REMOTE CONTROL OF A SEMI-AUTONOMOUS ROBOT VEHICLE OVER A TIME-DELAYED LINK

A Thesis

Submitted to the College of Graduate Studies and Research

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in the

Department of Electrical Engineering

University of Saskatchewan

by

MAYNARD R. OLSON

Saskatoon, Saskatchewan, Canada

Fall 2001

PERMISSION TO USE

The author has agreed that the libraries of the University of Saskatchewan may make this thesis freely available for inspection. Moreover, the author has agreed that permission for extensive copying of this thesis for scholarly purpose may be granted by the professor or professors who supervised the thesis work recorded herein or, in their absence, by the Head of the Department or the Dean of the College in which the thesis work was done. It is understood that due recognition will be given to the author of this thesis and to the University of Saskatchewan in any use of the material in this thesis. Copying or publication or any other use of the thesis for financial gain without approval by the University of Saskatchewan and the author's written permission is prohibited.

Request for permission to copy or to make other use of material in this thesis in whole or in part should be addressed to:

Head of the Department of Electrical Engineering
University of Saskatchewan
57 Campus Drive
Saskatoon, Saskatchewan
Canada, S7N 0W0

ACKNOWLEDGEMENTS

The author wishes to sincerely thank his supervisor, Dr. H.C. Wood for his assistance and guidance through the various challenging situations during the development and final preparation of this thesis.

The author also wishes to thank his advisory committee members, Dr. M.M. Gupta, Dr. G.J. Schoenau and Dr. K. Takaya for their thought-provoking questions and valuable comments which have had significant impact on the final form of this thesis.

The author would also like to acknowledge the many helpful meetings with Mr. T. J. Nelson of Prairie Machine and Parts, Ltd. which were instrumental in formulating the design of the system used in this thesis.

Financial support provided by NSERC, Canadian Space Agency and Prairie Machine and Parts, Ltd. of Saskatoon is most gratefully acknowledged and appreciated, as is the financial support from the thesis supervisor, Dr. H.C. Wood.

ABSTRACT

Remote control of equipment or systems over communication links having loop time delays of about 0.5 seconds or more is known to be a significant problem for a human operator in teleoperation mode. "Move and wait " strategy is the normal approach employed by an operator. In order to improve upon the inefficiency of this strategy, a number of solutions can be employed, including use of predictor displays, supervisory control and Smith control. A predictor display aids the operator by presenting a non-delayed view of the remote system output. This allows commands to be issued before the actual delayed output is received via feedback. Supervisory control allows the remote system to operate in semi-autonomous fashion by providing autonomous capability that can be directed by the operator using high-level commands. Smith control provides stable, closed-loop control of systems with inherent delay by effectively moving the delay out of the loop. This thesis presents a robot vehicle control system that includes each of these techniques in the overall design.

The robot vehicle being controlled is intended for underground mining applications. This is a difficult environment for a control system, particularly when autonomous operation is required. Movement of a vehicle in such an unconstrained environment coupled with the problem of sensor readings suggests an excellent application for fuzzy control. Remote control of a robot vehicle basically involves control of speed and steering. A multivariable, fuzzy control system has been developed to accommodate this task. A simplified version of Smith control is used to compensate not only for the time delay but also for the human operator's dynamics. This simplified method does not require the usual estimate of non-delayed plant dynamics, only a reasonably accurate measure of the loop time delay. Semi-autonomous operation provided includes automatic tunnel-tracking and turning into intersecting tunnels. Obstacle-avoidance, based on the use of fuzzy "obstacle factors" forms an essential part of the control system. A neural network-based predictor is also included for the operator's assistance.

System simulation results prove that the simplified Smith control method can compensate for both the time delay and the human operator's dynamics to a very high degree. Given a stable linear plant with no disturbances, the length of the time delay

is essentially irrelevant in maintaining stable control. It is also demonstrated that the predictor allows the operator to control the remote vehicle over the time delay by driving a non-delayed model of the vehicle on a console display. Semi-autonomous operation has not been tested in animated simulation in a mine environment, but preliminary simulations of obstacle-avoidance, automatic tunnel-tracking and automatic intersection-turning have indicated that the design appears sound.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 General

Remote control of equipment is widespread and commonplace. Consumer products include hand-held transmitters used to control audio/video equipment in homes. Garage-door openers allow users to control opening and closing of garage doors from their vehicles. Hobbyists control model airplanes, cars and boats using portable transmitters. Home automation systems allow owners to monitor and control household systems from wherever there are suitable communication facilities

Remote control is used widely in commercial and industrial processes where the operators are physically isolated from the process being controlled. Aircraft, ships and submarines are controlled remotely from the cockpit or bridge, although this may not fit the normal perception of remote control. Power stations, petro-chemical plants, railroads and mining operations use remote control in their daily operations. Remotely-operated (ie. controlled) vehicles (ROV'S) are routinely used for various underwater applications. Spacecraft, including satellites, are controlled from the earth using radio communication links. Remote-controlled vehicles having various levels of autonomy are used by military forces throughout the world on land, sea and in the air for defence purpose such as surveillance and reconnaissance, as well as for offensive operations.

Most of these examples have no significant delay between the time that control signals are transmitted and the time that they arrive at the receiver. For these cases, remote control is relatively straightforward. The biggest problem is probably ensuring the integrity of the forward and reverse communication channels. Line-of-sight transmission (for radio or optical channels), sufficient transmit power and receiver sensitivity, filtering of noise and distortion are important considerations.

Time delay between the transmitter(s) and receiver(s) presents a whole different set of problems, especially when the dynamic characteristics of a human operator must be considered. Controlling a robot vehicle in real time over a

communication link having appreciable loop time delay (> ~0.5 sec) [1] is known to be a difficult task for a human operator. In order to improve system performance over the "move and wait" strategy adopted by human operators in the time-delayed remote-control environment, it is usually necessary to provide some degree of autonomy on the remote vehicle. A term which has been introduced to describe the interaction of humans with remote, intelligent, partly-autonomous systems is "teleautonomous" control [2]. This term includes the full range of autonomy in remote-control systems such as manual/teleoperation, computer-assisted/telerobotics, and semi-autonomous/supervisory control, although autonomous/fully automatic control may not strictly fit the definition. These descriptors are not rigid, since there is often overlap in function and definition. Teleoperation, for example, is often used in a generic sense to mean that there is a human operator in real-time, on-line control of the system. If there is computer-assistance, as will often be the case, telerobotics might be a better description. In the system described in this thesis, there is a combination of autonomies, from teleoperation through supervisory control.

1.2 Fundamentals of Remote-Control Systems

Remote control: "Control of an operation from a distance: this involves a link, usually electrical, between the control device and the apparatus to be operated. Note: Remote control may be over (A) direct wire (B) other types of interconnecting channels such as carrier – current or microwave (C) supervisory control, or (D) mechanical means" [3]. The type of carrier in (B) should be expanded to include transmission in other media such as optical or acoustical transmission. Another definition of a remote-control (or telecontrol) system is "a closed-loop system which must consist of at least the following elements [4]:

1. Sensors of information.
2. A transmission system to communicate information to a remote-control point.
3. A control point which includes a human or automatic decision-making system.
4. Devices to translate information into appropriate control signals.
5. Links to communicate information to actuators at remote-control points.
6. Actuators operated by control signals to effect responsive controlled operations

One of the first reported applications of remote control was the bridge telegraph system for ships [4] . Control commands from officers on the bridge were transmitted down to the engine room where the appropriate control action was performed by human operators. Eventually, with technological development, this process was automated so that the required control action could be accomplished directly from the bridge.

The definition of remote control, as used in this thesis, includes the presence of a human operator. The primary reasons for using remote control are convenience, economy/efficiency and/or safety, each of which has specific relevance to the human operator in the control loop. Garage-door openers are primarily a convenience; centralizing control operations at a remote station is done for reasons of economy and efficiency (and often safety); safety is a major motivator for remote control of operations in environments that are unpleasant, unhealthy or hazardous to humans. These environments include industrial plants (smoke and toxic fumes), nuclear plants (ionizing radiation), undersea and outer space (extreme pressures and temperatures), military theaters of operation (gunfire and explosions) and underground mines (falling rock, poisonous gas, insufficient oxygen, flooding).

1.3 Time Delay in Remote-Control Systems

"Remote" implies physical distance separating the control and controlled sites. Physically-separated sites require communication links to convey the forward control signals and to return the status information for proper closed-loop control. Depending on the type of system being controlled, and the environment, the communication channels can become significant factors in the overall control system. Information-carrying capacity is limited by bandwidth. Information reliability is affected by channel distortion and noise. Bandwidth, distortion and noise are generally not a major problem in remote control since communication links can usually be designed to provide acceptable performance. System sensitivity to these factors is not considered in this thesis. The most significant factor introduced by the communication channel in remote-control applications is time delay. As the total loop time delay becomes an appreciable fraction of the time constant of the process being controlled, the closed-loop control system can become unstable

Time delay, also called transportation lag in control systems, has been studied for years and its effects are well known and documented in standard control system textbooks. The basic problem with time delay in closed-loop control systems is a decrease in stability. Depending on system gain (and other factors), even small delays can render an otherwise stable system as being unstable and unworkable.

Consideration of time delay in feedback control systems with human operators in the loop introduces additional complication. Attempts at modeling human operators as active devices with a transfer function go back to World War II [5] and continue to the present day. It is well known that the human operator is an extremely difficult subject to model due to the complexity of the human information-processing system, inconsistencies in behavior from one person to another, or in the behavior of an individual from one time to another. References on the effects of time delay in human activities go back many years. Many of these original studies were performed by psychologists and medical researchers in their studies on human behavior. In 1960, W.M. Smith [6] reported the effects of a 520 ms feedback time delay when performing nine operator tasks including writing letters of the alphabet, tracing symbols, writing words, etc. The delay resulted in operator frustration, increased difficulty and time to perform otherwise simple and easy tasks, degradation of legibility and various kinds of errors. These reactions are indicative of the problems encountered by human operators in remote-control applications.

Most research into time-delayed remote control has focused on the control of robotic manipulators as opposed to the control of robotic vehicles. While there are many analogous features common to both types of systems, there are significant differences. Manipulators are devices that usually require a high degree of spatial precision in three dimensions; there may be many degrees of freedom depending on the design and the overall configuration governing the motion of the arms or "end effectors". Vehicle movement is essentially confined to two (or three, in some instances) dimensions and does not usually require comparable precision. Manipulators are normally fixed in place and operate in a relatively structured environment; vehicles are mobile and often operate in an unstructured environment. This exacerbates the problem of modelling both the system and the environment, which is normally required for the design and implementation of a traditional control system.

1.4 Time-delay compensation

Various techniques have been devised to compensate for, or ameliorate poor system performance due to time delay. Some of these techniques are specific to remote manipulation but others are more general and are applicable to remote driving as well. Force and velocity feedback in various forms from a slave manipulator back to the master terminal is a very common technique with remote control of robot manipulators. For remote control of robot vehicles, other techniques have been used including Smith control, predictor displays, and supervisory control.

Originally developed for the control of industrial processes having inherent time delays, Smith control [7] (also known as Smith's Method or Smith Predictor) continues to attract attention by researchers. While this technique saw limited use in the analog systems of the day, the ease of digital implementation has resulted in its increased use more recently in practical control systems [8]. The essence of this technique is to replace the conventional controller inside the closed loop with a new design to effectively move the time delay outside of the loop, thereby maintaining closed-loop control of the non-delayed portion of the plant. A traditional Smith controller requires an accurate estimate of the non-delayed plant dynamics and an accurate estimate of the (inherent) delay for good performance [9,10]. Remote control with time delay in the loop is just another example of a closed-loop system with time delay and can be controlled on an end-to-end basis by using a Smith controller. The time delay in this system is essentially comprised of the communication link delays, which can be measured explicitly and accurately. The plant has no inherent delay (presumably), so its dynamics are available for the Smith controller. These factors tend to suggest that Smith control may be ideally suited for closed-loop control of non-delayed plants, including direct teleoperation or telerobotics, over time-delayed links [11].

Predictor displays provide the human operator with an estimated (ie. predicted) output of the system as if no time delay were present. This is usually accomplished in one of two ways: by using a model of the system without time delay, or by running a model of the delayed system in faster time. A predictor display can allow the operator to drive the model in simulation to observe its output in real time. The same control signals used to drive the model can be sent to the remote site so that the actual system output should (ideally) behave exactly the same as that of the model but delayed in time. Obviously, for this to work as well as intended, the model of the system must be

very accurate. Having access to a model of the system can allow predictor displays to perform other functions in off-line mode such as: vary the speed of the simulation, freeze the simulation at any point in the response, make adjustments to modify the system response, etc. A predictor display can be a powerful assistant to a human operator and can provide significant improvement in the time it takes to perform remote-control tasks over time-delayed links. Increasing development of computer-generated graphics is allowing creation of virtual environments and virtual reality-based teleoperation [12]. This can be considered as being closely associated with the function of a predictor display, even providing a complementary and enhancing capability.

For long time delays, teleoperation in real time is not a viable option even if stability can be maintained, because the operator requires feedback from the remote site in order to determine appropriate control actions. In this situation, supervisory control is necessary. This implies semi-autonomous operation. The only other option is fully-automatic, totally-autonomous operation. This option is not of interest in this project since direct operator intervention is required in the control of the remote system. Supervisory control [13] is high-level or task-level control; it is not real-time, on-line control. Control commands are symbolic, requiring on-board intelligence and/or capability at the remote terminal for interpretation and execution. This is in contrast to direct teleoperation where analogic control commands are used to directly control the plant in real-time, on-line fashion. Supervisory control usually requires the transmission to the remote site of goals and subgoals along with necessary instructions for accomplishing them. Basically, this includes sending and updating computer programs to be used at the remote site. Since supervisory control is not real-time control, it is vital that some degree of autonomy be provided on the remote robot vehicle. The reason for this is that with any significant delay in the system, a human operator located at a distant site cannot respond quickly enough to make any necessary correction to the system. If a hazardous situation should occur at the remote site, the operator will not be aware of it until the system status information is received over the time delay. Any corrective action initiated by the operator and transmitted back to the remote site will be similarly delayed. This delayed response could be catastrophic to the robot vehicle or to anything/anyone in the vicinity. There may also be a variety of less critical tasks of an automatic nature required such as monitoring and adjusting onboard system parameters in a timely fashion to avoid system degradation or failure. Therefore, all time-delayed remote-control systems are normally provided with some degree of autonomy sufficient to circumvent such problems.

1.5 Underground Mines

The general area of application for this research is the remote control of robotic vehicles operating in underground mines. Interest in providing robotics and automation in underground mines is high, motivated primarily by economics and worker safety [14-16]. This is a very difficult environment for control systems, compounded by the sensory problems. The walls, ceiling and, to a lesser extent, even the floors are uneven and rough, composed of varying geological elements, resulting in complex reflections made even more difficult if moisture is present. Tunnel openings and intersections may be difficult to identify accurately. There may be a variety of obstacles and hazards – fallen rock or other debris, mine personnel, other machines, etc. Lighting may be low, resulting in poor and/or confusing visibility. Drivewheels may slip or skid. There may be exhaust smoke from other vehicles and dust in the air. Vibration and vehicle motion is continually present. Machine sensors such as video and sonar do not have an easy task, with the reflections, scattering and absorption of waves. For conditions like these, where sensory information may be vague, ambiguous and imprecise, fuzzy control has often been used successfully [17].

1.6 Intelligent Control

Intelligent control is a relatively recent term that has been introduced to describe control techniques that tend to emulate the decision-making process of humans, animals or biological systems when faced with a control problem [18]. Whereas traditional control is based on the use of mathematical models and requires differential equations in the modeling and solutions [19], intelligent control does not require mathemetical models or equations in the traditional sense . Essentially, it is a model-free, rule-based discipline employing "soft computing" techniques such as fuzzy logic and neural networks to accomplish the necessary control.

Fuzzy logic control (or simply, fuzzy control) requires three major components: a knowledge base, a rule base and an inference engine to perform the computation [20]. The knowledge base is comprised of a number of "membership functions" spanning the "universe of discourse" (i.e. dynamic range) of each of the input and output control variables. The membership functions are fuzzy sets that represent linguistic variables over the universe of discourse of each control variable. They decribe the degree of membership (from 0 to 1) in each linguistic variable of any numerical (ie. "crisp") value

on the universe of discourse. The rule base is a list of logic statements relating the input and output linguistic control variables, describing the resulting control action for all combinations. These statements are typically of the form:

IF (input A is X) AND (input B is Y), THEN (output C is Z).

The rule base is often represented in table form as a "fuzzy associative matrix" or FAM. The inference engine accepts the system inputs as "fuzzified" linguistic variables and degrees of membership, performs the fuzzy inference process and "defuzzifies" the result to produce a specific "crisp" output value. This method of taking input information in descriptive form, applying logical rules and inferring the output result is clearly comparable with the human reasoning process, albeit at a rudimentary level.

Neural networks are configurations of artificial neurons interconnected in various ways in order to accomplish a desired input-output mapping [21]. The most widely used application of neural networks is in pattern recognition and they can accommodate both static and dynamic patterns. The input pattern to be recognized is connected to a number of input nodes, usually having no associated weights. These input nodes are then connected to a second (hidden) layer of neurons via connections having adjustable weights. The neurons usually have an adjustable bias input as well. The weighted inputs are all summed together in each neuron and the resulting signal level is applied to a nonlinear activation function (usually sigmoidal). The output of this function represents the output of the neuron. The outputs of all hidden layer neurons are then usually fed to a subsequent layer of neurons via connections having adjustable weights. This layer may be another hidden layer or may be the output layer. Any number of layers can be used, but usually no more than three are used. The output layer neuron(s) often contain a linear activation function to effectively provide an output which is a linear combination of its inputs. Some neural networks have feedback connections with delay units from hidden or output layers back to previous layers creating recurrent networks for modeling dynamic processes. Many types of networks are possible, depending on the configuration and interconnection of the neurons. Neural networks are capable of emulating linear and nonlinear functions with great accuracy given an appropriate structure and sufficient number of neurons. By adjusting the weights using a method such as back-propagation, the network is trained to realize the required input-output mapping. This learning capability of neural

networks is another example (again, very rudimentary) of human/animal/biological system intelligent behaviour.

Using a neural network to model the remote-control system without time delay should be sufficient to act as a suitable predictor to provide the operator with a non-delayed estimate of the actual system output. This non-delayed output from the neural network model is available immediately, before the actual remote system can respond due to the time delay. Therefore, it is considered to be a "predicted" output and the neural network model is called a "predictor". The predicted output should ideally be identical to the actual system output if there was no time delay.

Both fuzzy logic and neural networks are currently being used to control complex nonlinear systems which may be difficult or impossible to model in the traditional mathematical sense [17,22,23]. Both fuzzy logic and neural networks can model nonlinear functions of arbitrary complexity. Both techniques can work with input data that may be vague, ambiguous or imprecise. Both techniques are robust in the sense that system failure is graceful, not catastrophic if a portion should fail. These are all powerful arguments for using intelligent control techniques.

The underground mine environment described in section 1.5 clearly illustrates its unconstrained nature and the difficulty of obtaining accurate, deterministic measurements. These factors all add to the complexity of the control problem for this application, making conventional control extremely complicated, if not impossible. This is another strong argument for using fuzzy control [17]. The use of linguistic variables and logic rules are particularly appealing for describing and defining control action in humanistic terms, particularly in a system controlled by a human operator, where a variety of control strategies may be employed depending on the (fuzzy) inputs. Lastly, the multivariable requirement of the speed and steering control systems of the vehicle seemed to lend itself to fuzzy control, largely due to the descriptive nature of the control functions. For these various reasons, fuzzy control was chosen for this project. Fuzzy control is becoming more common in many control applications. Digital processors and computers are inherently amenable to the use of "soft computing" techniques. Coupled with the humanistic variable descriptors and logic design process, it is easy to see that fuzzy control will continue to proliferate, even encroach upon traditional areas of conventional control.

1.7 Objectives

The objective of this thesis is the design of a system for the remote control, over a time-delayed link, of a robot vehicle operating in an underground mine environment and incorporating intelligent control techniques [24]. Included is a multivariable control system based on fuzzy control, a display showing the predicted system output from a neural network predictor, Smith control for time-delay compensation in teleoperation mode and supervisory control for semi-autonomous operation. Autonomy is provided by including obstacle-avoidance in the system design, as well as automatic tunnel-tracking and automatic steering into intersecting tunnels. System dynamics, often omitted in many robotic applications, are included. Human operator reaction delay and neuromuscular response are also included, as well as their compensation in the overall system design.

1.8 Outline of the Thesis

Chapter 1 introduces the subject of the thesis and briefly discusses the various topics that are fundamental to this project. These topics are elaborated on in subsequent chapters. Chapter 2 presents a literature review of relevant research. This includes performance results of human operators in control of various systems with time delay, methods of compensating for time delay and performance results of modeling human operators in control applications. Chapter 3 covers the basic theory of the problem with time delay in control systems. Smith control theory is included here, and the simplified Smith controller, particularly as it relates to remote control, as well as its use in compensating for human operator dynamics. The rationale for using equivalent circuits in simulations involving predictors is also explained here. Chapter 4 describes the complete system design. All subsystems are described along with their functions. These include the various fuzzy controllers, human operator models, Smith control and the semi-autonomous modes of operation under supervisory control. Chapter 5 presents the system simulations and their results. All system functions and operating modes are tested and the resulting performance is demonstrated. Chapter 6 presents the summary and conclusions. The contributions of this thesis are included here. Finally, work suggested for the future is identified. This includes work originally planned and intended for this thesis, but time did not permit.

2. LITERATURE REVIEW

2.1 Time Delay in Remote Manipulation

Interest in space exploration in the 1950's and 1960's resulted in a number of studies involving human operators in control systems having time delay in the loop. Since much of this space interest was concentrated on lunar exploration, the time delays being considered had to include at least the two-way (loop) transmission delay to the moon and back of 2.6 seconds.

In the early 1960's, Sheridan and Ferrell [25] and Ferrell [26] reported the effects of time delay in remote manipulation tasks. This was the result of a study for NASA [27] which was interested in teleoperation applications in outer space, and particularly on the moon. An operator at a master terminal controlled a slave manipulator through a communication link. Loop delays of 0, 1.0, 2.1 and 3.2 seconds were introduced in the link between the master and the slave terminals. Two types of experiments were performed: the first was basic move-and-grasp, at various levels of difficulty; the second was a more complex manipulation. As delay increased, the total time to complete the task increased. In addition, the completion time increased with increasing level of difficulty, as expected. It was reported that the operators experienced difficulty and fatigue when delays were present. An interesting discovery was that each operator would automatically adopt a "move and wait" strategy to accomplish the tasks in the presence of delay. This involves an open-loop move followed by a period of waiting to see the effect or response.

If time delays are appreciable and if the task is relatively complex involving a number of moves, "move and wait" teleoperation can become very time-consuming. Operator boredom can also become a factor, leading to inattentiveness and impaired judgement.

2.2 Time Delay in Remote Driving

One of the earliest experimental studies on the effects of time delay in teleoperation of a remote vehicle was performed by Adams [28,1] for NASA. Its purpose was to investigate performance of systems using human operators in the remote control of machines at long transmission distances - specifically, in remotely driving a vehicle on the moon from the earth. There were two phases to the study: Phase I investigated the performance of human operators in simple simulated tracking tests with loop delays up to 6 seconds using various tracking speeds and both velocity and acceleration controls; Phase II involved the teleoperation of an actual vehicle using loop delays of 0, 1/2, 1, 2 and 3 seconds and fixed velocities from 0.12 to 0.82 m/s (0.4 to 2.7 ft/s).

The tracking experiment used a simulated vehicle and various tracks on a video screen. A number of graphs were generated, clearly illustrating the degradation in tracking performance with increased time delay. For the actual vehicle control, both a continuous course and an obstacle course were marked out on a parking lot. Two-wheel and four-wheel drive vehicles were remotely-controlled using loop delays of 0, 1/2, 1, 2 and 3 seconds at various speeds. A television camera was mounted on the vehicles to provide visual feedback. Top speed was 0.82 m/s (2.7 ft/s) with other test speeds ranging down to 1/8 of top speed.

After a learning period of driving with no delay, operators found a delay of 1/2 second to be "a bit of a shock", requiring more concentration for satisfactory driving. One second delay required extreme concentration, both on the present situation (on the display) and the future situation (deciding on steering control action to be performed). This concentration was comparable to "real physical work". A delay of 2 seconds (and top speed) resulted in total loss of control. Operators would also lose their vehicle orientation completely. By reducing the vehicle's speed, more delay could be accommodated. Satisfactory control comparable to full speed with no delay could be attained at 3 seconds delay if speed was reduced to 1/7 of full speed (0.12 m/s). At 1/3 of full speed (0.27 m/s) and 3 seconds delay, performance was comparable to that with full speed and 1/2 second delay. In light of the results obtained, an absolute maximum speed of 4.5 m/s (10 mi/hr) was recommended for remote lunar driving from the earth, with speed reduced to ~0.06m/s (~1/8 mi/hr) for complex terrain navigation.

Results of the actual vehicle tests produced reasonably good correlation with the simulated vehicle tracking tests. Comments from the operators were very enlightening including the suggestion for using prediction to aid the operator.

In a more recent study [29], seven operators took turns driving a robot vehicle through a maze containing obstacles. Two runs were first made with no delay in the remote-control system. One of these runs was performed with the operators viewing the robot and maze directly; the second run was made immediately following the first using a video camera and monitor, with the operators and monitor in a separate room. The average time of each run was 53.1 seconds and 52.4 seconds, respectively, indicating no degradation in performance when using a video display. A third run was performed on the following day with a delay of 3 seconds added to the control loop. On their first attempt with delay in the loop, four of the seven operators experienced failures soon after starting. All runs, successful and unsuccessful, were timed to completion or failure, whichever occurred. Although identification of which runs were successful was not provided, the average time for all was about 245 seconds, ranging from 170 to 350 seconds. The degradation due to time delay is apparent.

2.3 Time-Delay Compensation in Remote-Control Systems

2.3.1 Force/Velocity Feedback

Before examining remote control of vehicles under time delay, we will briefly look at some of the recent research on techniques for compensation of time delay in the remote control of robotic manipulators. The vast majority of research papers on robotics with time delay involve manipulators rather than vehicles. While this section is applicable only to manipulators it is informative to see how much delay can be accommodated in this area and the various techniques involved.

When there is no delay in the system, good manipulator control requires force feedback from the slave to the master controller. However, delays as short as 0.1 second can result in instability [30].

Using principles of passivity and scattering theory, stability of a telerobotic manipulator system was established at delays of 40 ms, 200 ms and 2 seconds [30]. For that system, the 2-port communication link was designed to look like a 2-port

lossless transmission line. This guaranteed passivity regardless of the amount of time delay. The authors indicated that the communication circuit cannot destabilize the teleoperator, regardless of the time delay. Two tests were conducted; first, the slave was commanded to move until contact was made, then the force was ramped up and down; second, backdrivability was tested with the master swinging freely and the slave being hammered twice and then being moved sinusoidally. In all cases, stability was maintained using the passive control law that was derived.

A variation of this technique, while retaining the principle of passivity, involved transforming the system force and velocity variables into wave variables [31]. This also enabled treating the system like a transmission line which allowed matching the end terminations to the "characteristic impedance" to prevent reflections and instability. Stable tracking was achieved with a delay of 1 second up to the point of contact of the manipulator. At that point, force feedback was used. It was indicated that, in using this type of technique, system performance degrades smoothly with increasing delay.

Shared compliant control (SCC) is a recent development for manipulator telerobotics [32]. With SCC, a human operator controls a telerobot having a compliant telerobot hand. The compliant hand softens collisions and allows for contact using little force. Using SCC, kinesthetic force feedback (ie. sensed through human nerves and tissues) to an operator resulted in stable telerobotic control at delays up to almost 1 second [32]. Increasing delay up to 4 seconds did not cause instability if the force feedback was removed as long as the SCC was retained. It was demonstrated that, by using this SCC method, tasks could be completed with up to 8 seconds delay. Tests included maintaining a constant force and peg-in-hole insertion at the slave site. This system effectively used two control loops: an outer loop conveying force information (end-to-end) and an inner (local) loop at the remote site controlling compliance and damping. By removing the force feedback to achieve control at delays longer than 4 seconds, the overall system is open-loop on an end-to-end basis despite having closed-loop control at the slave site.

Another unique procedure incorporated both delayed force and delayed position information to provide a "modified force feedback" (MFF) at the master terminal [33]. The predictive effect resulted in stable teleoperator control with short time delays up to 2 or 3 seconds for both position and force tracking tasks.

Ideal teleoperation would provide the operator with the feel of directly manipulating the end effector in performing a task [34]. Since the forces F and velocities V at each end are related by the end impedance Z = F/V, perfect transparency requires that the impedances of each end be equal, Zmaster = Zslave. Lawrence [34] has studied the performance of teleoperators on the basis of transparency. Four information channels conveyed the forces and velocities between each end. In a manner similar to [30] and [31], a transformation of variables was used to modify the architecture so as to ensure passivity (and therefore stability) in the presence of delays. Transparency vs. stability was studied over a range of impedances for three architectures: position-position, position-force and "passivated" position-force. Transparency and stability were found to be conflicting objectives, requiring tradeoff. Transparency varied with impedance. A delay of 50 ms was used in the study. It was reported that the maximum acceptable delay for this system was "probably" around 2 seconds.

