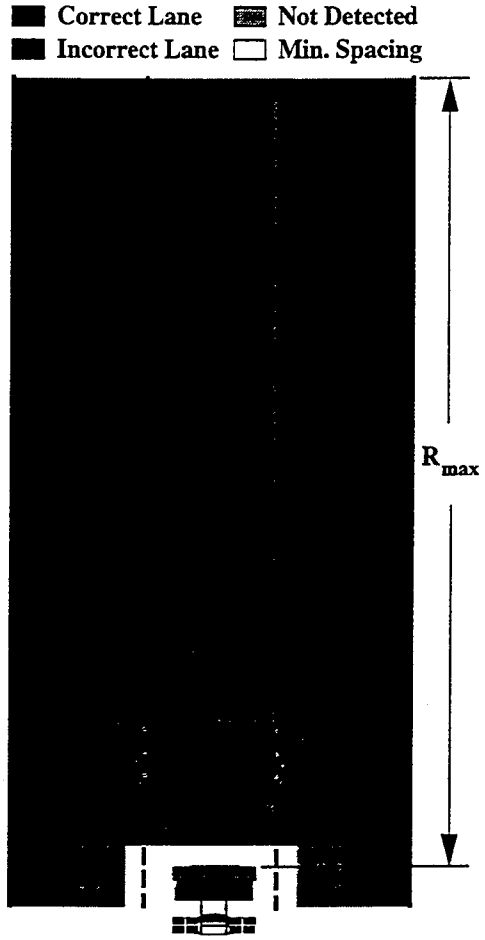


Figure 4.24: Lane identification plot using configuration 4A.

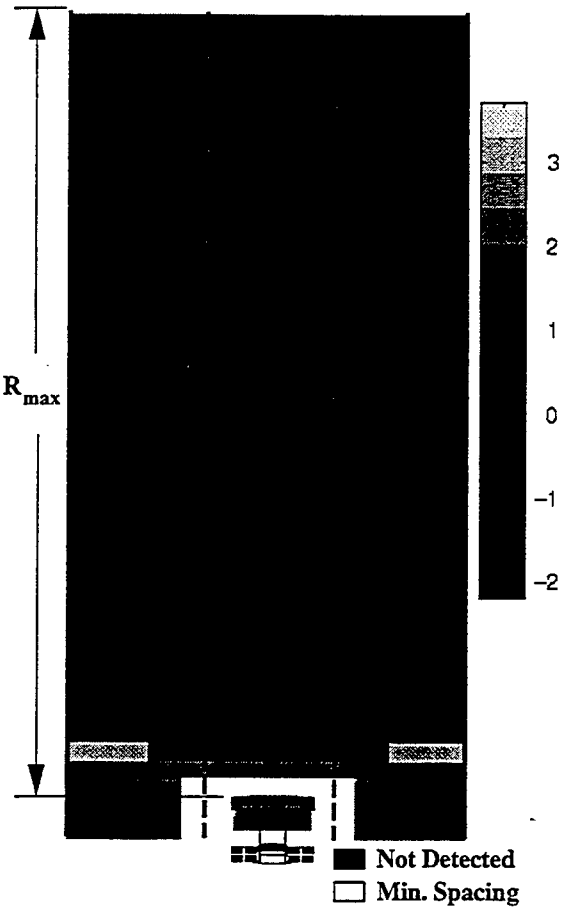


Figure 4.25: Longitudinal error plot for configuration 4A.

target width. Recall from earlier $L_{id_{error}}$ plots (Figure 4.24) that there is a narrow region in the center of the host's lane in which the target's lane is correctly identified. If a wider target is used in the evaluation, this region would become narrower and may even disappear. This is because as the target becomes wider, more sensors detect it. If the four main forward facing sensors detect the target, then the occupied region can not be reduced to the center lane. This same effect occurs if a narrower host vehicle is used.

Another limiting factor is that the sensor mounting tolerance required for the selected configuration is very small. The four forward looking sensors would have to be mounted with approximately a $\pm 0.5°$ tolerance on the orientation. This would be very difficult to achieve considering the orientation angle is measured from the vehicle coordinate system. This tolerance includes the tolerance of the alignment of the sensor beam with the sensor enclosure so an elaborate

calibration method is required to ensure proper mounting of sensors.

This approach also requires an accurate sensor model. The side-lobes of the sensor are ignored in this evaluation but these must be characterized in order to understand how these sensors would work in a real implementation. Also, since the evaluation of measurements relies heavily on a geometric model of the sensor, this model must be verified in order to achieve the desired results.

There is also an issue of "tolerance-stacking" in placing the targets on to the road. The position and orientation of the host vehicle on the road is measured with one sensor (i.e. DGPS/map-based system, machine vision, etc.) which has an associated error. Using range sensors, the position of the target relative to the host is measured. Then the target is placed on the road based on the position of the host. This means that the position of the target on the road includes the error from the road sensing system in addition to the error of the target sensing system. This "tolerance-stacking" does not exist for vision systems which sense the road as well as the target. For these systems, the road's position and the target position on the road can be determined from the same images. Then the error associated with the road position can be differentiated from the error of the target on the road. This is a strong advantage of vision-based road/target sensing systems.

In this chapter, the sensor placement issues have only been evaluated for straight road segments. The performance of this sensing system will degrade when operating on curved roads. How much the performance will degrade is not known but this could be evaluated by using the techniques outlined here for curved road segments. This issue requires serious investigation prior to implementing such a sensing approach.

Finally, in this evaluation, sensor cross-talk is not evaluated. Cross-talk may or may not be a issue. To evaluate this, tests must be done with at least two real sensors to determine how these sensors operate in the presence of other radar signals. This issue is perhaps the most significant limiting a practical implementation.

4.10 Conclusions

In this chapter, we introduced a new methodology for selecting the desired host vehicle sensor placement for a highway driving environment. Here we developed three plots which facilitated in identifying the ability of a given sensor configuration to meet the virtual bumper sensing requirements. These plots were used to iterate the placement of sensors in order to find a sensor configuration with the desired characteristics. We showed that this type of approach can be used to evaluate the placement of various types of sensor models. Following this design methodology, a sensor configuration was identified that meets the requirements of the virtual bumper using fourteen range radar sensors.

Chapter 5

Simulation

5.1 Overview

In the previous chapters we documented the development of the virtual bumper collision avoidance control strategy. The theoretical background has been discussed and the implementation process has been defined. Sensing requirements have been addressed and the development of the sensor array configuration has been defined. In this chapter, we discuss the testing performed on the virtual bumper. First we discuss the simulation environment developed for this implementation. This simulation is unique in that it test the actual real-time data acquisition and control code using a high fidelity model of the sensors and the truck dynamics. We document the driving scenarios tested and why they were selected. Finally, we present the results for each of the driving scenarios tested.

5.2 Real-Time Simulation Environment

Implementing a control strategy on even a small vehicle can be a dangerous task. When the vehicle being controlled is a Navistar semi-tractor, the risks are even greater. Therefore, it makes sense to develop the control strategy through simulation first and only implement the control algorithms on the real vehicle after they has been fully tested. However, traditional simulation approaches evaluate the control algorithm using a simulation package and after the simulation is successful, the engineer has to then generate the software for the actual application. The simulation often does not help the engineer develop and debugging any of the actual software. Worst yet, the simulation provides the engineer with no insight as to how the practical issues of task scheduling and latency affect the control algorithms. To address these shortcomings, we have developed a real-time simulation environment that simultaneously tests the algorithms and the code in which the algorithms are written.

The real-time simulation environment is outlined schematically in Figure 5.1. In this approach, the virtual bumper algorithm is executed on a VME bus based computer and only the host dynamics, the sensors and the roadway are simulated in a 3D environment. The 3D graphical environment is developed using World Tool Kit (WTK) and runs on an SGI workstation. This simulation calculates sensor readings (GPS, vision and/or range sensors) and passes this data through a network connection to a control computer. The virtual bumper algorithm executing on the control computer takes sensor information from the 3D simulation as if it came from sensors on board the host vehicle.

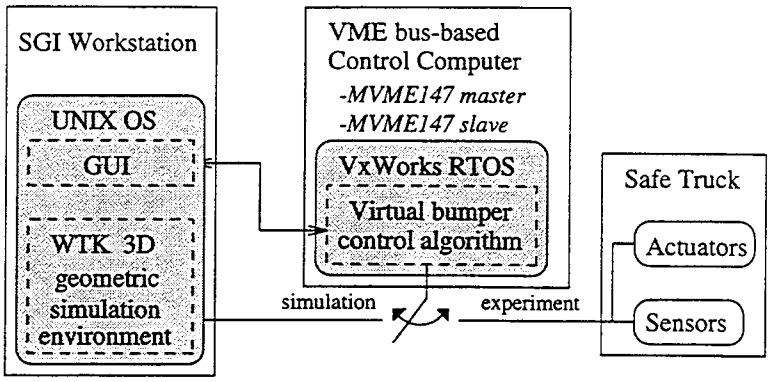


Figure 5.1: Schematic of the real-time simulation environment.

Sensors that read the truck's dynamics are simulated using a lateral dynamic model of the truck and on models of the sensors that have previously been experimentally validated. The algorithm determines the appropriate steering and velocity commands for the situation and passes these commands to the sub-system controllers. The control is then applied to a dynamic model for the host vehicle to determine its position and the position update is broadcast back to the 3D simulation environment. The host is moved to the new position in the simulated world and sensor readings are calculated and transmitted to the control computer. This loop operates continually allowing the virtual bumper algorithm to control the simulated vehicle in real-time.

The VME control computer is nearly identical to the control computer for the Safe Truck testbed (the Navistar 9400). The VME computer is equipped with two Motorola MVME-147 single board computers. Each of these Motorola boards runs the VxWorks real-time operating system. Using the same operating system and architecture ensures that the software developed through simulation will operate correctly when implemented on the real vehicle.

The simulation environment is designed for easily switching from the simulation to the experiment test mode. To facilitate this, the simulated sensor and the real sensor information are formated identically and placed in the same data structure for passing to the virtual bumper algorithm as input. This means that the algorithm uses both the real and the simulated sensor data in the same manner. Also, the control outputs of the virtual bumper go to the velocity and the lateral position controller for both simulation and experiment modes of operation. For simulations, these controllers control a dynamic model of the Navistar truck. For experiments, the outputs of these controllers go instead to the actuators on the Navistar truck. These features in the simulation environment allow for easily switching the virtual bumper from controlling a simulated vehicle to controlling the real truck.

The 3D simulated environment has been developed to allow for realistic sensing of the road

environment. The simulated environment is "sensed" using DGPS, vision and/or range sensors. The lateral position sensor for the truck can be based on any technology. For example, a vision sensor might detect points on the border of the road and passes this information to the VME computer to perform road tracking. The simulated DGPS receiver may pass lane centerline information back to the system, a yaw gyro might send heading angles back, etc. Sensing of other vehicles and obstacles is done using a range sensor. The sensors provided in the simulation can have any specified length and cone angle and can be placed arbitrarily on the host vehicle. Each range sensor determines the range and range rate to any object within its cone angle. These sensor measurements are passed to the VME computer for use by the virtual bumper algorithm.

The simulation environment is also capable of controlling the configuration of the driving scenarios. Various driving scenarios can be created through the simulation by adding target vehicles as desired. Each target vehicle is assigned its own path and velocity. This allows the simulation environment to be used to evaluate how the virtual bumper algorithm works for various driving situations.

From Figure 5.1 we see that there is also a graphical user interface (GUI) on the host for the control computer. This GUI provides the user with a method of controlling the software executed during a simulated run. The interface allows for selection of control parameters, control algorithms and also displays relevant parameters in real-time. The GUI is very useful for controlling and evaluating the performance of the developed software.

Several groups have developed simulation environments to test vehicle control strategies, however few of these simulations possess the advantages of the real-time simulation described here. Two groups have developed very similar approaches. The first group includes Sukthankar, Pomerleau and Thorpe at Carnegie Mellon University [Sukthankar95]. Their simulation is also based on a 3D graphical environment. The sensor models are based on actual sensor implemented on their Navlab II vehicle. The software has been developed such that it uses standard controller function calls to allow for straight forward implementation on the Navlab II. This simulation, however, does not control the simulated vehicle using the actual control computer. They have instead placed their emphasis on detailed sensor models and more advanced driving scenario control. The other simulation environment was developed by Khaled et al [Khaled94] in conjunction with the PROMETHEUS project in Europe. This simulation also uses a 3D graphical environment. Sensors used in this simulation are modeled from sensors used on a working vehicle. The simulated vehicle interfaces with an embedded control computer running a real-time operating system. From the citation it is difficult to say how different this environment is from that described here except that the implementation is developed for a driver advisory system rather than for a vehicle control

algorithm.

5.3 Driving Scenarios

The virtual bumper control algorithm has been developed to operate in a highway driving environment. In this environment, the virtual bumper has a wide range of capabilities. The driving scenarios discussed here were selected so that all of the algorithm's capabilities can be tested.

The capabilities of the virtual bumper can be divided into three main driving functions or tasks. These three functions are associated with the three virtual target forces and can be categorized as follows:

1) Headway control function (associated with F_{long}).

2) Maintaining lateral vehicle spacing function (associated with F_{reflex}).

3) Lane change function (associated with F_{LC}).

Our control algorithm is capable of performing these driving functions under both normal and emergency driving situations. Therefore, the driving scenarios have been selected such that all three of these functions are tested under both normal and emergency driving conditions.

For testing of this control algorithm, eight driving scenarios were selected. The first four of the scenarios represent normal driving conditions and are shown schematically in Figure 5.2. These were selected because they represent scenarios which should test the ability of the virtual bumper to respond as desired under normal situations. The next four are emergency driving scenarios and are shown schematically in Figure 5.3. These again should result in "desirable" collision avoiding maneuvers. Table 5.1 shows which driving functions are tested by each of these driving scenarios. From the table we see that each of the driving functions is tested under both normal and emergency conditions. Therefore, these driving situations should fully test the virtual bumper control algorithm's capabilities across a wide range of situations.

All of the driving scenarios tested are shown in Figures 5.2 and 5.3. For all of the situations, the simulation always begins with the host in the right lane traveling at 25 m/s (55.9 mph, 90 kph). The velocity of the targets was always set to a constant for the tests; these are listed in the figures. In the following discussion, we will describe each of the driving scenarios in detail. We will first outline the normal driving scenarios and then we will discuss the emergency driving scenarios.

5.3.1 Normal driving scenarios

Normal driving scenarios are tested to determine if the virtual bumper provides the desired performance under typical driving situations. For this simulation, four normal driving scenarios

Function	Passed by car	Passing parked car	Overtaking	Waiting to overtake	Drifting car	Passed and cutoff	Stalled car on highway	Approaching traffic jam
	Driving Scenario							
Headway Control				•		△	△	△
Lateral Vehicle Spacing	•	•			△	△		
Lane Change			•	•			△	

• Normal driving conditions.
△ Emergency driving conditions.

Table 5.1: Functions of the virtual bumper tested by each driving scenario.

were selected (see Figure 5.2). These four scenarios are titled: 1) Passed by car, 2) Passing parked car, 3) Overtaking and 4) Waiting to overtake.

The first driving scenario is "Passed by car" (see Figure 5.2a). In this situation, a target vehicle traveling at a constant velocity of 30 m/s (67.1 mph, 108 kph) passes the host in the left lane. The target is initially behind the host outside of the sensor range. The desired response is that the host vehicle should not change its course. It is important to test this scenario to verify that the virtual bumper algorithm is not too sensitive to other vehicles in its environment.

The next driving scenario is "Passing parked car" (see Figure 5.2b). In this situation, a target vehicle is parked on the right shoulder and is initially outside of the host's sensor range. The host approaches the target and passes. The desired response is that the host vehicle does not change its course. Again this situation is important in order to verify that the algorithm is not too sensitive to stationary obstacles in regions adjacent (e.g. the shoulder) to the lane of interest.

The third scenario is "Overtaking" (see Figure 5.2c). In this case, the target is traveling at a constant velocity of 20 m/s (44.7 mph, 72 kph) in the right lane. The target is initially in front of the host outside of the sensor range. Here, the desired response is that the host should approach the target and perform a lane change without any velocity adjustments. This test verifies that the longitudinal and lateral impedance controllers are working together properly.

The last normal driving scenario is "Waiting to overtake" (see Figure 5.2d). For this scenario there are two targets. The first target (T1) is traveling at a constant velocity of 20 m/s (44.7 mph, 72 kph) in the right lane. Target T1 is initially in front of the host outside of the sensor range. The second target (T2) is traveling at a constant velocity of 25 m/s (55.9 mph, 90 kph) and is initially

along side the host in the left lane. Here the desired response should be that the host slows down and tracks the desired headway behind T1 until T2 is far enough ahead to provide room for a lane change. The host then performs a lane change and follows T2. This test verifies proper operation of the headway controller as well as proper operation of the set of lane change heuristics.

Additional scenarios could be developed to more fully test the longitudinal impedance controller. However, this controller has already been tested and presented in the implementation discussion (see chapter 3).

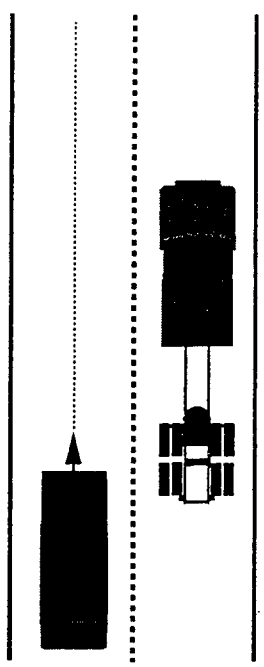
5.3.2 Emergency driving scenarios

Once the virtual bumper exhibits the desired performance for typical driving conditions, then it can be tested in emergency driving situations. Four emergency driving scenarios were selected (see Figure 5.3). These four scenarios are titled: 1) Drifting car, 2) Passed and cut-off, 3) Stalled car on highway and 4) Approaching traffic jam.

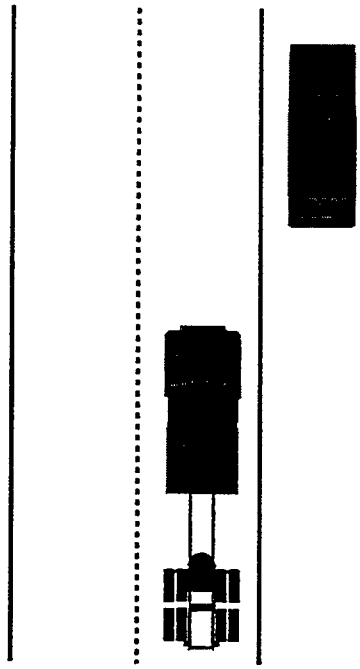
The first scenario is "Drifting car" (see Figure 5.3a). For this case, the target vehicle is traveling at 25 m/s (55.9 mph, 90 kph) in the left lane along side of the host. The target then drifts into the right lane and then returns to the left lane. The desired response should be that the host shifts in the right lane to maintain the desired spacing. Normally the host would also decelerate, but for this test the longitudinal controller was disabled. This test verifies that the lateral impedance controller is capable of maintaining a minimum lateral vehicle spacing.

The next emergency driving scenario is "Passed and cut-off" (see Figure 5.3b). Here the target vehicle travels at a contant velocity of 30 m/s (67.1 mph, 108 kph) and is initially behind the host outside of sensor range in the left lane. The target approaches the host and performs a lane change maneuver into the right lane while it is along side of the host. In order to avoid collision, the host must decelerate as well as swerve towards the shoulder. This test verifies the performance of the lateral impedance controller when working with the longitudinal controller in emergency conditions.

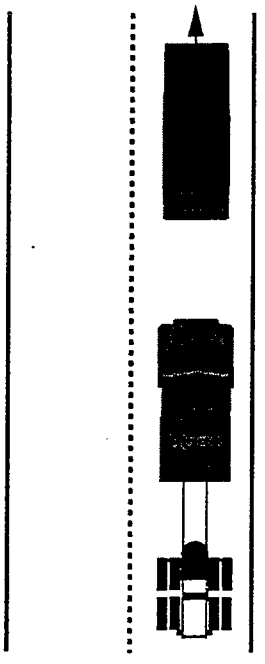
The third situation is "Stalled car on highway" (see Figure 5.3c). In this scenario, the target vehicle is stopped in the right lane of the highway initially outside of the host's sensor range. The desired response should be for the host to execute an emergency lane change and perform the appropriate headway control until the host is safely in the left lane. Headway control should be performed until the host is in the left lane because in this implementation of the virtual bumper, the longitudinal impedance controller operates independently of the lateral impedance controller and thus headway control is always performed when a target is in the longitudinal personal space. The headway control is performed regardless of whether a lane change maneuver is in progress or not. Here, the performance of the lateral and longitudinal impedance controllers during an emergency



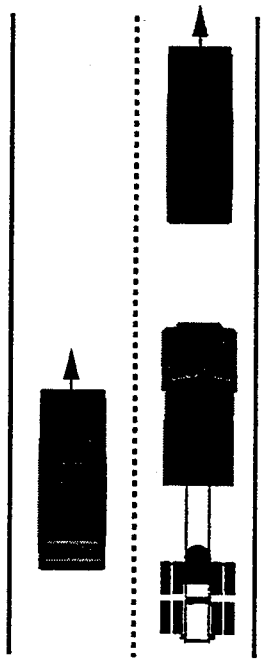
a) Passed by car ($V_{T1} = 30\ m/s$).



b) Passing parked car ($V_{T1} = 0\ m/s$).



c) Overtaking ($V_{T1} = 20\ m/s$).



d) Waiting to overtake ($V_{T1} = 20\ m/s$, $V_{T2} = 25\ m/s$).