Recently, a controller has been designed for a bilateral, force-reflecting teleoperator using H_∞ control and μ-synthesis techniques [35]. Using a model of the shuttle remote manipulation system shoulder joint, stable control was achieved at communication delays of 2 seconds in each direction. This was accomplished by considering the time delay as a perturbation to the system and designing the controller to compensate for perturbations of $| \leq 2 |$ seconds. This controller was shown to be unstable at a 3 second delay.

The previous results show that, for most cases, the maximum time delay that can be compensated in the remote control of robot manipulators using the techniques described is about 2 seconds. These are essentially passive systems, with no loop gain, and involving the transmission of force and velocity information between the master and slave terminals. They are not applicable to the remote control of a robot vehicle.

2.3.2 Smith Controllers

Triantafyllou and Grosenbaugh [36] have successfully used a Smith controller combined with a compensator based on LQG/LTR methodology in the remote control of a tethered submersible vehicle, ARGO. Linear quadratic gaussian (LQG) methodology with loop transfer recovery (LTR) was included to ensure sufficient robustness margins, with the overall control system being referred to as a "robust Smith

controller". The vehicle was controlled through 'dynamic positioning' of the surface vessel. The surface ship's thrusters were used to provide the control force and the hydrophones and submerged pingers provided position information. A vehicle weighing 17000 N in air was used, along with a 2500 m tether having a loop time delay of 40 seconds. Human operators are not able to control systems with such long time constants [36]. Simulation and actual tests resulted in good (ie. stable) overall performance.

There has been a considerable amount of research performed on the subject of Smith control over the years since it was introduced. The results of some additional research is included in Chapter 3 which discusses Smith control in detail.

2.3.3 Predictor Displays and Virtual Environments

Predictor displays and predictor instrumentation have been used in the control of submarines, aircraft and space vehicles [37]. They have assisted human operators in controlling diving and establishing depth control in submarines [37]. The prediction is based on running a model of the system in fast time. It is reported that a novice operator with only ten minutes of training and using a predictor can operate a system as well or better than a skilled operator using conventional instrumentation. Typical prediction time spans can be quite long: in the range of 3 to 6 seconds for helicopters and from 25 to 30 seconds for submarines. A number of simple studies were done using this technique to investigate the use of predictors in controlling space ship rendezvous, docking and landing [38]. Experimental results using a variety of people with different backgrounds showed significant improvement when using a predictor over the case with no predictor. Other studies involving submarine simulations have shown that collisions using standard instrumentation are nearly three times as many as when using predictor instruments [39].

One of the first major studies involving the use of a predictor in the remote control of an actual vehicle with signal transmission delays was by Arnold and Braisted [40]. A human operator was used to steer a robot vehicle first along a track and then through a maze using a communication link with adjustable time delays and a delayed television display of the landscape. Superimposed on the display was a marker showing the predicted location of the vehicle for the current steering commands. Without the predictor and marker, an operator must mentally estimate the future location of the

vehicle taking into consideration the nature of the terrain, implicit dynamics of the vehicle and any control commands made up to that time. Having a predictor frees the operator from the stress of performing such mental calculations and requires only that the operator steer the predicted marker in real (ie. operator) time. Test results showed that, for a loop delay of 2.6 seconds and vehicle speed of 2.16 m/s (7.1 ft/s) , an operator could drive the vehicle nearly as well by using the predictor as one could drive without time delay (and no predictor). The predictor was not found to be of great value during open field driving, but was useful when precise maneuvers were needed, such as driving through a maze.

Tests of delayed tracking with and without preview (prediction) were conducted by Stark et al [41]. One-dimensional tracking of a target on a video screen by a human operator was studied using delays from 0 to 0.5 seconds in 0.1 second increments. Preview experiments included the location of the target for the next 0.5 seconds. The target motion followed the pseudrandom path generated by the sum of fourteen nonharmonically-related sinusoids. Without delay, the gain response was flat (0 dB) to about 1.6 Hz; the phase curve revealed the human response delay of about 200 ms. Preview had virtually no effect on the delayed gain response, but significantly improved the delayed phase response by about 2/3 of the delayed phase.

Conway, Volz and Walker [2] have introduced the term "teleautonomous system" to include the whole range of human involvement with control systems. They used prediction in conjunction with time and position "clutches" to overcome the degrading effects of time delay in the control of a remote manipulator. The time clutch was used to disengage synchrony between operator specification time and telerobot manipulation time. This allowed the operator to plan and generate path segments in an off-line manner. The position clutch was used to disengage position synchrony between the simulator and manipulator paths. This allowed control of the simulation to test possible paths in different situations. When the simulation was in a good position, engaging the position clutch provided a short, smooth path to the previous path. By using these clutches, task completion time was significantly reduced over systems using pure prediction which, in turn, provided lower completion times than systems having no prediction. Time delays of 0, 2 and 4 seconds were used in the tests.

van de Vegte, Milgram and Kwong [42] used a modified optimal control model of a human operator to control the arm of a simulated submersible robot in a welding task. Operators had to contend with underwater turbulence while positioning a cursor at the end of a robot arm. Delays of 2, 4 and 6 seconds were used in the control loop. No prediction, Taylor prediction and "dynamical" prediction were all tested. The prediction took the form of having the cursor at the end of the robot arm move relative to the submersible. A two-dimensional display of a three-dimensional environment was then presented to the operator. Comparing the model to real operators revealed some difficulties in the modelling process. In some instances the human operator seemed to perform better than the expected optimum. It was hypothesized that the order of the internal model increases as the operator continues to gain experience. It was also determined that a human operator will always utilize some predictive behavior in time delay cases, even when requested not to.

In a paper summarizing the state-of-the-art, Sheridan [43] discussed the use of predictor displays. For relatively short predictions, a Taylor series extrapolation of current state and time derivatives is satisfactory. For longer predictions, and to account for nonlinear dynamics (e.g. saturation), a better method is to use current state and time derivatives plus expected near-future control signals to run in a model much faster than the process being controlled. This second type is better for controlling single-entity (ie. rigid body) devices, such as a vehicle. It is not as good for telemanipulators having multiple degrees of freedom [43].

Hirzinger et al [44] used a predictor display in the telerobotic control of a space-based manipulator over loop delays up to 7 seconds. Due to the long delay, bilateral control (forces fed back to the operator) was not used. A multisensory gripper provided position, touch and video sensor information that continually updated a world model at the ground-based control center. Using predictive stereographics simulation based on the world model, delays of up to 7 seconds were compensated for without major problems. A floating object was caught by the manipulator despite a 6 second round trip delay. It was reported that the system at the remote site and the time delays had to be modelled very accurately for satisfactory prediction. This would not work so well for unknown or unstructured environments. Using data bus speeds up to 6 MBaud the system was very computationally intensive.

Kim and Bejczy [45] also demonstrated the successful use of a predictor display in the remote control of a robot manipulator with round-trip delay up to 8 seconds. Three fixed television cameras at the slave terminal transmitted video images back to the master terminal 2500 miles away. Computer-generated graphics were used to show predicted motion of the manipulator by overlay on the video displays. The operator generated the manipulator motion in simulation, which was visually checked for correctness before actual commands were sent to the slave terminal. At that site, motion was assisted by automatic compliance control. The system required four components:

1. High-fidelity graphics models of the robot arm and objects.
2. High-fidelity calibration of 3D models to 2D views.
3. Overlay of graphic models over actual images of arm and objects.
4. Control of motion of graphic images to be the same as for the real robot arm.

Excellent results were obtained (positioning errors in order of 1%).

Lanteigne [46] used a predictor display in the remote control of a two-dimensional robot manipulator over a one-way delay of 5 seconds. The operator was required to track a target moving for up to 20 seconds in a straight line in two dimensions. Both kinematics and dynamics of the manipulator were modelled in the predictor display. Each component in the teleoperation loop was modelled, including the operator, and tested first in simulation. Actual tests with a robot were then performed. After a settling time of approximately 3 seconds, zero steady-state tracking was attained, proving the benefit of predictor displays. Tracking a straight-line is probably one of the easiest tracking tasks with or without delay. The model of the manipulator for the predictor was well known. Manipulator inverse dynamics was also used to decouple the manipulator dynamics and linearize the control system.

In general, predictor displays allow operators to "lead" the actual feedback and take larger control actions with confidence, improving time by up to ~50% [43]. A predictor display effectively decreases the time constant of the controlled process by an amount equivalent to the prediction span, reduces the gain of the human transfer function and the control action and increases the damping of the closed-loop system [43]. There are also a number of limitations in the use of predictors [43]: plant

dynamics must be modelled accurately; position, scale and perspective must be carefully calibrated to the video display; depth prediction in the direction of the video image plane is not very useful due to the two-dimensional display. These limitations are particularly applicable to the remote control of robot manipulators, where precision is essential. The same degree of precision is not required for the remote control of robot vehicles, but the general advantage of seeing the road ahead to assist the operator cannot be questioned, as reported as early as 1963 [40].

A recent area of research that is receiving increased attention is virtual reality-based teleoperation. A computer-generated virtual environment can be considered to be in the same general category as a predictor display, even though the one does not necessarily imply nor preclude the other. An excellent discussion of the state-of-the-art of virtual reality-based teleoperation is presented in [12], with particular emphasis on coping with time delay, and accounting for human factors such as: enhancing computer-aided teleoperation, improving bilateral and shared control, developing human-centered architectures, and enhancing sensory feedback to the human operator. The primary application is the control of remote manipulators in general and for special applications such as telesurgery and micro- and nano-scale teleoperation. Mobile robot teleoperation is also discussed, but real-time control is limited to the delay-free case. Examples illustrating control of mobile robots over time delay invariably utilize some form of supervisory control using a virtual reality-based master station. Another term that is used almost interchangeably with virtual reality is telepresence. While virtual reality attempts to provide the operator with the "illusion of presence" at the remote site, telepresence attempts to achieve an "actual feeling of presence" at the remote site [12]. From the point of view of controlling a physical system, the latter description would seem to be preferred. For the particular project described in this thesis, there is no significant use for virtual reality (or telepresence), other than its possible use as an enhanced predictor display.

2.3.4 Supervisory Control

All of the foregoing developments in the remote control of processes under conditions of time delay have retained the human operator at the master site in the direct control loop. The various techniques discussed have indeed provided various degrees of improvement over the "move and wait" strategy that operators tend to adopt under "straight delay" control situations. However, the operator still provides

on-line control, albeit delayed, and albeit under predicted conditions at times. The only way of avoiding such continuous control by the operator is to provide the slaved robot with increased autonomy, such that it can assume control over its own actions for a certain period of time. If this period of time is long enough, the destabilizing effect of the delay can be circumvented. Instead of sending low-level analogic control signals to the slave terminal, the master terminal can send goals, subgoals or complete sequences of subgoals using high-level symbolic commands. This "supervisory control" method was proposed back in 1967 [47]. A remote computer at the slave site would then proceed to carry out the required control action autonomously. A local control loop at the slave site would provide delay-free closed loop control at that point. A feedback loop from the slave to the master terminal would provide a (delayed) supervisory loop. At the master terminal, a control loop could be closed through a local computer providing non-delayed "quasi-feedback" for prediction purposes. Sheridan [13] has described a spectrum of control modes (ie. teleautonomous control) which range from total manual control (teleoperation) through computer assistance (telerobotics) and supervisory control (semi-autonomy) to fully automatic control (autonomy). He provides a summary of the state-of-the-art in 1993 in [43].

Information about the Soviet lunokhod program in the early 1970's has revealed a number of problems they encountered in remote driving on the moon [48]. Besides visibility difficulties caused by glare from the sun, black shadows and lack of depth perception, there was operator boredom associated with the 2.6 second loop delay. They did not use direct teleoperation. Using rudimentary supervisory control, operators sent commands which onboard computers used to steer the vehicle and to control rotation of the wheels. Movement was slow and the lunokhods stopped between commands to avoid getting into trouble.

Most of the recent research on time-delayed remote-control systems tends to focus on some form of supervisory control. Lindsay [49] used supervisory control to remotely control a Puma manipulator robot over a 5 second delay. Hybrid force/ position control was used to position an end effector having a compliant, sensor-based wrist. A kinematic model of the slave environment was established at the master site from sensor information at the slave site. Inaccuracies in this model, coupled with noisy or imprecise commands were compensated for at the remote site using the compliant wrist and "guarded moves" (described as moves "until some expected sensory event occurs"). The human operator interacted with the kinematic model in

real time and commands were automatically generated and transmitted to the slave site at least once every second. The command conveyed information on forces and motions, time durations and co-ordinate frames which the slave computer used to initiate actions. This system is able to operate semi-autonomously in partially-known, unstructured environments over time delays greater than 20 seconds. The model serves as a predictor for the operator to work in real-time. Since the operator's control commands are based on the local predictor model, model accuracy is paramount. This can work only in well-structured environments.

Stein [50] extended Lindsay's "tele-programming" [49] supervisory control technique by incorporating behavior-based control at the slave site as well as enhanced error diagnosis and recovery techniques. A Puma manipulator robot, having six degrees of freedom was remotely controlled over a 10 second loop delay. Tasks involved puncturing and slicing blanket material and bolt extraction and re-insertion. Individualized skills, which behavior-based control provides, worked well for the system described. However, it was questioned whether or not this technique would be appropriate for controlling systems having long time delays (e.g. Mars rover control with 8-20 minute one-way delays [51]). It was also felt that the techniques described in this study would not be as effective for tasks involving little or no force feedback. Having an operator available "allows for simplified and robust construction of the remote site semi-autonomous controller." As with Lindsey's technique, this would work only in well-structured environments, where manipulator robots tend to be used.

Graves [52] described a generalized architecture for teleautonomous systems based on the object modeling technique (OMT). An extensive review of system-specific architectures used in telerobotics and autonomous systems was presented to identify the various requirements for a more generalized scheme. Using OMT, he developed a design for an action selection mechanism (ASM), which is the "heart" of his proposed teleautonomous control system. The ASM interfaces with sensors, various command input entities, planner, controller and robot. The ASM is a software-intensive subsystem using 12000 lines of ANSI C code to transform a series of inputs from various sources into streams of action requests. These requests are then forwarded to other blocks for subsequent action. The complete system is very flexible, hierarchical and supervisory in nature. Simulation of the ASM was performed to verify its functional performance. Time delay in teleautonomous systems was not explicitly addressed or tested. No hardware system tests were performed.

Human supervision, if not direct teleoperation, is mandatory at the present state-of-the-art of robotic systems. This control could range from direct teleoperation in manual mode to supervisory control in semi-autonomous mode. Based on the current state-of-the-art as suggested in this section, we can identify the approximate maximum time delay that can be accommodated using each of the four specific teleautonomous control modes (type and mode of control). A table showing these modes and approximate delay limits is presented below for comparison purposes.

Table 2.1. Approximate maximum delay limits in teleautonomous control

Type of Control	Control Mode	Delay	Limit
Teleoperation	Manual	Short	~ 4 s
Telerobotic	Computer-Assisted	Moderate	~ 10 s
Supervisory	Semi-Autonomous	Long	~ 20 s
Fully Automatic	Autonomous	Very Long	~ 18+ s

2.3.5 Other Techniques

There is constant research into the general problem of compensating for time delay or dead-time in control systems. Some of the recent techniques which have been developed are based on the use of H_∞ control [53], neural networks [54], fuzzy control and adaptive fuzzy logic control [55,56]. While these techniques may well see application in remote manipulation or remote driving, they have not been examined in detail and have not been considered for the remote control system described in this thesis.

It should be mentioned that another active area of research is involved with the remote control of both manipulators and mobile robots over the Internet [57-61]. A major problem at present is the unpredictability of the transmission latency (ie. time delay) and throughput making modeling difficult [60]. While research efforts are being made in the use of predictive-aiding techniques for direct, on-line control, the emphasis seems to be on providing intelligence (on mobile robots) for increased autonomy [61]. This can take the form of behavior-based reactive control and/or supervisory control. Since the Internet is a public medium, however, it is not likely to be available for

commercial purposes such as the remote control of mining machines, so this particular medium has not been considered in this thesis.

2.4 Modeling Human Operators in Control Systems

Modeling human operators in control systems is a very difficult problem, compounded by many factors. Part of the problem is consistency – there is no way of guaranteeing that a particular operator will respond to a control situation the same way every time. Adequate operator training will overcome this problem to a large degree. Many human operators doing the same task yield results that are similar in terms of reaction delay and response characteristics, but there is often considerable variation among these results – a uniformity problem. There is also a strong task-dependent aspect. Human operators tend to approach different tasks in different manners, frustrating attempts to determine convenient transfer function models of the operators.

Human operators in closed-loop control systems are usually involved in performing a tracking function of some type. Two types of display are commonly used for this purpose: a pursuit-type or compensatory-type. The pursuit-type of display shows both the "target" (desired output) and the "follower" (actual output), with the operator controlling the movement of the follower to match the target; the compensatory-type of display shows only the difference between the target and the follower as an error signal which the operator attempts to minimize. The target and follower are not explicitly shown.

2.4.1 Transfer Function Model

Reports of studies [62] showed that early researchers concentrated on identifying and modeling specific characteristics of human operators, including:

1. Gain, K – degree of amplitude response to stimuli.
2. Reaction time delay, r – time delay in initiating response to stimuli.
3. Neuromuscular time lag, T_N – time constant of neuromuscular system.
4. Compensation parameters – phase lead, T_L to accommodate anticipation/prediction ability and phase lag, T_I to accommodate deliberate holdback action.

The general transfer function model, including all of the above terms is [63]:

$$H(s) = \frac{Ke^{-rs}}{T_N s+1}\left\{\frac{T_L s+1}{T_I s+1}\right\} \tag{2.1}$$

This model has been used by many researchers over the years [1,28,46,62,64-69] with some variation depending on the particular study. The non-bracketed portion represents the inherent response characteristics of the human operator, while the equalization terms in the bracket represent the characteristic response that an operator will tend to automatically adopt depending on the control task.

Adams [1,28] made a comprehensive study of the effects of time delay in the remote control of vehicles by human operators. Tests included both simulated driving using pursuit-tracking and actual vehicle driving under various time-delayed conditions. Results showed the significant problem that time delay presents to human operators and system performance (described in section 2.2). The simulations used the following transfer function for the human operator/display combination:

$$H(s) = Ke^{-0.25s}\frac{(1+T_L s)}{(1+0.25s)} \tag{2.2}$$

This assumed a nominal operator reaction delay time r of 0.25 seconds and a neuromuscular lag T_N of 0.25 seconds. The overall system gain K was unity and lead times T_L of 0, 0.5,1 and 2 seconds were tested. Results of these simulations were compared with those using a human operator (both in simulated driving tests). Various speeds were used with both no preview (ie. no forward vision) and simulated preview (unobstructed and unimpaired forward vision). Overall results varied dramatically with speed, type of control (velocity or acceleration) and lead time. In general, simulated preview provided best tracking, as he expected. At lower speeds, and time lags less than about 0.5 seconds, the transfer function model had relatively good results, especially with zero lead times. Increasing the lead times did not seem to offer much improvement, even at larger delays or higher speeds. The model did tend to exhibit very general shape characteristics similar to human operators, but with less accuracy. No experiments were performed using the model in actual vehicle tests.

Leslie [62] found that during pursuit-tracking tasks, human operators switch back and forth between open-loop and closed-loop control. This was modeled as shown in Figure 2.1(a). His tests established that, while the open-loop model was a necessary part of pursuit-tracking, the most prominent mode is closed-loop control as one would expect. On this basis, the block diagram was simplified as shown in Figure 2.1(b) and using Equation (2.1) by letting T_L and T_I equal zero and letting $K_h = K$, or

$$H(s) = \frac{K_h e^{-rs}}{T_N s + 1}. \tag{2.3}$$

He determined that human operators have cutoff frequencies between ~ 0.56 Hz and ~ 0.78 Hz (ie. ~ 4 to 5 r/s) depending on course complexity and no time delay τ in the system. The resulting closed-loop response is

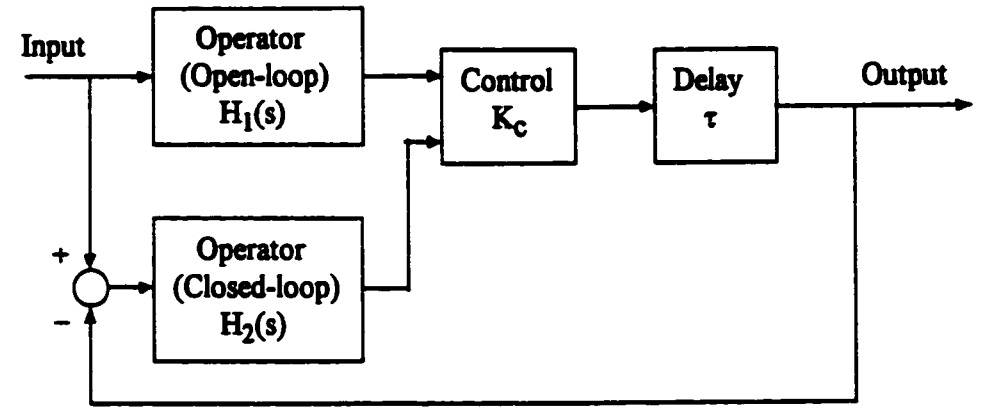
$$\frac{O(s)}{I(s)} = \frac{K}{e^{ds}(T_N s + 1) + K} \tag{2.4}$$

where $K = K_c K_h = 0.4$ to 0.7,

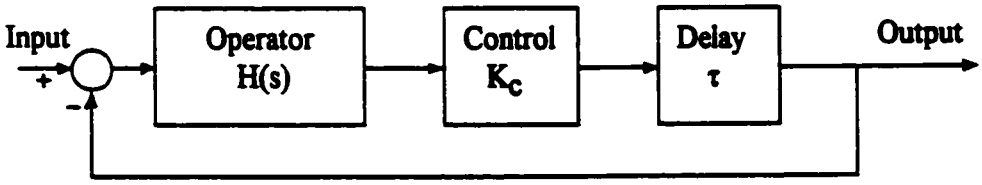
$d = r + \tau = 0.3 + \tau$

and $T_N = 0.45$.

K_c was the transfer function for manual control, assumed (by Leslie) to be constant over all frequencies. By plotting the response and comparing it with experimental results for the case of zero delay τ, parameter values were determined that yielded the best fit. These values for K, d and T_N are shown above. For this zero delay condition, a plot of the frequency response showed a cutoff frequency of 0.53 Hz (3.33 r/s). Below this frequency, tracking performance is theoretically perfect and falls at a rate of 6 dB/octave above. This compares well with the crossover frequency (discussed in section 2.4.2) model for human operators in vehicle-driving or similar tasks where crossover frequency values of 4.75 r/s (0.75 Hz)[70], 3.5 r/s [71] and 4 r/s (0.64 Hz) [72] have been reported. It should be noted that these model results did not approximate the experimentally determined cutoff frequencies as closely as hoped, but the general shape and characteristics were considered (by Leslie) to be good.



(a)



(b)

Figure 2.1. Pursuit-tracking block diagram: (a) open- and closed-loop models, and (b) simplified model (After Leslie [62]).

Earlier studies of human operators in control systems [63] found values for parameters to be:

$$r = 0.1 \text{ to } 0.3$$
$$T_N = 0.1 \text{ to } 0.6$$

with $T_N = \sim 0.1$ being typical. The ranges of values of these two parameters span the values found in many more recent studies [28,62,64-66,72-78]. Reaction delays r tend to range from 0.1 second [66] to 0.3 seconds [62,72] with the majority in the range of 0.15 – 0.25 seconds [1,46,64,65,73-77].

2.4.2 Crossover Model

Experimental studies of human operators in control systems have established a fundamental characteristic response referred to as the "crossover model " [70-72,79]. This model is based on single-input, single-output tracking , on a continuous basis, of random or quasi-random inputs. For this situation, a human operator tends to adopt a response $Y_h(j\omega)$ such that

$$Y_h(j\omega).Y_p(j\omega) \approx \frac{\omega_c e^{-j\tau\omega}}{j\omega} \tag{2.5}$$

where Yp(jω) is the response of the plant, ω_c is the crossover frequency, τ represents the total time delay in the human information processing system, including muscular response. This model has been found to be valid over a frequency range of 1 to1.5 decades around the crossover frequency. This model is considered to be fundamental for human operators and is often used in studies [66,77,80-82].

2.4.3 Precognitive Control and Preview

McRuer [70] has developed a structural model hypothesizing a number of pathways in the human perceptual/neuromuscular system to account for different types of control used by operators. These types include compensatory, pursuit and precognitive control modes. Depending on which pathway is used, the control can be open-loop, combination open-loop/closed-loop or totally closed-loop to visual stimuli. Proprioceptive feedback is also included from other sensory sources such as joint receptors. This is shown in Figure 2.2. Closed-loop compensatory control is concerned with response to errors or plant output only. When system inputs can be distinguished from system outputs via the display, the open-loop pursuit control pathway is active. Under conditions of total familiarity with plant dynamics and the environment, a skilled operator can generate neuromuscular commands which are appropriately timed, scaled and sequenced to generate the plant output exactly as desired. This open-loop behaviour is called precognitive control. Precognitive control is often combined with compensatory control in a dual-mode type of control where the precognitive control component initiates the (appropriate) control action and the compensatory control component completes the control action by minimizing the error

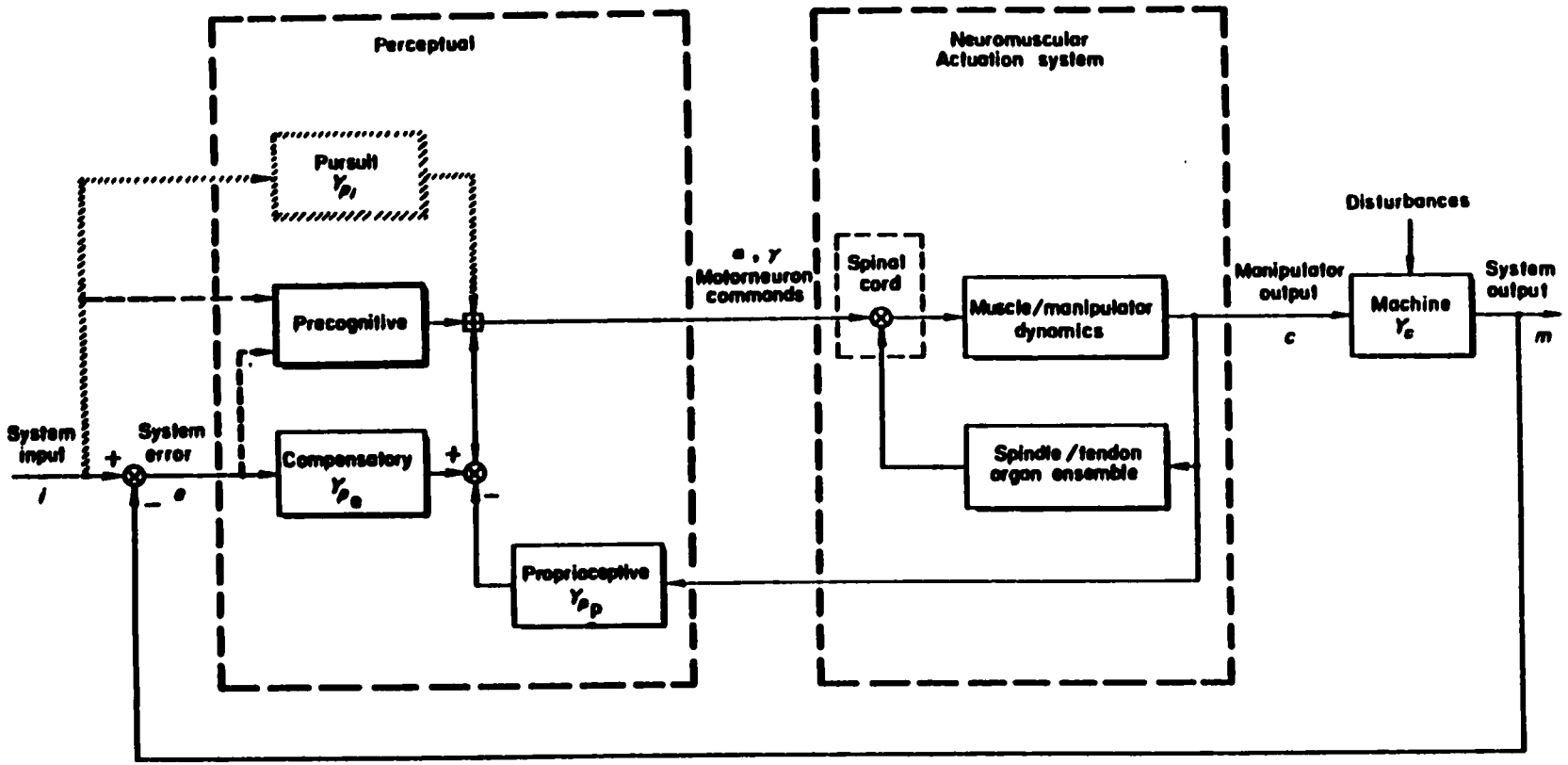


Figure 2.2. Major human operator pathways in a man-machine system (After McRuer[70]).

and maintaining control. Driving a motor vehicle is an example in which all three of the control pathways are present [70]. This accounts for a skilled operator's ability to make use of preview of the road ahead and knowledge of the vehicle's output response in generating feedforward control [71]. Aside from the initial delayed response to an input, this feedforward control can effectively compensate for the operator's reaction delay.

Tomizuka and Fujimora [65] studied "extended signal quickening" (including derivatives of output signal in the display) for manual control of pursuit-type tracking by incorporating modified future reference trajectory components into the displayed signal. Their model of the human operator was the familiar form

$$H(s) = \frac{Ke^{-0.15s}}{T_N s + 1} \tag{2.6}$$

where $K = 5$ to 10
and $T_N = 0.125$ to 0.25.

They determined that ~ 5 seconds of preview of the reference trajectory was sufficient to control a triple-integral plant having a step input. Without signal quickening, human operators were not able to control this plant using either compensatory-type or pursuit-type of displays. While extended signal quickening had its own merit, the effect of preview was dramatic. The preview essentially allows the operator to use precognitive control and effectively shift the output response ahead in time. It should be noted that signal quickening (or extended) is useful only when an explicit reference trajectory to be followed, including sufficient preview, is available. That is not exactly the case for the system described in this thesis; the reference trajectory is in the operator's mind.

Hess [79] studied models of human pilots using compensatory, pursuit and precognitive control. These models were basically extensions of the models developed by McRuer [70,71] (see Figure 2.2) and described typical response characteristics in the control of a second-order plant. He developed a structural model of an adaptive human pilot for compensatory-pursuit-precognitive behavior showing the various paths for each type of control. He considered the effect of preview by using a preview display as shown in Figure 2.3. This display shows the reference trajectory as it moves

from right to left. The objective in this case is to keep the small circular tracking symbol on the trajectory curve (ie. the input) by moving the symbol up and down vertically. If the preview and postview portions were not shown, the display would be a pure pursuit display. He indicates the total open-loop time delay of the human neuro-muscular sysem could theoretically be completely eliminated using preview. He reported that previous work has shown that significant improvement in tracking using preview over no preview occurred during the first 0.4 seconds, with marginal improvement after that.

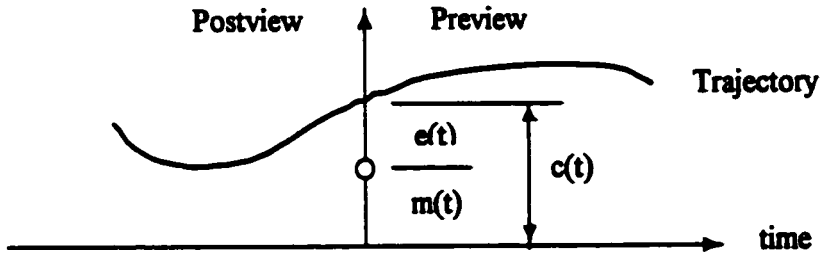


Figure 2.3. Schematic view of preview display (After Hess[79]).

Hess and McNally [77] studied human operators in multiloop manual control of systems such as helicopters. The multiloop model included an effective time delay of 0.3 seconds for the human operator. The model was used to test response to rate command, attitude command, velocity command and position command. A square wave input command signal (for each test) was displayed in preview fashion with the signal moving across the screen from right to left. Results indicated no apparent time lag between the fundamental components of the input and the response. This implies complete compensation of the human reaction time delay with preview.

Boer and Kenyon [83] studied a human operator tracking a 0.6 Hz bandlimited random signal with a time-varying gap between the tracking cursor and the initial appearance of the signal. The gap was varied between zero and 1 second using a signal with a bandwidth of 0.18 Hz. It was determined that human operators were capable of predicting the future fairly accurately up to about one-eighth of the period of the highest frequency component in the signal, or to about 0.2 seconds for this particular experiment.

2.4.4 Other Models

Other methods have been used to model human operators in control systems. These include optimal control models [5,73,80], structural models [5,79,84], auto-regressive moving average models (ARMA) [66,68,78,83] and fuzzy models [5,68]. These have not been considered in this thesis. Hess [5] has compared the crossover model, structural model and optimal control model using a particular example. Gain and frequency responses are all very similar, particularly in the region of the crossover frequency validating the various methods.

2.5 Summary

This chapter has presented results of research into the problems of time delay as it affects the remote control of robotic manipulators and the remote driving of vehicles over a time-delayed link A number of solutions have been presented. Remote control of manipulators is not directly applicable to the problem of remote driving but the same general problem of time delay is faced. While most solutions to the remote control of manipulators involve force and velocity feedback, many of the other solutions are also applicable to remote driving. These include the use of prediction, supervisory control and Smith control (which will be discussed in more detail in the next chapter). Each of these techniques offer their own unique advantages and are included in the system designed for this project.

The modeling of human operators in control systems has been studied by various researchers and has also been included in this chapter. Transfer function models have been emphasized since this is one of the most common approaches and also since a transfer function is a good representation of the fundamental response of a human operator in a control system. The neuromuscular response is very basic and can be modeled quite well. Of particular interest is the normal reaction delay, which can be compensated for if sufficient preview is provided. All indications suggest that the transfer function model of the human operator used in this system should be reasonable and that, given the (pre)view of the road ahead of the vehicle, there should be no reaction delay to compensate for. This assumes that the operator is making appropriate use of a predictor display.

3. COMPENSATION OF DELAY IN CONTROL SYSTEMS

3.1 Time Delay in Control Systems

The adverse effect of time delay on system stability is covered adequately in control system textbooks. Briefly, the problem is as follows. In the complex domain, a time delay of τ seconds is represented by the Laplace transform

$$G(s) = e^{-\tau s}.$$

The frequency response of this transfer function is determined by substituting $s = j\omega$. Then,

$$\begin{aligned} G(j\omega) &= e^{-j\omega\tau} \\ &= 1\angle -\omega\tau \ . \end{aligned} \tag{3.1}$$

The constant magnitude of 1 implies that a time delay will not affect the magnitude of any signal (at any frequency) passing through it. The phase term $-\omega\tau$ indicates a phase lag that increases linearly with frequency. Since this delayed system phase characteristic must be added to the non-delayed system phase characteristic, the result is a non-minimum phase system. This will affect the overall system phase and gain margins as shown in the Bode diagrams in Figure 3.1.

Both the gain and phase margins are observed to be reduced by the presence of delay. This means that, if delay is present, the overall system gain would have to be reduced to maintain the same amount of margin (ie. degree of stability) as there was without delay. By reducing the gain, the magnitude curve shifts to the left lowering the crossover frequency and reducing the bandwidth of the system. If the delay is sufficiently small that the resulting margins remain large enough for required system stability, no further action need be taken. Conventional delay-free control techniques (e.g. PID control) should be satisfactory. If the delay is large, the margins may become

too small or even negative indicating instability. Reducing system gain and/or system bandwidth may be a solution.

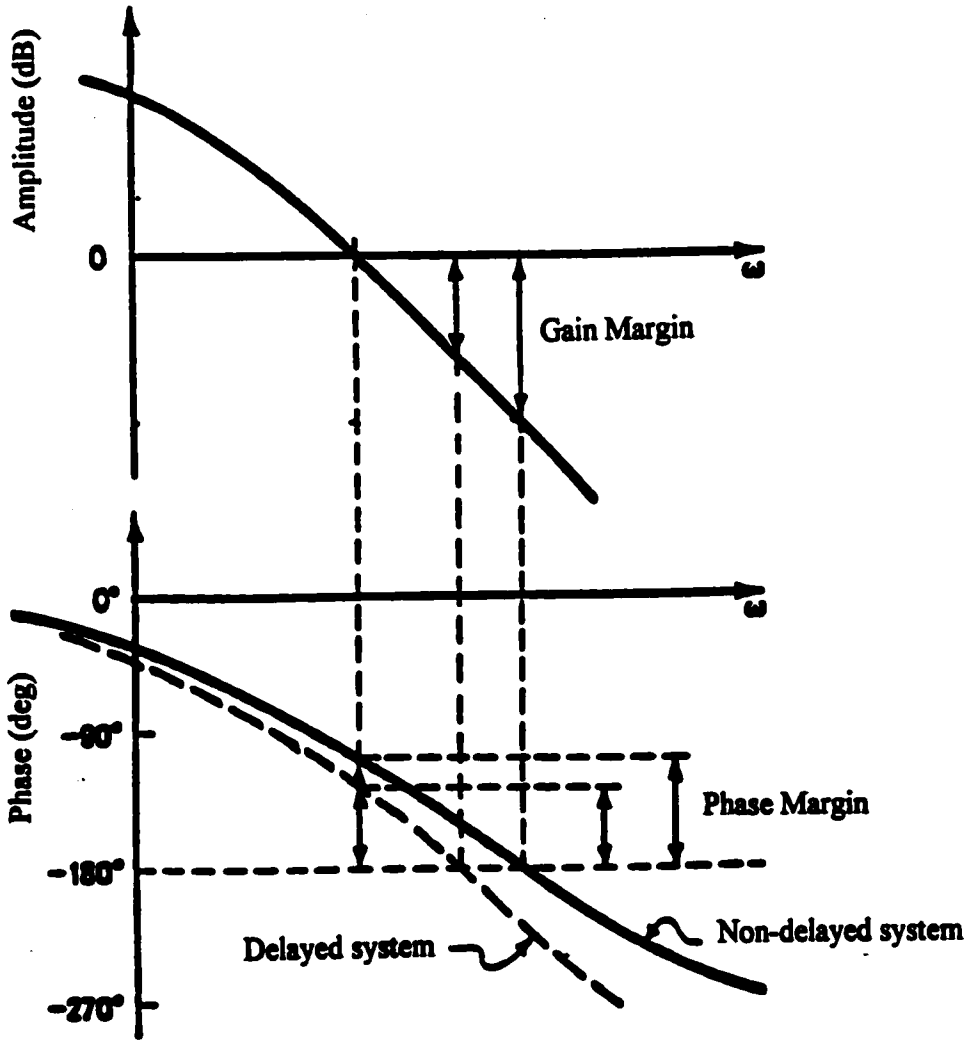


Figure 3.1. Bode diagrams of non-delayed and delayed systems.

Bahill [85] has examined the use of different approximations in a simple unity-feedback system containing a first-order plant and time delay in the forward path. For a plant with transfer function $G(s) = K/(Ts + 1)$, the closed-loop response is

$$\frac{C(s)}{R(s)} = \frac{Ke^{-s\tau}}{(Ts+1+Ke^{-s\tau})} \tag{3.2}$$

where K is the plant gain,

τ is the time delay

and T is the plant time constant.

The first type of approximation used was a Taylor series expansion,

$$e^{-s\tau} = 1 - s\tau + \frac{(s\tau)^2}{2!} - \frac{(s\tau)^3}{3!} + \ldots \; . \tag{3.3}$$

For small values of τ and $s\tau << 1$ for frequencies of interest, the system response becomes

$$\frac{C(s)}{R(s)} = \frac{Ke^{-s\tau}}{(T-K\tau)s+(K+1)} \; . \tag{3.4}$$

From the Routh-Hurwitz criterion this system is stable if the denominator coefficients are all positive. This single-pole approximation is then stable if

$$-1 < K < T/\tau. \tag{3.5}$$

Similarly, a two-pole approximation has been shown [85] to be stable if

$$0 < K < T/\tau. \tag{3.6}$$

The second type of approximation used was a Padé approximation,

$$e^{-s\tau} = \frac{1-s\tau/2}{1+s\tau/2} \; . \tag{3.7}$$

By inserting this into Equation (3.2) the resulting transfer function becomes

$$\frac{C(s)}{R(s)} = \frac{K(2-s\tau)}{\tau Ts^2 + (2T+\tau-K\tau)s + 2(K+1)} \; . \tag{3.8}$$

For positive coefficients in the denominator, the system is stable if

$$-1 < K < 1 + 2T/\tau. \tag{3.9}$$

Bahill [85] then provides the following stability criteria for Equation (3.2) based on the Nyquist Criterion:

$$\text{For } \tau << T: \quad -1 < K < [1 + (1.57T/\tau)^2]^{1/2}. \tag{3.10}$$

$$\text{For } \tau = T: \quad -1 < K < [1 + (2T/\tau)^2]^{1/2}. \tag{3.11}$$

$$\text{For } \tau >> T: \quad -1 < K < [1 + (\pi T/\tau)^2]^{1/2}. \tag{3.12}$$

From the stability criteria described in Equations (3.5),(3.6),(3.9),(3.10),(3.11) and (3.12) based on a variety of methods and approximations, Bahill [85] has indicated that large gains can be used in time-delay systems only if the time delay τ is small compared to the plant time constant T.

While lowered values of gain may be sufficient to maintain system stability, low gain may well result in unacceptable transient response behavior. A teleoperation system, for example, usually requires high gain for a slave manipulator to respond to the commands from a master controller in a timely manner. Other remote-control systems may similarly require high values of gain. In addition, the type of system being considered for remote control in this project will have relatively short time constant for good transient response. If the transmission delay between the local and remote terminals is significant (ie. in the order of seconds), as is presumed in this project, the restriction on gain based on the above criteria would undoubtedly result in unsatisfactory performance. In the extreme, a system may be rendered unworkable.

3.2 Smith Control in Remote-Control Systems

3.2.1 Time-Delay Compensation

In 1957, O.J. M. Smith [7] introduced a new technique for overcoming instability problems in process control systems experiencing time delays, or dead-time. The essence of his technique was to replace the conventional controller C(s) with a new

design C*(s) such that closed-loop control of the non-delayed process dynamics was accomplished. Given a process (or plant) having a time delay of τ seconds, Smith proposed his new controller C*(s) to be installed within the closed-loop with the time-delayed process $G(s)e^{-\tau s}$ as shown in Figure 3.2(a).

The effect of the Smith controller C*(s) is to remove the time delay from within the loop as shown in Figure 3.2(b) and to realize effective closed-loop control of the non-delayed process dynamics G(s) using a conventional controller C(s). It should be noted that the time delay may be shown at the input or output of the closed-loop.

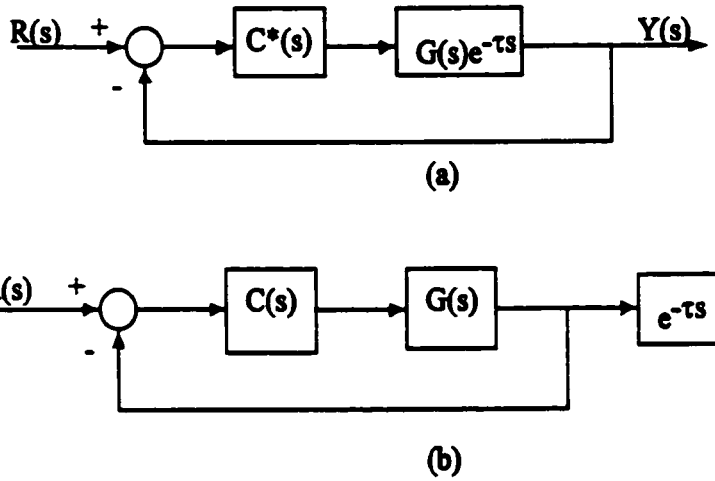


Figure 3.2. Smith control of time-delayed process: (a) Smith controller C*(s) in the loop, and (b) equivalent system with conventional controller C(s).

The same procedure can be applied to a remote-control system to accomplish stable, closed-loop control on an end-to-end basis. A basic remote-control system including a Smith controller can be represented as shown in Figure 3.3(a). The local and remote terminals are physically separated such that the feedforward control and feedback status signals each suffer a transmission delay of τ seconds. The intent of the controller is to effectively remove the time delays from within the loop as shown in Figure 3.3(b). Note that only the one-way delay is moved out of the loop. The time delay is shown at the input where it is more appropriate for a remote-control system. By equating these two systems [9], we obtain the transfer function

$$\frac{Y(s)}{R(s)} = \frac{C^*(s)G(s)e^{-\tau s}}{1+C^*(s)G(s)e^{-2\tau s}} = \frac{C(s)G(s)}{1+C(s)G(s)}e^{-\tau s} \qquad (3.13)$$

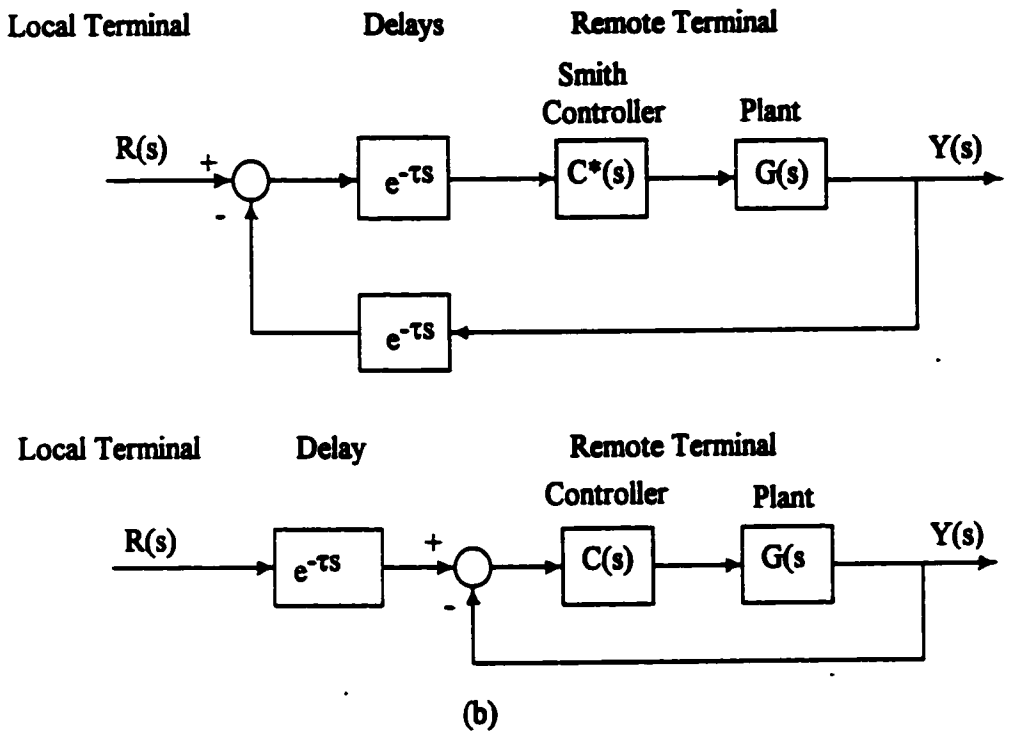


Figure 3.3. Smith control of remote-control system: (a) Smith controller C*(s) in the loop, and (b) equivalent system with conventional controller C(s).

so the Smith controller for remote control becomes

$$C^*(s) = \frac{C(s)}{1+C(s)G(s)(1-e^{-2\tau s})} \tag{3.14}$$

Substituting this controller into Figure 3.3 and rearranging Smith controller blocks, the system shown in Figure 3.4 is obtained. It must be noted that the time delay τ' and non-delayed dynamics G'(s) terms in the Smith controller of Figure 3.4 are estimates, since these parameters are imbedded in the process. This is the typical configuration for the Smith controller [9,10] except that in this remote-control case there is delay in both forward and reverse paths, so the delay block in the Smith controller contains a $2\tau'$ term for the total loop delay. It is quite critical in the usual applications of the Smith controller to provide accurate estimates τ' and G'(s) of the plant delay τ and the non-delayed dynamics G(s), respectively, to minimize mismatch problems [9,10,31]. In this application, the plant being controlled does not have any delay so we do not need to model G(s) for the Smith controller. We can take the output of the actual process

G(s) and reconfigure the delay block, $1-e^{-2\tau's}$. This results in the simplified Smith-controlled system shown in Figure 3.5 [11].

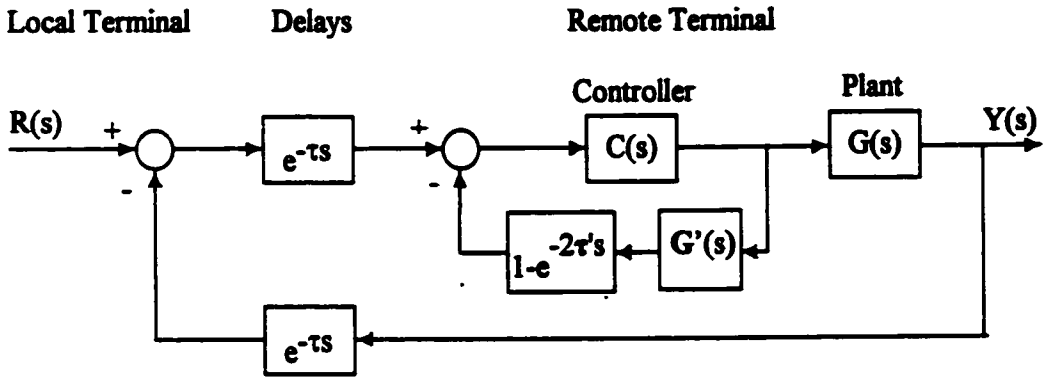


Figure 3.4. Remote-control system using a traditional Smith controller.

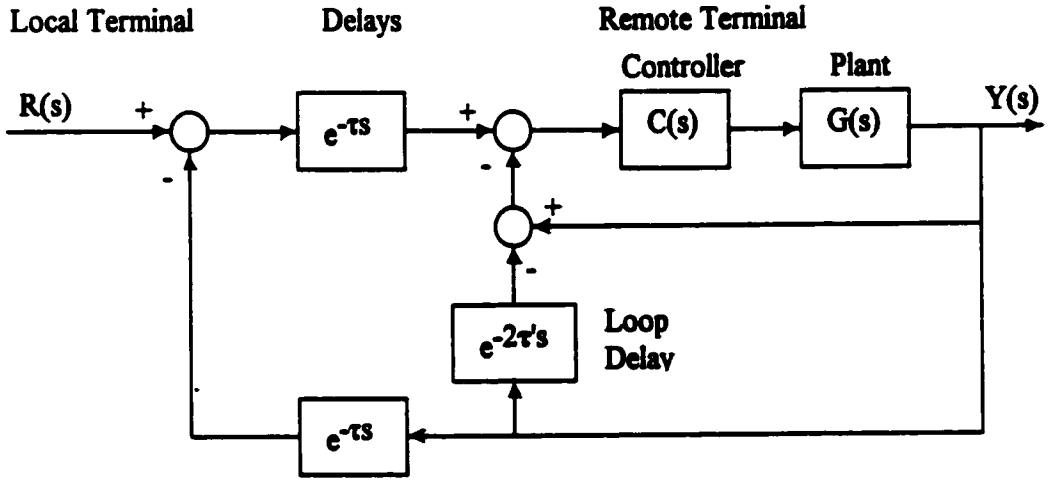


Figure 3.5. Remote-control system using simplified Smith control.

While this "Smith" controller does not require a model of the non-delayed plant, it still requires a good estimate $2\tau'$ of the loop delay 2τ. This delay essentially represents the sum of all transmission delays between the local and remote terminals. This loop delay can be measured accurately using any of a number of techniques within the communication system. Depending on the communication system used to carry the feedforward and feedback signals, it is probably prudent to measure the system loop delay on a continuous basis and update the simplified Smith controller accordingly.

Properties of Smith-controlled systems have been studied by various researchers over the years. Some of those results are included here.

System stability studies with Smith controllers have identified sensitivity to modeling errors as mentioned above. Since the plant parameters must be modeled very accurately for normal applications of Smith control (ie. control of plants with dead-time delay), adaptive control has been proposed to continuously adjust the estimated time delay and non-delayed dynamics [85]. It has also been established that Smith control cannot stabilize an unstable plant [86]. The robot vehicle plant in this project is stable.

For some types of plants, the performance of Smith-controlled systems can be somewhat sluggish and have poor load disturbance rejection. This has initiated a number of modifications to the basic structure [87-93] for improved performance. These have not been found to be necessary for this project.

From the point of view of robustness, it has been established that the nominal stability of a time-delayed system using Smith control is equivalent to that of the corresponding non-delayed system [94]. Robust stability is also the same for both delayed and non-delayed systems, based on guarantees for additive uncertainty, input multiplicative uncertainty and output multiplicative uncertainty [94]. Robustness has not been examined for the system proposed for this project.

Many of the concerns with Smith control identified and studied above are relevant only in the usual applications involving plants having dead-time delay, and where there is mismatch between actual system and estimated model parameters. The remote control application using the simplified Smith controller in this thesis only requires an estimate of the loop delay which can be measured accurately. The effects of mismatch will be examined in Chapter 5.

An interesting and thought-provoking statement relating Smith control with predictor displays and supervisory control was made in [12]: "In terms of pure control theory, predictive displays and 'continuous' teleprogramming could be likely seen as an implementation, in the frame of teleoperation, of the well-known Smith Predictor, proposed by Smith in 1957". Teleprogramming (or tele-programming) was discussed in Section 2.3.4 as part of supervisory control.

The operation of the Smith-controlled remote-control system can be described simply using the following time-domain explanation along with Figure 3.6. The error signal $e_1(t)$ is applied to the system at the local terminal and arrives at the remote terminal after the transmission delay τ. This delayed input error signal $e_1(t-\tau)$ becomes the input signal to the closed-loop system at the remote terminal. A negative feedback signal $s(t-\tau)$ provides the dual function of ensuring closed-loop control around the controller c(t) and plant g(t) and providing a delayed output which will be used to cancel the end-to-end closed-loop delay. This is apparent from the following analysis:

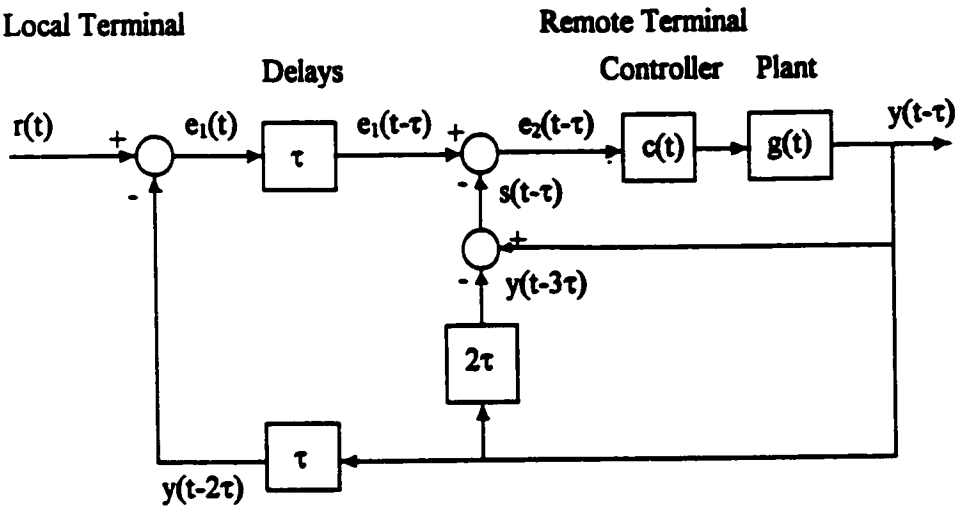


Figure 3.6. Time-domain signals in Smith-controlled remote-control system.

The input error transmitted to the remote terminal is:

$$e_1(t) = r(t) - y(t-2\tau) \tag{3.15}$$

Note that this includes an output feedback term which is delayed by two time delays (relative to r(t)). Arriving at the remote terminal, this signal becomes:

$$e_1(t-\tau) = r(t-\tau) - y(t-3\tau) \tag{3.16}$$

The error signal which is applied to the plant (and controller) is

$$e_2(t-\tau) = e_1(t-\tau) - s(t-\tau) \tag{3.17}$$

The feedback term is the Smith control signal

$$s(t-\tau) = y(t-\tau) - y(t-3\tau). \tag{3.18}$$

Substituting Equations (3.16) & (3.18) into (3.17), we get

$$e_2(t-\tau) = r(t-\tau) - y(t-3\tau) - y(t-\tau) + y(t-3\tau). \tag{3.19}$$

The two delayed output terms $y(t-3\tau)$ cancel, leaving

$$e_2(t-\tau) = r(t-\tau) - y(t-\tau). \tag{3.20}$$

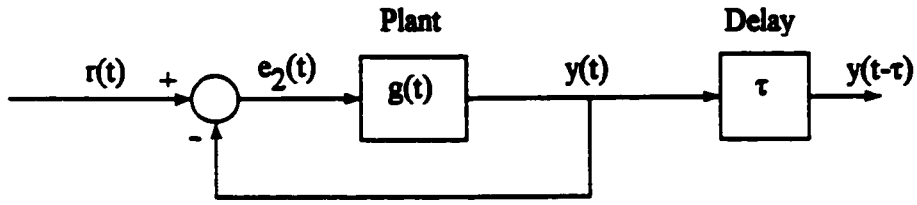
Since τ is a constant delay in all three terms, it is superfluous and can be removed, so

$$e_2(t) = r(t) - y(t). \tag{3.21}$$

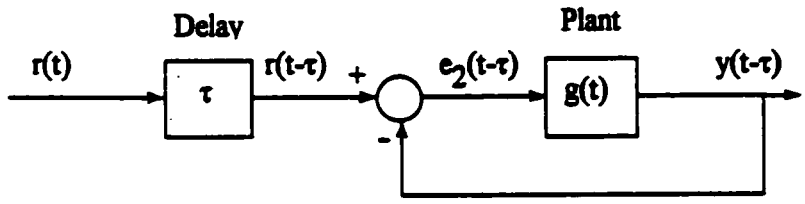
Equation (3.21) is the equation that would describe a non-delayed closed-loop system. The actual output $y(t-\tau)$ is of course delayed relative to the input so the overall equivalent system can be represented as shown in Figure 3.7(a). However, since the input r(t) to the closed-loop is actually delayed in a remote-control system, Figure 3.7(b) is a more accurate representation and results directly from Equation (3.20).