Figure 5.2: Normal driving scenarios.

lane change is verified.

The last scenario tested is "Approaching traffic jam" (see Figure 5.3d). Here there are two target vehicles. The first target (T1) is stopped in the right lane and is initially outside of the host's sensor range. The second target (T2) is along side of the host vehicle and moves at the same velocity ($V_{T2} = V_{host}$). As the host approaches T1, it decelerates and the second target also decelerates, shadowing the host's position. This is similar to the scenario which would occur when approaching a traffic jam. The desired response should be that the host perform emergency braking and come to a complete stop. This test demonstrates how the longitudinal controller works during emergency braking conditions.

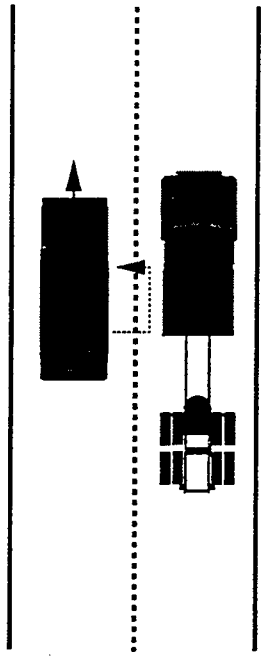
5.4 Simulation Results

Now that we have defined the driving scenarios, we will evaluate the performance of the control algorithms using the simulation. To describe and analyze the results of this simulation, we will use the following four plots:

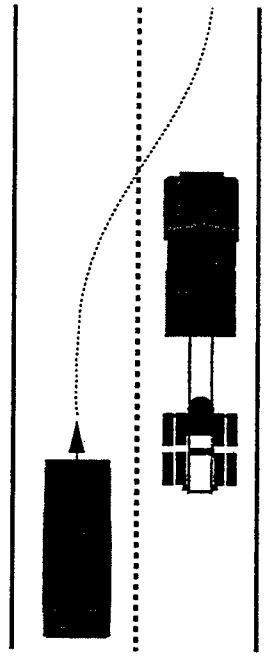
1) Time history of vehicle-road configuration.
2) (R,$\dot{R}$) phase plot.
3) Time history of longitudinal parameters.
4) Time history of lateral parameters.

The (R,$\dot{R}$) phase plots have already been described in section 2.2 and are used for documenting the performance of the longitudinal impedance controller. The time history graphs plot the relevant parameter variations vs. time to which is useful for understanding the system performance as well as the margins of safety during the maneuvers. The time history of the vehicle-road configuration provides a sketch of the vehicles' relative orientation over time on the road and allows a better understanding of system performance relative to the other vehicles. Not All of the plots contain useful information for every driving scenario, so only the appropriate set is presented for each driving scenario.

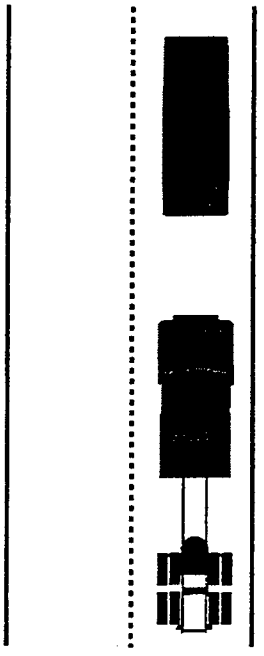
Before discussing the simulation results, the time history of vehicle-road configuration plot (configuration plot for short) needs to be described in more detail. Figure 5.4 shows an example of the configuration plot. This configuration plot is for the "Waiting to overtake" driving scenario. This figure provides an overhead "bird's eye" view of the road segment. In the right margin, there are the labels "0 m" and "285 m" which identify the length of the road segment (285 meters in this case). On the road there are three types of symbols which identify the host (the rectangle), the first target (the triangle) and the second target (the diamond). The length of each symbol is set



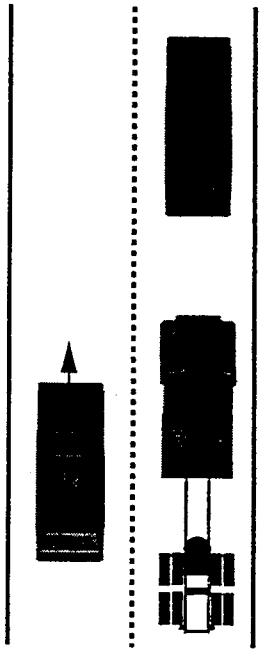
a) Drifting car ($V_{T1} = 25\ m/s$).



b) Passed and cut-off ($V_{T1} = 30\ m/s$).



c) Stalled car on highway ($V_{T1} = 0\ m/s$).



d) Approaching traffic jam ($V_{T1} = 0\ m/s$, $V_{T2} = V_{host}$).

Figure 5.3: Emergency driving scenarios.

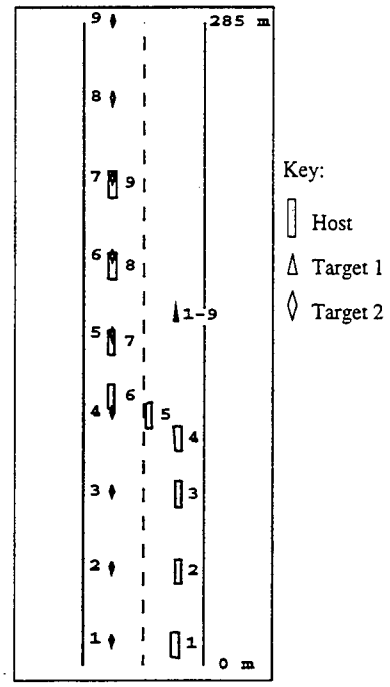


Figure 5.4: Example of a time history of the vehicle-road configuration for "Waiting to Overtake" scenario. $V_{const} = V_{T1}$ and $\Delta T = 6.6$ seconds.

such that the vehicle length is scaled with the length of the road segment and the symbol width is set to maintain the aspect ratio of the vehicle. Next to each of the symbols is a number which is a time stamp. Every ΔT a symbol with the current time stamp is drawn for each vehicle. In this plot, $\Delta T = 6.6$ seconds (see the figure caption). To keep the plot to a reasonable length, the distance traveled for each vehicle is based on the relative velocity between that vehicle and a defined constant velocity (V_{const}), rather than on the absolute velocity. This constant velocity is usually set to the velocity of a target vehicle but can be set to any value. In this plot, $V_{const} = V_{T1}$ (see the figure caption).

A quick explanation of this example of a configuration plot will aid to better understand the information. This plot shows that initially (at time stamp 1) the host is approximately 140 meters behind target 1 (T1) and target 2 (T2) is along side the host in the left lane. The host approaches T1 with little or no deceleration between time stamps 1 and 3. This is shown by approximately constant spacing between the rectangle symbols. Between time stamps 3 and 4 the host begins to decelerate to track the desired headway. The host begins a lane change maneuver between time stamps 4 and 5 which is completed by time stamp 6. Then the host accelerates (expanding spacing between symbols) and follows T2. The time stamps from the configuration plots are also shown on the (R,$\dot{R}$) and time domain plots to allow the information from each of the plots to be correlated.

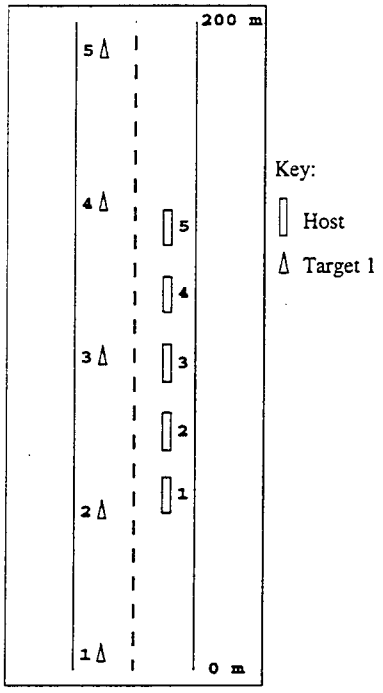


Figure 5.5: Time history of the vehicle-road configuration for "Passed by car" scenario. $V_{const} = 20\ m/s$ and $\Delta T = 4.0$ seconds.

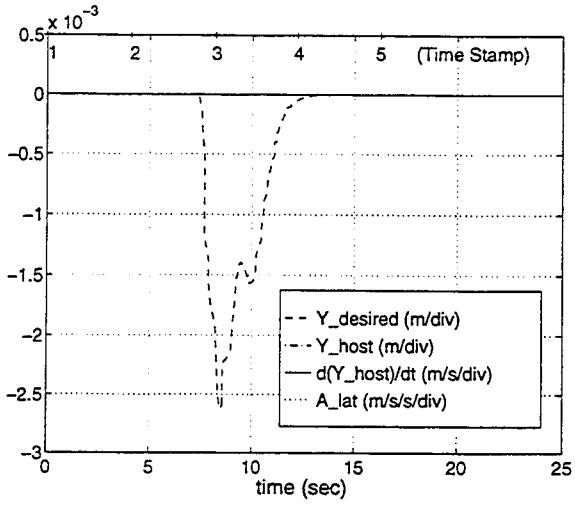


Figure 5.6: Time history of lateral parameters for "Passed by car" scenario

5.4.1 Normal driving scenario results

In this section we document the performance of the virtual bumper control strategy for the normal driving scenarios. These scenarios are documented in section 5.3.1.

"Passed by car" scenario results

The first normal driving scenario is "Passed by car." Figure 5.5 shows the configuration plot for this scenario. From this plot we see no noticeable movement of the host in response to the passing target. The time history for the lateral parameters is shown in Figure 5.6. This figure shows that when the host is passed around time stamp 3, the desired lateral position shifts from the center of the lane a minimal distance (less than a centimeter). The lateral controller does not track this distance because it is too small. In fact, the desired lateral position shift is so small that in a real implementation it would be far less than the noise used to measure the lateral position. All of the other lateral parameters remain at zero throughout this driving scenario test. For this example, the longitudinal time history and the $(R,\dot{R})$ plots are not shown because there is no longitudinal response. From these plots we can conclude that the virtual bumper is not sensitive to passing target vehicles.

"Passing parked car" scenario results

The next driving scenario is "Passing parked car." The configuration plot for this scenario is shown in Figure 5.7. Here we see that the host swerves slightly in the lane as it passes by the target vehicle on the shoulder at time stamp 6. From the time history of the lateral parameters (Figure 5.9), we see that the commanded lateral position ($Y_{desired}$) hits a maximum of about 0.4 m and the actual lateral position (Y_{host}) reaches a maximum of approximately 0.6 m. The vehicle's lateral acceleration (A_{lat}) stays well below the safe limit of $A_{lat_{roll}} = 2.0\ m/s$ defined in section 3.3.1.

The virtual reflexive force which shifted the vehicle in the lane also had a longitudinal component. This is apparent from the negative longitudinal force shown in Figure 5.8. This force decelerates the host down to approximately 24 m/s (53.7 mph, 86.4 kph). As the target is passed, a force is applied to accelerate the host (see section 3.2.1) back to the desired user set velocity of 25 m/s (55.9 mph, 90 kph). Notice that the velocity controller does not track this exactly and the longitudinal force turns off when $V_{host} < 25.0\ m/s$. This response to passing a parked car is similar to that of a human and is an acceptable response for the virtual bumper.

"Overtaking" scenario results

The third situation tested is the "Overtaking" scenario. Figure 5.10 shows the configuration plot for this situation. The host approaches the target and performs a lane change maneuver between time stamps 3 and 5. Then the host continues driving in the left lane and passes the target vehicle.

The (R,$\dot{R}$) phase plot for this situation is shown in Figure 5.11. From this figure we see that the (R,$\dot{R}$) point enters into the personal space for the lane change (crosses the R_{PS_lat} line) between time stamps 3 and 4. Once the target enters the personal space, the lane change maneuver is initiated. At approximately time stamp 4, the target is no longer in front of the host which is shown by the (R,$\dot{R}$) which is shown by the straight line from time stamp 4 upward to the maximum range. Note that this last portion of the R,$\dot{R}$ represents the discontinuity when the target is no longer sensed in front of the host. Remember that the (R,$\dot{R}$) phase plots are only for targets directly in front of the host. Notice from the phase diagram that while the target is in front of the host, the trajectory stays at a constant $\dot{R}$ which means that the host velocity is not adjusted.

Figure 5.12 shows the time history of the lateral parameters. This plot shows that $\dot{Y}_{host} \approx 1.0\ m/s$ during the lane change which means that this is a nominal lane change maneuver. As expected for a nominal lane change maneuver, A_{lat} stays within the safe acceleration limits. The lane change occurs between time stamps 3 and 5 which corresponds to the results from the other plots. This scenario verifies that under normal driving conditions, a lane change can be performed without any adjustments in velocity for headway control.

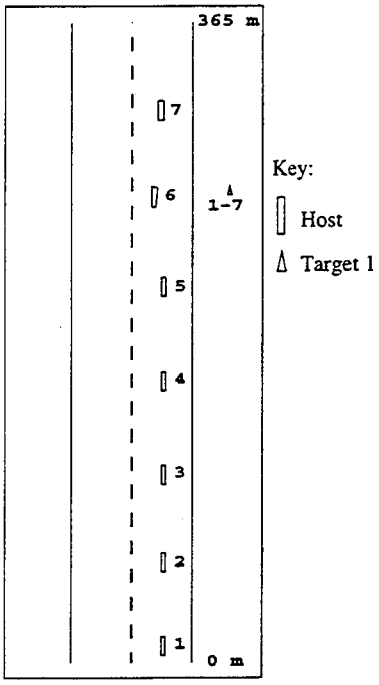


Figure 5.7: Time history of the vehicle-road configuration for "Passing parked car" scenario. $V_{const} = 0$ and $\Delta T = 2.0$ seconds.

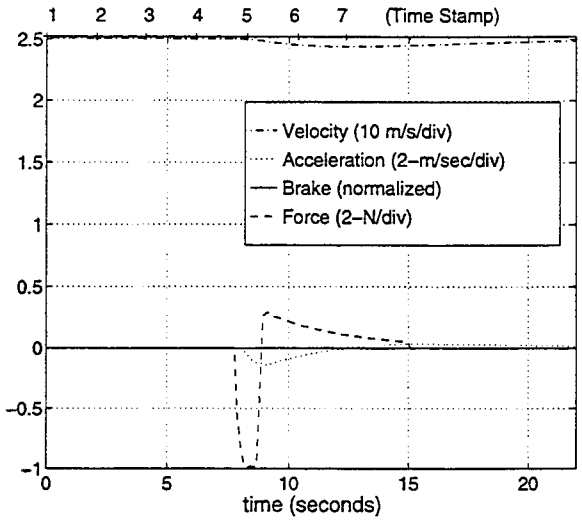


Figure 5.8: Time history of longitudinal parameters for "Passing parked car" scenario.

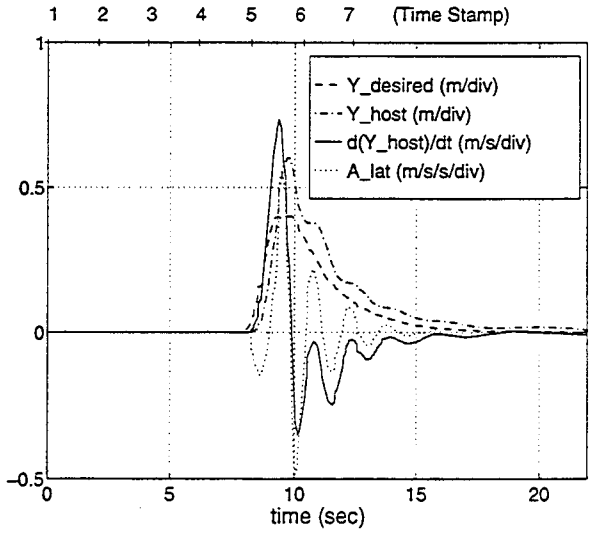


Figure 5.9: Time history of lateral parameters for "Passing parked car" scenario.

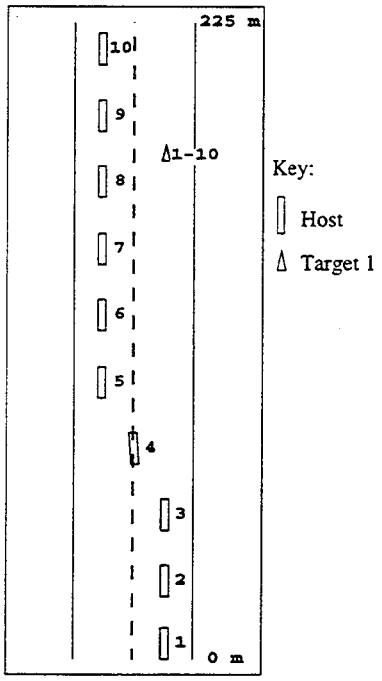


Figure 5.10: Time history of the vehicle-road configuration for "Overtaking" scenario. $V_{const} = V_{T1}$ and $\Delta T = 4.5$ seconds.

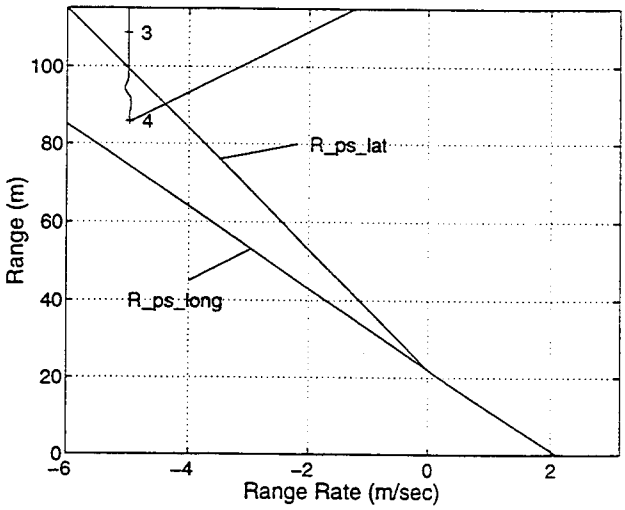


Figure 5.11: $(R,\dot{R})$ phase plot for "Overtaking" scenario.

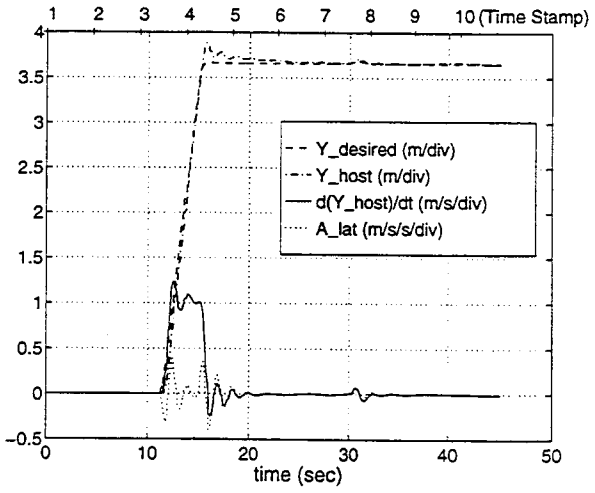


Figure 5.12: Time history of lateral parameters for "Overtaking" scenario.

"Waiting to overtake" scenario results

The last normal driving scenario is "Waiting to overtake." The configuration plot for this scenario is shown in Figure 5.13. The host approaches the first target, T1, from behind with little or no deceleration between time stamps 1 and 3. Between time stamps 3 and 4 the host begins to decelerate to track the desired headway and allows the second target, T2, to pass. Between time stamp 4 and 5, T2 reaches a sufficient distance ahead of the host and the host begins a lane change. The host then follows T2 and accelerates back to the desired velocity.

Figure 5.14 show the time history for the lateral parameters. Up to about the fourth time stamp we see that the host has a slight offset in the lane. This is because T2 is a van which is slightly wider than the car used to tune the lateral personal space. Therefore, for the van the host must shift in the lane to maintain the desired lateral spacing. This figure also shows that a nominal lane change is performed between time stamps 4 and 6. Again, this is a nominal lane change because the lateral path velocity ($\dot{Y}_{host}$) is approximately constant at a value of 1.0 m/s during the lane change maneuver.

The (R,$\dot{R}$) phase plot is shown in Figure 5.15. Here we see that the (R,$\dot{R}$) trajectory enters the lane change personal space (defined by R_{PS_lat}) and then enters the longitudinal personal space (defined by R_{PS_long}) since a lane change maneuver could not be performed (T2 blocked the lane change). At approximately time stamp 3 the host begins to track the desired headway and at time stamp 5 the trajectory jumps to the (R,$\dot{R}$) measurement for target 2. The vehicle settles at a following distance of approximately 60 meters behind T2.

The time history for the longitudinal parameters is shown in Figure 5.16. This plot corresponds to what we would expect based on the (R,$\dot{R}$) plot. Just before time stamp 3 the headway tracking begins and a force is applied which decelerates the host. The velocity decays smoothly down to approximately 21 m/s (50.0 mph, 75.6 kph). At approximately time stamp 5, an acceleration force profile is used to bring the host's desired velocity back to the nominal velocity set by the user at 25.0 m/s (55.9 mph, 90 kph). Notice that the velocity controller does not track the velocity exactly and the force turns off while $V_{host} < 25.0\ m/s$.