3.2.2 Human Operator Compensation

To account for the human operator in the system, the transfer function model given by Equation (2.3) has been used in the remote-control system described in this thesis. The effect is to add another transfer function block following the first error detector as shown in Figure 3.8(a). During the development of the system, it was found that the inclusion of the human operator dynamics caused totally unacceptable performance. This problem made it necessary to provide compensation. Since Smith control was already included in the design to compensate for the time delay, it seemed logical to use it for the human operator dynamics as well. Specific results of system response before and after Smith control compensation will be discussed in section 5 but it can be noted that resulting success of the system simulations proved that this concept is sound. The usual restrictions and sensitivities (ie. model accuracy) with

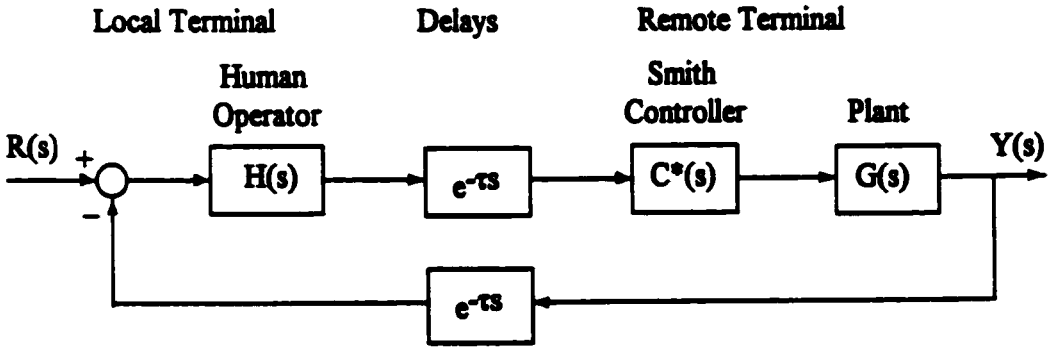


(a)

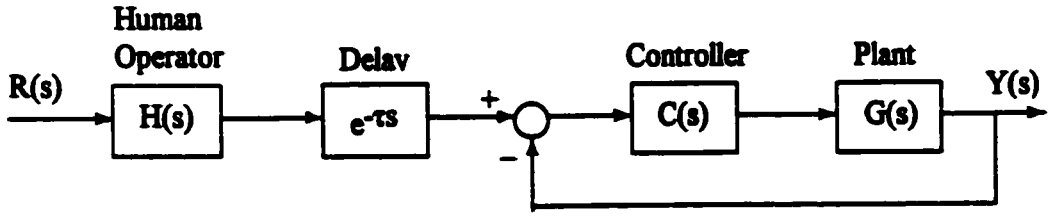


(b)

Figure 3.7. Smith-controlled equivalents of remote-control system: (a) time delay at system output, and (b) time delay at system input



(a)



(b)

Figure 3.8. Smith control of remote-control system including time delays and human operator dynamics: (a) block diagram with Smith controller, and (b) equivalent system.

Smith control still hold true. The same analysis as in section 3.4.1 is applicable and is presented here in brief.

If we equate these two configurations in Figure 3.8, the Smith controller will effectively remove not only the time delay but the human operator dynamics from within the closed loop as shown in Figure 3.8(b). Equating the two systems and simplifying gives the resulting Smith controller

$$C^*(s) = \frac{C(s)}{1 + C(s)G(s)(1 - H(s)e^{-2\tau s})} \quad . \tag{3.22}$$

The only difference from Equation (3.14) is the presence of the human operator dynamics H(s) in the denominator. Therefore, in conformance with Figure 3.5, the resulting system including the simplified Smith controller to compensate for time delay and human operator dynamics is as shown in Figure 3.9.

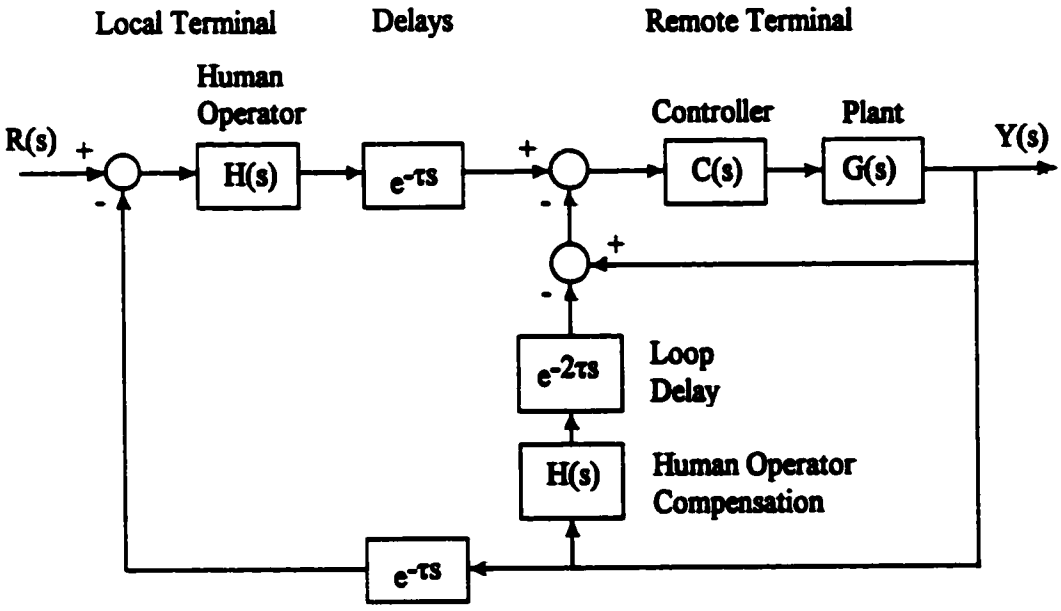


Figure 3.9. Remote-control system using simplified Smith control for both time-delay and human operator compensation.

3.3 System Equivalents Using Predictors

The purpose of a predictor is to provide the operator with an estimate of the output of the system as if there were no time delays. If we remove the time delays from the system shown in Figure 3.9, we are left with the non-delayed system shown in Figure 3.10. A predictor P_B modeling the non-delayed system that the operator would see comprises the portion of the system enclosed by the dotted lines. This would result in the closed-loop system shown in Figure 3.11. This predictor was developed using a neural network model as will be described in section 5.4 and the system shown in Figure 3 11 was tested. Technical difficulties in simulation using Matlab/Simulink were encountered so alternate but equivalent system configurations were investigated and developed.

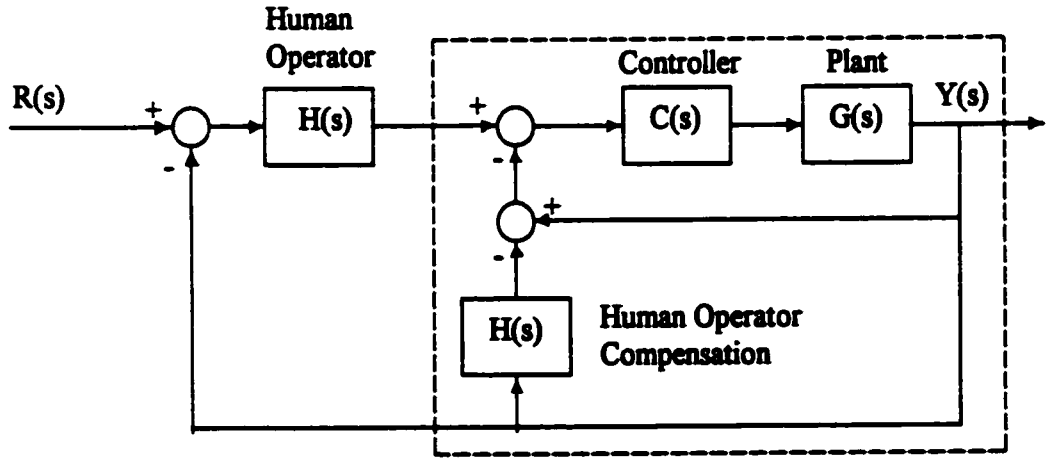


Figure 3.10. Remote-control system with no time delay, but using simplified Smith control to compensate for human operator dynamics.

By using the Smith control equivalent, we know that Fig. 3.10 can be shown as in Figure 3.8(b) without the time delay block (for the non-delayed case). In other words, we are using the simplified Smith controller to remove just the human operator dynamics out of the closed-loop in the same manner as it removed the time delay out of the loop in Figure 3.5. By generating a new predictor P_C to model the closed-loop portion of Figure 3.8(b), we have the new equivalent circuit shown in Figure 3.12. In principle, we are modeling different system dynamics with each predictor P_B and P_C, but the resulting systems are equivalent on an input-output basis.

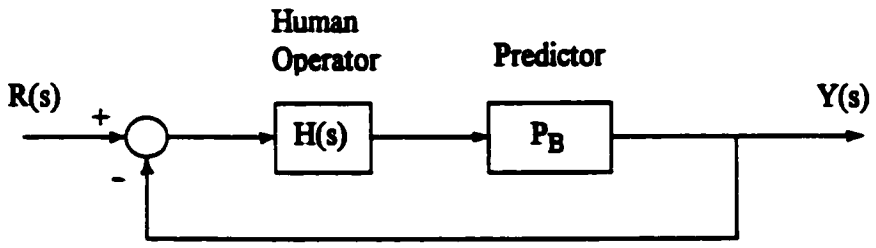


Figure 3.11. Equivalent system to Figure 3.10 using predictor P_B.

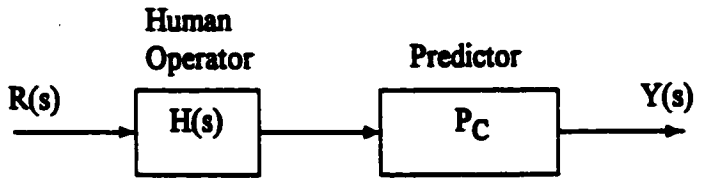


Figure 3.12. Equivalent system to Figure 3.10 based on Figure 3.8(b) with no time delay and using predictor P_C.

The system configuration that was originally envisioned for the use of the predictor by the operator in driving the robot vehicle is as shown in Figure 3.13 where H(s) represents the human operator's dynamics, C(s) represents the controller and G(s) represents the vehicle's dynamics. There is, of course, no need for Smith control in this configuration since it is open-loop on an end-to-end basis. The operator would issue command U(s) which would drive predictor P_A in closed-loop control as shown. The same command U(s) would also drive the robot vehicle once the command reached the remote terminal over the time delay. However, it has already been mentioned that technical difficulties were encountered when simulating this predictor loop. An examination of this system shows that, at steady-state, the error signal E(s) = 0.

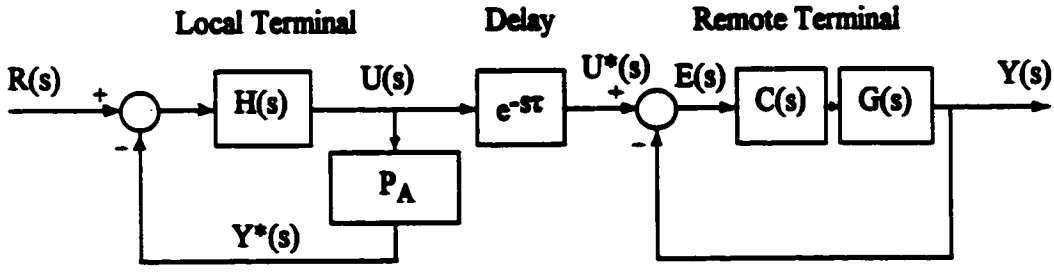


Figure 3.13. Configuration model for driving robot vehicle using predictor P_A.

Therefore,

$$U^*(s) = Y(s). \tag{3.23}$$

But, $U^*(s)$ is just the delayed command signal, or

$$U^*(s) = U(s)e^{-s\tau} \tag{3.24}$$

and we want the non-delayed signal U(s) to drive predictor P_A to give us $Y^*(s)$, the non-delayed (ie. predicted) estimate of Y(s), or

$$Y^*(s) = Y(s)\, e^{+s\tau}. \tag{3.25}$$

But, from Equation (3.24), U(s) is the non-delayed version of $U^*(s)$, or

$$U(s) = U^*(s)e^{+s\tau}. \tag{3.26}$$

Substituting Y(s) for $U^*(s)$ from Equation (3.23) and rearranging, we have

$$U(s) = Y(s)\, e^{+s\tau}. \tag{3.27}$$

Therefore, from Equations (3.27) and (3.25), at steady-state, U(s) is equal to $Y^*(s)$, the non-delayed estimate of Y(s), which is what is wanted from the predictor, so it would appear that the predictor is irrelevant. This situation did not seem logical, and there were problems with the simulation as mentioned, so the configuration was changed to that shown in Figure 3.14. The human operator still drives a predictor in closed-loop but the predictor is now P_B as in Figure 3.11 and the output of this predictor now drives the remote robot vehicle. The rationale for this configuration is as follows:

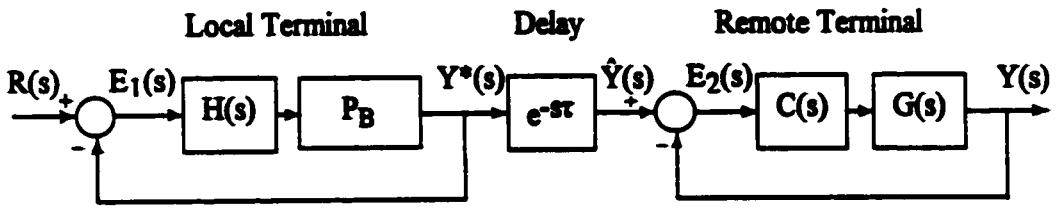


Figure 3.14. Revised configuration model for driving robot vehicle using predictor P_B.

At steady-state, the error signal $E_1(s) = 0$ and, therefore

$$Y^*(s) = R(s). \tag{3.28}$$

For this case, $Y^*(s)$ becomes the command signal transmitted to the remote terminal. The delayed command signal at the remote terminal then is

$$\hat{Y}(s) = Y^*(s)e^{-s\tau} \tag{3.29}$$

At steady-state, $E_2(s) = 0$ and the system output

$$Y(s) = \hat{Y}(s). \tag{3.30}$$

Therefore, by substituting Equation (3.30) into Equation (3.29) and solving for $Y^*(s)$, we get

$$Y^*(s) = Y(s)e^{+s\tau} \tag{3.31}$$

indicating that $Y^*(s)$ is a true predicted estimate of $Y(s)$. This cascade system should work in a manner that is virtually identical to the system shown in Figure 3.13 since the basic system time constant (ie. with no delays and no human operator in the system) is much shorter than that of the human operator's dynamics, which are controlling.

Just as with the model in Figure 3.12, problems were encountered with system simulations of this system. However, it can be seen that the closed-loop with predictor P_B in Figure 3.14 is the same structure as shown in Figure 3.11. This, in turn, is equivalent to Figure 3.12 with predictor P_C. Therefore, we can model yet another configuration using predictor P_C as shown in Figure 3.15. It should be noted that, while this configuration is open-loop on an end-to-end basis, this is an equivalent of the closed-loop system of Figure 3.14. Predictor P_C includes feedback. System tests using

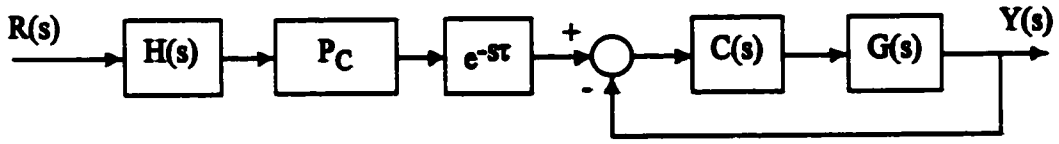


Figure 3.15. Final configuration model for driving robot vehicle using predictor P_C.

this model should therefore serve to verify the equivalent performance of the system model in Figure 3.14. The model in Figure 3.15 will be used for testing the system with the predictor in the loop in Section 5.4.3.

4. DESCRIPTION OF PROPOSED CONTROL SYSTEM

4.1 Introduction

The application which has guided the design and development of this project is the remote control, over a time-delayed link, of a robot vehicle which would operate in a tunnel environment constrained by the walls, general topography and environment of an underground mine. The primary function of this machine would be to traverse the mine, follow the tunnels, avoid obstacles and make appropriate turns at intersecting tunnels. These activities are to be automated to a high degree. Having a human operator in the loop to make every adjustment in speed and steering would result in a prohibitively slow process due to the long time delays in conveying system status information from the machine back to the operator and in conveying control command information from the operator to the machine. Nevertheless, teleoperation/telerobotic capability in real time is an important part of this system. A human operator must be able to control the forward and reverse motion and the steering of the machine over the time-delayed link. The extent to which direct teleoperation/telerobotics can be used is obviously a function of the time delay as indicated in Table 2.1. Therefore, the type of system described in this thesis should accommodate a total loop delay in the order of a few seconds (~10 seconds or less) for a machine travelling up to ~10 km/h (~3 m/s). This implies that the maximum distance allowed for the robot vehicle to travel autonomously is about 30 metres. By the time the operator receives status information from the remote site, the machine will already have travelled up to 15 metres from the location where the status was sent. The operator's resulting command is then based on the machine's speed, steering angle, position and heading at that previous location. Depending on the environment, a speed of ~3 m/s with a loop delay of ~10 seconds should be a reasonable objective. This delay should accommodate any remote robot vehicle operation on the earth using radio or optical communication links, including geostationary satellite links. It should also accommodate robot vehicle motion control on the moon from the earth, although lunar mining activities may require increasing the time delay beyond 10 seconds. Depending on the communication links, delays between Earth and shallow space can be quite long, reportedly as long as 18 seconds [49].

4.2 General

The generalized architecture for the remote control of a robot vehicle over a time-delayed link showing all major elements is shown in Figure 4.1. The human operator, using the console at the local terminal, controls the movement of the robot vehicle in its environment over the (time) delays via two major interface blocks, the local and remote teleautonomous controllers. The operator uses the system information provided by the console's display to plan, make decisions and issue commands. The operator's commands are used in the local teleautonomous controller (LTC) to generate a predicted response of the vehicle for the operator's assistance. These commands may also be transmitted as control commands to the remote site over a (time-delayed) communication link, depending on configuration. This link may be a complex communication system either dedicated to the remote-control system or shared with other users. For the purposes of this thesis, it is simply a transparent link conveying the control and status information between the two terminals at a satisfactory performance level. Its only degrading feature being considered for the remote-control system is its inherent propagation time delay. At the remote site, the delayed control commands are processed in the remote teleautonomous controller (RTC) which generates the necessary control signals to the robot vehicle's actuators. Closed-loop control is provided on the vehicle for stability purposes using on-board sensors. These sensors also return status information about the vehicle and its environment to the operator's display console.

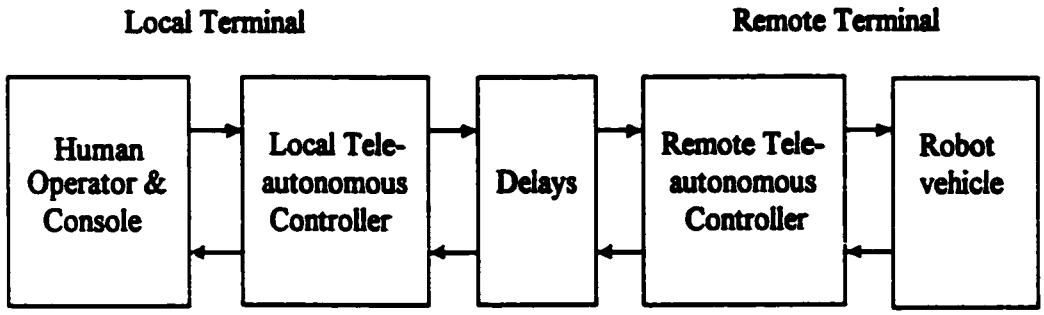


Figure 4.1. Time-delayed remote-control system block diagram

In addition to the continuous manual control by the operator in teleoperation mode, semi-autonomous modes are provided using supervisory control. The supervisory control used in this project does not include sending goals and directives to

the remote site as normally inferred by the term supervisory control. In this case, it refers to sending appropriate commands to open or close switches at various locations in the control system. These switches enable and disable the particular semi-autonomous mode as required. The two semi-autonomous modes provided with this system are tunnel-tracking and intersection-turning. In tunnel-tracking mode, the robot vehicle simply proceeds down the tunnel independently, avoiding obstacles as best it can. In the event of major difficulty (e.g. unable to avoid an obstacle), the machine will stop. The operator will then intercede to take the necessary action.

In either semi-autonomous mode, the overall end-to-end loop is retained but the operator is not required to drive the system. Closed-loop feedback control is provided by closing low-level loops at the remote terminal in real time. This avoids any instability problems with the vehicle due to time delay. This was a concern of one of the original partners involved in the definition of the system described in this thesis. For long loop delays, supervisory control is required to allow the remote vehicle to operate semi-autonomously while retaining control from the local terminal. This is accommodated by the multivariable nature of the control system. The manual control loops for both speed and steering are never disabled although the operator can set the speed to a fixed value and remain essentially "hands off" by leaving the steering control device (e.g. joystick) at zero angle. The steering would be controlled automatically to either follow the tunnel or turn into an intersecting tunnel. Likewise, steering would automatically avoid obstacles. In order to avoid excessive speed during steering maneuvers (either manual or automatic), the system is designed so that the speed is dependent on the steering angle. Beyond a narrow range around the zero steering angle, the speed will automatically be reduced proportional to the angle of the wheels.

4.3 Local Terminal

Teleoperation and supervisory control commands are generated by the operator via the console controls and are multiplexed together in the local teleautonomous controller (LTC) for transmission as a control command to the remote terminal over the communication link. This is shown in Figure 4.2. The most important component in the LTC is the predictor. In teleoperation mode, the operator's command is fed to the predictor which generates a (predicted) system output signal. This signal is made available to the console display showing the expected output of the system (ie. speed and steering angle of the robot vehicle) as if no delay were present in the system. The

operator uses this closed-loop control information to provide appropriate commands to the remote site by effectively driving the predicted system in real time. Since the predictor is considered to be a valid representation of the subsequent system without delays (ie. RTC and robot vehicle), the robot vehicle's actual motion should follow the predicted motion τ seconds later, where τ is the one-way time delay. This has been discussed in Chapters 1 and 3.

Since the predictor is a model of the RTC and robot vehicle in a pristine environment, any real-world anomaly (e.g. obstacle, wheel slip) that the actual robot vehicle experiences cannot be predicted. Therefore, there will often be an error between the predicted and actual robot vehicle's position and heading once the status feedback is received from the remote terminal. This feedback information would be used to update the actual robot vehicle's position and heading on the console display.

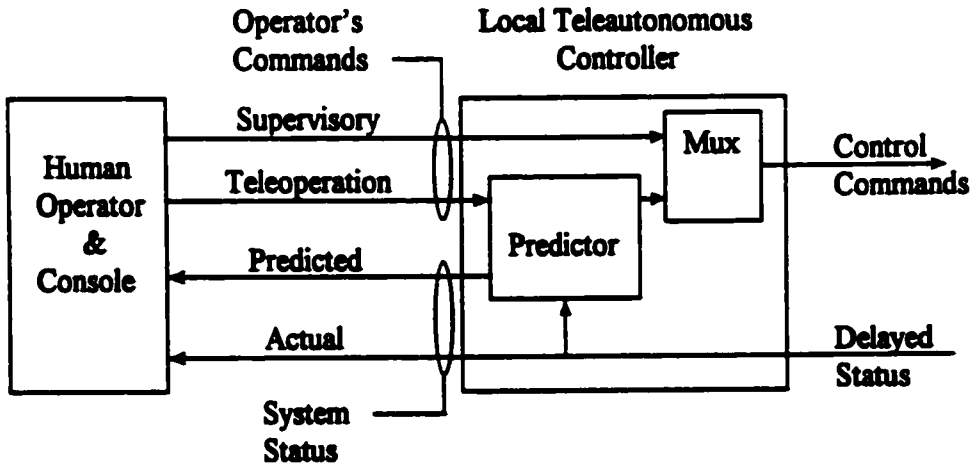


Figure 4.2. Local terminal block diagram

4.4 Remote Terminal

4.4.1 General

The heart of the remote terminal, and indeed of the entire system is the remote teleautonomous controller (RTC) shown in Figure 4.3. The real-time control of the robot vehicle in its environment is performed by this subsystem. Incoming control commands from the local terminal are demultiplexed to separate the supervisory

control component from the teleoperation control. The latter is further separated into its basic constituents - speed and steering commands. Both of these commands are processed using simplified Smith controllers to compensate for the time delay and for the human operator dynamics before being fed to the respective speed and steering control stages.

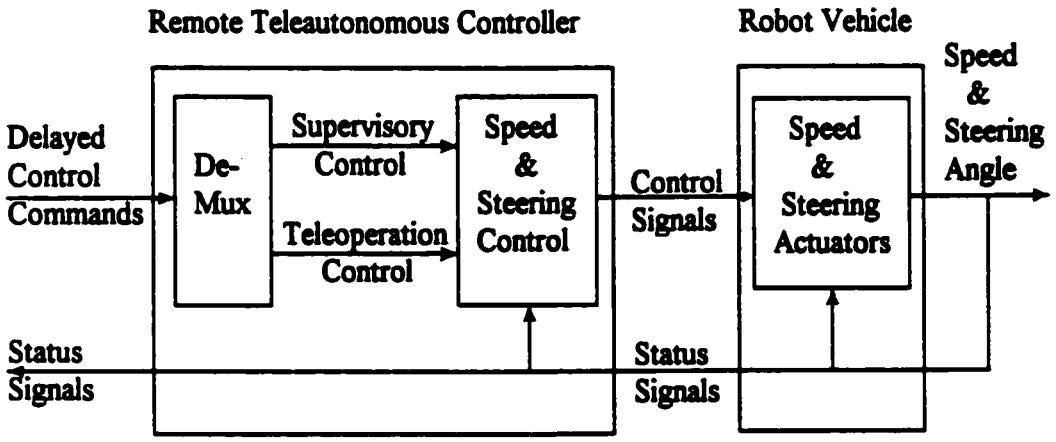


Figure 4.3. Remote terminal block diagram

The main subsystem of the RTC is the speed and steering control block. This subsystem is further comprised of three main subsystems: obstacle-avoidance, RTC speed control and RTC steering control as shown in Figure 4.4. Each of these three subsystems are in turn comprised of another layer of subsystems which are based on fuzzy control. The speed and steering control signals generated in the RTC are used to drive the speed and steering actuators on the robot vehicle. The intersection control, obstacle detection and tracking blocks, while shown as part of the RTC, are assumed to be part of a video/sonar sensor system considered to be outside of the scope of this thesis. The signals provided by that system are the angle and range to intersection, angle and range to obstacles and tunnel-tracking error as shown in Figure 4.4.

A description of a control system should start with its functions. This system supports two major functions – teleoperation (including telerobotics) and supervisory control. The teleoperation requirement is straightforward. Teleoperation over time delay is a fundamental objective of this thesis. The human operator must have the capability of manually intervening in the control of the vehicle at all times, even if that control is delayed. The operator's (delayed) control commands shown in Figure 4.4 are

actually error signals based on the difference between the desired reference input and the delayed system output fed back to the local terminal. Actual output speed and output steering angle from the robot vehicle are also required for proper functioning of the obstacle avoid, speed control and steering control subsystems.