From this discussion we see that the virtual bumper provides a very desirable response for this driving scenario. Also, notice here that the lane change maneuver is approximately the same as that for the "Overtaking" scenario even though the circumstances are much different. This test demonstrates that when the lateral and longitudinal impedance controller are combined, they can respond appropriately to a relatively complicated driving scenario.

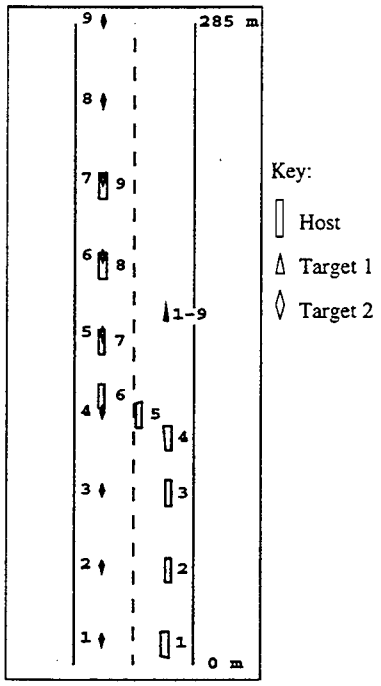


Figure 5.13: Time history of the vehicle-road configuration for "Waiting to overtake" scenario. $V_{const} = V_{T1}$ and $\Delta T = 6.6$ seconds.

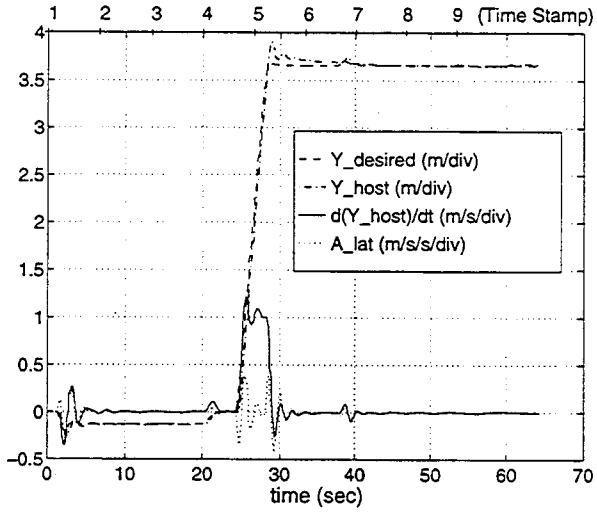


Figure 5.14: Time history of lateral parameters for "Waiting to overtake" scenario.

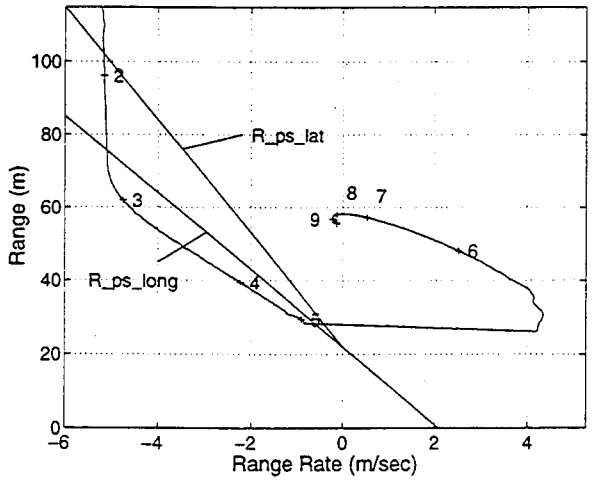


Figure 5.15: $(R,\dot{R})$ phase plot for "Waiting to overtake" scenario.

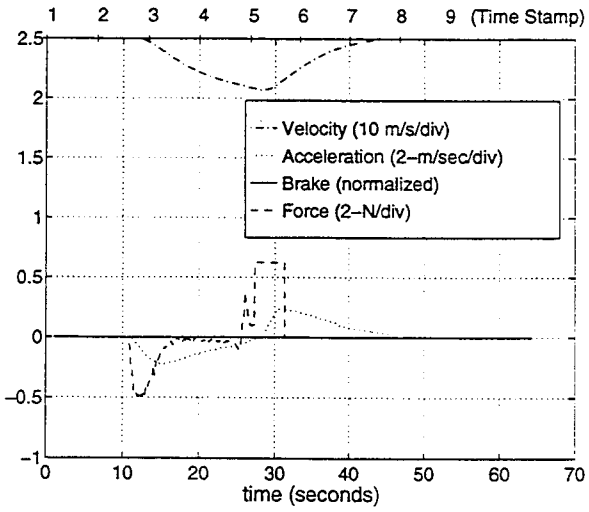


Figure 5.16: Time history of longitudinal parameters for "Waiting to overtake" scenario.

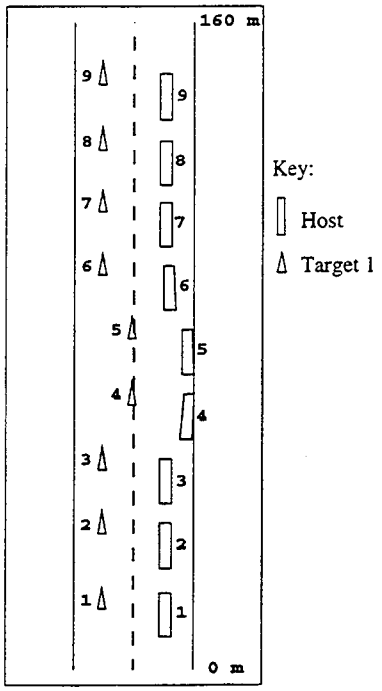


Figure 5.17: Time history of the vehicle-road configuration for "Drifting car" scenario. $V_{const} = 20\ m/s$ and $\Delta T = 3.0$ seconds.

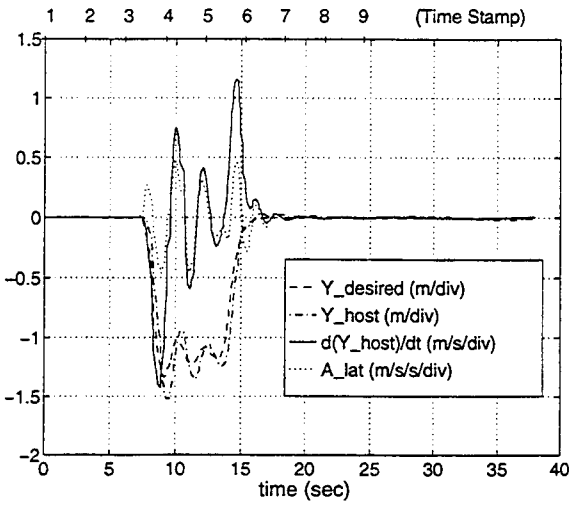


Figure 5.18: Time history of lateral parameters for "Drifting car" scenario.

5.4.2 Emergency driving scenario results

In this section we document the performance of the virtual bumper control strategy for the emergency driving scenarios. These scenarios are documented in section 5.3.2.

"Drifting car" scenario results

The first emergency driving scenario is referred to as the "Drifting car." Figure 5.17 shows the configuration plot for this scenario. From this plot we see that the target vehicle drifts over and stays in the right lane between time stamps 3 and 6. The position of the target is approximately centered at the centerline of the road for time stamps 4 and 5. This causes the host to shift toward the shoulder to maintain the desired lateral spacing. The exact lateral positioning of the host is shown in the time history for the lateral parameters (see Figure 5.18). Here we see that the desired lateral position ($Y_{desired}$) and the actual lateral position (Y_{host}) oscillate around $-1.2\ m$ from the center of the lane between time stamps 3 and 6. This oscillation is due in part to the fact that there is no damping coefficient in the lateral reflexive force (F_{reflex}) due to sensor placement limitations (see chapter 4). Even with the oscillation, the host still does avoid collision and the vehicle is controlled in a stable manner.

"Passed and cut-off" scenario results

The next emergency driving scenario is "Passed and cut-off." Figure 5.19 shows the configuration plot for this scenario. In this figure, the target is overtaking the host up to time stamp 5. Between time stamps 5 and 6 the target begins a lane change while the host is in its "blind-spot." The host swerves to the shoulder to avoid collision and moves back into the lane once the target has passed by time stamp 7.

The time history of the lateral parameters are presented for this scenario in Figure 5.21. Here we see that the host shift towards the shoulder between time stamps 5 and 7. In this situation, the desired lateral position does not oscillate because the lateral force is applied and removed as the target passes. The oscillation in the lateral position in this test is caused by the lateral position controller (see appendix B). This maneuver is considered to be safe because the lateral acceleration (A_{lat}) reaches a maximum magnitude of 0.6 m/s^2 which is well below the safe limit of $A_{lat_{roll}} = 2.0\ m/s$.

Figure 5.20 shows the time history for the longitudinal parameters. From the plot we see that the host begins to decelerate when the target is in the lateral personal space. Then there is a moment when the target is not in front of the host and is not in the lateral personal space (remember that when $\dot{R} > 0$ that the lateral personal space extends in front of the host, see section 2.4.2). This occurs around time stamp 6 and a positive force is applied to the host during this time. Then as the target pulls in front of the host, the headway range is below a minimum desired range and a constant deceleration force is applied. Once the target is outside of this minimum range, the host is accelerated back to the desired velocity. Even with this variation in forces applied to the host notice that the actual deceleration of the host stays relatively constant at about $-.2\ m/s^2$. This type of deceleration level would be very comfortable for a passenger.

From this discussion, we see that the virtual bumper provides safe vehicle control for the "Passed and cut-off" scenario. Here we also see that there is some undesirable fluctuation in the longitudinal forces during the actual "cut-off" portion of the maneuver. These fluctuations are due to the criteria used for deciding when headway tracking should be performed for a target (i.e. currently the criteria is that headway is tracked only for targets directly in front of the host). The fluctuation could be reduced some through tuning this criteria to include vehicles which appear to be shifting into the host's lane. Tuning this criteria is not investigated here because the system is already achieving the desired response.

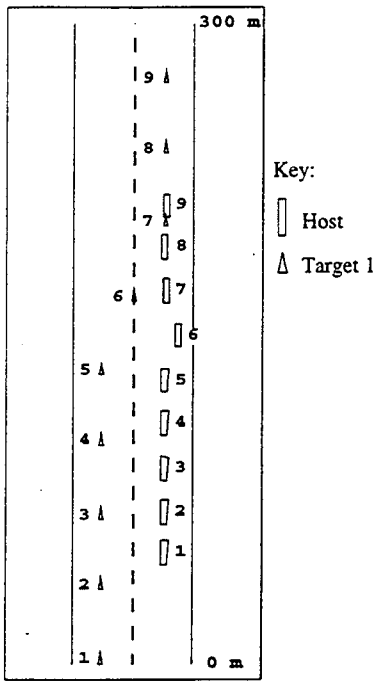


Figure 5.19: Time history of the vehicle-road configuration for "Passed and cut-off" scenario. $V_{const} = 15\ m/s$ and $\Delta T = 2.0$ seconds.

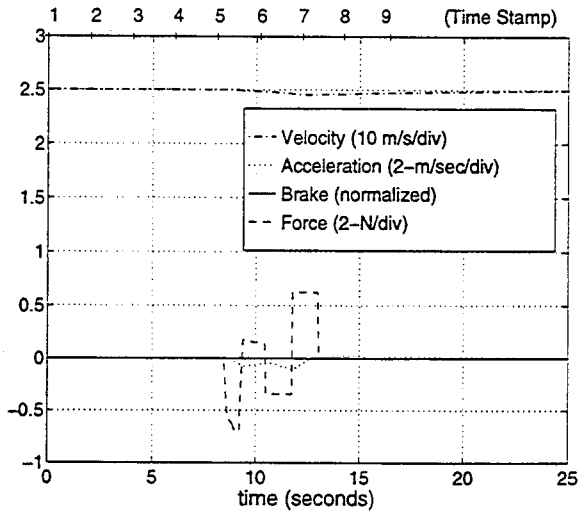


Figure 5.20: Time history of longitudinal parameters for "Passed and cut-off" scenario.

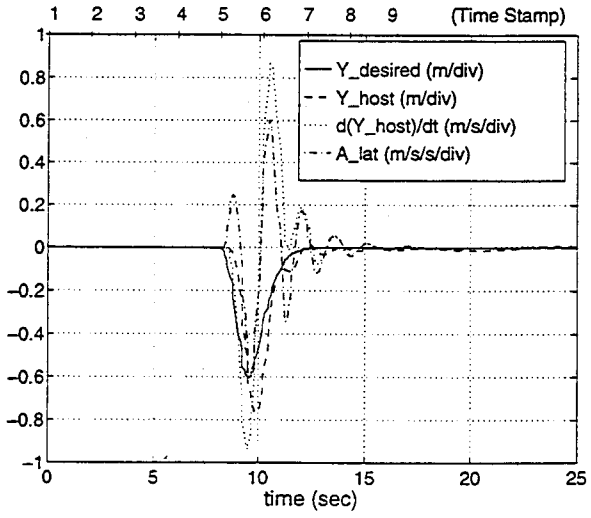


Figure 5.21: Time history of lateral parameters for "Passed and cut-off" scenario.

"Stalled car on highway" scenario results

The third emergency driving scenario is the "Stalled car on highway." Figure 5.22 shows the configuration plot for this scenario. In this figure, the host approaches a target which is stopped in the right lane. Between time stamps 5 and 7 the host performs a lane change to avoid collision and passes the target using the left lane.

Figure 5.23 presents the time history for the lateral parameters. This figure shows that an emergency lane change maneuver is performed between time stamps 5 and 7. It is an emergency lane change because the lateral path velocity during the maneuver is approximately 2.0 m/s. Also notice that there is an additional shift in the lane as the host passes the target vehicle. This is caused by the lateral reflexive force. The additional lateral shift is similar to the lateral shift during the "passed parked car" scenario. This makes intuitive sense as the relative positioning and velocities of the target and the host is the same during the shifting for both scenarios.

The $(R,\dot{R})$ plot for this scenario is shown in Figure 5.24. Here we see that when the target is first sensed, the $(R,\dot{R})$ point is inside the region were the nonlinear longitudinal force is applied. This is also within the region which triggers an emergency lane change maneuver. Then at about time stamp 6 the target is no longer in front of the host. This discontinuity in the range sensor measurements is shown by the straight line from approximately time stamp 6 to the right and up to the maximum range.

The time history for the longitudinal parameters are presented in Figure 5.25. The plot shows that the a maximum deceleration force is applied to the host around time stamp 5. The velocity controller begins to track these changes in velocity around time stamp 6 and approximately a 50% braking level is applied. Now the host is in the left lane and a positive longitudinal force is applied to accelerate the host. As the host passed the target, the lateral spacing is too small (for the high relative velocities) and a deceleration force is applied as the host swerves in the lane. Then a positive longitudinal force is again applied to the host to bring it back to the desired velocity. During this maneuver, the host's minimum velocity is 20 m/s (44.7 mph, 72 kph).

These plots show that the virtual bumper is capable of performing emergency lane change maneuvers while keeping the vehicle within its safe control limits.

"Approaching traffic jam" scenario results

The last driving scenario evaluated is "Approaching traffic jam." Figure 5.26 shows the configuration plot for this scenario. In this figure, the host approaches target one (T1) which is stopped in the right lane. During the maneuver, a second target (T2) shadows the velocity of the host and is in the left lane. In this situation, the host is unable to perform a lane change and must apply emergency

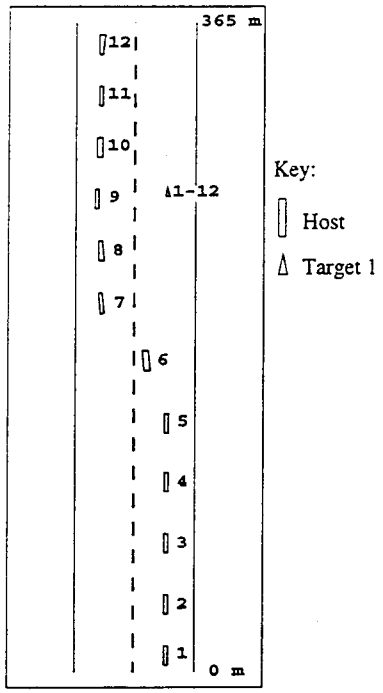


Figure 5.22: Time history of the vehicle-road configuration for "Stalled car on highway scenario. $V_{const} = 0\ m/s$ and $\Delta T = 1.3$ seconds.

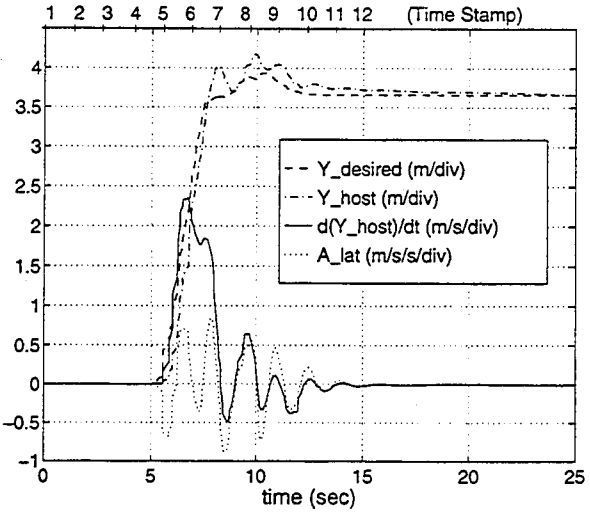


Figure 5.23: Time history of lateral parameters for "Stalled car on highway" scenario.

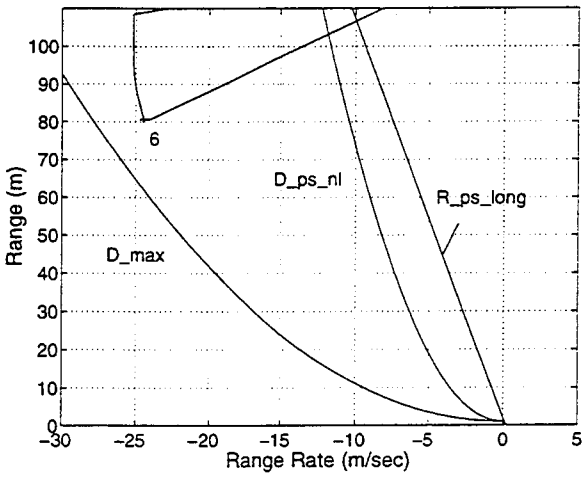


Figure 5.24: $(R,\dot{R})$ phase plot for "Stalled car on highway" scenario.

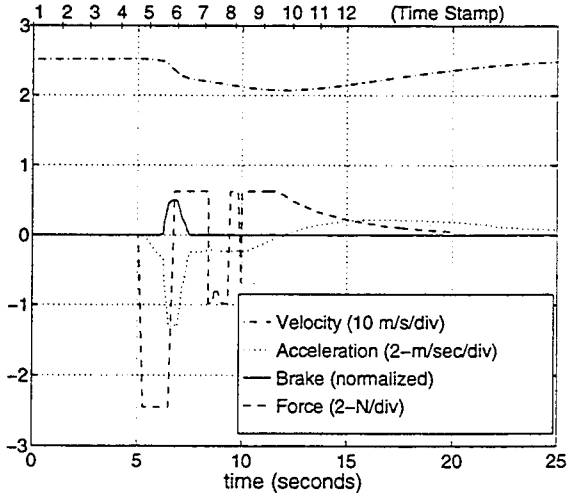


Figure 5.25: Time history of longitudinal parameters for "Stalled car on highway" scenario.

braking levels to avoid collision. From the figure it is difficult to determine at which point the host begins to decelerate. However, it is clear that host comes to a stop prior to collision with the target.

Figure 5.27 is the lateral parameter plot in the time domain. The only significance of this plot is that the host shifts slightly in the lane (about $-.12\ m$) to maintain the desired lateral spacing with T2. In this scenario, T2 is a van which is slightly larger than the car and accounts for the additional spacing requirement.

The $(R,\dot{R})$ phase plots are presented in Figure 5.28. Here we see that the stopped target enters sensor range between time stamps 4 and 5. When the target is first sensed, the $(R,\dot{R})$ point is in the nonlinear force region. Since a lane change is not possible, the host is decelerated along approximately a constant deceleration curve (represented by the parabolic shape, see section 2.2) using nonlinear longitudinal forces. Between time stamps 8 and 9, the trajectory crosses into the linear force region and the vehicle is brought to a stop ($\dot{R} = 0$).

The longitudinal parameter plot in the time domain is presented in Figure 5.29. The figure shows that at approximately time stamp 4, a maximum deceleration force is applied to the host. After some delay, the velocity controller begins to track the velocity commands by applying full braking levels to the vehicle. This results in deceleration levels which peak at approximately $-4.5\ m/s^2$ and average approximately $-4.0\ m/s^2$. The host velocity is brought to zero short of the desired final headway so an acceleration force is applied to reduce the headway. After the vehicle is stopped for a moment, a low level of braking is applied due to a nonlinearity in the velocity controller.

This test verifies that the virtual bumper is capable of avoiding collisions through application of emergency braking.

5.5 Conclusion

In this chapter we have documented the performance of the virtual bumper. For testing this algorithm a unique real-time simulation environment was developed. The driving scenarios evaluated that were selected so that the functionality of the control strategy could be fully tested. These scenarios were defined to allow for testing of both normal and emergency situations. The system performance was documented using four plots. One of these, the time history of the vehicle-road configuration (or configuration plot) was introduced. This plot allows for correlation of data in all of the plots to assist in a better understanding of the results. The evaluation demonstrates that the virtual bumper provides the desired system response in both normal and emergency driving conditions.

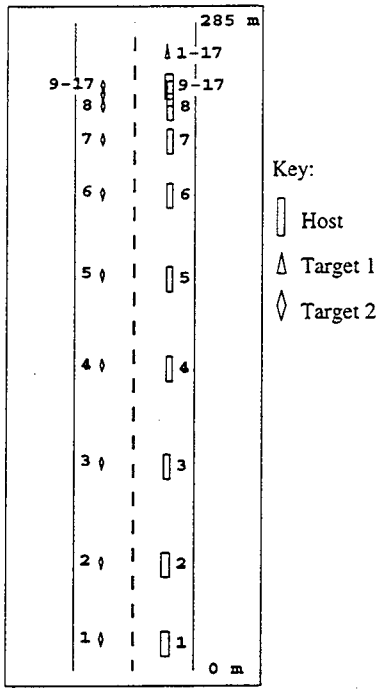


Figure 5.26: Time history of the vehicle-road configuration for "Approaching traffic jam" scenario. $V_{const} = 0\ m/s$ and $\Delta T = 1.6$ seconds.

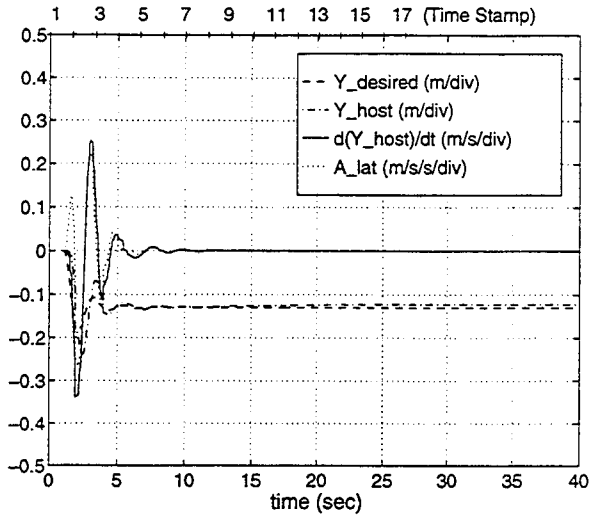


Figure 5.27: Time history of lateral parameters for "Approaching traffic jam" scenario.

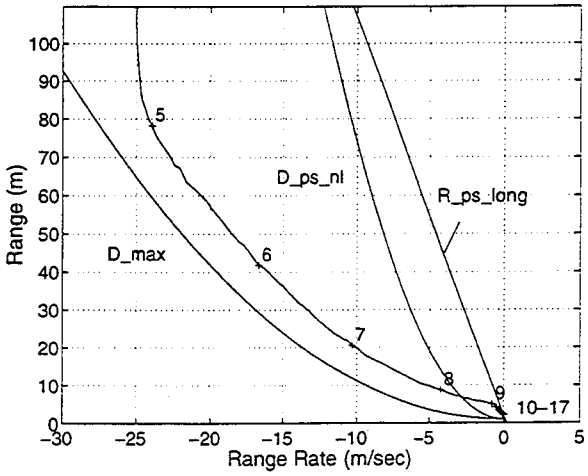


Figure 5.28: $(R,\dot{R})$ phase plot for "Approaching traffic jam" scenario.

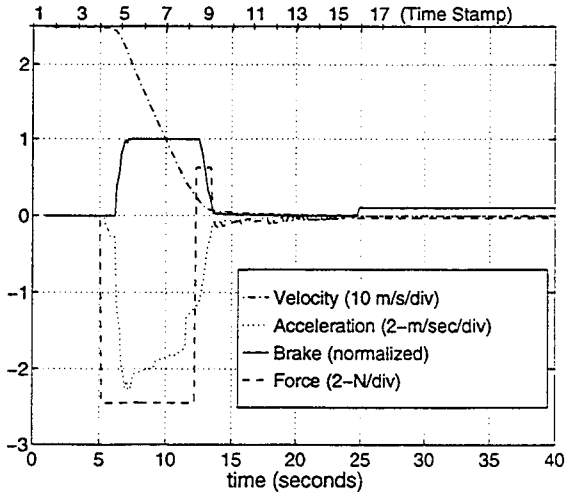


Figure 5.29: Time history of longitudinal parameters for "Approaching traffic jam" scenario.

Chapter 6

Discussion and Conclusion

Summary

In this document we have reported on our development of the virtual bumper algorithm for highway vehicles. This algorithm is a 2-dimensional control strategy that provides both steering and throttle/braking actuation to maneuver a vehicle in order to avoid obstacles and other vehicles. The approach is based on impedance control combined with a set of heuristics. We have designed the virtual bumper to be capable of responding to both normal and emergency driving conditions. Under all circumstances, the vehicle dynamic limits are considered and the vehicle executes appropriate combinations of steering, braking and throttle control within constraints.

The virtual bumper is implemented using two separate impedance control loops. One loop is the longitudinal impedance control loop and controls the velocity of the host. This loop acts to maintain the desired headway to target vehicles ahead as well as to provide emergency braking. The other loop is the lateral impedance control loop with a set of heuristics integrated into its feedback loop. This loop is responsible for maintaining lateral spacing between the host and adjacent target vehicles as well as performing lane change maneuvers. The heuristics provide a mechanism for deciding which is the desired lane for travel and controls the lane change maneuver for both normal and emergency driving conditions. When these loops execute together the host is provided with a control subsystem that steers and accelerates/brakes the vehicle to avoid collisions (or at least to mitigate the effect of a collision).

Most of the work done to date on control strategies for highway vehicles has been developed in a traditional simulation environment. In order to expedite development, the virtual bumper was developed in a real-time simulation environment. In this simulation environment, the control software executes on a control computer running a real-time operating system. The simulation allows us to evaluate much more than the virtual bumper algorithm. For example, with this approach we were able to evaluate sensor latency issues, controller bandwidth requirements, process timing issues, etc. In addition, when the simulation testing is completed, this software is ready and debugged for implementation on the real vehicle. Our simulation environment moves us rapidly towards a real world implementation.

For this first investigation of the virtual bumper, objects in the environment are sensed using an array of radar range sensors. We introduce a methodology for evaluating the sensor placement on a vehicle driving in a highway environment. This approach involves a graphical tool which characterizes the ability of a given sensor configuration to meet our sensing requirements. We then use an iteration process to determine the final sensor configuration – an array of fourteen radar

units.

To test the functionality of the virtual bumper, several driving scenarios were evaluated. The scenarios consider both normal driving situations as well as emergency driving conditions. The normal driving scenarios demonstrated that the control algorithm operates the vehicle similar to the way a human would. This is important because a comfortable and a predictable (i.e. intuitive) system response is required for achieving driver acceptance. From the emergency scenarios we see that this strategy is capable of reacting appropriately while maintaining safe acceleration/deceleration levels for the vehicle. From the evaluation it is apparent that the virtual bumper can provide safe vehicle control for a broad range of driving situations.

Future directions

The next phase of this work is to move the virtual bumper towards an implementation on the SafeTruck. This is a major task and there are still many issues which must be addressed to facilitate success. These issues are related to the longitudinal controller, the lateral controller and the sensing approach.

The virtual bumper algorithm has been developed adequately for normal driving conditions through simulation and is ready for implementation on a real vehicle. When it is desired to operate the virtual bumper in emergency driving scenarios, the algorithm needs further development in the simulation environment for safety considerations. The lateral dynamic model used in the simulation environment is very accurate but it does not address dynamic changes which occur during emergency braking (i.e. brakes fully engaged, see Appendix B for more detail on the dynamic model). Therefore, for these future simulations to be more accurate, the lateral dynamic model should be updated to consider longitudinal decelerations (i.e. braking).

In order for the longitudinal impedance control loop to be fully functional, a velocity controller which applies braking, needs to be developed for the SafeTruck testbed. Also, the fact that the SafeTruck testbed does not as of yet have the capability to apply variable levels of braking needs to be addressed. The preferable solution is to incorporate a hardware change to allow for variable braking levels. If this is not an option, then as a minimum the effect of the braking controller on the control algorithm should be studied.

One of the most important topics to investigate for the lateral impedance controller is the effect of longitudinal accelerations on the lateral dynamic model. Here we need to develop a lateral dynamic model that is cross-coupled with the longitudinal dynamic model. This model is required for a better understanding of how safe the virtual bumper really is in emergency driving situations. Using this model, the lateral acceleration levels can be evaluated for emergency lane changes for

various velocities and braking levels. Considering the new model, the lateral path limits should be re-evaluated. Based on the results, new lateral force limits should be determined. The new force limits should then be set considering both the forward velocity and acceleration instead of only the velocity. Developing this dynamic model is an essential step towards safely implementing a virtual bumper algorithm capable of emergency driving maneuvers.

The lateral controller currently operating on the SafeTruck is satisfactory for initial implementations of the virtual bumper. However, if we desire that the algorithm to be capable of avoiding side-swipes or reacting appropriately to being cut-off, a new lateral controller is required. This controller must be capable of maintaining the desired lateral position of the host in the lane rather than aiming the vehicle at a point ahead in the road. The controller must also be capable of operating across a range of velocities from 0 mph up to the maximum velocity.

The heuristics guiding the lane change system also requires some further investigation. As designed, this system is capable of operating under a wide range of driving conditions. However, we believe that as requirements of the system become more involved, the complexity of this system will grow and lose its effectiveness. We recommend that future research investigate replacing this system with either a neural network system, a fuzzy logic controller or a combination of both. It appears to be a problem to which these technologies are well suited. A more complicated interaction between the lateral and longitudinal controllers could then be developed. For example, the longitudinal force decelerating a vehicle could be inhibited if we know that the lane change maneuver alone is adequate to avoid collision.

Also more complicated driving environments for the virtual bumper need to be investigated. The first environment to consider is the highway near an exit or on ramp. The strategy documented here operates on a highway but does not address the ramps. If the vehicle is to travel more than a few miles, this situation must be addressed. Future work should also look into the city driving environment and the intersection problem. Initial studies should investigate how applicable this concept is for such an environment. Consideration of the merging scenario is a natural one for the next step.

In order to implement the virtual bumper, the host must be provided with information about its environment. In this document the environment is sensed using range radar sensors. We have documented several of the limitations of using radar sensors. If we move forward with this approach, we must investigate these limitations. First, we need an accurate sensor model. The side-lobes of the sensor are ignored in the placement evaluation but these must be characterized for a better understanding of how these sensors work in a real implementation. Second, we must evaluate whether sensors can be mounted with specified orientation angles and still meet the desired

accuracy to provide useful information for the algorithm. This will require developing a sophisticated calibration process. Next, we must evaluate the effects of cross-talk between the sensors. If cross-talk reduces the signal to noise ratio too low, this approach may not be feasible. Finally, sensor placement for curved road segments needs to be evaluated. It may prove that adequate coverage requires too many sensors for curved roads. Using a radar sensor to sense the environment has several implementation issues to overcome. Even if this approach proves to be the best in the end, it will take a long time to work through all of the design issues. A near term solution to environment sensing needs to be investigated.

A solution that appears to be more plausible in the near future is using a Differential Global Positioning System (DGPS) and a local area network (LAN). Here, all of the vehicles in the environment would be equipped with DGPS and connected to the same LAN. The vehicles would be in constant communication with one another, providing position information as well as trajectory. Safer vehicle control would be facilitated because we could also have access to a vehicle's acceleration levels which are difficult to measure. The accuracy of the DGPS system is high enough ($\pm$10 cm) that it would meet the sensing requirements of the virtual bumper. Within this sensing scheme, fewer radar units would be needed to provide information on targets in specific locations (i.e. cars parked on shoulder, detection of pedestrians, etc.). The sensor data could be fused together to produce a more accurate measurement of the vehicle's environment Currently, we are only capable of accurately testing radar accuracy on static targets. The DGPS/LAN sensing system would allow for accurate dynamic tests of these sensors capabilities.

With proper sensing of the environment, the virtual bumper could be implemented on the SafeTruck testbed in the near future. The algorithm has been developed to a level that with a good sensing system, such as a DGPS/LAN system, the vehicle could be adapted with the virtual bumper for normal driving situations with minimal modifications. However, more rigorous study and validation of dynamic models is required prior to implementation for the emergency driving conditions.

References

[Adams94] L. Adams, "Review of the Literature on Obstacle Avoidance Maneuvers: Braking versus Steering", Tech. Report UMTRI-86649, Research Information and Publication Center, University of Michigan Transportation REsearch Institute, August 1994.

[Alexander96] L. Alexander, M. Donath, M. Hennessey, V. Morellas, and C. Shankwitz, "A Lateral Dynamic Model of a Tractor-Trailer: Experiment Validation", Tech. Report, Mn/DOT Office of Research Administration, November 1996.

[Ame95] Amerigon, "Amerigon Radar Products: A Technology & Product Primer", 1995.

[Anderson90] T. Anderson and M. Donath, "Animal Behavior as a Paradigm for Developing Robot Autonomy", *Robotics and Autonomous Systems*, vol. 6, pages 145–168, 1990.

[Bajikar97] S. Bajikar, A. Gorjestani, P. Simpkins, and M. Donath, "Evaluation of In-Vehicle GPS-Based Lane Position Sensing for Preventing Road Departure", *IEEE Conference on Intelligent Transportation Systems, Boston*, November 1997.

[Bodor97] R. Bodor, V. Morellas, M. Donath, and D. Johnson, "In-Vehicle GPS-Based lane Sensing to Prevent Road Departure", *Proceedings of the Third Annual World Congress on Intelligent Transportation Systems, ITS America, Orlando*, November 1997.

[Chakraborty95] S. Chakraborty and D. Smedley, "Adaptive Cruise Control for Heavy Duty Vehicles", *ITS America 1995 Annual Conference*, pages 145–150, 1995.

[Chapius95] R. Chapius, A. Potelle, J.L. Brame, and F. Chausse, "Real-Time Vehicle Trajectory Supervision on the Highway", *International Journal of Robotic Research*, vol. 14, pages 531–542, December 1995.

[Choi95] S. Choi and J. Hendrick, "Vehicle Longitudinal Control Using an Adaptive Observer for Automated Highway Systems", *Proceedings of the American Control Conference*, vol. 5, pages 3106–3110, 1995.

[Dickmanns86] E. Dickmanns and A. Zapp, "A Curvature-Based Scheme for Improving Road Vehicle Guidance by Computer Vision", *In Proceedings of SPIE*, pages 161–168, October 1986.

[Dickmanns90] E. Dickmanns, B. Mysliwetz, and T. Christians, "An Integrated Spatio-Temporal Approach to Automatic Visual Guidance of Autonomous Vehicles", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, no. 6, pages 1273–1284, 1990.

[Dickmanns94] E. Dickmanns, R. Behringer, D. Dickmanns, T. Hildebrandt, M. Maurer, F. Thomanek, and J. Schiehlen, "The Seeing Passenger Car 'VaMoRs-P'", *Proceedings of the Intelligent Vehicles '94 Symposium*, pages 68–73, 1994.

[Doyle92] J. Doyle, B. Francis, and A. Tannenbaum, "Feedback Control Theory", Macmillan Publishing Company, 1992.

[Eat96] Eaton Vorad Technologies, "EVT-200 Radar System Specifications", 1996.

[Ervin86] R. Ervin, "Dependence of Truck Roll Stability on Size and Weight Variables", *International Journal of Vehicle Design*, vol. 7, pages 192–208, 1986.

[Fancher93] P. Fancher, Z. Bareket, and G. Johnson, "Predictive Analysis of the Performance of a Headway Control System for Heavy Commercial Vehicles.", *In Proceedings of the 13th IAVSD Symposium on the Dynamics of Vehicles on Roads and Tracks*, pages 128–141, 1993.

[Fancher94a] P. Fancher and Z. Bareket, "Controlling Headway Between Successive Vehicles.", *In Proceedings of the Association for Unmanned Vehicles Systems Annual Meeting*, pages 149–157, 1994.

[Fancher94b] P. Fancher and Z. Bareket, "Evaluating Headway Control Using Range Versus Range-Rate Relationships", *Vehicle System Dynamics*, pages 576–596, 1994.

[Fancher94c] P.S. Fancher and R.D. Ervin, "Intelligent Cruise Control: Performance Studies Based Upon An Operating Prototype", *Proceedings of the IVHS America 1994 Annual Meeting*, pages 391–399, 1994.

[Fraichard92] T. Fraichard and C. Laugier, "Kinodynamic Planning in a Structured and Time-Varying 2D Workspace", *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, pages 1500–1505, 1992.

[Franke94] U Franke, S. Mehring, A. Suissa, and S. Hahn, "The Daimler-Benz Steering Assistant - A Spin off from Autonomous Driving", *Proceedings of The Intelligent Vehicles '94 Symposium*, pages 120–124, 1994.

[Garnier95] P. Garnier, C. Novales, and C. Laugier, "An Hybrid Motion Controller for a Real Car-Like Robot Evolving in a Multi-Vehicle Environment", *Proceedings of the Intelligent Vehicles '95 Symposium*, pages 326–331, 1995.

[Gillespie92] T. Gillespie, "Fundamentals of Vehicle Dynamics", Society of Automotive Engineers, Inc., 1992.

[Grupen95] Roderic Grupen, Chris Connolly, Kamal Souccar, and Wayne Burlenson, "Toward a Path Co-Processor for Automated Vehicle Control", *Proceedings of the Intelligent Vehicles '95 Symposium*, pages 164–169, 1995.

[Hassoun93] M. Hassoun and C. Laugier, "Towards a Real-Time Architecture to Control an Autonomous Vehicle in Multi-Vehicle Environment", *Proceedings of the 1993 International Conference on Intelligent Robots and Systems*, pages 1134–1140, 1993.

[Hatipoglu95] C. Hatipoglu, U. Ozguner, and K. Unyelioglu, "On Optimal Design of a Lane Change Controller", *Proceedings of the Intelligent Vehicles '95 Symposium*, pages 436–441, 1995.

[Hennessey93] M. Hennessey and M. Donath, "Generalized Impedance Control: Experiments on Constrained Motion in the Presence of Obstacles", *1993 Symposium on Mechatronics, ASME, NY*, November 1993.

[Hennessey94] M. Hennessey, M. Donath, C. Shankwitz, J. Swanson, and T. Morris, "Simulations of Lateral Guidance Control of a Semi Tractor-Trailer on the Mn/Road Pavement Testing Facility: Basic Control Strategy", Tech. Report TOC#71783, Mn/DOT Office of Research Administration, November 1994.

[Hennessey95] M. Hennessey, C. Shankwitz, and M. Donath, "Sensor-based Virtual Bumpers for Collision Avoidance: Configuration Issues", *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 2592, pages 48–59, 1995.

[Howss94] Alfred Howss, "Realization of an Intelligent Cruise Control System Utilizing Classification of Distance, Relative Speed and Vehicle Speed Information", *Proceedings from the Intelligent Vehicles '94 Symposium*, pages 7–12, 1994.

[Jochem95] T. Jochem, D. Pomerleau, and C. Thorpe, "Vision Guided Lane Transition", *Proceedings of the Intelligent Vehicles '95 Symposium*, pages 30–35, 1995.

[Jossi95a] D. Jossi, "Six Degree of Freedom Impedance Controlled Robot Manipulation For Tasks Involving Uncertainty and Rigid Contact", Master's thesis, Dept of Mech Eng, University of Minnesota, 1995.

[Jossi95b] D. Jossi and M. Donath, "Implementing Contact Tasks Involving Position Uncertainty with Impedence Controlled Robots", *Proceedings of the 1995 IEEE International Conference on Systems, Man and Cybernetics, Vancouver*, vol. 2, pages 1699–1704, October 1995.

[Kageyama95] Ichiro Kageyama and Yoshiyuki Nozaki, "Control Algorithm for Autonomous Vehicle with Risk Level", *Proceedings of the Second World Congress on Intelligent Transport Systems, '95 Yokohama*, pages 1284–1288, 1995.

[Kamiya96] H. Kamiya, Y. Nakamura, Y. Fujita, S. Matsuda, T. Tsuruga, and K. Enomoto, "Intelligent Technologies of Honda ASV", *Proceeding of the 1996 IEEE Intelligent Vehicles Symposium*, pages 236–241, September 1996.

[Kawashima95] Shinobu Kawashima, Kajiro Watanabe, and Kazuyuki Kobayashi, "Traffic Condition Monitoring by Kalman Filtering of the Incomplete Laser Radar Signals", *Proceedings of the Second World Congress on Intelligent Transport Systems, '95 Yokohama*, pages 1019–1022, 1995.

[Kempf92] R. Kempf, "MVMA/NHTSA/SAE Heavy Truck Round Robin Brake Test", *SAE Technical Paper Series*, no. 922484, pages 1–12, 1992.

[Khaled94] O. Khaled, N. Fort, M. Hassoun, and C. Laugier, "Real-Time Decision System for the Prolab II Demonstrator", *Proceedings of the Intelligent Vehicles '94 Symposium*, pages 485–491, 1994.

[Khatib86] O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", *International Journal of Robotics Research*, vol. 5, no. 1, pages 90–98, 1986.