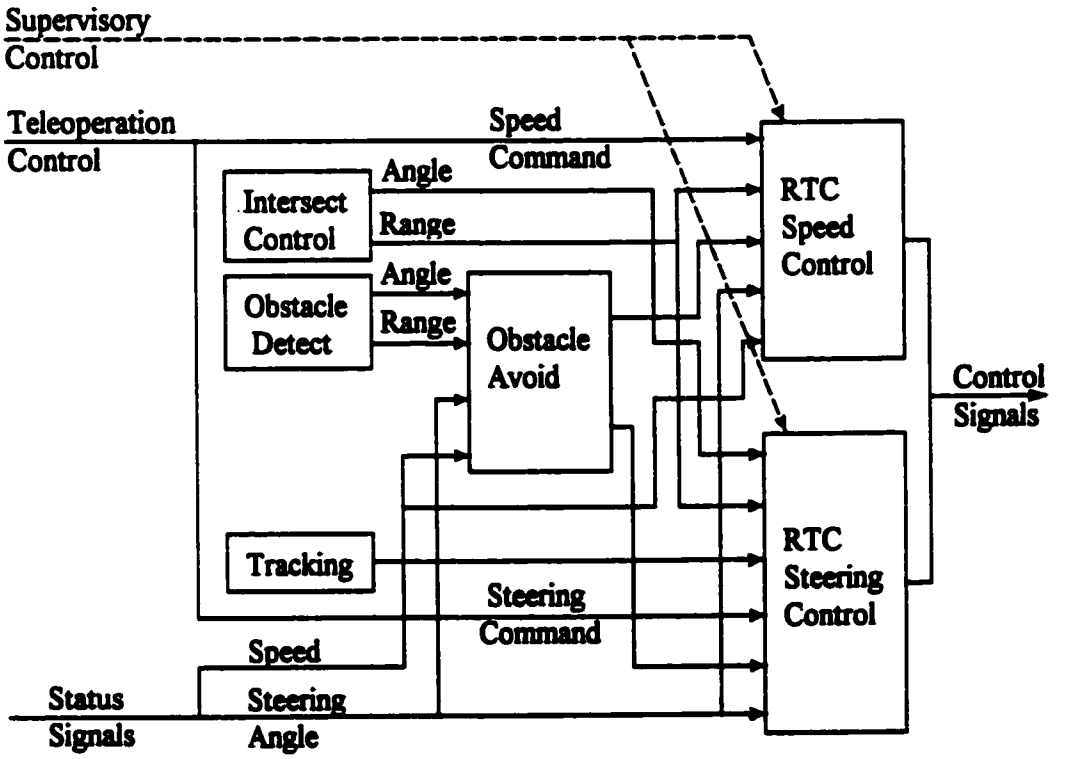


Figure 4.4. Speed and steering control block diagram

Supervisory control is provided to accommodate semi-autonomous operation mode. Video and sonar sensors on the vehicle provide information about the location of tunnel intersections as well as the range and angle relative to the vehicle's position and heading. This is necessary to accommodate automatic turning at intersections. Another function requiring video and sonar sensor information is tracking. A basic requirement for semi-autonomous operation is for the robot vehicle to be able to drive down a tunnel on its own. This requires a good estimate of the centreline of the tunnel. This should be an another excellent application for fuzzy control, given the complex nature of the tunnel environment. Accuracy of the centreline location is not of critical importance, and linguistic descriptors used in fuzzy control should easily suffice. Video

and sonar information is also required for obstacle detection and avoidance. This is vital for a vehicle being remotely-controlled over a time delay, regardless of whether the control is semi-autonomous/supervisory or manual/teleoperation. The vehicle must be able to respond to hazardous situations in a timely manner and take appropriate corrective action. The signals required to accommodate these various features as shown in Figure 4.4 are: angle to intersection, range to intersection, angle to obstacle, range to obstacle and tracking error.

Video and sonar sensor fusion/processing is a major research and development area and is not addressed in this project. It is assumed that the range and angle to intersections and obstacles, possibly based on fuzzy logic, will be available. It is similarly assumed that the tracking error in the tunnel, also possibly based on fuzzy logic, will be made available when required.

4.4.2 Obstacle-Avoidance Using Fuzzy Logic

Range and angle to obstacle is combined with the speed and steering angle of the vehicle to determine an "obstacle factor" and an "obstacle avoid" signal as shown in Figure 4.5. The obstacle factor is a measure of the likelihood of collision and is used by the speed control system to reduce the vehicle's speed as necessary. The obstacle avoid signal is an adjustment to the steering control system proportional to the likelihood of collision. The obstacle factor and obstacle avoid signals are used by the speed and steering control systems to safely navigate around hazards or to simply stop the vehicle if there is no clear path available. The decision-making process to determine these two factors is based on fuzzy logic. This was done primarily for two reasons: linguistic variables seem more appropriate in describing the control parameters than conventional "crisp" values, and it seems appropriate to use linguistic rules in determining the extent or significance of the resulting hazard. To say that the range to an obstacle, for example, is "close" or "far" means more (to a human) than, say, 14.23 metres or 56.18 metres, respectively. It is also easier to establish a control scheme, since the designer can say, for example, "If the range to obstacle is 'close', and the angle to obstacle is 'small right', then the hazard situation is 'high'". The designer also has control over the shape and distribution of the membership functions to enable shaping of the output control surface. The control surface represents the entire input-output space in three dimensions showing the output value for all possible combinations of the two inputs over their full universes of discourse.

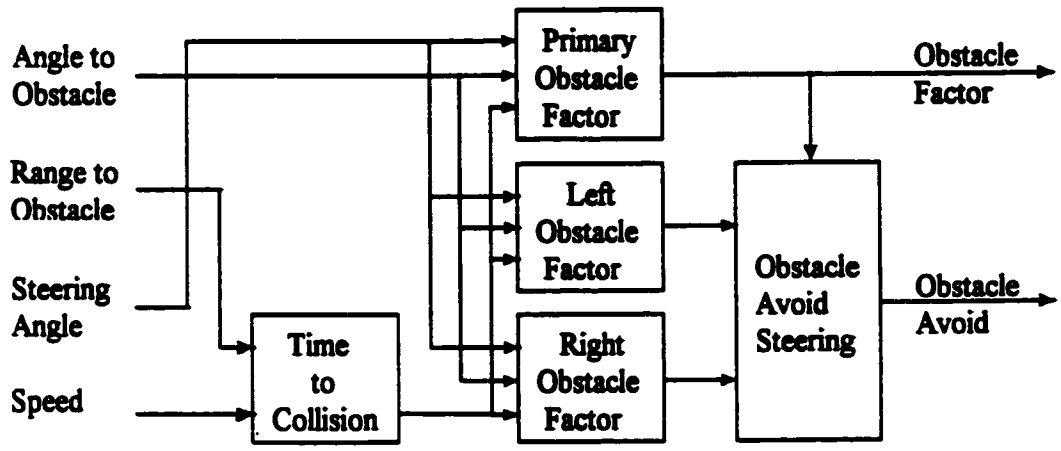


Figure 4.5 Obstacle avoidance block diagram

All obstacles in the general forward direction of the vehicle will be detected and tracked. range and angle to the obstacle will be measured relative to the vehicle's heading and represented as fuzzy (linguistic) variables. The range information will be used to determine the time to collision using the vehicle's present speed. Time to collision is a better choice of variable than using the range to obstacle since it also includes the vehicle's speed. Two fuzzy controllers are used to combine the steering angle, angle to obstacle and time to collision as shown in Tables 4.1 and 4.2 to produce the obstacle factor. Since there are three input variables, two 2-input fuzzy controllers are used. There are two reasons for this: firstly, Matlab's fuzzy logic simulation toolbox can only accommodate 2-input fuzzy controllers; secondly, it is more efficient. This can be explained as follows: If we assume that each of the three fuzzy variables is represented by five membership functions, there are clearly $3^5 = 125$ combinations available. This means 125 rules in the rule base for a 3-input fuzzy controller. A 2-input fuzzy controller would require $2^5 = 25$ rules. The output of this controller is then combined with the third remaining input variable. Assuming that the output from the first controller is also described using five membership functions, we only require a second controller with 25 more rules. The total number of rules using two 2-input fuzzy controllers is then 50, one-third as many as the 3-input controller. Table 4.1 is the fuzzy associative matrix (FAM) relating the steering angle of the wheels and the angle to obstacle to an output variable referred to as the "hazard situation". This is an indication of the likelihood of steering into the obstacle strictly on the basis of the angles. Table 4.2 is the FAM relating this hazard situation and the time to collision to the obstacle factor. This is an indication of the likelihood of actually hitting the obstacle by including speed and distance.

The obstacle factor varies from safe to caution, warning and danger. For example, let us assume that the robot vehicle is presently turning right by a small amount. In fuzzy control terminology, we can call this steering angle "small right". If an obstacle is detected which is somewhat to the left of our present heading, this angle to obstacle can be called "close left". Using Table 4.1, we find that the intersection of these two angles is in the third column from the right and third row from the top. This results in a "small left" hazard situation. This is represented in Table 4.2 by the second column from the left. This tells us that the obstacle factor will be classified as "caution" if the time to collision is "very short" and will be considered "safe" for any longer times to collision. This second fuzzy controller could have been simplified and given a shorter rule base by eliminating twelve combinations on the basis of symmetry. This would require the hazard situation descriptors to simply remove any reference to left or right. This was considered a fairly low priority and was not done.

The membership functions for each input and output variable are shown in Figure 4.6. Seven membership functions were selected based on a human operator's fuzzy description of angles, such as: "dead ahead", "medium left" and "hard right". Five symmetrical triangular functions uniformly spaced across the [-1 1] universe of discourse plus a bisected triangle at each end are used to define the steering angle and angle to obstacle. Seven membership functions also define the output hazard situation, since this parameter can be well described by terms such as: "high", "medium", "low" and "zero". The symmetry was discussed in the previous paragraph. Four of these are symmetrical triangles and three are trapezoids. The functions at each end are bisected trapezoids with small shoulders to indicate the fact that there is little change at the ends [95]. Similarly, a trapezoidal function was used at the centre of the universe of discourse to minimize the effect of small variations. This region represents the "high" hazard situation so it is prudent to maintain an extra degree of safety for the system. Since the control surface in Figure 4.8 was very smooth over much of it, no tuning was felt necessary. The same membership functions as used for the hazard situation output are used for the hazard situation input for the second fuzzy controller as shown in Figure 4.7. The final input, time to collision, is defined by two triangular and two bisected trapezoidal membership functions. The non-symmetry over the universe of discourse is an attempt to provide higher priority for shorter times to collision. The obstacle factor output membership functions were tuned to improve the shape and smoothness of the control surface shown in Figure 4.8

Table 4.1 Fuzzy associative matrix (FAM) for hazard situation

	Steering Angle						
Angle to Obstacle	Hard Left	Med Left	Small Left	Zero	Small Right	Med Right	Hard Right
Far Left	High	Med Left	Small Left	Zero Left	Zero Left	Zero Left	Zero Left
Medium Left	Med Right	High	Med Left	Small Left	Zero Left	Zero Left	Zero Left
Close Left	Small Right	Med Right	High	Med Left	Small Left	Zero Left	Zero Left
Dead Ahead	Zero Right	Small Right	Med Right	High	Med Left	Small Left	Zero Left
Close Right	Zero Right	Zero Right	Small Right	Med Right	High	Med Left	Small Left
Medium Right	Zero Right	Zero Right	Zero Right	Small Right	Med Right	High	Med Left
Far Right	Zero Right	Zero Right	Zero Right	Zero Right	Small Right	Med Right	High

Table 4.2 Fuzzy associative matrix (FAM) for obstacle factor

	Hazard Situation						
Time to Collision	Zero Left	Small Left	Medium Left	High	Medium Right	Small Right	Zero Right
VShort	Safe	Caution	Warning	Danger	Warning	Caution	Safe
Short	Safe	Safe	Caution	Warning	Caution	Safe	Safe
Medium	Safe	Safe	Safe	Caution	Safe	Safe	Safe
Long	Safe	Safe	Safe	Safe	Safe	Safe	Safe

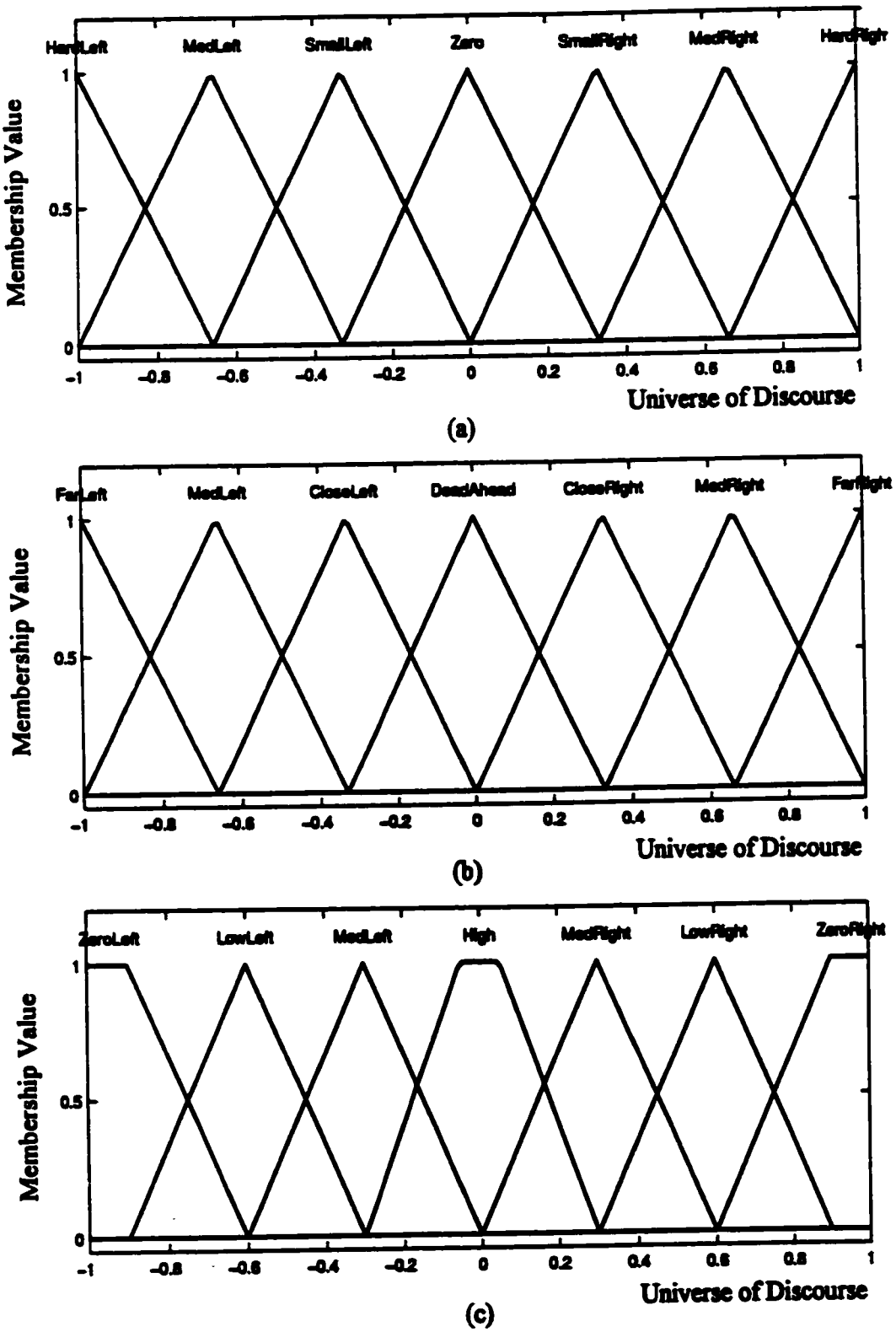


Figure 4.6. Membership functions for hazard situation fuzzy controller: (a) steering angle input, (b) angle to obstacle input, and (c) hazard situation output.

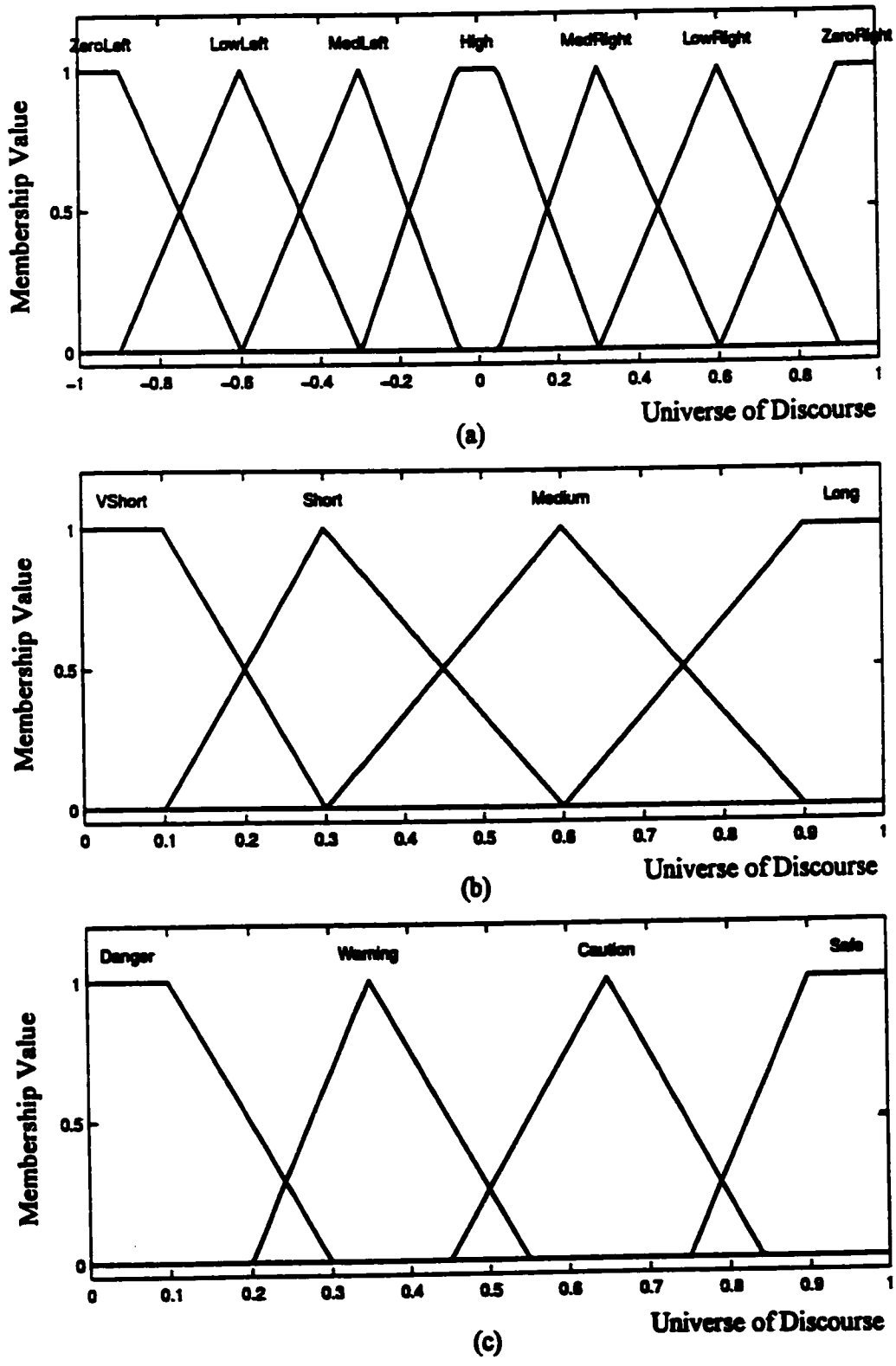
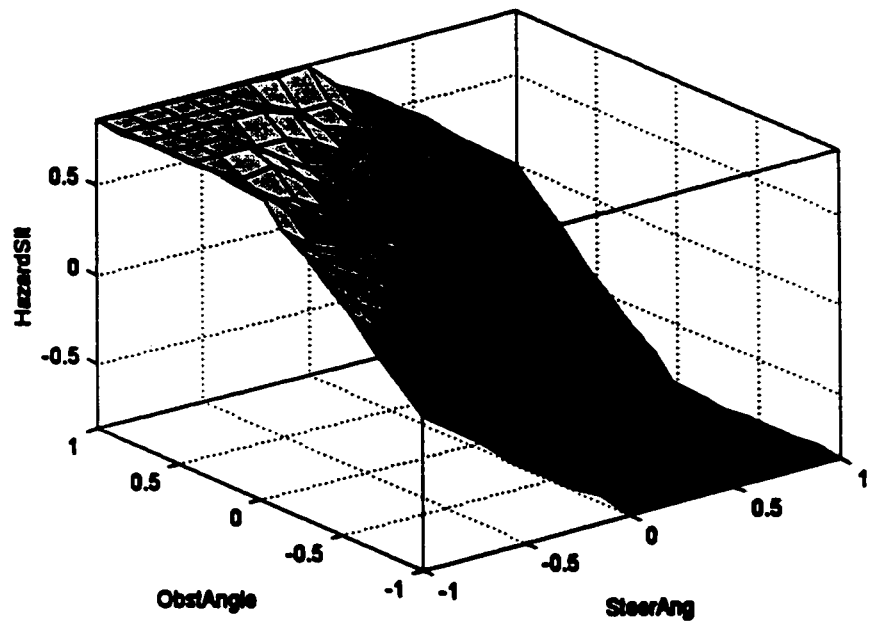
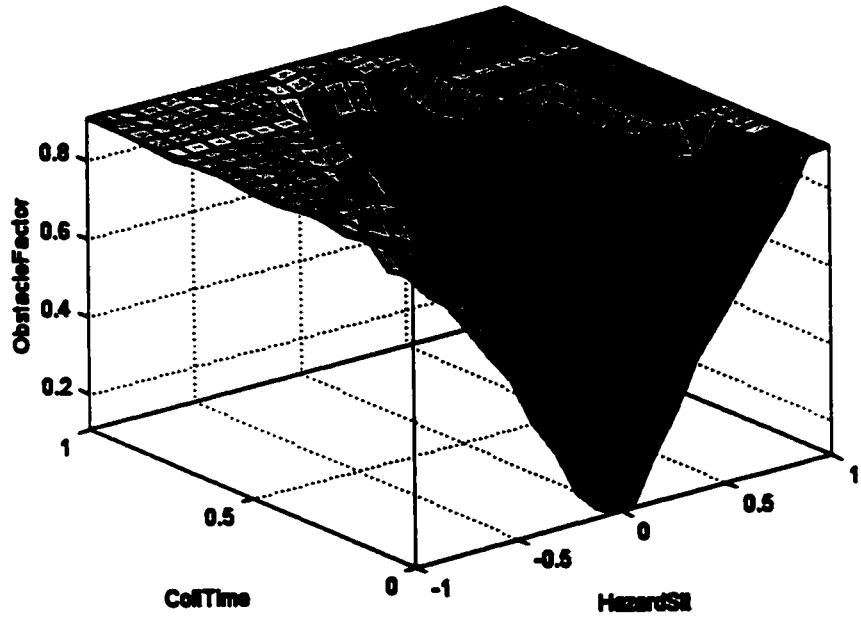


Figure 4.7. Membership functions for obstacle factor fuzzy controller: (a) hazard situation input, (b) time to collision input, and (c) obstacle factor output.



(a)



(b)

Figure 4.8. Control surfaces for obstacle avoidance subsystem: (a) hazard situation as a function of steering angle and angle to obstacle, and (b) obstacle factor as a function of hazard situation and time to collision.

The control surfaces for both the hazard situation as a function of angle to obstacle and steering angle and for the obstacle factor as a function of time to collision and the hazard situation are shown in Figure 4.8. The obstacle factor classified as the "danger" condition is represented by the small region when both the fuzzy hazard situation and fuzzy time to collision inputs are zero.

The obstacle factor has an immediate effect on the robot vehicle's speed, reducing it to a safe level if necessary. Steering around obstacles is more involved. The obstacle factor is fundamentally associated with the robot vehicle's steering angle. Regardless of the number and location of obstacles, the vehicle's steering will be adjusted in the direction of improved obstacle factor wherever possible. This is done by first examining the obstacle factors relating to the steering angles directly adjacent to (ie. to the left and to the right of) the current angle of the wheels and then by making any appropriate steering adjustments. In an actual implementation of this system, obstacle factors would be determined for all obstacles in the forward path of the vehicle. The primary obstacle factor is the one having the lowest value (between 0 and 1) of all the obstacles in front of the vehicle and therefore representing the highest danger level. For the system block diagram and simulations done for this project, it is assumed that this has already been determined. Left and right obstacle factors are then calculated for steering angles 20% above and below the current steering angle (20% has been arbitrarily chosen to represent a reasonable increment from the current steering angle to look for a safe path around or by the obstacle). If the primary obstacle factor is larger (and safer) than each of the adjacent obstacle factors then no steering adjustment will be made. However, if either left or right obstacle factor is higher (and safer) than the primary obstacle factor, then the robot vehicle will be steered in that direction.

This obstacle-avoidance method is expected to be adequate for all but the most extreme situations. In such an event the obstacle factor would soon generate a "danger" condition and the vehicle would stop. For the overall semi-autonomous system, this is considered acceptable since the operator is always ultimately available.

The detailed Matlab/Simulink configuration used in the system simulations is included in Appendix A.

4.4.3 Speed Controller Using Fuzzy Control

The RTC speed control block in Figure 4.4 contains the simplified Smith control subsystem and the fuzzy speed control subsystem, as shown in Figure 4.9. The five inputs are speed command, range to intersection, obstacle factor, steering angle and speed. The speed command is the control signal sent from the local terminal. The other four inputs are generated on-site at the remote terminal. The range to intersection input is used to ensure that the speed is reduced when approaching, crossing and automatically turning at intersections. The obstacle factor input is used to slow or stop the robot vehicle in the presence of obstacles. The steering angle input provides for the speed to be reduced appropriately as the angle of the wheels increases to avoid any problem with skidding and to maintain good control during turns. The steering angle, obstacle factor and range to intersection inputs are all considered to have equal priority. Only when all three permit a "fast" condition (as defined in the fuzzy controllers) will the operator be able to activate full speed on the vehicle. Any other condition (medium, slow or very slow) will result in a proportional reduction in speed. For example, if the obstacle factor reaches the danger level, the speed control output goes to zero to stop the vehicle. The range to intersection, obstacle factor and steering angle are all used to restrict the speed of the vehicle in accordance with their individual requirements. They are all considered equal such that the particular input with the minimum value will determine the "speed factor". The speed control philosophy is that the human operator has full control over the speed of the vehicle via the speed command unless the speed factor requests a lower speed.

The fuzzy speed control block diagram in Figure 4.9 shows the simple method of selecting the minimum value of the range to intersection, obstacle factor and steering angle to determine the speed factor. This subsystem uses two 2-input fuzzy controllers, one to generate the speed factor error and the other to generate the speed error control signals. The inputs to the speed factor error block are speed factor and speed. The FAM is shown in Table 4.3. This block can be viewed as an error detector with speed being the positive input and speed factor being the negative. For example, if the actual speed of the vehicle is "fast" and the speed factor is also "fast" (ie. no reduction required), the speed factor error is "zero", indicating no change in speed is required. However, if the speed factor requires a reduced speed of "slow", the speed factor error is "NMed" (negative medium) telling the fuzzy speed

error control to slow down. Similarly, if actual speed was "slow" and speed factor was "fast", the speed factor error is "PMed" (positive medium) telling the fuzzy speed error control to speed up.

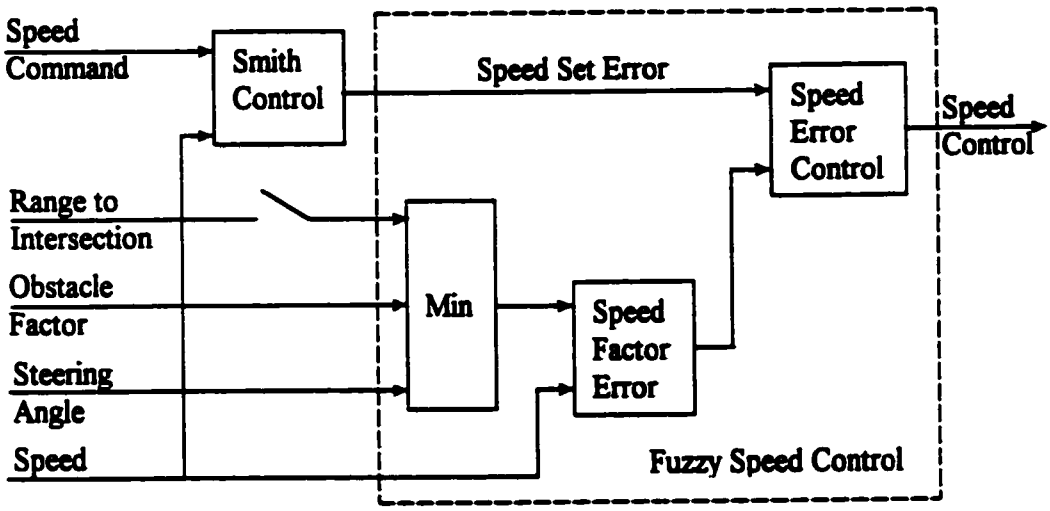


Figure 4.9. RTC speed control block diagram.

Table 4.3 Fuzzy associative matrix (FAM) for speed factor error

	Speed			
Speed Factor	VSlow	Slow	Med	Fast
Vslow	Zero	NSmall	NMed	NLar
Slow	PSmall	Zero	NSmall	NMed
Med	PMed	PSmall	Zero	NSmall
Fast	PLar	PMed	PSmall	Zero

The FAM for the fuzzy speed error control block is shown in Table 4.4. The inputs are speed set error and speed factor error. The speed set error is the speed command received from the local terminal. The speed set error is the difference between the speed set desired (or set) by the operator and the actual delayed speed fed

back from the remote terminal over the time-delayed link. The speed error control block is designed to output the minimum of either the speed set error or the speed factor error. This is to ensure that regardless of which input requests a lower speed, the speed will be lowered accordingly.