[Krogh84] B.H. Krogh, "A Generalized Potential Field Approach to Obstacle Avoidance Control", *Proceedings of SME Conference on Robotics Research*, pages 1–15, 1984.

[Kyriakopoulos95] K. Kyriakopoulos, P. Kakambouras, and N. Krikelis, "Potential Fields for Nonholonomic Vehicles", *Proceedings of the 10th IEEE International Symposium on Intelligent Control*, pages 461–465, 1995.

[Lenarchand94] C. Lenarchand and F. Coffin, "Conception Method of Man Machine Interface for an Automotive Vehicle Transversal Driving Assistance System", *Proceedings of 1st World Congress on Applications of Transport Telematics and Intelligent Vehicle Highway Systems*, vol. 4, pages 1820–1827, 1994.

[Liao97] C. Liao and M. Donath, "Generalized Impedance Control of a Redundant Manipulator for Handling Tasks with Position Uncertainty While Avoiding Obstacles", *Proceedings of the 1997 IEEE ICRA, Albuquerque*, pages 1073–1079, April 1997.

[MacAdam85] C. MacAdam and P. Fancher, "Study of the Closed-loop Directional Stability of Various Commercial Vehicle Configurations", *Dynamics of Vehicles on Roads and on Tracks, Proceedings of IAVSD Symposium (International Association for Vehicle System Dynamics) 9th*, pages 367–382, 1985.

[Morellas95] V. Morellas, J. Minners, and M. Donath, "Implementation of Real Time Spatial Mapping in Robotic Systems Through Self-Organizing Neural Networks", *Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '95), Human Robot Interaction and Cooperative Robots, Pittsburgh*, vol. 1, pages 277–284, August 1995.

[Morellas97] V. Morellas, T. Morris, L. Alexander, and M. Donath, "Preview Based Control of a Tractor Trailer Using DGPS for Preventing Road Departure", *IEEE Conference on Intelligent Transportation Systems, Boston*, November 1997.

[Najm95] W. Najm, M. Nironer, J. Koziol, J. Wang, and R. Knipling, "Synthesis Report: Examination of Target Vehicular Crashes and Potential ITS Countermeasures", Tech. Report DOT HS 808 263, DOT-VNTSC-NHTSA-95-4, U.S. Department of Transportation, National Highway Traffic Safety Administration, June 1995.

[Najm96] W. Najm, M. Mironer, and P. Yap, "Dynamically-Distinct Precrash Scenarios of Major Crash Types", Tech. Report DOT-VNTSC-HS621-PM-96-17, U.S. Department of Transportation, National Highway Traffic Safety Administration, June 1996.

[Navin86] F. Navin, "Truck Braking Distance and Speed Estimates", *Canadian Journal of Civil Engineering*, vol. 13, no. 4, pages 412–422, 1986.

[Niehaus94] A. Niehaus and R. Stengel, "Rule-Based Guidance for Highway Driving in the Presence of Uncertainty", *Journal of Dynamic Systems Measurement and Control - Transactions of the ASME*, vol. 116, pages 668–674, December 1994.

[Noll95] Detlev Noll, Martin Werner, and Werner von Seelen, "Real-Time Vehicle Tracking and Classification", *Proceedings of The Intelligent Vehicles '95 Symposium*, pages 101–105, 1995.

[Palmquist93] U. Palmquist, "Intelligent Cruise Control and Roadside Information", *IEEE Micro*, vol. 13, no. 1, pages 20–28, 1993.

[Peng90] H. Peng and M. Tomizuka, "Vehicle Lateral Control for Highway Automation", *Proceedings of the American Control Conference*, pages 788–794, 1990.

[Peng94] H. Peng, W. Zhang, M. Tomizuka, and S. Shladover, "Reusability Study of Vehicle Lateral Control System", *Vehicle System Dynamics*, vol. 23, no. 4, pages 259–278, 1994.

[Pomerleau91] D. Pomerleau, "Neural Network-Based Vision Processing for Autonomous Robot Guidance", *SPIE, Application of Artificial Neural Networks II*, vol. 1469, pages 121–128, 1991.

[Pomerleau95] Dean Pomerleau, "RALPH: Rapidly Adapting Lateral Position Handler", *Proceedings of the Intelligent Vehicles '95 Symposium*, pages 506–511, 1995.

[Radlinski86] R. Radlinski, "Issues in Truck Braking", *Proceedings – Society of Automotive Engineers P-181*, pages 15–31, 1986.

[Reed87] W. Reed and A. Keskin, "A Comparison of Automobile and Truck Decelerations During Emergency Braking", *Proceedings – Society of Automotive Engineers P-193*, pages 79–84, 1987.

[Reichardt94] D. Reichardt and J. Schick, "Collision Avoidance in Dynamic Environments Applied to Autonomous Vehicle Guidance on the Motorway", *Proceedings of the Intelligent Vehicles '94 Symposium*, pages 74–78, 1994.

[Reichardt95] D. Reichardt, "A Real-Time Approach to Traffic Situation Representation from Image Processing Data", *Proceedings of the Intelligent Vehicles '95 symposium*, pages 500–505, 1995.

[Reichart94] B. Reichart and K. Naab, "Heading Control and Active Cruise Support Driver Assistance System for Lateral and Longitudinal Vehicle Guidance", *Proceedings of the 1st World Congress on Application of Transport Telematics and Intelligent Vehicle Highway Systems*, vol. 4, pages 2126–2133, 1994.

[Sal94] G. Sal and F. Pressi, "Strategies for Intelligent Cruise Control (ICC) Application", *Proceedings of the First World Congress on Applications of Transport Telematics and Intelligent Vehicle-Highway Systems*, pages 2039–2046, 1994.

[Satoh94] H. Satoh and I. Taniguchi, "A New Control Method of Adaptive Cruise Control", *Proceedings of 1st World Congress on Application of Transport Telematics and Intelligent Vehicle Highway Systems*, pages 2094–2101, 1994.

[Schumacher95] R. Schumacher, R. Olney, R. Wragg, F. Landau, and G. Widmann, "Collision Warning System Technology", *The Second World Congress on Intelligent Transport Systems '95 Yokohama*, vol. 3, pages 1138–1145, 1995.

[Schwerberger94] W. Schwerberger, "Autonomous Intelligent Cruise Control in Commercial Vehicles", *Proceedings of 1st World Congress on Applications of Transport Telematics and Intelligent Vehicle Highway Systems*, vol. 4, pages 1887–1891, 1994.

[Shiller95] Z. Shiller and P. Fiorini, "Autonomous Negotiation of Freeway Traffic", *SPIE*, vol. 1592, pages 83–91, 1995.

[Shladover95] S. Shladover, "Review of the State of Development of Advanced Vehicle Control Systems", *Vehicle System Dynamics*, vol. 24, no. 6–7, pages 551–595, 1995.

[Sienel94] W. Sienel and J. Ackermann, "Automatic Steering of Vehicles with Reference Angular Velocity Feedback", *Proceedings of the American Control Conference*, pages 1957–1958, 1994.

[Sukhtankar97] Rahul Sukhtankar, "Situational Awareness for Tactical Driving", Ph.D. thesis, Carnegie Mellon University, January 1997.

[Sukthankar95] R. Sukthankar, D. Pomerleau, and C. Thorpe, "Simulated Highways for Intelligent Vehicles Algorithm", *Proceedings of the Intelligent Vehicles '95 symposium*, pages 332–337, 1995.

[Tribe95] R. Tribe, K. Prynne, I. Westwood, N. Clarke, and M. Richardson, "Intelligent Driver Support", *Proceedings of the Second World Congress on Intelligent Transport Systems, '95 Yokohama*, pages 1187–1192, 1995.

[Ulmer94] B. Ulmer, "VITA II - Active Collision Avoidance in Real Traffic", *Proceedings of the Intelligent Vehicle '94*, pages 1–6, October 1994.

[Usami95] M. Usami, T. Suzuki, M. Nishida, and M. Ohta, "A Vision-Based Environment Recognition System for Automated Vehicles", *Proceedings of the Second World Congress on Intelligent Transport Systems, '95 Yokohama*, pages 1260–1265, 1995.

[Werner95] M. Werner and C. Engels, "A Nonlinear Approach to Vehicle Guidance", *Proceedings of the Intelligent Vehicles '95 symposium*, pages 136–140, 1995.

[Wissing95] T. Wissing, "Field Operational Test of an Intelligent Cruise Control System", Tech. Report, Eaton Corporation, Corporate Research and Development – Detroit Center, February 1995.

Appendix A

Longitudinal Controller

A.1 Introduction

In this appendix we will discuss the design of the velocity controller for the host vehicle. The control law is based on a proportional plus integral (PI) gain for the error in desired velocity. First we will develop the dynamic model for the plant based on the Navistar semi-tractor. Then we will discuss the controller designs followed by some plots demonstrating the performance of the system.

A.2 Longitudinal Vehicle Dynamic Model

The dynamic model developed for the host is based on the parameters for the Navistar semi-tractor. The dynamic model is a first order system with inputs of engine and braking forces applied through the tires and output of forward velocity. The dynamic equation for the system is shown in Equation A.1 Note that all of the dynamic equations for the vehicle described in this appendix are based on those developed by Gillespie [Gillespie92].

$$ma + F_{roll} + F_{drag} = F_{engine} + F_{brake} \tag{A.1}$$

where:

m	= Mass of the truck
a	= Acceleration of the truck.
F_{roll}	= Force due to rolling resistance of the tires.
F_{drag}	= Force due to aerodynamic drag of the vehicle.
F_{engine}	= Force applied by the engine through the tires and transmision.
F_{brake}	= Force applied by the brakes through the tires.

Here in this equation we have ignored the force term due to road grade. The resistance to motion for the vehicle is based on more than the actual mass. There are also rotational inertia effects (due to the transmission, engine, etc.) which must be considered. The rotational inertia is accounted for here by replacing the vehicle mass in Equation A.1 with an effective mass (m_{eff}) defined by Equation A.2. The effective mass is dependent on the gear ratio each gear for the vehicle.

$$m_{eff} = m_{factor} m \tag{A.2}$$

where:

m_{eff} = Effective mass of the truck

m_{factor} = Mass factor based on the selected gear.

Now with the basic dynamic equation defined we need to develop the equations for each of the forces to fully define the dynamic equation.

The rolling resistance force is based on the following equation:

$$F_{roll} = K_r C_n m g \tag{A.3}$$

where:

K_r = Rolling resistance coefficient.

C_n = Road surface coefficient.

g = Acceleration of gravity.

The rolling resistance is only dependent of the weight of the vehicle so here m is used instead of m_{eff}. The rolling resistance coefficient does change linearly with velocity (V_{long}). This coefficient is determined per Equation A.4.

$$K_r = K_{r_1} + K_{r_2} V_{long} \tag{A.4}$$

where:

K_{r_1} = Rolling resistance coefficient when velocity is zero.

K_{r_2} = Rolling resistance coefficient rate of increase with velocity.

The road surface coefficient is approximately constant over velocity and is dependent on the road type (i.e. bituminous, concrete, etc.).

The force due to aerodynamic drag of the vehicle is based on the velocity squared. This force is described by Equation A.5.

$$F_{drag} = \frac{1}{2} \rho C_D A V_{long}^2 \tag{A.5}$$

where:

ρ = Density of air.

C_D = Drag coefficient.

A = Front surface area of the truck.

The density of air changes based on temperature as defined by the following equation.

$$\rho = 1.225 \left(\frac{P_r}{101.325} \right) \left(\frac{288.16}{273.13 + T_r} \right) \tag{A.6}$$

where:

P_r = Atmospheric pressure in kpa.

T_R = Air temperature in degrees C.

The density of air determined by Equation A.6 is then used in the drag force Equation A.5.

The force from the engine is based on a simple model of a diesel engine. Engines produce a torque (T_{engine}) based on the rotational speed and the level of throttle actuation (δ_t). Here, this torque vs. speed relationship is determined based on the model of a dc servomotor (see Equation A.7). This is very inaccurate at lower speeds but is a reasonable model at higher speeds (see Figure A.1). Since in our simulation we are primarily concerned with higher speed tests, this model is adequate.

$$T_{engine} = \left(K_c - \frac{K_n V_{long}}{r_w}\right)\delta_t \tag{A.7}$$

where:

K_c = Engine stall torque coefficient (Nm).

K_n = Engine slope coefficient ($\frac{Nms}{rad}$).

r_w = Wheel radius in meters.

δ_t = Throttle actuation level (0 - full off, 1 - full on).

This engine torque is then transmitted to an engine force (F_{engine}) through the transmission and tires as shown in Equation A.8.

$$F_{engine} = \frac{N_{tf}\eta_{tf}}{r_w}\left(K_c - \frac{K_n V_l ong}{r_w}\right)\delta_t \tag{A.8}$$

where:

N_{tf} = Combined ratio of transmission and final drive.

η_{tf} = Efficiency of combined transmission and final drive.

Here there is an efficiency term associated with the transmission which reduces the force actually applied. This efficiency and the transmission ratio are set based on the current gear for the vehicle.

The brakes apply a force to the vehicle through the wheels to decelerate the host. This braking force is determined by Equation A.9.

$$F_{brake} = -m d_{brake_{max}} \alpha_b \tag{A.9}$$

where:

$d_{brake_{max}}$ = Maximum deceleration capability of brakes.

α_b = Brake actuation level (0 - full off, 1 - full on).

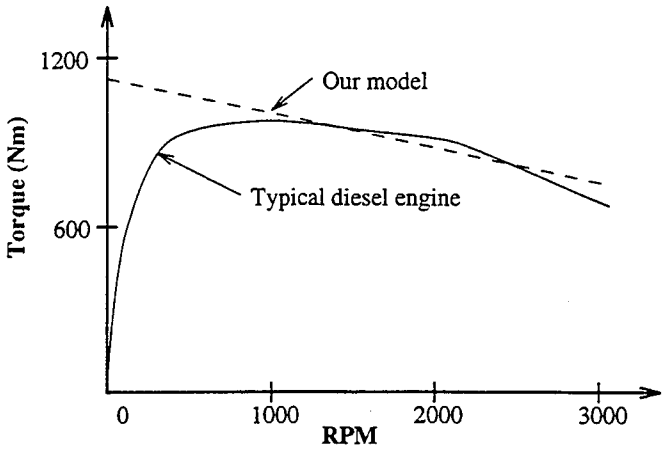


Figure A.1: Engine torque vs. speed for typical diesel engine and for our engine model. The torque levels are based on full throttle actuation ($\delta_t = 1.0$).

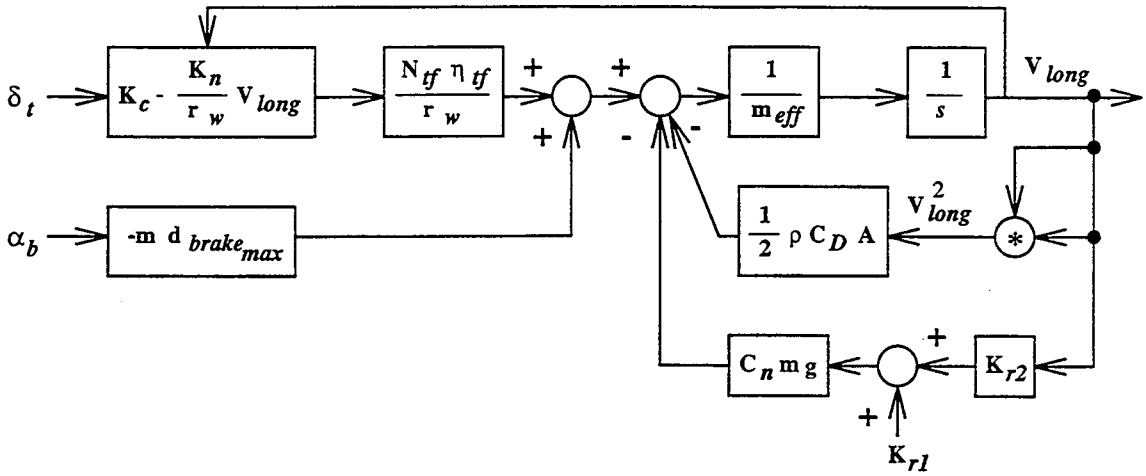


Figure A.2: Block diagram for longitudinal dynamic model of vehicle.

From the equation we see that this force which will impose a deceleration level is modeled as approximately constant across velocities and is based linearly on the level of brake application.

The dynamic longitudinal model for the vehicle can be represented in a block diagram model. This block diagram is shown in Figure A.2. The parameters for the dynamic equations which are not dependent on the current gear are shown in Table A.1. Parameters in this table without a reference are determined from measurements off of the truck. The parameters which change based on the gear are shown in Table A.2. These parameters are automatically changed during acceleration or deceleration based on the given velocity to simulate the automatic transmission shifting. The velocity range for each gear and N_{tf} are determined from measurements off of the Navistar semi-tractor. The other parameters are based on typical values listed in [Gillespie92]. The dynamic model defined with these parameters is used in the design of the velocity controller.

Parameter	Value	Source
m	9053 kg	measurement
K_{r_1}	.0066	[Gillespie92]
K_{r_2}	.000103 $\frac{s}{m}$	[Gillespie92]
C_n	1.0 (typical road surface)	[Gillespie92]
C_D	0.85	[Gillespie92]
A	10.0 m^2	estimate
T_r	25.0° C	typical
P_r	101.325 kpa	[Gillespie92]
ρ	1.184 $\frac{kg}{m^3}$	Equation A.6
K_c	1125.0 Nm	[Gillespie92]
K_n	1.1937 $\frac{Nm}{rad/sec}$	[Gillespie92]
r_w	0.5 meters	measurement
$d_{brake_{max}}$	4.904 $\frac{m}{s^2}$	[Kempf92]

Table A.1: Parameter values for the longitudinal dynamic model for parameters that are not dependent on the selected gear.

Gear	V_{long} Range	N_{tf}	η_{tf}	m_{factor}
1	0 - 4.4 m/s	28.11	.96	2.50
2	4.4 - 7.9 m/s	15.62	.96	1.60
3	7.9 - 13.2 m/s	9.37	.96	1.47
4	13.2 - 19.8 m/s	6.25	.96	1.34
5	19.8 - 24.2 m/s	4.69	.96	1.20
6	> 24.2 m/s	4.02	.96	1.09

Table A.2: Parameter values for the longitudinal dynamic model for parameters that are dependent on the selected gear.

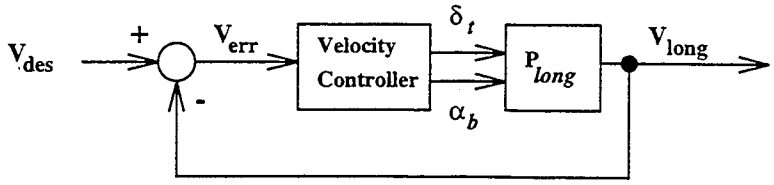


Figure A.3: Block diagram for velocity control loop.

A.3 Longitudinal Controller Design

The velocity controller developed for the this dynamic model is based on a PI control law. The controller input is the error in measured velocity (V_{error}) and the outputs are throttle actuation level (δ_t) and brake actuation level (α_b). The block diagram for this controller is shown in Figure A.3.

For this design, we have devised a simple approach for generating two outputs (δ_t and α_b) from one control law. This approach is described by the block diagram for only the controller shown in Figure A.4. Here, we see that the controller outputs are each set from the output of the PI control law (u). The output for the throttle is defined by Equation A.10.

$$\delta_t = \begin{cases} 0 & u \leq 0 \\ u & 0 < u \leq 1 \\ 1 & 1 < u \end{cases} \tag{A.10}$$

Here we have defined that when the controller output is greater than zero, the throttle output is set to its value. If the controller output is greater than 1, the output saturates and $\delta_t = 1.0$. When the controller output is less than zero, the throttle is turned off allowing the vehicle to coast to decelerate. The output for the brake application level is defined in Equation A.11.

$$\alpha_b = \begin{cases} 1 & u \leq -.5 \\ -2u & -.5 < u \leq -.2 \\ 0 & -.2 < u \end{cases} \tag{A.11}$$

Here we see that the brake actuation is zero (or off) when the controller output is greater than .2. This means that there is a dead-band for $-.2 < u \leq 0$ where the controller applies neither throttle or braking to the plant. In this dead-band the vehicle will decelerate by coasting alone. For controller output values less than $-.2$ brake application is applied at double the controller output value. Therefore, the gains for braking are approximately double the gains for throttle actuation. The doubling of the gains is done to meet the design requirements discussed below. When the control output is less than $-.5$ the brake level saturates at 1.0 which corresponds to full brake application.

Special note should be made of the limits set on the integration error term. In this controller,

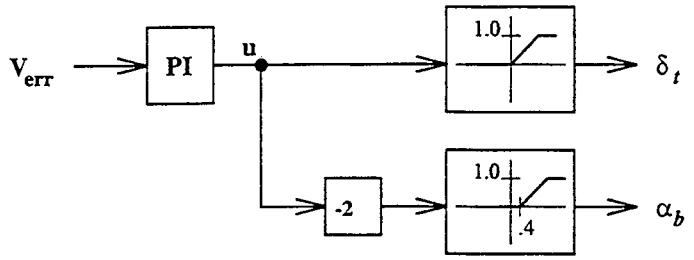


Figure A.4: Block diagram for only the velocity controller.

Gain	Value
K_p	0.2051
K_i	0.0256

Table A.3: Gains selected for the velocity controller.

the outputs, δ_t and α_b, sometimes saturate. This occurs usually in tracking acceleration commands (i.e. step input from 0 to 25 m/s) and during emergency braking maneuvers. When either of the controller outputs saturate, the integration error term is limited to its current value. This keeps the integral term from building up large values due to the saturation nonlinearity. Limiting the integral error term eliminates excessive over-shoot due to controller output saturation.

With the control law defined, we are now ready to tune the controller to meet our requirements. The controller was designed to meet the following two requirements:

1) Minimal over-shoot of desired velocity when accelerating.

2) Track the deceleration profile of the input velocity.

The first requirement is listed as a general requirement needed for passenger comfort. The second requirement is needed to allow the controller to be useful for the virtual bumper algorithm. Remember that the virtual bumper will send velocity commands to the velocity controller which follow a desired deceleration profile (see section 2.3.2). If the velocity controller can track the deceleration profile determined by the velocity commands, safe control of the vehicle can be realized.

Controller gains which meet the above design requirements are shown in Table A.3. The performance of this controller is demonstrated in the next section.

A.3.1 Controller performance

In this section we will discuss the performance of the designed velocity controller. Here we will test the response to the following inputs: 1) Positive step input, 2) Negative step input and 3) Constant deceleration input profile.

First, let's look at the system response for a positive step input. Our controller response to a

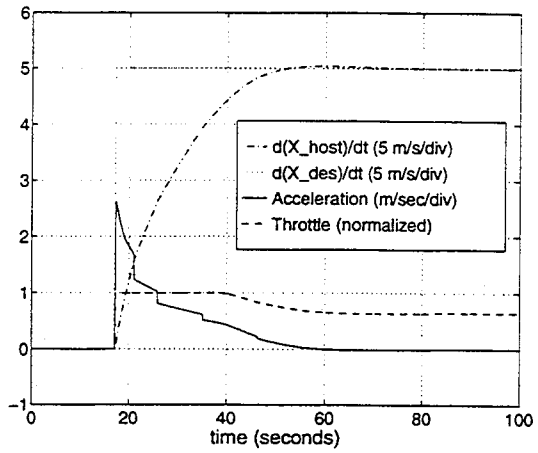


Figure A.5: Velocity controller response to a 25.0 m/s step input.

step input from 0.0 to 25.0 m/s is shown in Figure A.5. From the figure we see the slight shifts in the acceleration profile as the vehicle shifts gears. For this input, the velocity over-shoots the desired velocity by only 0.5 m/s. This is a reasonable response and shows that the controller meets the first design requirement listed above.

Next, let's look at the response to a negative step input to the system. For this test, the system starts at 25.0 m/s and the input steps are commanded. Here, we evaluate the response to a -2.0 m/s step (shown in Figure A.6) and to a -5.0 m/s step (shown in Figure A.7). From these figures we see that the controller has a much faster response in tracking deceleration commands. For the -2.0 m/s step input we see that braking levels up to XX are applied and for the -5.0 m/s step input the braking levels are up to YY. These tests demonstrate that large steps in the commanded velocity should not be allowed if we desire smooth velocity adjustments to insure driver comfort.

Finally, we look at the velocity controller for constant deceleration input profiles. Figure A.8 show the response of the controller to a constant deceleration profile of $-.2$ $g's$ (≈ -2.0 m/s). Here we see that the actual velocity (V_{long}) does not track the commanded velocity ($V_{desired}$) exactly. However, V_{long} does track $V_{desired}$ with a phase lag. This shows that the controller is capable of decelerating the vehicle at approximately the desired deceleration level. Figure A.9 shows the response of the controller when the desired deceleration level is $-.5$ $g's$ (≈ -4.9 m/s) which is an emergency deceleration profile. In this situation, V_{long} tracks $V_{desired}$ with a phase lag and the vehicle is decelerated applying maximum braking levels. These tests demonstrate that this velocity controller is capable of applying smooth moderate braking levels as well as applying full level braking to meet emergency driving requirements. These tests also demonstrate that the controller meets the second design requirement of tracking the deceleration profile of the input velocity.

Through these tests we have shown that this controller meets our design requirements. It is

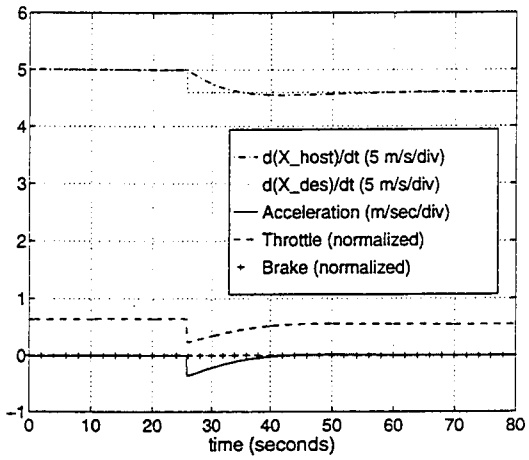


Figure A.6: Velocity controller response to a $-2.0\ m/s$ step input.

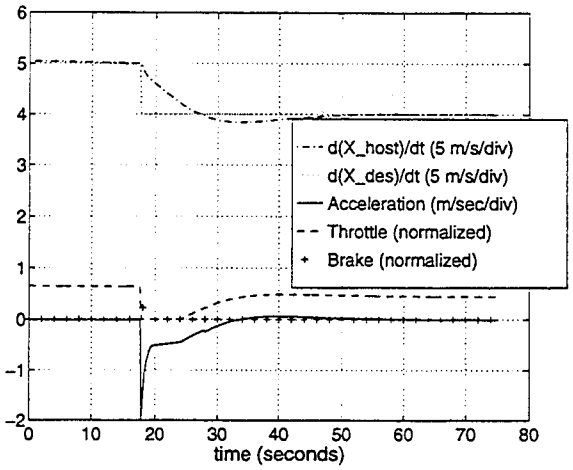


Figure A.7: Velocity controller response to a $-5.0\ m/s$ step input.

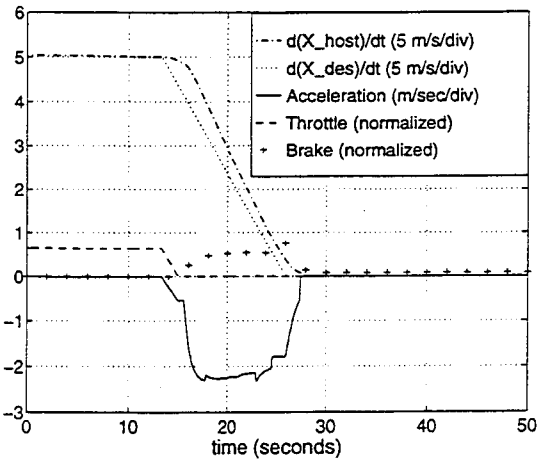


Figure A.8: Velocity controller response to a constant deceleration profile of $-.2\ g's$.

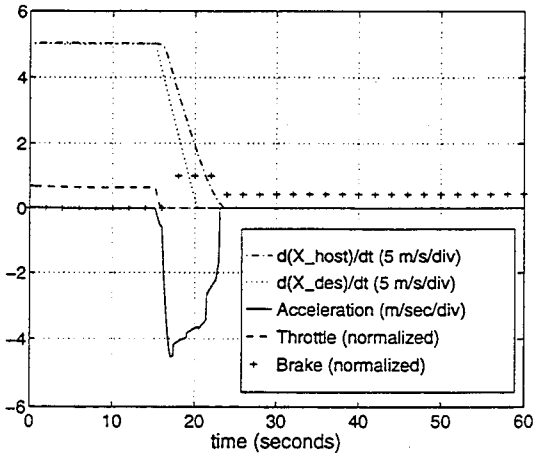


Figure A.9: Velocity controller response to a constant deceleration profile of $-.5\ g's$.

capable of accelerating the vehicle to the desired velocity with a minimal over-shoot. This controller can also decelerate the vehicle at a desired rate determined by the input velocity. The controller described here can be used in conjunction with the virtual bumper to provide collision avoidance control for the host vehicle.

Appendix B

Lateral Controller

B.1 Introduction

In this appendix we will discuss the design of the lateral position controller for the host vehicle. First, we develop the lateral dynamic model for the Navistar semi-tractor used in the simulation. Then the lateral controller for this dynamic model is designed. We then document the performance of the controller by testing the system response for lane change inputs.

B.2 Lateral Vehicle Dynamics

The lateral dynamic model used in the simulation environment is based on that developed by Alexander et al [Alexander96]. The model developed by [Alexander96], however, is based on a constant forward velocity of 15.0 m/s. In the development of the virtual bumper, the vehicle can travel any speed from 0 to 25 m/s so this dynamic lateral model needs some modifications. Here, we have modified the dynamic model to allow it to be applied for velocities from 3.5 to 25.0 m/s. For low velocities, the dynamic model is replaced with a kinematic model. In the following discussion, we will first review the dynamic model outlined in [Alexander96] and then we will discuss the kinematic model and the dynamic model modifications.

B.2.1 Dynamic model

The dynamic model developed in [Alexander96] uses four states to model the lateral motion for the semi-tractor. These states are the lateral velocity (V_{lat}), the yaw rate (or rate of rotation, r) and two states for the slip angle of the front wheel (α_1 and $\dot{\alpha}_1$, where α_1 is the front wheel slip angle). Dynamic equations are developed describing the derivative of each of these states and the model is presented in state-space form.

The equation describing the dynamics of the front wheel slip angle is shown in Equation B.1.

$$\ddot{\alpha}_1 = -V_{long}^2\omega_n^2\alpha_1 - 2\zeta V_{long}\omega_n\dot{\alpha}_1 - V_{long}\omega_n^2 V_{lat} - V_{long}\omega_n^2 ar + V_{long}^2\omega_n^2\delta \qquad \text{(B.1)}$$

where:

V_{long} = Forward velocity of truck.

V_{lat} = Lateral velocity with respect to truck coordinate system.

r = Yaw rate of the truck.

ζ = Damping ratio for the slip angle dynamics.

ω_n = Natural frequency for the slip angle dynamics.

a = Offset of front axle from center of mass.

From this equation, we see that the dynamics of the front wheel slip angle is dependent on the other states of the lateral dynamics (V_{lat} and r) as well as on the forward velocity (V_{long}). The values of ζ and ω_n are selected to fit the dynamic model based on tests conducted on the real truck [Alexander96].

The remainder of the dynamic equations are derived based on a simple sketch shown in Figure B.1. In this figure we see that lateral forces are applied to the truck by the tires. Here, each axle is modeled as a single tire so only three forces are shown. The force for each tire is shown in the following equations:

$$F_{y1} = C_1\alpha_1 \tag{B.2}$$

$$F_{y2} = C_2\alpha_2 \tag{B.3}$$

$$F_{y3} = C_3\alpha_3 \tag{B.4}$$

where:

C_i = Cornering stiffness for the i^{th} axle.

α_i = Slip angle for the i^{th} axle.

The slip angle for the front tires (axle 1) is determined by the dynamics described in Equation B.1. The slip angles for the 2nd and 3rd axle are determined by Equations B.5 and B.6, respectively.

$$\alpha_2 = -\frac{V_{long} - b_1 r}{V_{lat}} \tag{B.5}$$

$$\alpha_3 = -\frac{V_{long} - b_2 r}{V_{lat}} \tag{B.6}$$

where:

b_1 = Distance of axle 2 from the center of mass.

b_2 = Distance of axle 3 from the center of mass.

Now we can apply Newton's Second Law to derive the dynamic equations for the lateral and rotational motion. These dynamic equations are shown in Equation B.7 and B.8.

$$m(\dot{V}_{lat} + V_{long} r) = F_{y1} + F_{y2} + F_{y3} \tag{B.7}$$

$$I\dot{r} = aF_{y1} - b_1 F_{y2} - b_2 F_{y3} \tag{B.8}$$

where:

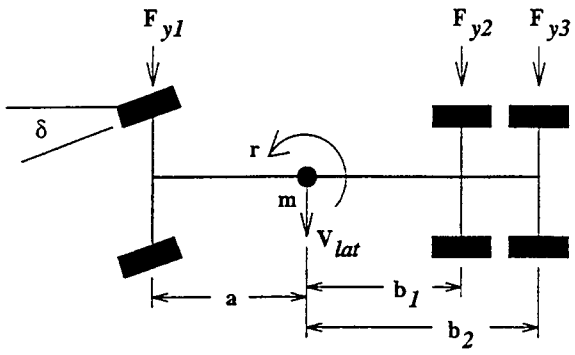


Figure B.1: Simple lateral model of truck with applied forces.

m = Mass of the truck.

I = Moment of inertia of the truck.

Then if we substitute the force Equations B.2 through B.4 and the slip angle equations (B.1, B.5 and B.6) into the lateral motion equations, we arrive at the following equations:

$$m\dot{V}_{lat} = C_1\alpha_1 + C_2\left(-\frac{V_{lat}-b_1 r}{V_{long}}\right) + C_3\left(-\frac{V_{lat}-b_2 r}{V_{long}}\right) - mV_{long}r \quad \text{(B.9)}$$

$$I\dot{r} = aC_1\alpha_1 - b_1C_2\left(-\frac{V_{lat}-b_1 r}{V_{long}}\right) - b_2C_3\left(-\frac{V_{lat}-b_2 r}{V_{long}}\right) \quad \text{(B.10)}$$

Next we solve for $\dot{V}_{lat}$ and $\dot{r}$ to result in:

$$\dot{V}_{lat} = \frac{C_1}{m}\alpha_1 + \left(\frac{-C_2-C_3}{mV_{long}}\right)V_{lat} + \left(\frac{C_2b_1+C_3b_2-mV_{long}^2}{mV_{long}}\right)r \quad \text{(B.11)}$$

$$\dot{r} = \frac{aC_1}{I}\alpha_1 + \left(\frac{b_1C_2+b_2C_3}{IV_{long}}\right)V_{lat} + \left(\frac{-b_1^2C_2+b_2^2C_3}{IV_{long}}\right)r \quad \text{(B.12)}$$

Then Equations B.1, B.11 and B.12 can be rewritten in a state-space form.

$$\begin{bmatrix} \dot{V}_{lat} \\ \dot{r} \\ \dot{\alpha}_1 \\ \ddot{\alpha}_1 \end{bmatrix} = \begin{bmatrix} \left(\frac{-C_2-C_3}{mV_{long}}\right) & \left(\frac{C_2b_1+C_3b_2-mV_{long}^2}{mV_{long}}\right) & \frac{C_1}{m} & 0 \\ \left(\frac{b_1C_2+b_2C_3}{IV_{long}}\right) & \left(\frac{-b_1^2C_2+b_2^2C_3}{IV_{long}}\right) & \frac{aC_1}{I} & 0 \\ 0 & 0 & 0 & 1 \\ -V_{long}\omega_n^2 & -V_{long}\omega_n^2 a & -V_{long}^2\omega_n^2 & -2\zeta V_{long}\omega_n \end{bmatrix} \begin{bmatrix} V_{lat} \\ r \\ \alpha_1 \\ \dot{\alpha}_1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ V_{long}^2\omega_n^2 \end{bmatrix} \delta \quad \text{(B.13)}$$

This defines the form of the lateral dynamic equations for the truck. To fit these equations to match the dynamics of the Navistar semi-tractor, the parameters are set as shown in Table B.1. These parameters are fit based on a forward velocity of 15.0 m/s.

B.2.2 Dynamic model varying with V_{long}

Now we have the dynamic model fully defined for a forward velocity of 15.0 m/s. However, as stated earlier, the dynamic model needs to be used across a range of velocities from 3.5 to 25.0 m/s. The

Parameter	Value
m	9,053 kg
I	52,161 $kg\text{-}m^2$
a	2.59 m
b_1	2.70 m
b_2	4.02 m
C_1	180,000 N/rad
C_2	350,000 N/rad
C_3	350,000 N/rad
ζ	0.4
ω_n	4.5 rad/sec

Table B.1: Parameters for the lateral dynamic equations for $V_{long} = 15.0\ m/s$.

most accurate way to develop this model would be to repeat the tests discussed in [Alexander96] for forward velocities throughout our specified operating range and fit the parameter values for each velocity. Unfortunately, due to limited time and resources, this approach was not possible.

Instead, the approach used here is to scale the coefficients of the state space matrices with velocity. Notice from the state space representation (Equation B.13) that several of the coefficients of the A and B matrices are a function of the forward velocity (V_{long}). Therefore, the dynamic equations can be adjusted for velocity by re-calculating the A and B matrices coefficients for each velocity. This is the approach chosen here. For the simulation environment, the velocity is used at each time step to adjust the dynamic equations. Then for that time step, the adjusted dynamic equations are used. This approach is used for forward velocities from 3.5 to 25.0 m/s.

B.2.3 Kinematic model

For velocities from 0 to 3.5 m/s a kinematic model is used instead of a dynamic model to model the vehicle's motion. This is a reasonable approach because at low velocity, the vehicle motion is accurately defined by a kinematic model. However, there is an issue in using a kinematic model. The issue is that when we switch from the dynamic to kinematic model, there is a sudden discontinuity in the way the system responds. This discontinuity will degrade the performance of the lateral controller. Furthermore, in reality, this sharp discontinuity does not exist. Therefore, a new kinematic model was developed.

The kinematic model developed here eliminates the discontinuity when switching from the dynamic model. This model is based on the observation that at lower forward velocities (V_{long}), the lateral velocity (V_{lat}) and the yaw rate (r) are proportional to the steer angle (δ) and V_{long}. Therefore, at lower velocities, we can determine V_{lat} and r based on functions of δ and V_{long}. These

functions are defined as follows:

$$V_{lat} = K_{V_{lat}}\delta \tag{B.14}$$

$$r = K_r\delta \tag{B.15}$$

where:

$K_{V_{lat}}$ = Gain for calculating V_{lat} which is dependent on V_{long}.
K_r = Gain for calculating r which is dependent on V_{long}.

With these equations defined, we then determine $K_{V_{lat}}$ and K_r based on a second order polynomials.

For $K_{V_{lat}}$ the polynomial is fit to ($V_{lat}/delta$,V_{long}) data points for V_{long} = 4.0, 5.0 and 6.0 m/s. Here, we make the additional assumption that there is no lateral velocity for $V_{long} < 1.0$ m/s. Therefore we add the additional data point of (0.0,1.0) to our data set. This results in the following data fit:

$$K_{V_{lat}} = -0.9521 + 1.0304V_{long} - 0.0740V_{long}^2 \tag{B.16}$$

Then to prevent $K_{V_{lat}}$ from being non-zero for $V_{long} < 1.0$ m/s, we determine determine its value using Equation B.17.

$$K_{V_{lat}} = \begin{cases} 0 & \text{if } V_{long} < 1.0\ m/s \\ -0.9521 + 1.0304V_{long} - 0.0740V_{long}^2 & otherwise \end{cases} \tag{B.17}$$

The value of $K_{V_{lat}}$ based on velocity is then used in Equation B.14 to determine a lateral velocity. The plot of the values of $K_{V_{lat}}$ over velocity is shown in Figure B.2.

Likewise, the polynomial for K_r is fit to ($r/delta$,V_{long}) data points for V_{long} = 4.0, 5.0 and 6.0 m/s. In this case, we assume that the vehicle will have a yaw rate which decreases to zero at $V_{long} = 0.0$ m/s. Therefore we add the additional data point of (0.0,0.0) to our data set. This results in the following equation for K_r:

$$K_r = 0.1932V_{long} - 0.0099V_{long}^2 \tag{B.18}$$

The value of K_r based on velocity is then used in Equation B.15 to determine a yaw rate for the vehicle. The plot of the values of K_r over velocity is shown in Figure B.3.

The values for V_{lat} and r determined by this kinematic model match up smoothly with the values determined by the dynamic model at low velocities. This eliminates the discontinuity when switching between these models as the vehicle accelerates or decelerates through the switching velocity. The smooth transition makes our model more closely match the lateral dynamics of a real vehicle across all velocities.

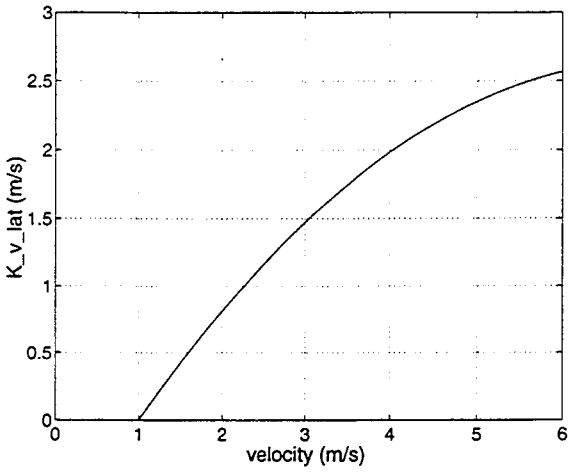


Figure B.2: Plot of $K_{V_{lat}}$ versus V_{long}.

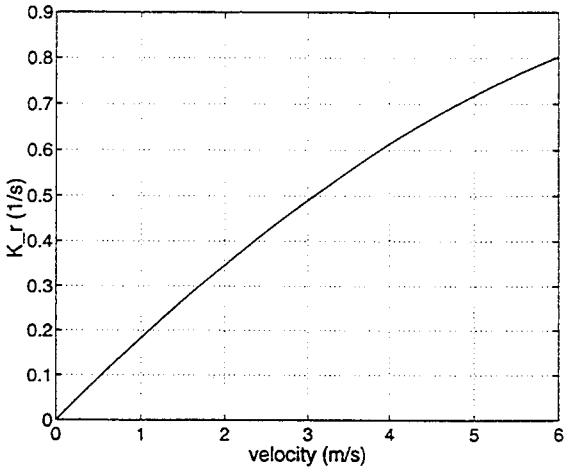


Figure B.3: Plot of K_r versus V_{long}.