Table 4.4 Fuzzy associative matrix (FAM) for speed error control

Speed Set Error	Speed Factor Error						
	NLar	NMed	NSmall	Zero	PSmall	PMed	PLar
NLar	NLar	NLar	NLar	NLar	NLar	NLar	NLar
NMed	NLar	NMed	NMed	NMed	NMed	NMed	NMed
NSmall	NLar	NMed	NSmall	NSmall	NSmall	NSmall	NSmall
Zero	NLar	NMed	NSmall	Zero	Zero	Zero	Zero
PSmall	NLar	NMed	NSmall	Zero	PSmall	PSmall	PSmall
PMed	NLar	NMed	NSmall	Zero	PSmall	PMed	PMed
PLar	NLar	NMed	NSmall	Zero	PSmall	PMed	PLar

The membership functions for the fuzzy speed factor error block are shown in Figure 4.10. The speed factor and speed inputs have identical membership functions. Shouldered trapezoids are again used at the extreme ends of the universes of discourse to reflect the desire for minimum change in these regions [95]. Speed factor error output has five triangular and two shouldered trapezoidal membership functions. Narrower triangles are used toward the centre (zero) to increase the sensitivity in this region. The control surface is shown in Figure 4.11(a) reflecting is symmetrical nature. The membership functions for the speed error control block are all identical to the speed factor error membership function and are not shown. The final control surface in Fig. 4.11(b) shows the speed error control as a function of speed factor error and speed set error. Each of these inputs is seen to have an identical effect on the the output. If both inputs are at the same level to request a particular value of output, that value will be accepted; if either one is set to a lower level, then that lower value of

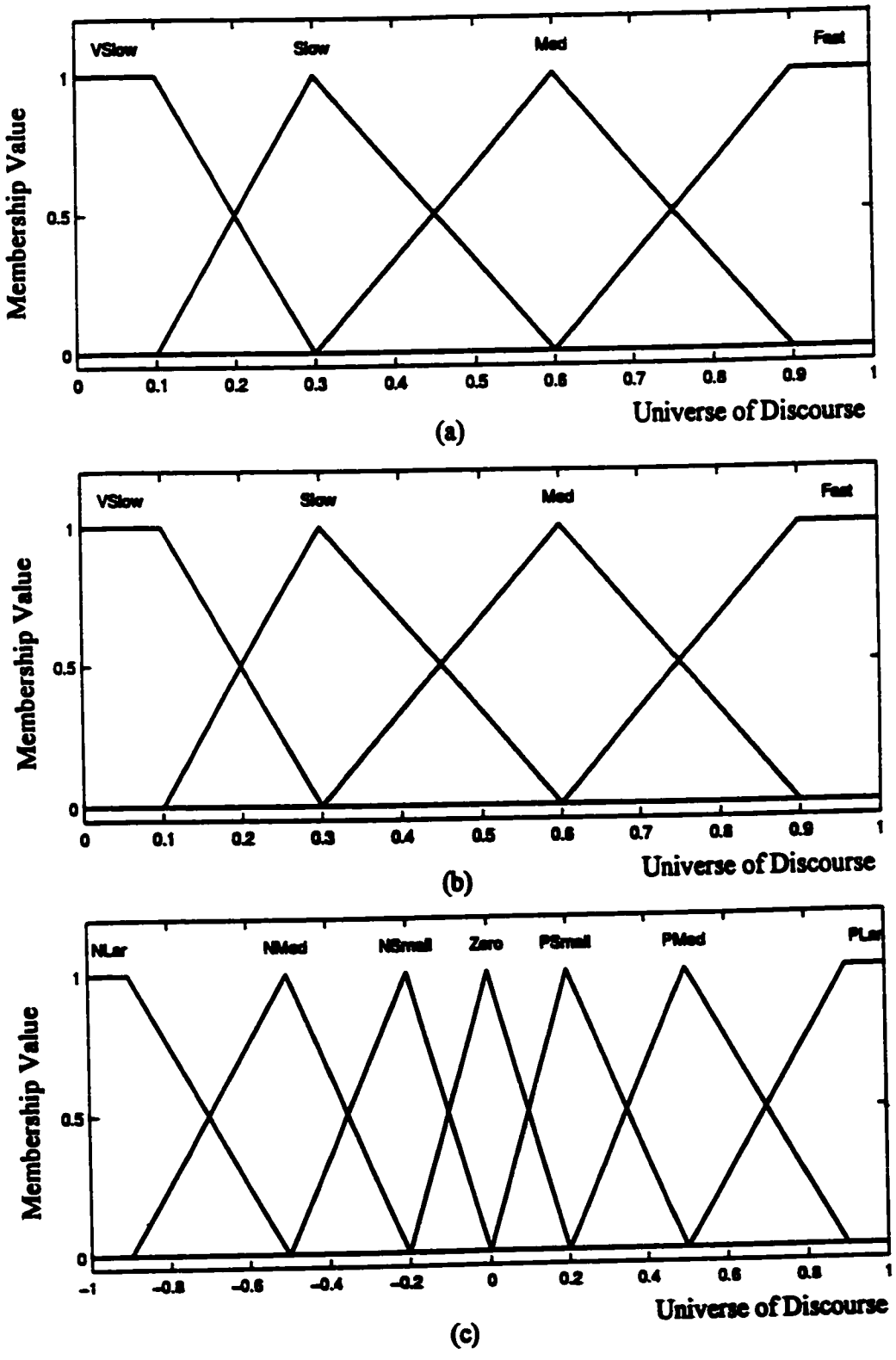
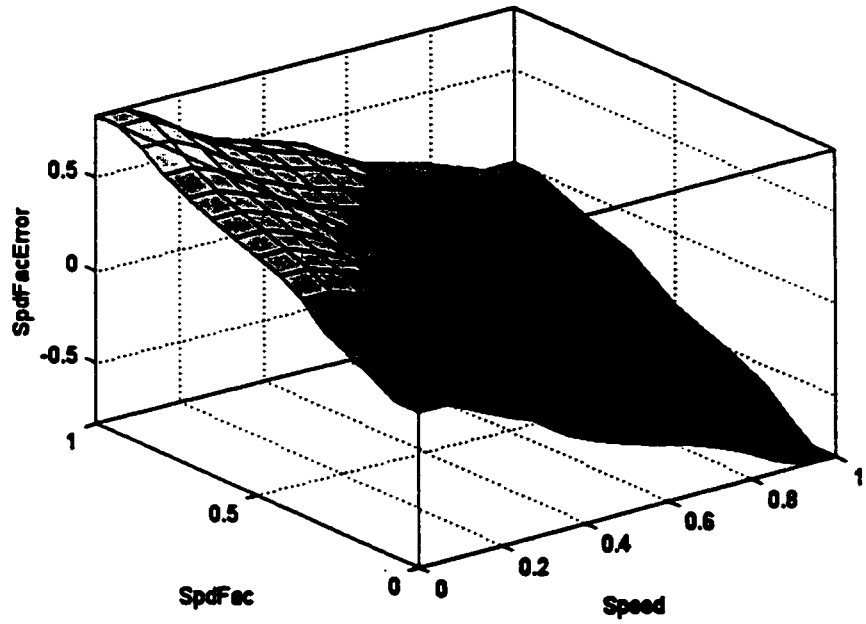
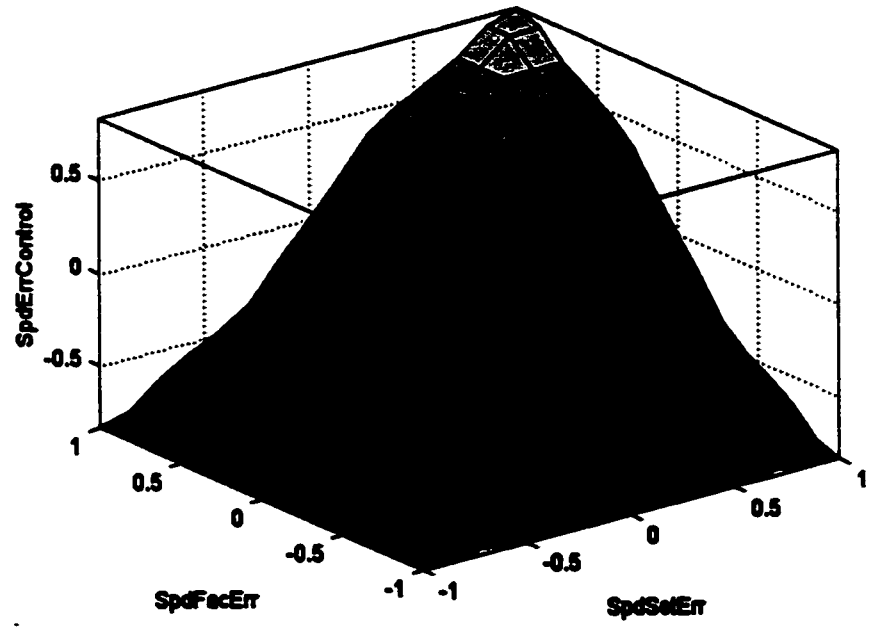


Figure 4.10. Fuzzy membership functions for speed factor error: (a) speed factor input, (b) speed input, and (c) speed factor error output.



(a)



(b)

Figure 4.11. Control surfaces for speed control subsystem: (a) speed factor error as a function of speed factor and speed, and (b) speed error control as a function of speed set error and speed factor error.

output will be accepted. Both of these inputs have identical control over the output, such that the lower value can always reduce the speed of the vehicle.

The Matlab/Simulink diagram used for simulations is included in Appendix A.

4.4.4 Steering Controller Using Fuzzy Control

The RTC steering control block in Figure 4.4 contains the simplified Smith control subsystem and the fuzzy steering control subsystem blocks, as shown in Figure 4.12. The six inputs are angle to intersection, range to intersection, tracking error, steering command, obstacle avoid and steering angle. The first three inputs are assumed to be provided by an onboard sensor processing subsystem. The steering command is the control signal sent from the local terminal. The obstacle avoid signal is the adjustment sent from the obstacle-avoidance subsystem to ensure safe navigation around obstacles. The steering angle is the angle of the wheels referenced to zero (ie. "straight ahead") and measured directly. The first three inputs are used for the semi-autonomous modes of operation (intersection-turning and tunnel-tracking). While all inputs can directly control the steering, the hierarchy has been designed with priority in mind as seen in Figure 4.12. Due to the time delay in separating the local and remote terminals and the option of using semi-autonomous modes, obstacle-avoidance is the highest priority. Next, we want the operator to be able to intervene in the control of the machine whenever required or desired via the steering command. The order of the other three inputs is determined by the semi-autonomous functions associated with each input. This will be discussed during the description of the autosteering control.

4.4.4.1 Teleoperation Mode

Teleoperation mode uses direct operator control via the steering command input or steering set error as it is also called. The steering set error is the difference between the steering set desired (or set) by the operator at the local terminal and the actual delayed steering angle fed back from the remote terminal over the time-delayed link. In teleoperation mode, only the steering set error and obstacle avoid inputs are able to control the steering. It should be noted that the steering set error has a range [–2 2] since it is based on the difference of two signals, each of which has a range [-1 1]. The steering set error and autosteering control signals are combined using the fuzzy hybrid control block. Autosteering control is used for semi-autonomous

operation. The FAM for the hybrid control block is shown in Table 4.5. It can be seen that, for teleoperation mode (autosteering control signal is zero), the hybrid control output has the same linguistic variables as the steering set error input. In the same manner, if a semi-autonomous mode was active, and the steering set error input was zero, the hybrid control output would have the same linguistic variables as the autosteering control input. The fuzzy membership functions are shown in Figure 4.13. The autosteering control and steering set error functions are both identical bipolar functions having five triangles and two bisected shouldered trapezoids similar to those in the speed control section. There is some "bunching" of the membership functions toward the centre for additional sensitivity around zero. The hybrid control output membership function is similar but the triangles are uniformly spaced. The control surface is shown in Figure 4.14. The hybrid control output signal is then combined with the obstacle avoid input in the fuzzy obstacle avoid control block. Just as for the previous fuzzy controller, the FAM shown in Table 4.6 is designed such that the output linguistic variables are identical to the input if either input is zero. The input and output fuzzy membership functions are identical to those of the fuzzy hybrid control block and are not shown. The identical control surface is likewise not shown.

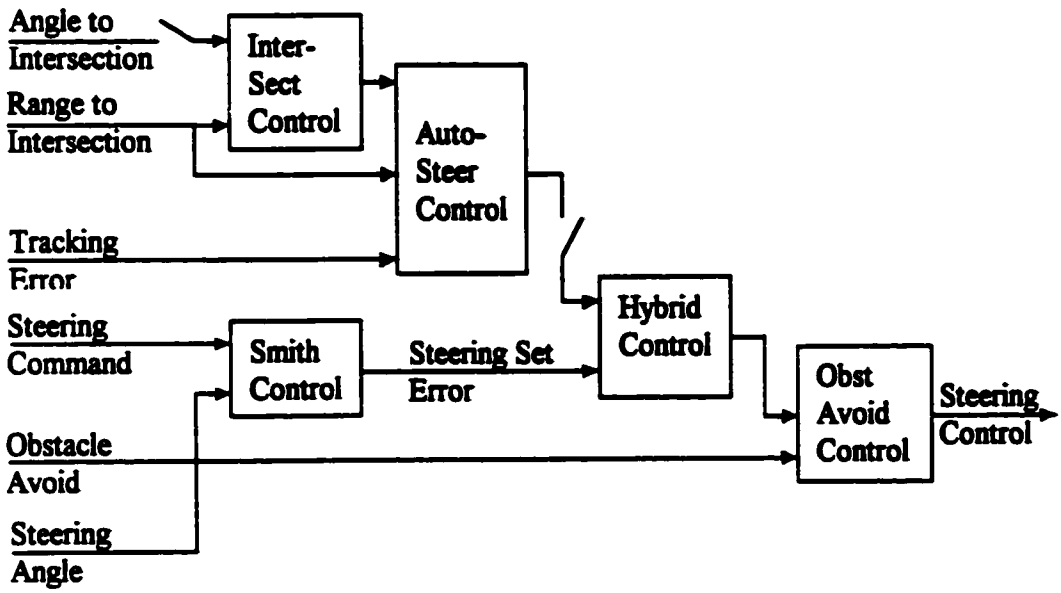


Figure 4.12. RTC steering control block diagram.

Table 4.5 Fuzzy associative matrix (FAM) for hybrid control

	Autosteering Control						
Hybrid Control	Large Left	Medium Left	Small Left	Zero	Small Right	Medium Right	Large Right
Large Left	Large Left	Large Left	Large Left	Large Left	Medium Left	Small Left	Zero
Medium Left	Large Left	Large Left	Large Left	Medium Left	Small Left	Zero	Small Right
Small Left	Large Left	Large Left	Medium Left	Small Left	Zero	Small Right	Medium Right
Zero	Large Left	Medium Left	Small Left	Zero	Small Right	Medium Right	Large Right
Small Right	Medium Left	Small Left	Zero	Small Right	Medium Right	Large Right	Large Right
Medium Right	Small Left	Zero	Small Right	Medium Right	Large Right	Large Right	Large Right
Large Right	Zero	Small Right	Medium Right	Large Right	Large Right	Large Right	Large Right

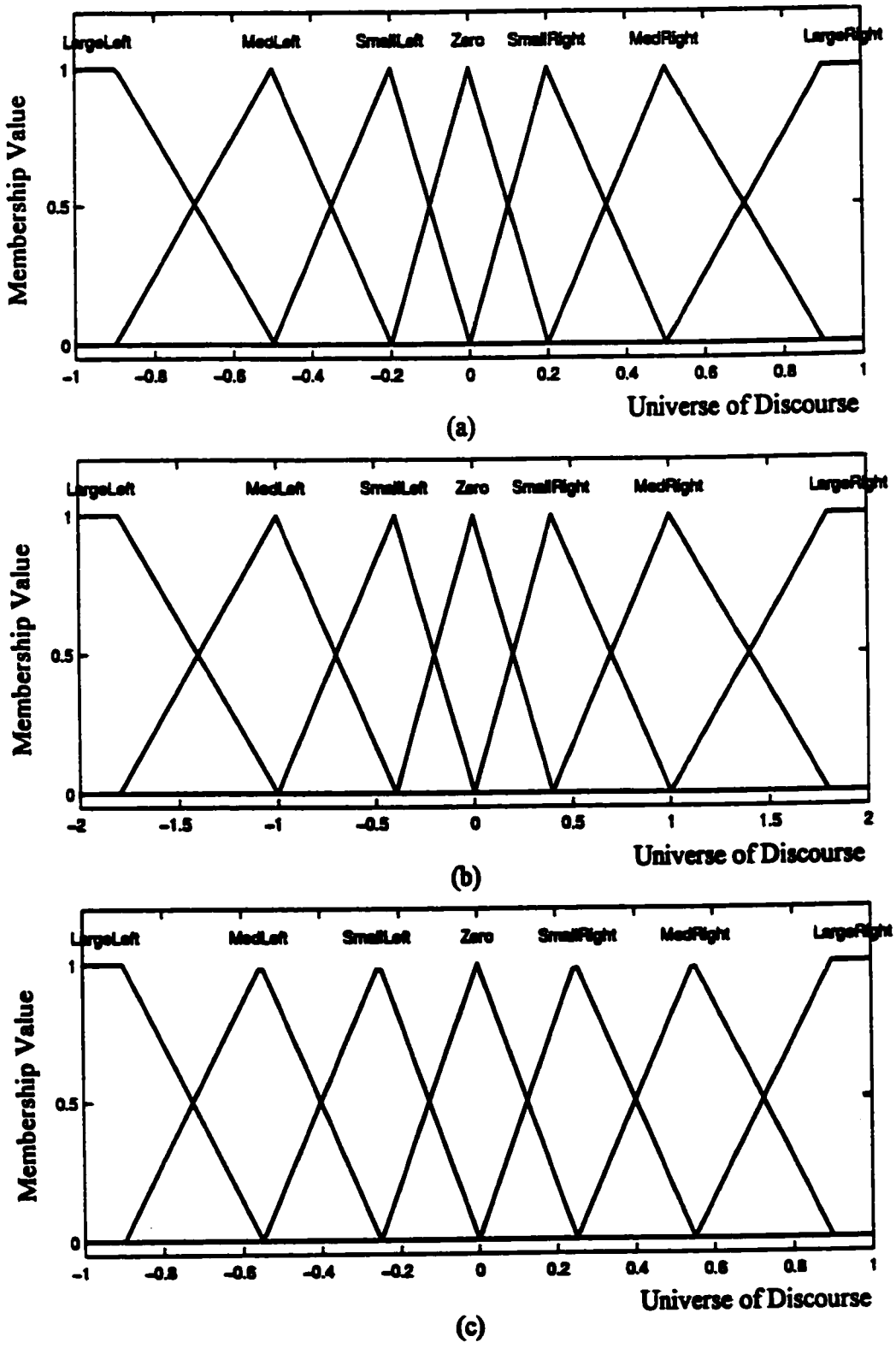


Figure 4.13. Membership functions for hybrid control: (a) autosteering control input, (b) steering set error input, and (c) hybrid control output.

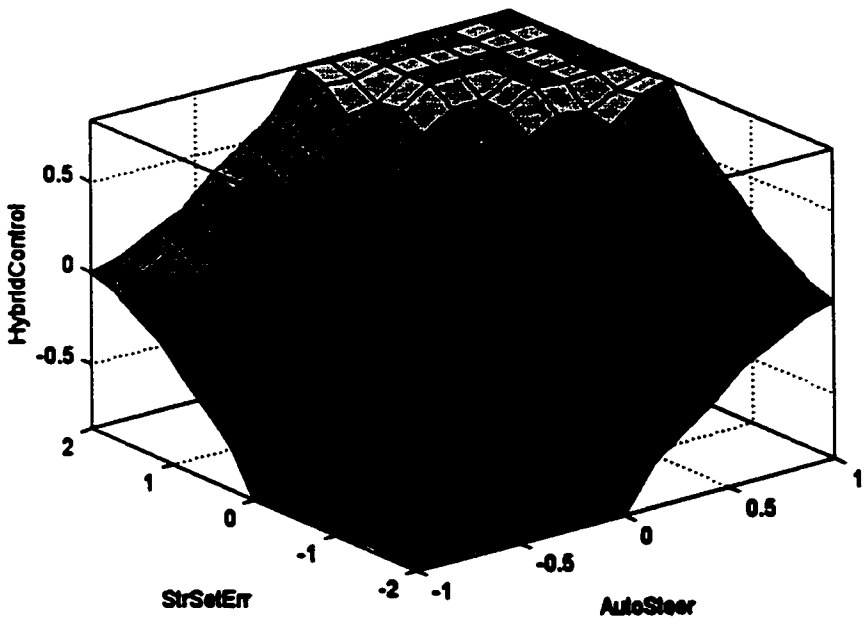


Figure 4.14. Control surface for hybrid control as a function of autosteering control and steering set error.

4.4.4.2 Semi-Autonomous Modes

The two semi-autonomous modes are primarily concerned with automatic steering. Tunnel-tracking is self-explanatory: the robot vehicle proceeds down the tunnel on the basis of the approximate centreline. That is, it uses an estimate of the centreline of the tunnel as the desired reference, and any deviation from it of the vehicle's heading will be used as the tracking error for control purposes. In order to enable this mode, the operator would send a supervisory control command to close the switch which is directly following the autosteering control block shown in Figure 4.12. It would then be normal for the operator to release the steering control device (e.g. joystick) thereby sending a steering command (ie. steering set error) of zero to the remote terminal and the steering control subsystem. This frees the operator from having to make all of the real-time control decisions and responses, but retains the capability to intercede at any time since the steering command loop is still connected on-line. In tunnel-tracking mode, the range to intersection is also used. Speed is reduced for safety reasons in case of cross-traffic through the intersection as discussed in section 4.4.3.

Table 4.6 Fuzzy associative matrix (FAM) for obstacle avoid control

	Obstacle Avoid						
Hybrid Control	Large Left	Medium Left	Small Left	Zero	Small Right	Medium Right	Large Right
Large Left	Large Left	Large Left	Large Left	Large Left	Medium Left	Small Left	Zero
Medium Left	Large Left	Large Left	Large Left	Medium Left	Small Left	Zero	Small Right
Small Left	Large Left	Large Left	Medium Left	Small Left	Zero	Small Right	Medium Right
Zero	Large Left	Medium Left	Small Left	Zero	Small Right	Medium Right	Large Right
Small Right	Medium Left	Small Left	Zero	Small Right	Medium Right	Large Right	Large Right
Medium Right	Small Left	Zero	Small Right	Medium Right	Large Right	Large Right	Large Right
Large Right	Zero	Small Right	Medium Right	Large Right	Large Right	Large Right	Large Right

Intersection-turning mode is enabled by closing both switches shown in Figure 4.12 from the local terminal. Tunnel-tracking is required as part of the intersection-turning mode. Intersection-turning mode is accomplished by using the fuzzy intersection control block. The FAM for this block is shown in Table 4.7. This is designed to give an output of zero for all ranges to intersection greater than zero, unless there is an appreciable angle between the vehicle's heading and the centre of the intersection. The reason for this is that there must be no significant steering into the intersection until the robot vehicle has "reached" the intersection in the fuzzy sense. (The location of the intersection for the determination of range is considered to be defined by the closest corners.) This occurs when the range to intersection decreases to zero. In this region the intersection control output effectively is defined by the

Table 4.7 Fuzzy associative matrix (FAM) for intersection control

	Angle to Intersection						
Range to Intersection	Large Left	Medium Left	Small Left	Zero	Small Right	Medium Right	Large Right
Zero	Large Left	Medium Left	Small Left	Zero	Small Right	Medium Right	Large Right
Close	Medium Left	Small Left	Zero	Zero	Zero	Small Right	Medium Right
Medium	Small Left	Zero	Zero	Zero	Zero	Zero	Small Right
Far	Zero	Zero	Zero	Zero	Zero	Zero	Zero

angle to intersection by using the same linguistic variables. The automatic steering signal, intersection control, will then be proportional to the angle between the vehicle's heading and the intersection. (The location of the intersection for the determination of angle is considered to be defined by the midpoint between the corners of the intersecting tunnel entrance.) Only then will the intersection control output begin to make the turn and continue until it has been completed. This can be seen from the control surface in Figure 4.15. It can be seen that if the range to intersection is "far" (ie: close to 1), the output control signal is zero, regardless of the angle to intersection. However, as the vehicle approaches the intersection, the range to intersection becomes shorter. This fuzzy range and angle information serves to begin steering the vehicle into the intersection. This should maximize the likelihood of steering into the centreline of the intersecting tunnel and minimize the likelihood of hitting the closest corner. The membership functions are shown in Figure 4.16. The angle to intersection is seen to be comprised of five trianglar and two bisected triangles at the ends. In this case, it was felt that the triangles should be spread at the centre to allow less sensitivity in the centre region and to increase it toward the ends of the universe of discourse. The range to intersection has two triangular and two bisected shouldered trapezoidal functions. The shoulders are intended to provide small regions of minimal change in

effect. The intersection control output has five triangular and two bisected trapezoidal membership functions.

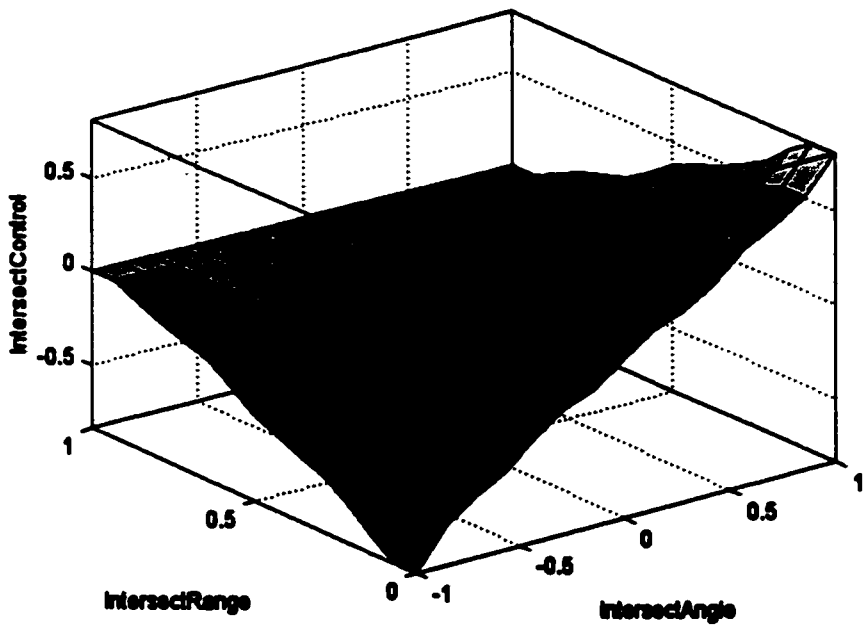


Figure 4.15. Control surface for intersection control as a function of angle to intersection and range to intersection.

Tunnel-tracking is also designed to form part of the intersection-turning mode so activating this latter mode involves closing both the teleoperation/ tracking mode and the tunnel-tracking/intersection-turning mode switches shown in Figure 4.12 from the local terminal. The general philosophy is that the robot vehicle would approach an intersection in tunnel-tracking mode until the intersection is reached. At that time, the tracking-error control input is diminished and the intersection control input begins to take control. The autosteering control scheme as shown in Figure 4.17 has been designed to perform this function, which is simulated in Matlab in Figure A.15. Once the turn has been accomplished and the vehicle is in the new tunnel, tunnel-tracking mode would be re-enabled and the vehicle will proceed down the new tunnel.