B.2.4 Position Estimates

Now we have developed a model for the Navistar's lateral motion for velocities ranging from 0 to 25 m/s. Next, we determine the vehicle's actual position based on the outputs of our model (V_{lat} and r) and V_{long}.

In this discussion, the vehicle's position is determined relative to its starting point. Initially, the host is located with its control point at the origin of our global coordinate system. The x-axis is positive directly ahead of the vehicle and the y-axis is positive to the left. The vehicle's orientation or yaw angle (θ) is measured positive counter-clockwise from the x-axis and is initially zero. The approach here is to determine the value of θ by integrating r over time and then to use theta to calculate the components of the velocity along the global coordinate system.

$$\theta = \int_0^t r dt \tag{B.19}$$

$$\dot{x} = V_{long}\cos\theta - V_{lat}\sin\theta \tag{B.20}$$

$$\dot{y} = V_{long}\sin\theta + V_{lat}\cos\theta \tag{B.21}$$

Then these components of velocity in the global coordinate system are integrated to determine the global position of the vehicle.

$$x = \int_0^t \dot{x} dt \tag{B.22}$$

$$y = \int_0^t \dot{y} dt \tag{B.23}$$

It is important to note that this is not the exact position, it is only an estimate of the position. This is an estimate because when the wheels are not in perfect contact with the road and there is some

wheel slippage. This introduces error into the vehicle position. For the purposes of simulation, this estimated position is used to place the host at an exact position in the simulation environment. Therefore, for the purposes of simulation, the inaccuracy of the estimate is not an issue.

B.3 Lateral Controller Design

The lateral controller developed in this section is a proportional plus derivative (PD) controller with a separate yaw rate feedback loop. This controller is designed to controller the lateral position of the vehicle on the road without any preview information. The lack of preview information limits the lateral controller to operating only on straight road segments. This is reasonable for our work here as the lateral controller is primarily developed to demonstrate the capabilities of the virtual bumper algorithm.

In the following sections, we will discuss the reasons for using yaw rate feedback. Then we will define the controller gains selected and we will finish by demonstrating the system performance across the range of forward velocities.

B.3.1 Advantages of yaw rate feedback

As stated, this lateral controller uses a separate yaw rate feedback loop. The use of a separate yaw rate feedback loop is not a new idea. Similar controllers with this feedback have already been developed by [Sienel94] and [Peng90]. The yaw rate loop implemented in this controller has no reference input value. Therefore, this loop acts to drive the rotation of the host vehicle to zero. This smoothes out the oscillations for the lateral controller so we can think of yaw rate feedback as adding damping to our system.

For the development of this controller, the gains are selected using standard controller tuning methodologies. However, a more involved controller structure looks as if it holds promise. To demonstrate this, let's look at the open loop gain (L) for this lateral controller, shown in Figure B.4. This open loop gain is equal to the transfer function, Y_{host}/Y_{des} when the feedback portion of the loop is removed. L has great significance in designing the controller and determining the system response.

Before deriving L, lets make a note of some of the assumptions made for simplifying the block diagram structure to that shown in Figure B.4. First, the road is assumed to be straight with the host initially aligned pointing down the road in the center of the right lane. With this assumption, the global y-position and the road y-position are the same. This eliminates any coordinate transformations in the control loop. The second assumption is that at highway speeds,

the yaw angle (θ) is sufficiently small such that $\sin\theta \approx \theta$ and $\cos\theta \approx 1$. This assumption eliminates nonlinearities introduced by the trigonometric functions in calculating the lateral position, y. This allows for a simpler loop gain in the following derivation.

To determine this gain, we start with the equation for Y_{host} and work our way back to Y_{des}. The equation for Y_{host} is determined from the block diagram as:

$$Y_{host} = \frac{1}{s}\left(V_{lat} + \frac{1}{s}V_{long}r\right) \tag{B.24}$$

Next we eliminate the yaw rate from the above equation by deriving the r/δ_1 transfer function.

$$r = P_{\delta r}\delta_{1+2} \tag{B.25}$$

where:

$P_{\delta r}$ = The plant transfer function from steer angle to yaw rate.

δ_{1+2} = Total steer angle commanded from both controllers.

Then, considering that:

$$\delta_{1+2} = \delta_1 + \delta_2 = \delta_1 + K_2 r = \delta_1 + K_2 P_{\delta r} \tag{B.26}$$

which leads to:

$$\delta_{1+2} = \frac{1}{1 - K_2 P_{\delta r}}\delta_1 \tag{B.27}$$

Now, substituting Equation B.27 into Equation B.25 leads to a new yaw rate equation.

$$r = \frac{P_{\delta r}}{1 - K_2 P_{\delta r}}\delta_1 \tag{B.28}$$

Next we eliminate the lateral velocity from Equation B.24. Using the same approach as shown above the transfer function for V_{lat}/δ_1 can be derived to the form shown in Equation B.29.

$$V_{lat} = \frac{P_{\delta V_{lat}}}{1 - K_2 P_{\delta r}}\delta_1 \tag{B.29}$$

where:

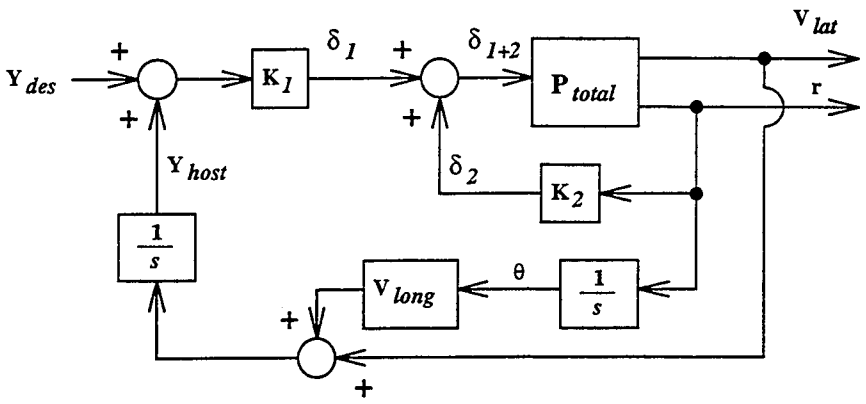
$P_{\delta V_{lat}}$ = The plant transfer function from steer angle to lateral velocity.

Then we substitute Equations B.28 and B.29 into Equation B.24 results in the following equation.

$$Y_{host} = \frac{1}{s}\left(\frac{P_{\delta V_{lat}} - \frac{1}{s}V_{long}P_{\delta r}}{1 - K_2 P_{\delta r}}\right)\delta_1 \tag{B.30}$$

And finally, knowing that with no feedback that $\delta_1 = K_1 Y_{des}$, we can solve for the open loop gain equation.

$$Y_{host} = \frac{K_1}{s}\left(\frac{P_{\delta V_{lat}} - \frac{1}{s}V_{long}P_{\delta r}}{1 - K_2 P_{\delta r}}\right)Y_{des} \tag{B.31}$$



Y_{des}	= Desired lateral position on the road.
Y_{host}	= Host's lateral position on the road.
K_1	= Lateral position controller.
K_2	= Yaw rate controller.
δ_1	= Commanded steer angle from lateral position controller.
δ_2	= Commanded steer angle from yaw rate controller.
δ_{1+2}	= Total commanded steer angle.
P_{total}	= Lateral dynamic model of plant including steering controller.
V_{lat}	= Lateral velocity of host.
V_{long}	= Longitudinal velocity of host.
r	= Yaw rate of host vehicle.
θ	= Yaw angle of host vehicle.
s	= Laplacian operator.

Figure B.4: Block diagram for the lateral position controller.

or:

$$L = \frac{Y_{host}}{Y_{des}} = \frac{K_1}{s}\left(\frac{P_{\delta V_{lat}} - \frac{1}{s}V_{long}P_{\delta r}}{1 - K_2 P_{\delta r}}\right) \tag{B.32}$$

Notice from the loop gain in Equation B.32 that there are two independent controllers (K_1 and K_2) controlling the shape of a single loop. This means that the shape of the loop in the frequency domain can be "shaped" using two sets of independent parameters. The shape of the loop determines how robust the controller design is to plant uncertainties as well as how sensitive the controller is to noise [Doyle92]. The use of two controllers here can allow for controller designs which achieve high robustness measures and low sensitivity to noise whereas this might not be possible for a single control loop for some plant models. The yaw rate feedback loop makes developing a robust controller much easier for the design engineer. Although our controller structure is quite simple (PD with a P gain on yaw rate feedback loop) this derivation demonstrates the advantage and the potential of this design approach for a lateral controller.

B.3.2 Controller gain selection

The control law for this design is a PD gain on the lateral position error with a P gain on the yaw rate. This law is summarized in Equation B.33.

$$\delta = K_p Y_{error} + K_d \dot{Y}_{error} - K_{p_{yaw}} r \tag{B.33}$$

where:

K_p = Proportional lateral error gain.

K_d = Derivative lateral error gain.

$K_{p_{yaw}}$ = Proportional yaw rate gain.

In our implementation, δ is the "steered wheel" angle in radians and is zero when the wheels are pointed straight ahead with positive angles steering the wheels to the left (CCW). Also, notice from the control law equation that $K_{p_{yaw}}$ is subtracted from the PD controller output. This is required to make the yaw rate feedback loop a negative feedback loop and allows for this loop to be stable.

The gains for this controller are selected through a tuning process. For tuning the controller, the input signal is a lane change profile where the width of the lane is 3.65 m. The process for tuning the gains is not cleanly defined in our case. The issue is in part that the simulation is implemented on a real-time embedded computer. For instance, our plant can be controlled in a very stable manner using only a PD controller if it is developed in a simulation package such as Simulink. This same controller is only marginally stable in the real-time implementation. The reason is that the real time implementation introduces variations in the controller cycling rate as well as sensor latency. These changes cause the stable controller to become marginally stable. For the real-time implementation we were unable to find gains for a PD controller which provide stable control for our plant. We were, however, able to tune a PD controller with yaw rate feedback to provide stable control for our system. The gains for this controller were adjusted following these rules of thumb:

1) K_p is required to keep the system stable but should be kept relatively small. Too large of K_p value makes the system too sensitive to error and will cause excessive over-shoot.
2) K_d is increased to allow the system a fast response to changes in the desired lateral position. Typical values for K_d are approximately five times the K_p values.
3) Increasing $K_{p_{yaw}}$ reduces the oscillation in the system. It also reduces the over-shoot. However, as $K_{p_{yaw}}$ is increased, the system response becomes more sluggish and to large of a value can cause the system to be too unresponsive.

These rules of thumb are used to determine controller gains for the system. The gains are tuned for forward velocities from 1.0 to 25.0 m/s at 1 m/s increments.

$V_{long}(m/s)$	K_p	K_d	$K_{p_{yaw}}$
0.0	0.400	3.700	0.00
1.0	0.400	1.700	0.00
2.0	0.207	0.750	1.00
3.0	0.187	0.900	4.60
4.0	0.169	0.800	4.20
5.0	0.150	0.700	3.75
6.0	0.131	0.630	3.35
7.0	0.112	0.560	2.95
8.0	0.089	0.490	2.55
9.0	0.080	0.420	2.15
10.0	0.070	0.350	1.75
11.0	0.064	0.324	1.60
12.0	0.058	0.298	1.45
13.0	0.052	0.272	1.30
14.0	0.046	0.246	1.15
15.0	0.040	0.220	1.00
16.0	0.037	0.204	0.92
17.0	0.034	0.188	0.84
18.0	0.031	0.172	0.77
19.0	0.028	0.156	0.70
20.0	0.025	0.140	0.63
21.0	0.024	0.134	0.60
22.0	0.023	0.128	0.57
23.0	0.022	0.122	0.54
24.0	0.021	0.116	0.52
25.0	0.020	0.110	0.50

Table B.2: Lateral controller gains across the range of forward velocities.

Our controller's gains are shown in Table B.2. For the implementation, this table of values is used as a look up table for the gains. The velocity is used to determine the nearest larger and smaller velocities on the table and these entries are then used to determine the controller gains through linear interpolation. The gains are calculated once for every cycle of the lateral controller.

From the table of controller gain values, notice the relationship between velocity and gain value. As velocity increases, the gains decrease. So the gains are inversely proportional to the velocity. This makes intuitive sense because as the vehicle travels faster, smaller steering angles are required to follow the commanded lateral position. The smaller steer angles are generated by using smaller gains. This relationship breaks down some at lower velocities ($V_{long} < 4.0\ m/s$) because of the switching between the kinematic model and the dynamic model.

B.3.3 Controller performance

In this section we document the performance of the lateral controller. To document the performance, we show the system response during a lane change maneuver. For this maneuver, the lateral path

V_{long} (m/s)	$\dot{Y}_{max}$ (m/s)	$\ddot{Y}_{max}$ (m/s^2)	Overshoot (m)	A_{lat} (m/s)
1.0	0.3	0.6	0.21	–
2.0	0.5	1.0	0.22	–
3.0	1.0	2.0	0.21	–
4.0	1.2	2.4	0.24	0.73
5.0	1.4	2.8	0.23	0.84
7.5	1.7	3.4	0.18	1.04
10.0	2.0	4.0	0.16	1.08
15.0	2.0	4.0	0.13	0.73
20.0	2.0	4.0	0.13	0.51
25.0	2.0	4.0	0.14	0.50

Table B.3: Summary of lateral position controller performance for a range of longitudinal velocities.

profile is the same as that generated by the virtual bumper's lateral impedance controller. The system response is evalutated for velocities across the range of operating velocities.

The system response for the lane change maneuver is shown in Figures B.5 through B.14 for the range of forward velocities. Key parameters for these maneuvers are summarized in Table B.3 at each velocity. Notice that the lateral acceleration (A_{lat}) is not shown in the figures or table for $V_{long} < 3.5\ m/s$ because the kinematic model is used at these velocities and A_{lat} has no significance. The lane change maneuvers are performed at a

The values of maximum lateral path velocity ($\dot{Y}_{d_{max}}$) and maximum lateral path acceleration ($\ddot{Y}_{d_{max}}$) are summarized in Table B.3 for the range of forward velocities. These limits define the emergency lane change profile at this velocity (see section 3.3.2). These limits are set at each velocity by starting with these values set at there maximums ($\dot{Y}_{d_{max}} = 2.0\ m/s$ and $\ddot{Y}_{d_{max}} = 4.0\ m/s$). Then the system with the tuned controller gains is tested for the emergency lane change profile and the following two criteria must be met:

1) Overshoot must be less than 0.25 m.
2) Vehicle lateral acceleration must be less than $A_{lat_{roll}} = 2.0\ m/s$.

If either of these criteria are not met, both $\dot{Y}_{d_{max}}$ and $\ddot{Y}_{d_{max}}$ are reduced until these conditions are achieved. In the evaluation, the ratio of $\dot{Y}_{d_{max}}/\ddot{Y}_{d_{max}}$ is kept constant for reasons discussed in section 3.3.2 The limiting factor in our evaluation is condition 1. Here, from the table we see the reasonable result that at slower velocities, slower lane change profiles can be achieved.

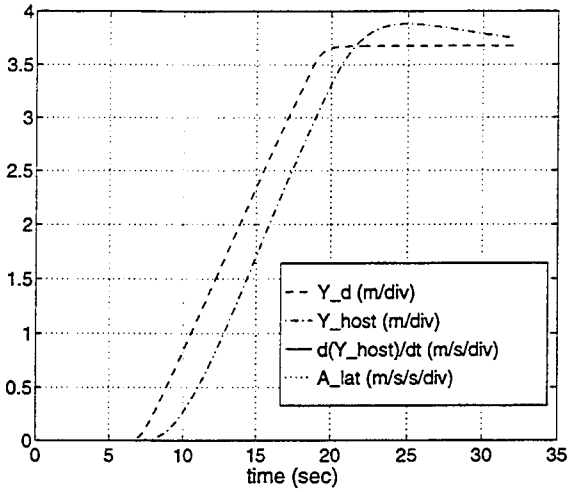


Figure B.5: Plot of lateral parameters for lane change maneuver with $V_{long} = 1.0\ m/s$.

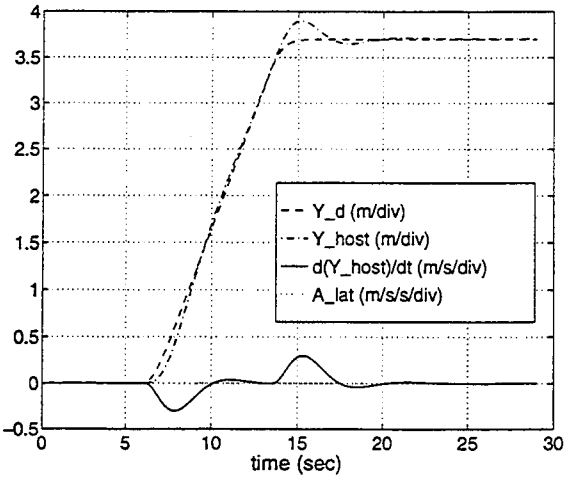


Figure B.6: Plot of lateral parameters for lane change maneuver with $V_{long} = 2.0\ m/s$.

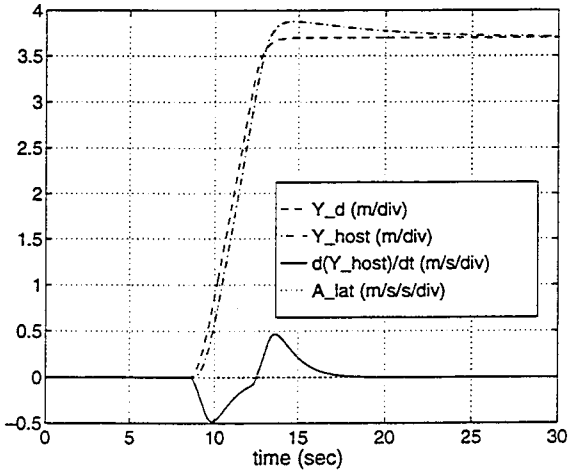


Figure B.7: Plot of lateral parameters for lane change maneuver with $V_{long} = 3.0\ m/s$.

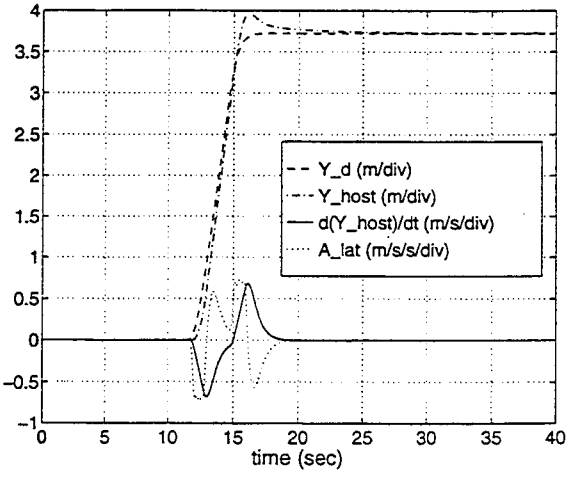


Figure B.8: Plot of lateral parameters for lane change maneuver with $V_{long} = 4.0\ m/s$.

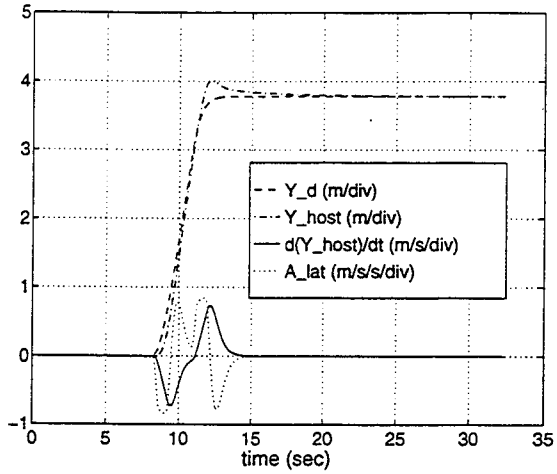


Figure B.9: Plot of lateral parameters for lane change maneuver with $V_{long} = 5.0\ m/s$.

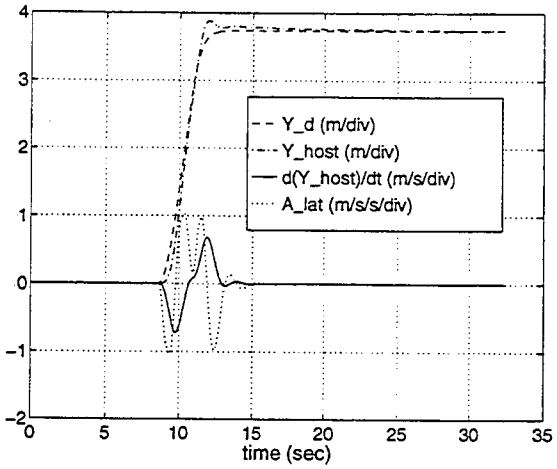


Figure B.10: Plot of lateral parameters for lane change maneuver with $V_{long} = 7.5\ m/s$.

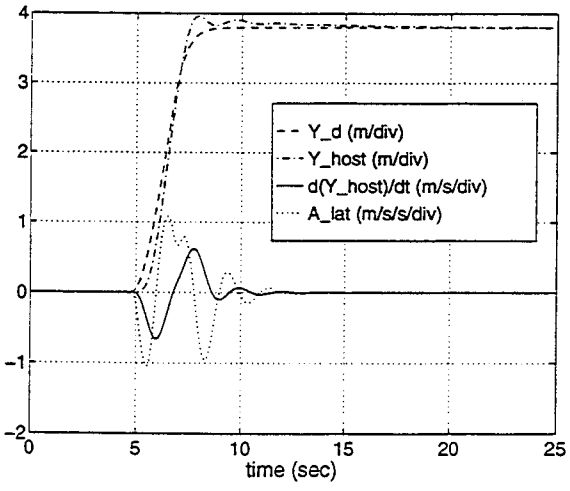


Figure B.11: Plot of lateral parameters for lane change maneuver with $V_{long} = 10.0\ m/s$.

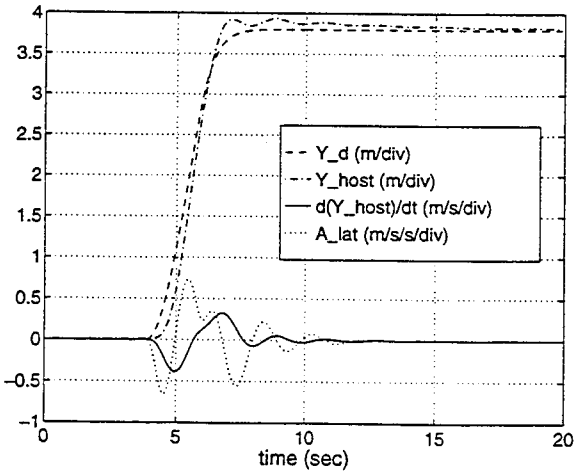


Figure B.12: Plot of lateral parameters for lane change maneuver with $V_{long} = 15.0\ m/s$.

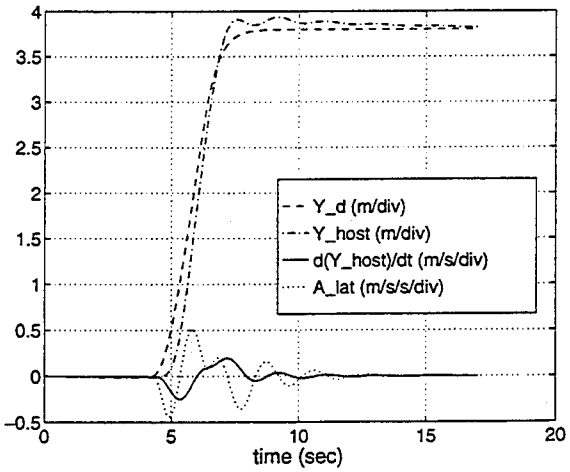


Figure B.13: Plot of lateral parameters for lane change maneuver with $V_{long} = 20.0\ m/s$.

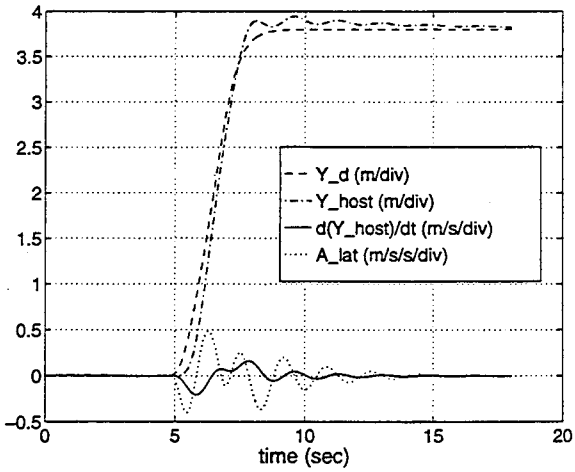


Figure B.14: Plot of lateral parameters for lane change maneuver with $V_{long} = 25.0\ m/s$.

Appendix C

Software Listing

C.1 Introduction

In this appendix we document the software developed for the simulation environment as well as the virtual bumper. This documentation is not meant as an exhaustive explanation of the software and the functionality. Rather, it is meant to give future researchers a reference for finding file locations as well as general functionality. Further documentation is listed in the files as appropriate.

C.2 Source code summary

All of the software has been developed on "tinman", the SGI workstation in the Robotics Research Laboratory. This software is resident in the "golf" account and is all in the "/usr/people/golf/virt_bumper/simulation/" directory. This directory contains the following sub-directories:

WTK/
alx/
dv/
unixcmdsocket/
unixconsumer/
unixdataconsumer/
unixproducer/
vxcmdsocket/
vxconsumer/
vxdataproducer/
vxproducer/

There is also a "bat" file in this directory which is a batch file which recompiles all of the software for the entire simulation environment.

The software for the simulation falls into four main categories. The first category is the embedded software. All of the embedded software is designed for the VxWorks operating system and is in the "alx/" directory. The second category is the graphical user interface (GUI). The GUI software is developed using DataViews and is listed in the "dv/" directory. The third category is the 3D graphical simulation developed using World Tool Kit (WTK) and is listed in the WTK directory. The remainder of the software is developed for communication between the embedded software, the

Directory	Description
unixcmdsocket/	Used by Dataviews to send information to the VxWorks environment. Passed information is the vehicle control commands in data structure p_ControlMsg listed in p_ControlMsg.h.
unixconsumer/	Used by Dataviews to receive information from the VxWorks environment. Information received is vehicle states in the data structure p_ControlCommands listed in p_ControlCommands.h.
unixdataconsumer/	Used by Datavies to receive information from the VxWorks environment. Information received is processed sensor data in the RoadAndRangeMsg data structure listed in RoadAndRangeMsg.h.
unixproducer/	Used by WTK to pass information to VxWorks. Information passed is range and vision sensor data in the p_SensorData structure listed in p_SensorData.h.
vxcmdsocket/	Used by VxWorks to receive controller commands from the Dataviews GUI. Data is received in the data structure p_ControlMsg listed in p_ControlMsg.h.
vxconsumer/	Used by VxWorks to receive range and vision sensor data from the WTK simulation. The data is passed in the p_SensorData structure listed in p_SensorData.h.
vxdataproducer/	Used by VxWorks to send data to the Dataviews GUI. Data sent is the road tracking and the processed range data in the RoadAndRangeMsg data structure listed in RoadAndRangeMsg.h.
vxproducer/	Used by VxWorks to send data to the Dataviews GUI. Data sent is the vehicle state information in the p_ControlCommands data structure listed in p_ControlCommands.h.

Table C.1: Communication software description.

GUI and the WTK environment. All of the communication software is listed in the "unix*" and "vx*" directories.

In the remaining sections we will detail the software in the "alx/", "dv/" and "WTK/" directories. The software in the communication category listed in the "unix*" and the "vx*" directories is not detailed specifically here. This is because this software is automatically generated using the NDDS communications package and a good understanding of this software can be achieved by referencing the NDDS software manual. To aid in reviewing the communication software, a brief description of the type of data communicated by the code in each directory is shown in Table C.1.

C.2.1 Embedded software

In this application, the embedded computer is a VME bus based computer with two Motorola MVME-147 single board computers (one processor each). Each of these processors is running the VxWorks real-time operating system. The embedded software discussed here has been developed to split the computational load between these processors.

The embedded software is listed in four sub-directories under the "alx/mv147" directory. These

four directories are:

WTK/
golf0/
golf1/
h/
share/

The "golf0" directory contains software which is executed on the master processor only. The files in this directory are outlined in Table C.2. The "golf1" directory contains software which is executed on the slave processor. The files in this directory are outlined in Table C.3. The "h" directory contains all of the include files for both the "golf0", "golf1" and the "share" directory. Several of the include files are *.h files which define function prototypes for the similarly named *.c files and are not summarized in Table C.5. However, some of the files contain data structure definitions and/or macros and are listed in Table C.5. The "share" directory contains all of the source files which are used by both the master and the slave processor. These files are listed in Table C.4.

C.2.2 Graphical user interface software

The graphical user interface has a great range of functionality. First, it allows the developer to control the how the software is executing. This is done through control of algorithm and parameter selection as well as starting, stopping and initializing the embedded software. The GUI also provides the designer with the ability to monitor the software for evaluation of performance. Facilities have also been included for data collection for the purposes of debugging run-time errors. This environment has proven to be well worth the effort required for its development.

The GUI has been developed using the DataViews API package. The software for the GUI is in the "dv/" directory. This software is briefly outlined in Table C.6.

C.2.3 3D simulation software

The graphical portion of the simulation environment is developed using the World Tool Kit (WTK) API. WTK provides the developer with a library of function calls for creating a 3D graphical environment. It allows objects to be moved easily in the environment and provides for collision detection of graphical objects. The collision detection is especially useful for this application in modeling the radar sensor.

The software developed using the WTK libraries is listed in the "WTK/" directory. The files in this directory fall into five main categories:

File Name	Description
ADC.c	Performs AD/DA conversion for the hardware on ALX.
ALX_Dynamics.c	Model of motion for ALX when controlling golf cart model in simulation.
BrakeController.c	Controller for the ALX braking system.
DeadReck.c	Calculates estimate of ALX's position when performing experiments outside.
Encoder.c	Driver for reading the encoder for velocity control.
Golf.c	Initializes some of the messageQues.
InitDiscretes.c	Initializes the digital I/O hardware on ALX.
LatController.c	Lateral controller used in development of the lateral controller (not used by virtual bumper).
PositionUpdate.c	Receives vehicle states and makes them available to all processes on both processors.
ReadNav.c	Continuously reads sensors used for calculating position estimates during experiments.
RungeKutta.c	Solves differential equations using a fourth order Runge-Kutta approach.
Scope.c	Collects data for several processes using Stethoscope.
SteerController.c	Controller for the steering system.
TruckDynamics.c	Models the dynamic motion of the Navistar semi-tractor.
UserControlInterface.c	Processes information sent from the DataViews GUI.
VelocityController.c	Controller for the vehicle's forward velocity.
Watchdog.c	Monitors the control loop and initiates an emergency stop when appropriate.
exec	Executable file
makefile	Used to compile and link source files

Table C.2: Source code file listing for the master processor.

File Name	Description
ControlInput1.c	Passes the controller commands to the controllers executing on the master cpu.
CoordTransform.c	Performs several coordinate transformations for the vision data (i.e. camera to world, etc.).
GetPositionUpdate.c	Provides access the the vehicle states calculated by the master processor.
LatController.c	Performs the lateral position control for the host and is used by the virtual bumper.
ObstacleForces.c	Receives range readings and calculates the obstacle positions with respect to the road. The obstacle positions and relative velocities are then used to calculate the three forces (F_{long}, F_{reflex} and F_{LC}) for the virtual bumper.
RangeSensorInit.c	Initializes the data structures for the range sensor based on a input data file.
RangeVisSensorRead.c	Reads both the vision and the range sensor data from a socket when in simulation mode.
RoadForces.c	Generates a road force based on the lateral position of the vehicle.
Scope.c	Collects process data using Stethoscope.
SonarOp.c	Initializes the ultra-sonic sensors and continuously reads the sensors and makes the data available to ObstacleForces.c.
TrackRoadEdges.c	Uses the vision information passed to it by VisionDataProcess.c to track the edges of the lane.
TransformForce2Control.c	Takes the forces and transforms them into controller commands.
UserControlInterface1.c	Performs the appropriate action for commands received by UserControlInterface.c executing on the master processor.
VirtualBumper.c	Main execution loop for the virtual bummer.
VisionDataProcess.c	Receives vision data and transforms it to world coordinate system and passes it on to TrackRoadEdges.c.
VisionDataRead.c	Reads the vision data from the Datacube when operating ALX in experiment mode.
VisionFeedback.c	Provides the Datacube with screen positions of the vision data when operating ALX in experiment mode.
_SENSOR_CONFIG_	Data file which defines the sensor positions when operating ALX in experiment mode.
exec	Executable file
makefile	Used to compile and link source files

Table C.3: Source code file listing for the slave processor.

File Name	Description
CommLink.c	Wrappers for the VxWorks messageQue calls.
Filter.c	Contains some simple FIR filter functions.
MathFunc.c	Some miscellaneous mathematical functions.
Polyfit.c	Performs curve polynomial curve fits for a given set of input data.
TimingFunc.c	Clock functions for timing as well as functions to calculate tick delay values.
matrix.c	Matrix manipulation function calls.
linalg.c	Linear algebra functions calls.
targ.c	Contains functions required if using matrix.c and linalg.c.
makefile	Compiles and links the source code.
share	Executable file.

Table C.4: Source code file listing for files which are shared by both the master and the slave processors.

File Name	Description
CameraCalib.h	Calibration values for the camera mounted on ALX.
FitTypes.h	Data structures used for curve fitting functions.
GlobalDefines.h	Defines used by both of the processors.
GlobalIncludes0.h	Include files included for all source files on the master processor.
GlobalIncludes1.h	Include files included for all source files on the slave processor.
Golf.h	Defines some useful macros.
Messages.h	Defines most of the data structures used by the different messageQues as well as when passing data between platforms (i.e. Dataviews — VxWorks).
RoadEdgesType.h	Defines the data structures for tracking the edges of the road.
SM_Types.h	Data structures used for passing data between the processors using shared memory.
VXWORKS.h	Includes several useful VxWorks standard libraries.
VirtualForceTypes.h	Data structures for the virtual forces.
VisionDataType.h	Data structures for the vision information.
Z8536.h	Defines for the Z8536 chip on the O & R board.

Table C.5: Listing of include files for the VxWorks code.

File Name	Description
BSipc.c	Contain IPC software for WTK to DataViews communication.
BSipc.h	Data structures and defines for WTK to Dataviews communication.
alx.c	The main process loop which initializes the GUI, runs the event handler and closes the GUI.
alx.h	Function prototypes for alx.c and some defines.
var_declarations.h	Global variables which are linked to the GUI screens developed using DVdraw.
driving_scene.c	Draws the driving scenario plots used for documenting the performance of virtual bumper.
driving_scene.h	Function prototypes for driving_scene.c.
echo_functions.c	Functions which perform the desired action when a graphical object is manipulated through the GUI.
echo_functions.h	Function prototypes for echo_functions.c.
init_screens.c	Initializes the screens developed using DVdraw by attaching the appropriate variables.
init_screens.h	Function prototypes for init_screens.c.
playback.c	Records and plays back displayed data for analysis purposes.
playback.h	Function prototypes for playback.c.
plotscreen.c	Makes encapsulated postscript images of the GUI screens.
process.c	Redraws the interface screens if an event has occurred or if there is new data.
process.h	Function prototypes for process.c.
range_screen.c	Generates an overhead view of the vehicle with road and range sensor data shown graphically in real-time.
range_screen.h	Function prototypes for process.c.
rebind.c	Binds the global variables to the appropriate graphical object during initialization.
rebind.h	Function prototypes for process.c.
sigs.c	Generates and receives signals used to determine when to access the information from the WTK simulation.
sigs.h	Function prototypes for sigs.c as well as data structures for shared data and signal defines.
wtk_comm.c	Functions used to receive and send data to the WTK simulation.
wtk_comm.h	Function prototypes for process.c.
makefile	Compiles and links the source files.
alx	Executable file.

Table C.6: Source code file listing for the Dataviews graphical user interface.

Source code files

Configuration files

Graphical data files

Vehicle position data files

Script files.

For the source code files, we have C, C++ and include files. All of the source files are summarized in Table C.7. The configuration files are used to modify the simulation environment easily without re-compiling the software. These configuration files are listed in Table C.8. Of course, there are several data files associated with the graphical objects in the simulation. These data files for the graphical objects are shown in Table C.9. In the simulation, the obstacles and sometimes the host is maneuvered using a data file. These data files are listed in Table C.10. Finally, since the simulation takes several run time arguments, script files have been made which define a particular simulation environment or driving scenario. These script files are listed and described in Table C.11.

File Name	Description
BSipc.c	Contain IPC software for WTK to DataViews communication.
comm.c	Function callback needed for communication between WTK and Vxworks.
comm.h	Function prototypes for comm.c.
gethost.c	Determines the host name of the workstation.
gethost.h	Function prototypes for gethost.c.
gldraw.cc	Some useful GL stuff.
gldraw.h	Function prototypes for gldraw.c.
linklist.h	A C++ linked list object type.
makefile	Compiles and links the files for release 6 of WTK.
makefile_21	Compiles and links the files for version 2.1 of WTK.
makefile_beta	Compiles and links the files for beta release 6 of WTK.
makefile_r6	Compiles and links the files for release 6 of WTK.
radar.cc	Initializes the radar unit placement on the host.
radar.h	A C++ object which contains an array of radar units. This is where the radar measurement is performed and sensor data passed to VxWorks.
roadtrack.cc	Creates the road and terrain around the road.
roadtrack.h	A C++ object which is an array of road segments.
sigs.c	Signal handling functions to allow DataViews to notify WTK when shared memory has been updated.
sigs.h	Function prototypes and data structures and defines for the shared memory.
smarttruck.h	A C++ object which is a linked list of vehicles. This object controls the motion of all vehicles.
track_obstacles.cc	Initializes all of the obstacles in the environment.
track_obstacles.h	A C++ object which is a linked list of all the obstacles in the simulation environment.
truckrc.h	Defines a data structure for the host vehicle. This is needed when the vehicle is initialized using the standard x function calls.
trucksim	Executable file.
trucksim.cc	Main file for simulation. This makes calls to initialize the environment and then enters the simulation loop. Once in the loop, it performs the appropriate calls for each loop and handles any events. When signaled, this loop is exited and the WTK environment is deleted.
truckviews.h	A C++ object which is a linked list of the possible views for the simulation.
vehicle.cc	Initializes all of the vehicles with the appropriate graphical data files and adds them to the simulation.
vehicle.h	A C++ object which is a linked list of all the vehicles in the simulation.
vision_sensor.h	A C++ object which is the simulated vision sensor. This file calculates the roadway information and passes the data to VxWorks.

Table C.7: Source code and include file listing for the WTK simulation.

File Name	Description
ALXsim	Defines the graphical objects for ALX to be used in the simulation.
Template_Configuration	This is a template file for the range sensor configuration file. This states how to make a configuration file and identifies all of the appropriate fields.
Trucksim	Defines the graphical objects for truck to be used in the simulation.
Trucksim_l	Defines the graphical objects for truck which have a lower level of detail than those defined in Trucksim. Used to allow for faster simulation frame rates.
_TMP_CONFIG_	The range sensor configuration file that will be used when the simulation is run.

Table C.8: Configuration files used by the WTK simulation.

File Name	Description
ALX_VISION.nff	Defines the shape of the vision sensor object used for simulations of ALX.
DEFAULT_VISION.nff	Vision sensor object file that is used when the simulation is run.
TRUCK_VISION.nff	Defines the shape of the vision sensor object used for simulations of the truck.
_RADAR*.nff	3D object file created when _TMP_CONFIG_ is used to generate and mount the radar sensors. In the name, * is equal to the sensor number.
ALX files	All of the ALX graphical files are clearly defined in ALXsim.
Truck files	All of the truck graphical files are clearly defined in Trucksim (high detail) and Trucksim_l (low detail).
car_1.nff	Car object in the simulation.
obstacle*.nff	Obstacles in the simulation.
road1.rgb	Defines the color for the road object.
road*a.nff	Road segments which when combined are a full loop on the MnRoad track. In the name, the * is the number for the road segment.
road*z.nff	Road segments which are used for vision sensor. These are invisible and when combined make the full MnRoad loop in the right lane when traveling clock-wise. The * in the name is the number of the road segment.
van_1.nff	Van object in the simulation.

Table C.9: 3D object input files used by the WTK simulation.

File Name	Description
obs_traj*a.dat	Defines the first target vehicle's position during the simulation run. The * in the name is the number which identifies the driving scenario.
obs_traj*b.dat	Defines the second target vehicle's position during the simulation run. The * in the name is the number which identifies the driving scenario.
truck_*.dat	Position of truck which is recorded during a real-time simulation. This file is used for playing back simulation for demonstration purposes. The * in the name is the number which identifies the driving scenario.

Table C.10: Data files that define vehicle positions used by the WTK simulation.

File Name	Description
evalradar	Moves target through the points defined for the radar evaluation and records sensor data.
evalradar2	Same as above but rendering is disabled for higher frame rates.
testalx	Simulation for ALX on circular track with static obstacles.
testtr1*	Overtaking driving scenario. The * is blank for the real-time simulation and b for playing a simulation back.
testtr2*	Waiting to Overtaking driving scenario. The * is blank for the real-time simulation and b for playing a simulation back.
testtr3*	Passed driving scenario. The * is blank for the real-time simulation and b for playing a simulation back.
testtr4*	Passed and cutoff driving scenario. The * is blank for the real-time simulation and b for playing a simulation back.
testtr5*	Parked car on shoulder driving scenario. The * is blank for the real-time simulation and b for playing a simulation back.
testtr6*	Stalled car on highway driving scenario. The * is blank for the real-time simulation and b for playing a simulation back.
testtr7*	Approaching traffic jam driving scenario. The * is blank for the real-time simulation and b for playing a simulation back.
testtr8*	Drifting car driving scenario. The * is blank for the real-time simulation and b for playing a simulation back.

Table C.11: Script files for running the WTK simulation.