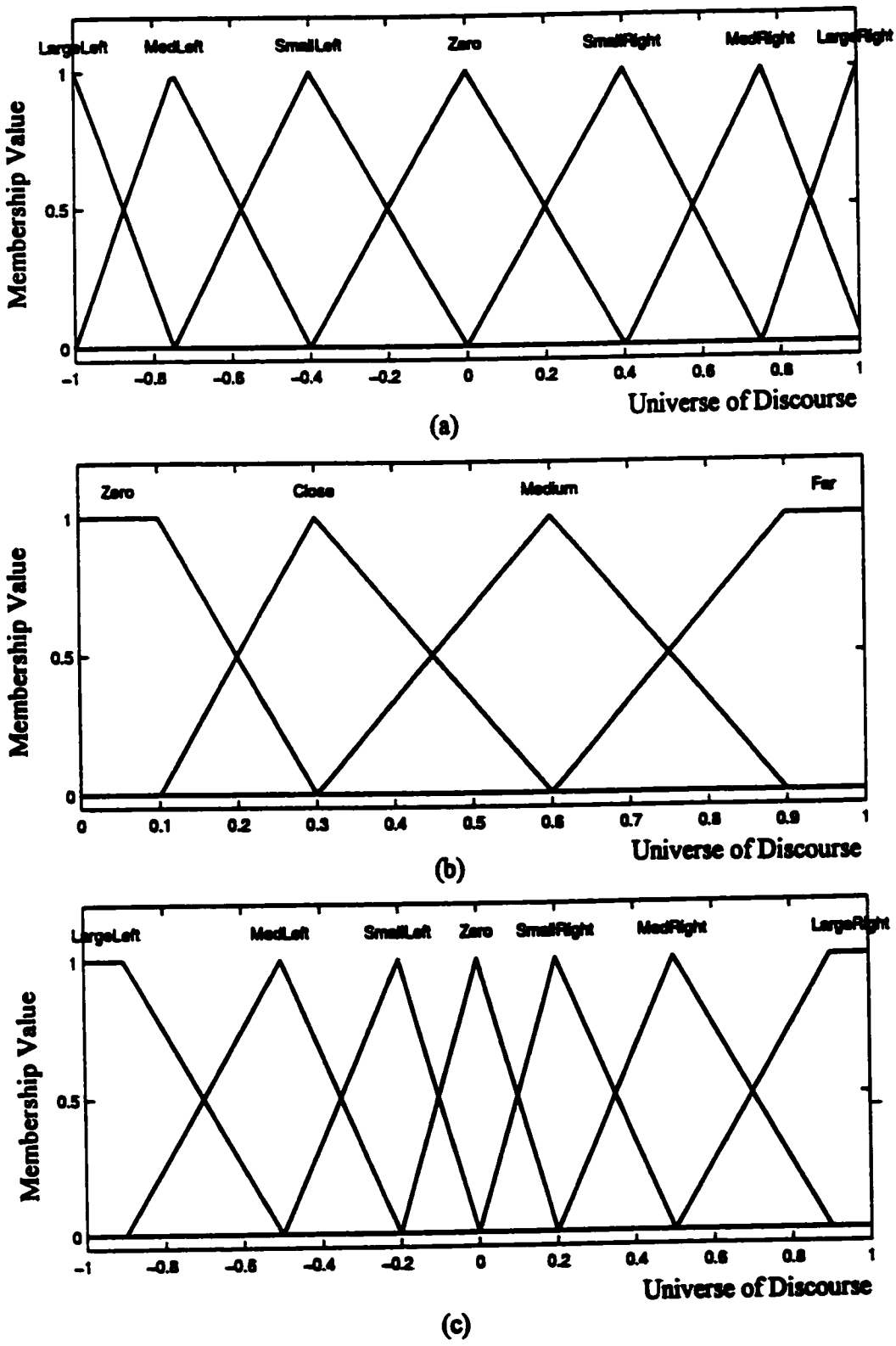
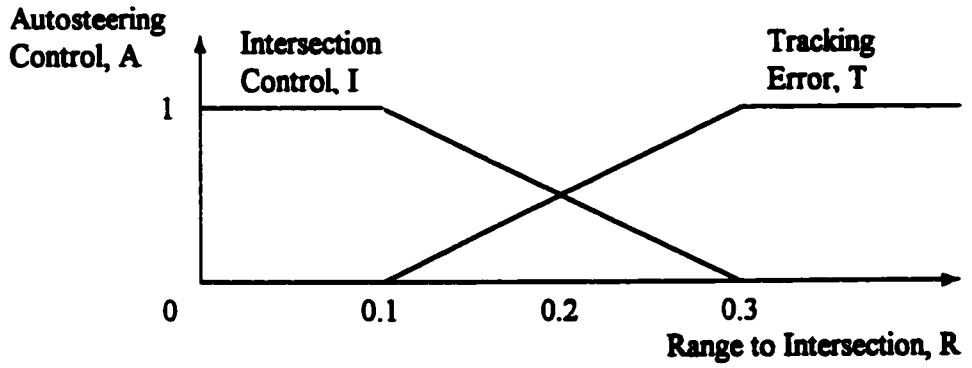


Figure 4.16. Membership functions for intersection control: (a) angle to intersection input, (b) range to intersection input, and (c) intersection control output.



$R < 0.1$: $A = I$

$R > 0.3$: $A = T$

$0.1 < R < 0.3$: $A = (1.5 - 5R)I + (5R - 0.5)T$

Figure 4.17. Autosteering control scheme.

4.4.5 Robot Vehicle Dynamics

The robot vehicle block diagram shown in Fig. 4.3 contains the speed and steering actuators. One of the original partners involved in this project, a local company with extensive experience with mining equipment, had requested that high-gain, proportional controllers driving the actuators in separate closed loops be provided on the vehicle. This became part of the design and was retained through all of its development which was done using Matlab/Simulink. In the early stages of the project, this portion of the speed control system included an amplifier of gain 100 and a first-order plant having a time constant of 1 second in a closed loop. The steering control system portion also used an amplifier of gain 100 but the first-order plant had a time constant of 0.25 seconds. These plant models were arbitrarily chosen on the basis of the time constants, since 1 second and 0.25 second seemed to be good representative values for the speed and steering response of the actual vehicle that was intended to be used in field testing of the system developed in this thesis. First-order plant models seemed reasonable for the vehicle. This implied that the time to reach steady-state would be ~ 4-5 seconds for the speed and ~ 1-1.25 seconds for the steering responses, respectively. Actual tests of the test vehicle were never done since it was never completed. While early development using the first-order plant models progressed satisfactorily with excellent performance results, it was suggested that using a higher-order plant model might be a better choice and a better test of the system. Even though

the primary objective of this project was to determine a solution to overcome problems due to time delay in the remote-control system, this suggestion was accepted and various higher-order plant models were tested. The criteria used in the selection of suitable models were threefold:

1. The models should be of (at least) second-order.
2. The damping coefficient ζ should be ~ 0.5 (or greater) for good response with minimal overshoot.
3. The cutoff frequency should be at least 10 times that of a human operator so that in teleoperation mode, the dynamics of the operator would be controlling. This would mean a cutoff frequency of at least 5 Hz (or 30 r/s). It should be noted that the reason for this criterion is to ensure accuracy of the system output when using the predictor. The predictor design used in this system provides only the steady-state output of the system. A predictor that can accommodate the transient response of the system for any speed/steering angle initial condition would not need this restriction.

The first two criteria were to be valid for each of the speed and steering control systems on a complete end-to-end basis for the non-delayed case with no human operator. This would then form the basic system response. A specific type of vehicle with specific dynamics has not been defined or demanded in this project. It is intended to be as general as possible while retaining the primary application described as a mining machine having the features and criteria indicated herein. Depending on the type of mine, its location (e.g. on the earth or the moon), and the function of the machine, a wide variety of dynamics is possible. If it is an ore-hauling vehicle, another consideration is whether it is empty or full. The third criterion implies restrictions on the vehicle. A high power-to-mass ratio is required. There may be restrictions on maximum allowable speed. The necessary dynamics to meet the three criteria outlined above are expected to be attainable and form the basis for the system described in this thesis. This is not meant to imply that vehicles meeting different criteria will not work in this type of system. For that situation, some modifications will likely be required, some possibly very minor and others possibly more extensive. It is beyond the scope of this thesis to address them here.

The input control device is assumed to be a joystick type. Therefore, the position of the joystick controls both the vehicle speed and its steering angle, that is, a velocity and a (angular) displacement, respectively. The speed model selected was a

basic second-order configuration having a proportional gain of 100 to conform to the original design. The steering model selected was a first-order plant in series with a proportional gain of 100 and an integrator. The separate integrator provides the (angular) displacement output for a (angular) velocity input. Both of these models in unity-feedback loops provide second-order closed-loop responses. Many models for speed and steering dynamics were investigated. The two open-loop models selected that met the criteria were:

$$\text{Speed dynamics model,} \quad G_V(s) = \frac{100}{s^2 + 65s + 2} \tag{4.1}$$

$$\text{Steering dynamics model,} \quad G_S(s) = \frac{100}{s(s + 100)} \tag{4.2}$$

The speed and steering control systems were tested with their respective vehicle dynamics models given in Equations 4.1 and 4.2 using their primary control loops. The primary control loop for the speed control system is shown in Figure 4.18. The general structure of the steering control system is identical, with the exception of the input gain, which is unity, and the vehicle dynamics which is $G_S(s)$. The Matlab/Simulink systems used in the simulations are shown in Appendix A. These simulations include all system elements except the delays and the human operator dynamics. The speed and steering control system responses using these models were determined using Matlab and are presented in Figures 4.19 and 4.20, respectively. These are based on linearized state-space models of the systems, as determined by Matlab, from which the Bode diagrams were plotted. From these plots, we can see that the damping coefficient

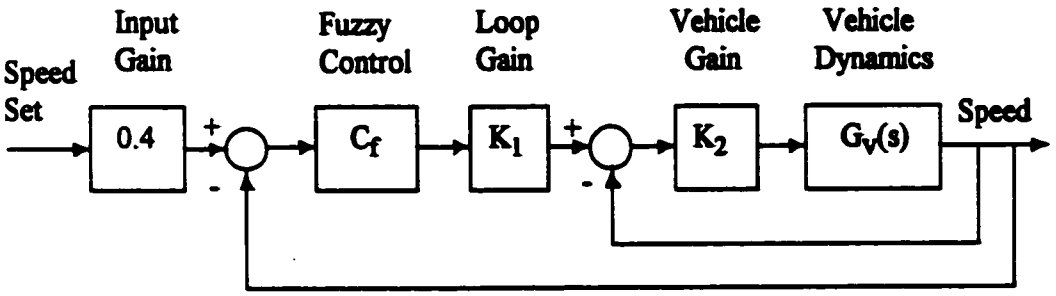


Figure 4.18. Speed control system primary control loop showing major gain and dynamics blocks.

$\zeta \approx 0.5$ since there is a slight overshoot for the steering response, while for the speed response, it looks to be more like $\zeta \approx 0.6$. The cutoff frequency is seen to be ~ 52 r/s (8.3 Hz) and ~ 90 r/s (14.3 Hz) for speed and steering, respectively, suggesting that, in general, the criteria have been met.

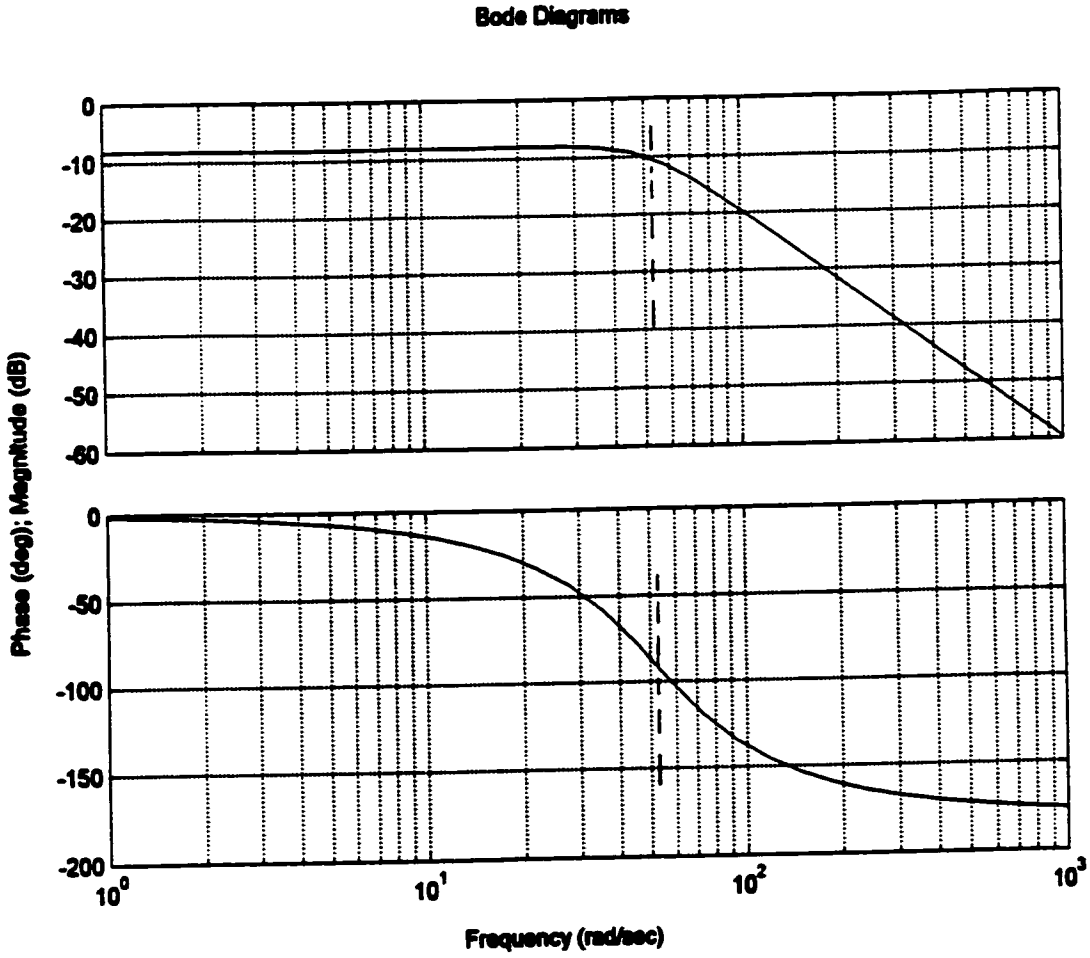


Figure 4.19. System speed response with no delay or human operator dynamics.

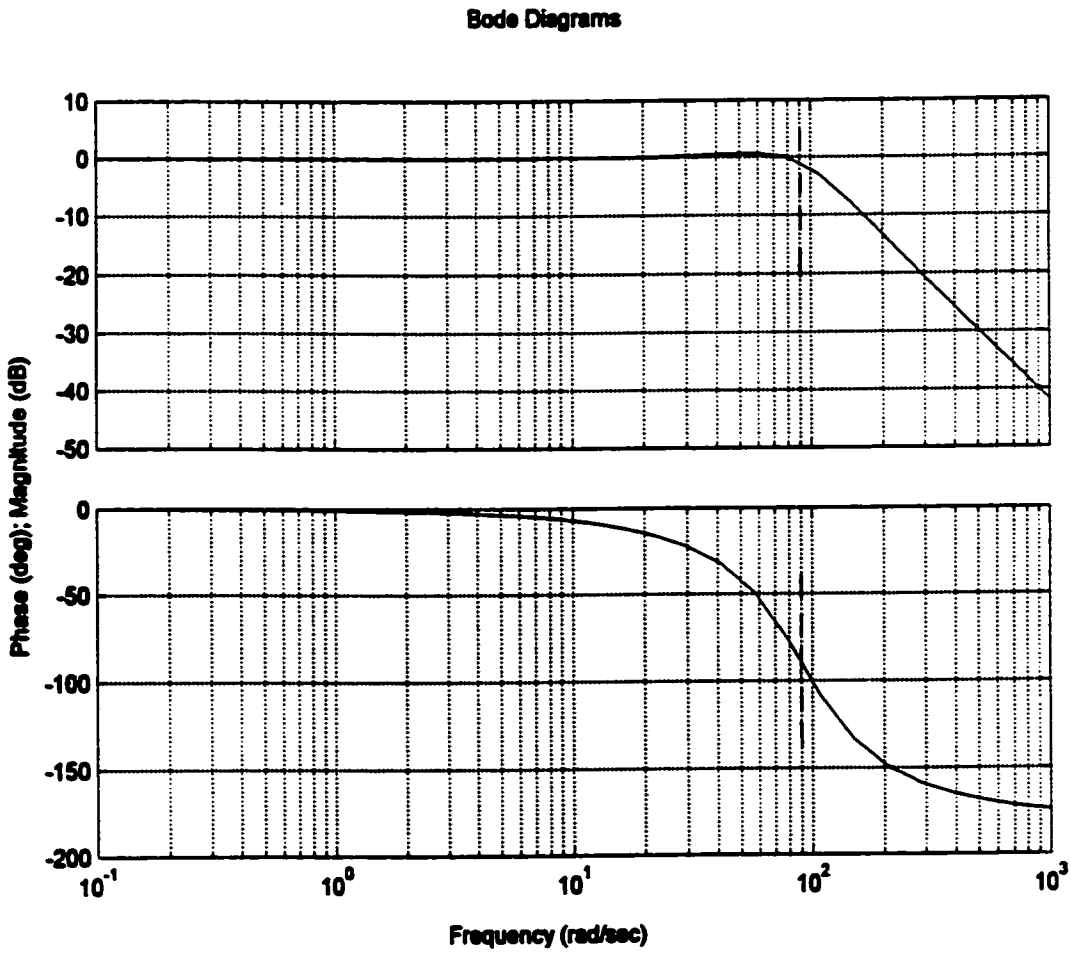


Figure 4.20. System steering response with no delay or human operator dynamics.

Models of automobile speed dynamics indicate that, while higher-orders are present, first-order dynamics are domininant [96]. Gillespie [97] shows that a vehicle's acceleration is essentially determined by the ratio of engine power to vehicle weight or

$$a = \frac{F}{m} = 550 \frac{g}{W} \frac{HP}{V} \tag{4.3}$$

where F is the tractive force at the drive wheels
m is the mass of the vehicle
g is the gravitational constant, 9.8 m/s^2
W is the weight of the vehicle
HP is the engine horsepower
V is the forward speed.

The velocity term in the denominator requires the acceleration to decrease with increasing speed. Since velocity is the integral of the acceleration, it can be easily shown that the velocity will be of the form

$$V = V_m (1 - e^{-t/T}) \tag{4.4}$$

where V_m is the maximum speed
T is the time-constant.

This implies a first-order speed response. A similar analysis by Wong [98] shows that the speed response of a passenger car follows a similar exponential rise to full speed.

Vehicle steering response is generally a more complex problem. Factors having a direct influence on the steady-state response include [99]:

- Wheelbase
- Turning radius
- vehicle speed
- Slip angles on front and rear tires

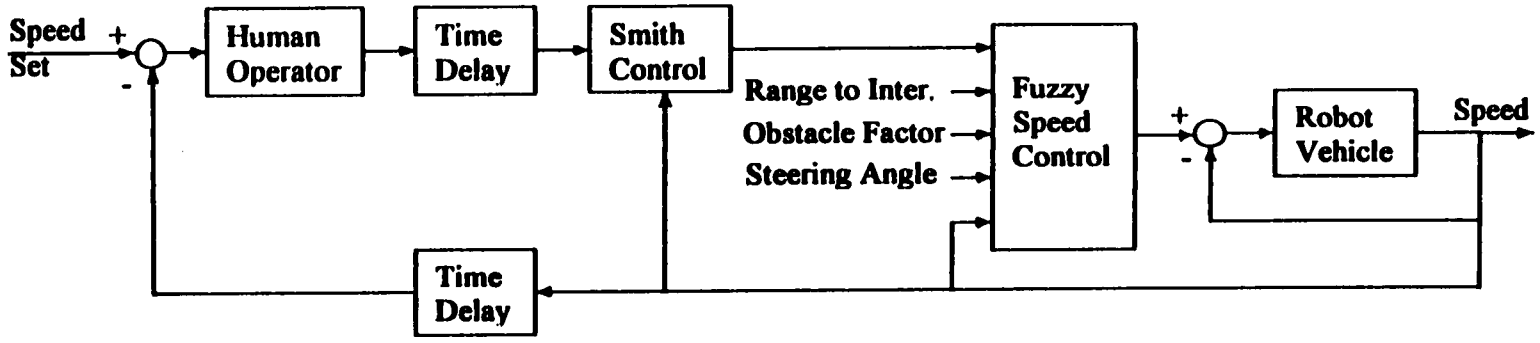
Transient-response must take into account additional factors such as translational and rotational inertia [99]. This is dependent on the vehicle's mass and physical design. An

analysis of a simplistic model by Ellis [100] considers oscillatory steering behavior using a second-order model. The optimum transient response is the fastest response with minimum oscillation [99]. A more in-depth analysis of the vehicle's steering response is not within the scope of this thesis, nor has a particular design been specified. On the basis of the foregoing discussion, it seems reasonable to use a steering model restricted to second-order. Furthermore, since oscillation or even overshoot in steering is highly undesirable, the model must have sufficient damping.

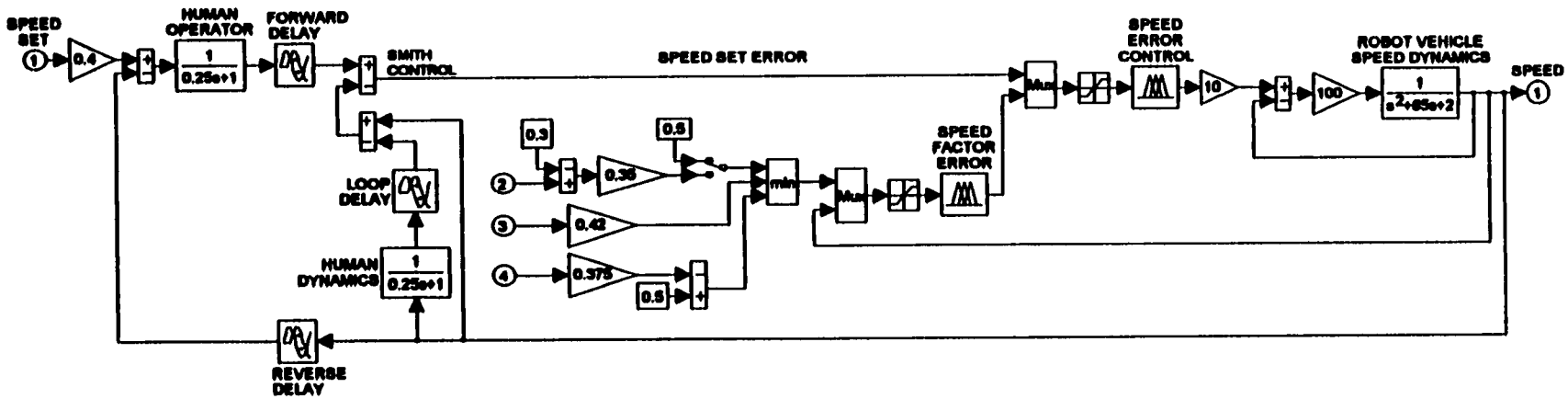
If we examine the closed-loop responses of the unity feedback speed and steering dynamics models, it is apparent that each is strongly overdamped. This was found to be necessary for overall system stability. While the results are not included here, the step-function responses can be approximated by a first-order system based on the dominant pole of each model. From the step responses, the time-constants of the closed-loop, unity-feedback speed and steering dynamics models are approximately 0.75 seconds and 1.1 seconds, respectively. These values are very close to the respective time constants, 0.625 seconds and 0.99 seconds, due to the dominant poles.

4.5 Speed and Steering Control System Diagrams

The subsystem block diagram schematic representation of the remote-control system as described in the foregoing sections is not the most convenient for appreciating the structure from an overall control system point of view. For this reason, the complete speed and steering control system diagrams are included in Figures 4.21 and 4.26, respectively. The multiloop, multivariable nature of the systems is apparent. These system diagrams show each control system based on their primary control loops. For example, the obstacle-avoidance subsystem is not included here. In order to relate these diagrams to the functional subsystem blocks described in the preceding paragraphs, the particular elements have been grouped into the same subsystem blocks and shown in Figures 4.22 through 4.25 and Figures 4.27 through 4.30 for each system, respectively. Only the major blocks are included. These subsystems have been described previously in this chapter. Both the block diagrams and their Matlab/Simulink equivalent are included in each figure to assist the reader who may not be familiar with Simulink nomenclature in making comparisons. Some of these diagrams are also included in the complete Matlab/Simulink configuration diagrams in Appendix A.

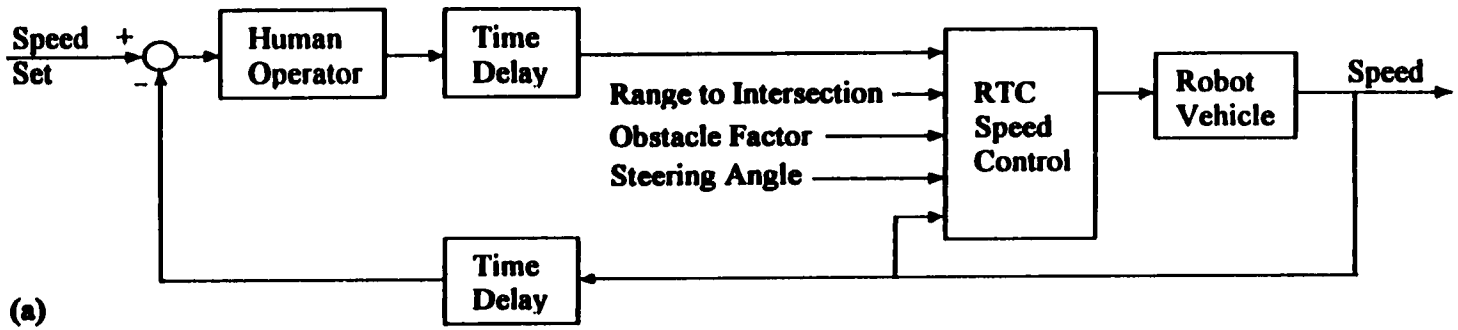


(a)



(b)

Figure 4.21. Speed control system: (a) block diagram, and (b) Simulink diagram.



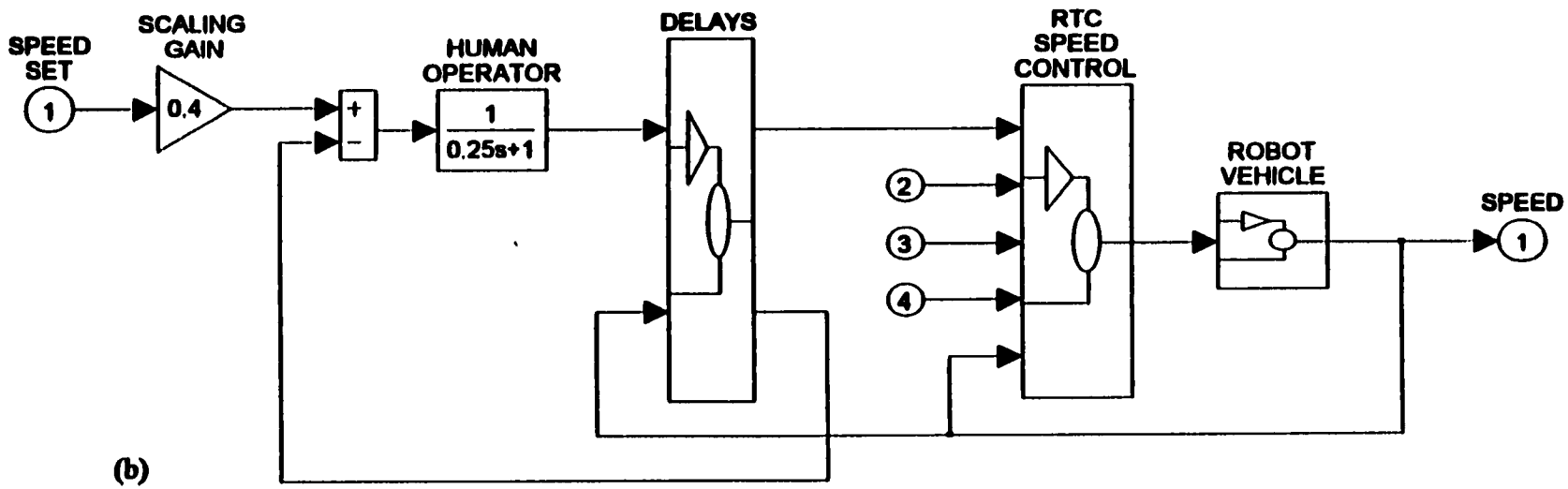
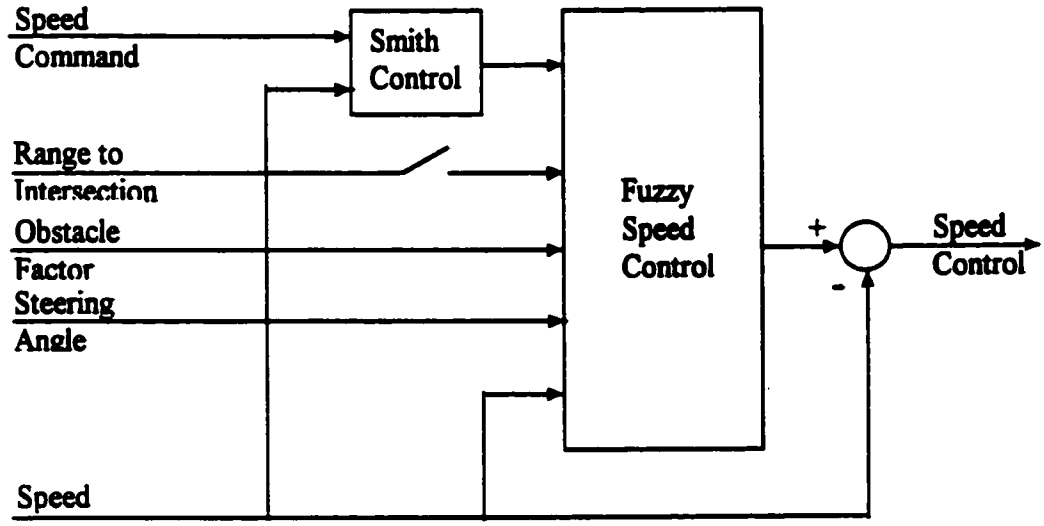
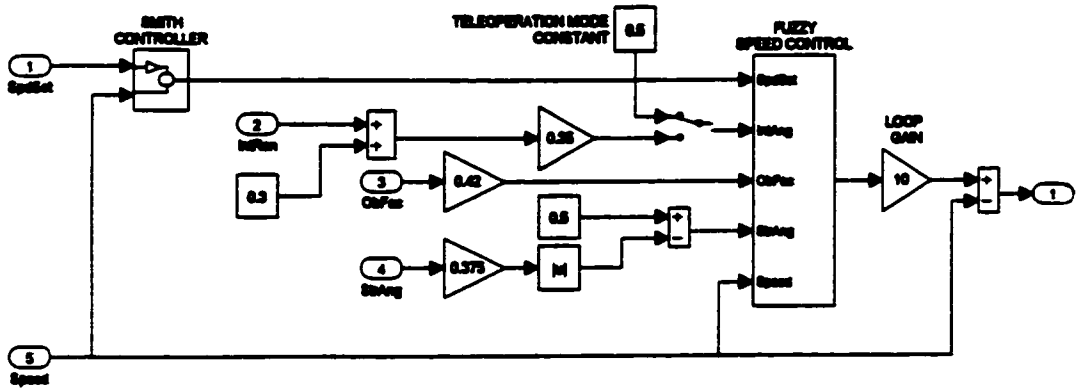


Figure 4.22. Speed control system: (a) subsystem diagram, and (b) Simulink diagram.



(a)



(b)

Figure 4.23. RTC speed control: (a) block diagram, and (b) Simulink diagram.

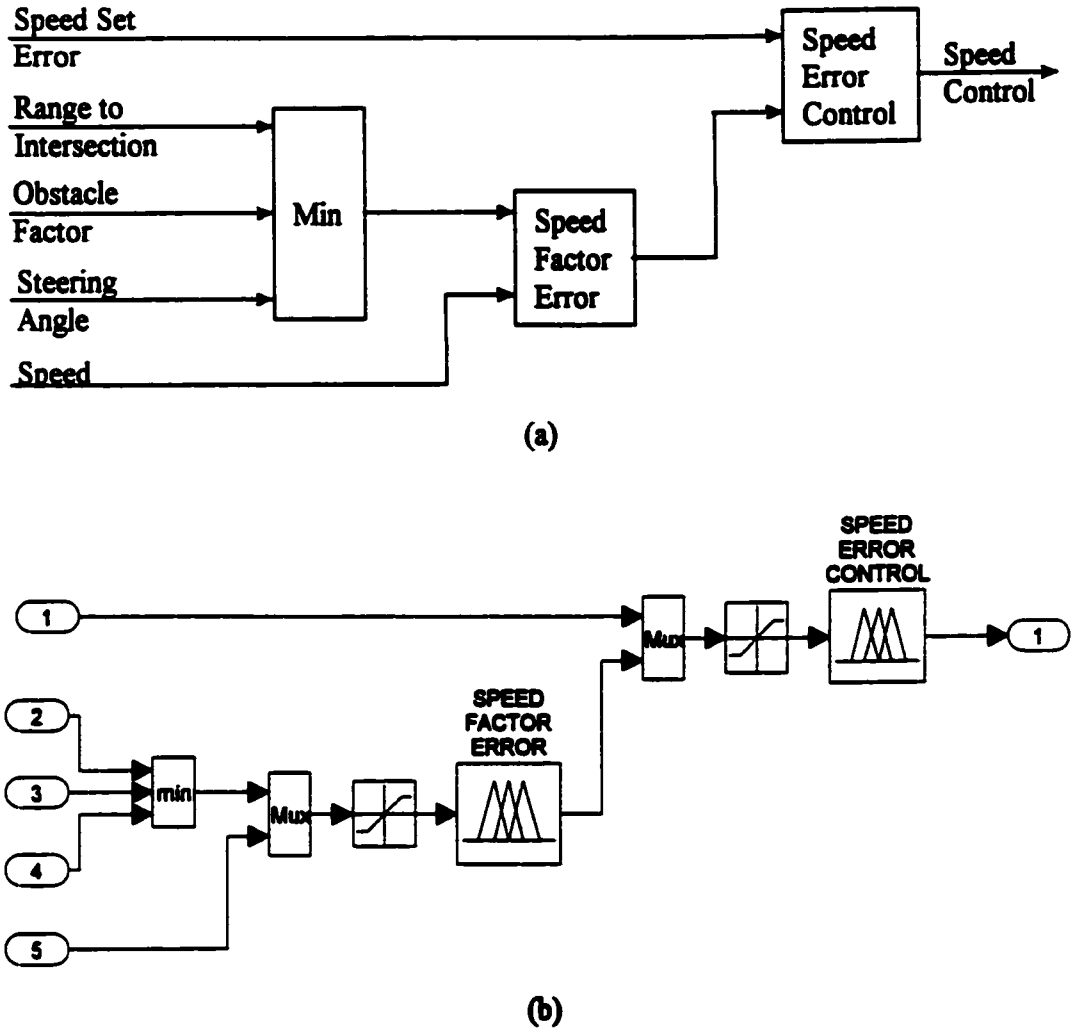
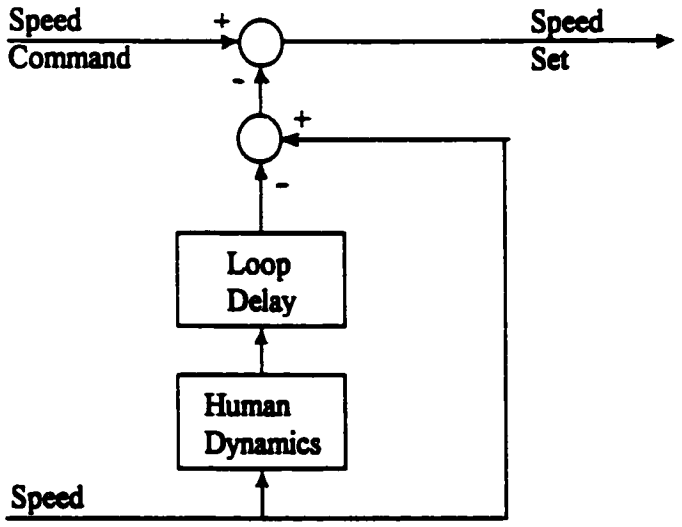
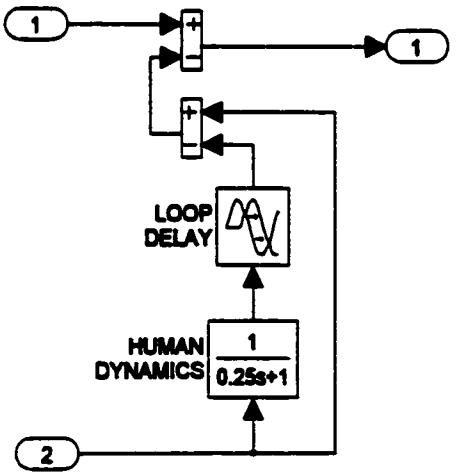


Figure 4.24. Fuzzy speed control: (a) block diagram, and (b) Simulink diagram.



(a)



(b)

Figure 4.25. Smith control: (a) block diagram, and (b) Simulink diagram.

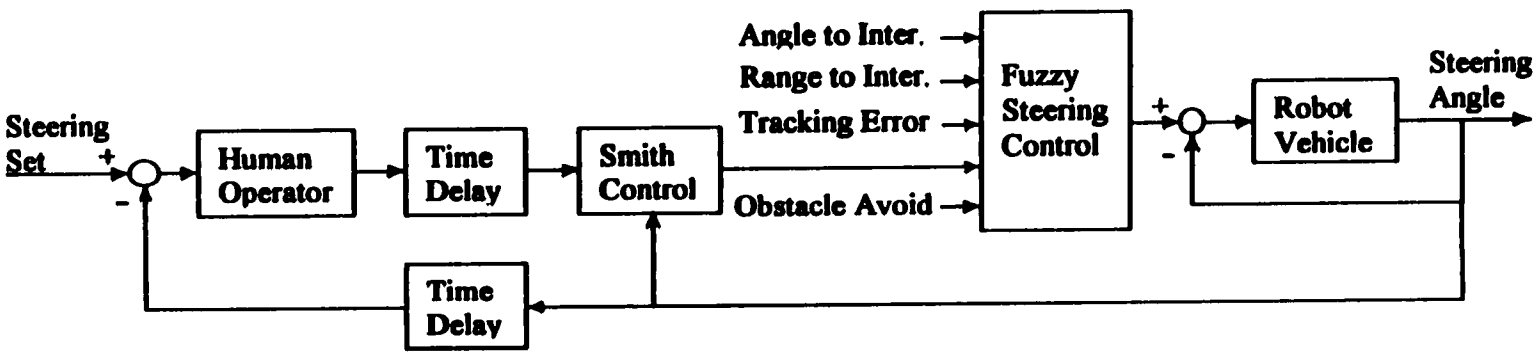


Figure 4.26. Steering control system block diagram.

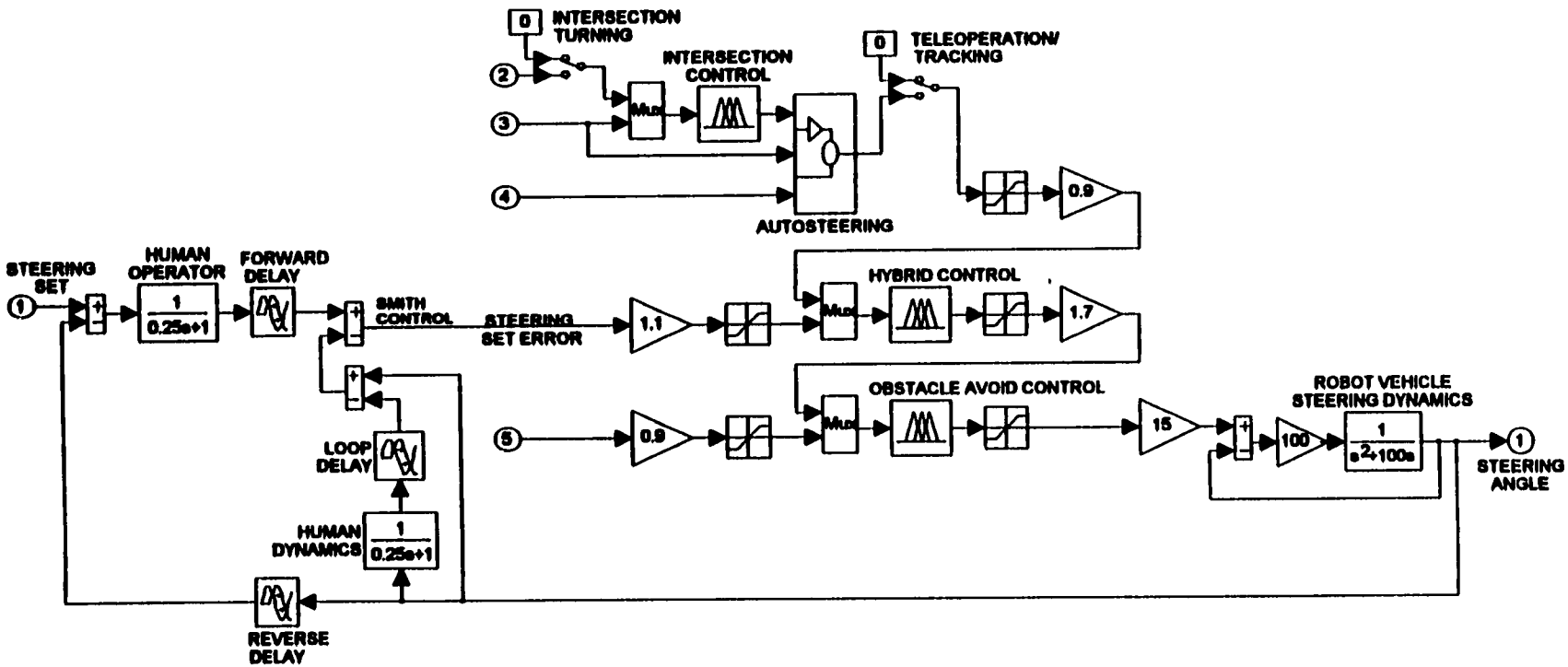
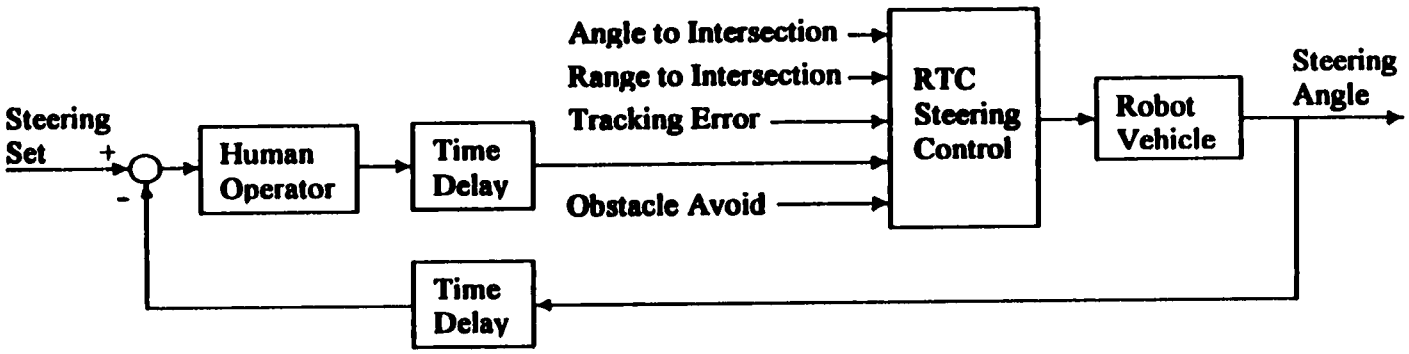
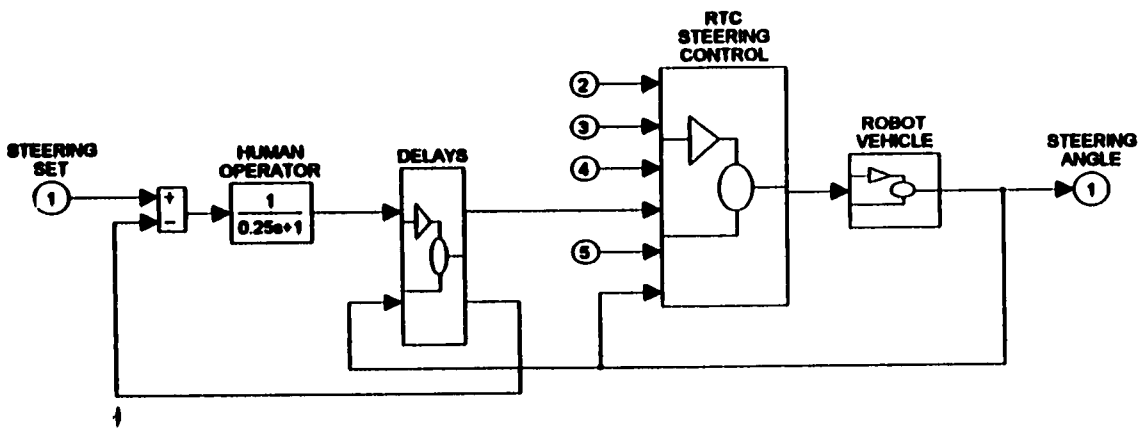


Figure 4.27. Steering control system Simulink diagram.

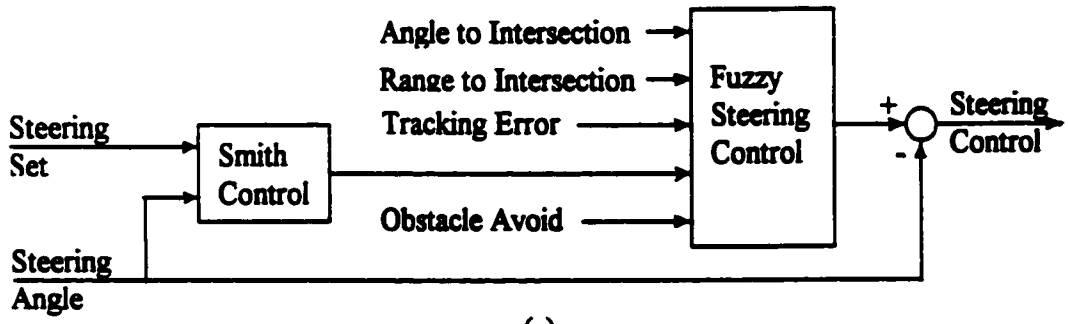


(a)

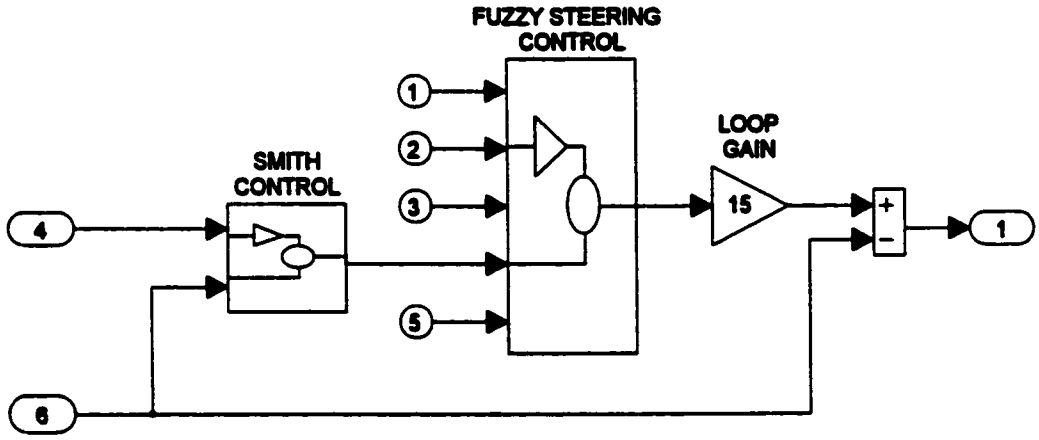


(b)

Figure 4.28. Steering control system: (a) subsystem diagram, and (b) Simulink diagram.

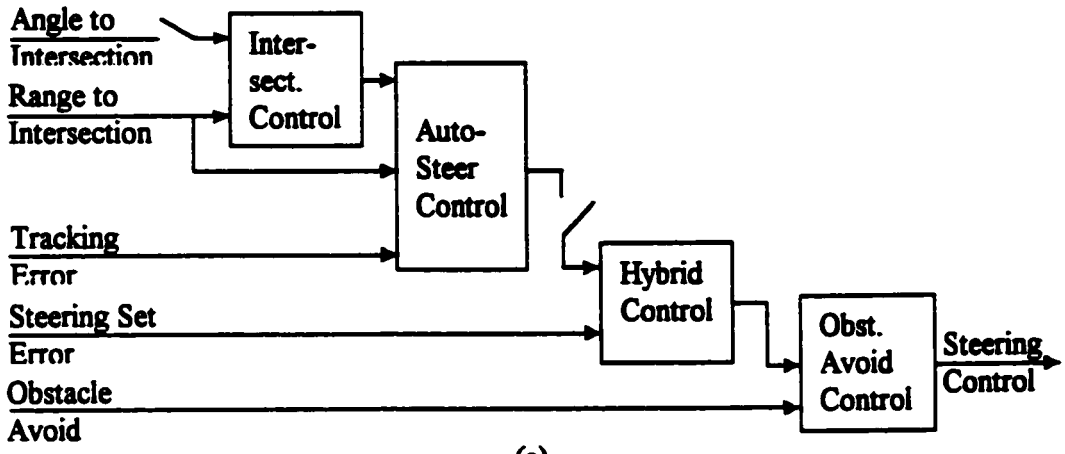


(a)

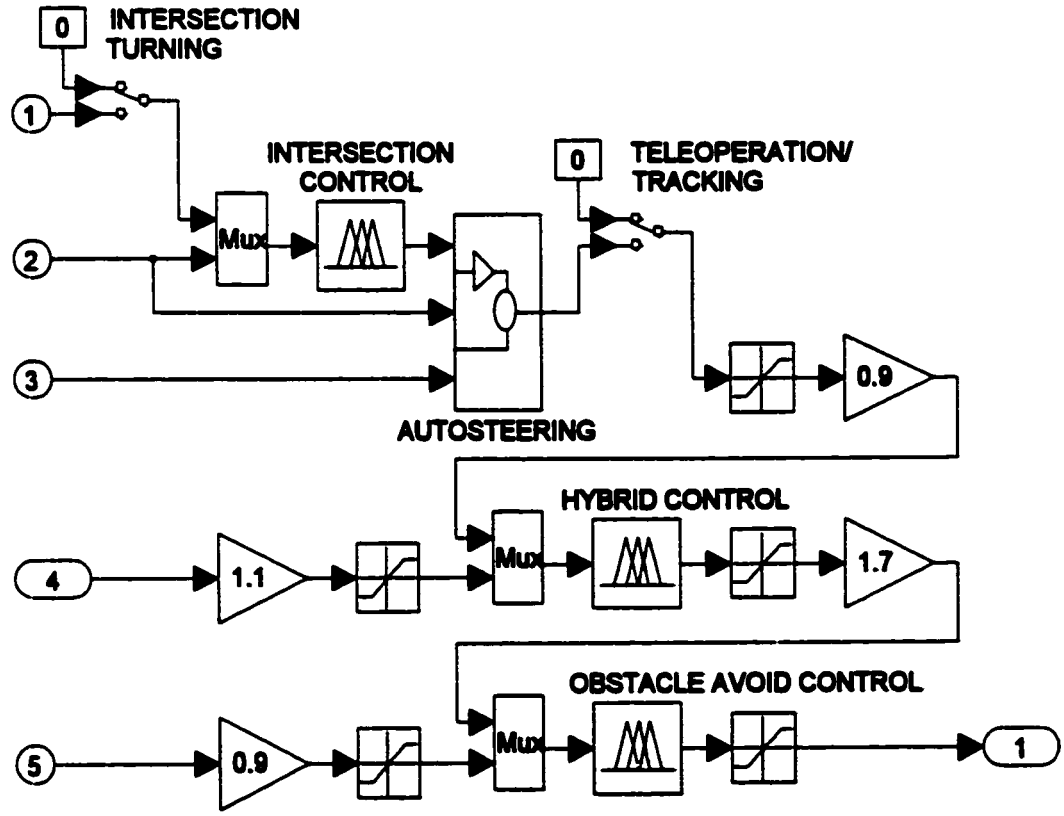


(b)

Figure 4.29. RTC steering control: (a) block diagram, and (b) Simulink diagram.



(a)



(b)

Figure 4.30. Fuzzy steering control: (a) block diagram, and (b) Simulink diagram.

4.6 Human Operator Considerations

The operator typically uses pursuit-mode tracking when using a pursuit-type display, although there is no guarantee of this [79]. It is possible that the operator will visualize the problem in terms of minimizing the error, thereby adopting compensatory tracking. This is not likely to be a major problem since a human operator will tend to do whatever is required to perform the tracking task by adopting any necessary lead or lag equalization [63]. This adds to the difficulty of modeling human operators.

While the human operator may be able to compensate for his/her neurophysiological limitations given sufficient preview, there are more complicating considerations The operator's control output is the output of the error detector, ie: the difference between the reference input and the system feedback signal. Given sufficient preview of the reference signal and the current feedback signal, it appears that the trained human operator can compensate for the inherent neurophysiological limitations (primarily reaction delay) to effectively allow an "ideal" error-detection function [77]. The main problem in this case is that the feedback signal is delayed. It is this delay which causes operators so much problem, leading to loss of control (ie. instability) with relatively short delays (<1 s) [1]. By providing simplified Smith control, however, the effect of the time delay can be compensated if the operator can provide ideal error detection. It is the contention in this thesis that, given sufficient preview of the road ahead, the operator can compensate for inherent human reaction delay and provide the required error detection.

Another scenario involves the use of a system predictor and predictor display. With a predicted system output appearing on the screen showing the output of the system without time delay, the operator now has a non-delayed "feedback" signal to use in the error-detection function. The resulting "non-delayed" error will now be different than when using the delayed system feedback. Therefore, this signal can not be used as the command signal to the remote terminal unless we operate in open-loop mode on an end-to-end basis and disable the simplified Smith controller. It is an option. This "non-delayed" error signal is used to drive the predicted model which provides the non-delayed output. The output of this model can also be used to drive the robot vehicle in open-loop on an end-to-end basis. This is another option. These options were discussed in Chapter 3 and will be examined with the simulations and results presented in Chapter 5. While this "open-loop" control of the robot vehicle will

be shown to perform in an excellent manner, it must be recognized that feedback to the local terminal is necessary to at least update the display continuously.

Supervisory, or semi-autonomous control operation is accommodated in the system by the provision of a number of switches throughout the system designed to be activated by the operator from the local terminal. In this case, the operator can exit the control loop by releasing a switch on the control device and just sit back and monitor the system. This switch would effectively bypass the operator and insert an actual error detector in the local terminal and also provide a bypass of the human dynamics blocks in the simplified Smith controllers at the remote terminal. The end-to-end closed-loop control would still be maintained using the Smith control to compensate for the time delay. The operator could still intervene in the system by activating the same switch on the control device to activate teleoperation mode and directly enter speed and steering set commands. Besides inserting the operator back into the loop, the switch would provide a supervisory control command to re-insert the human dynamics compensation in the simplified Smith controllers.

5. SYSTEM SIMULATIONS AND RESULTS

5.1 Simulation Environment

All simulations were made using The Mathworks, Inc. software package Matlab, simulation package Simulink and their various toolboxes as follows:

Matlab	v. 5
Simulink	v. 2
Fuzzy Logic Toolbox	v. 2
Neural Networks Toolbox	v. 2

The system simulations were run using the general system configurations shown in Appendix A and B. Some runs were made on simplified versions for convenience, such as removing the system delays for non-delayed system tests.

All speed set inputs and speed outputs are normalized to the range [0 1]; steering set inputs and steering angle outputs are normalized to the range [-1 1], except in instances where the single-sided range [0 1] is used, simply due to left-right symmetry. All speed and steering set step inputs are unit step-functions, except where noted; speed set sinusoidal inputs have a peak-to-peak amplitude of unity and occupy the range [0 1]; steering set sinusoidal inputs have a peak amplitude of unity and occupy the range [-1 1].

5.2 System Performance with No Time Delay in the Loop

5.2.1 No Human Operator in the Loop

The basic criterion for appraising system performance over the time delay is a comparison with the system performance with no delays in the system and no human operator present. The step-function response of the steering angle to steering set inputs from 0.1 to 1 in increments of 0.1 is shown in Figure 5.1. This can be considered as turning to the right. Negative inputs for left turns will give the same response using even symmetry and is not shown. The linearity of the steady-state

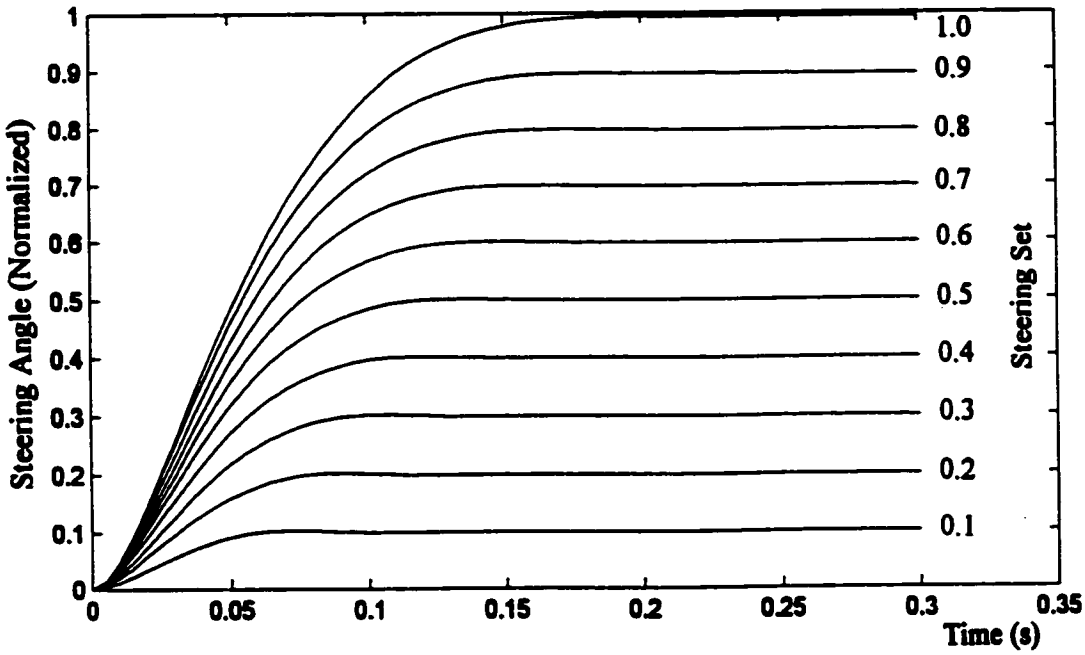


Figure 5.1. Steering angle response to step-function steering set inputs with no time delay and no human operator in the system.

output is apparent from the uniform spacing between the curves. Since we are using normalized values of steering angle as the desired steady-state response on the vertical axis, the accuracy of the output is readily apparent.

One of the main features of the control system design is the dependence of the speed on the steering angle. This has been described in section 4.4.3. Figure 5.2 shows this dependency clearly for maximum speed as a function of steering angle (using fixed increments of 0.1). The graph shows that the speed can remain at maximum until the steering angle reaches ~ +/- 0.25. At that point, the maximum speed begins to decrease. This reduction in speed with steering angle is relatively linear until it reaches its minimum value of ~ 0.32 at the maximum (positive or negative) steering angle. This is to ensure that sharp turns are made at appropriately safe speeds.

The values in the previous plot represent the steady-state condition. The unit step-function transient response is shown in Figure 5.3(a). While the steady-state

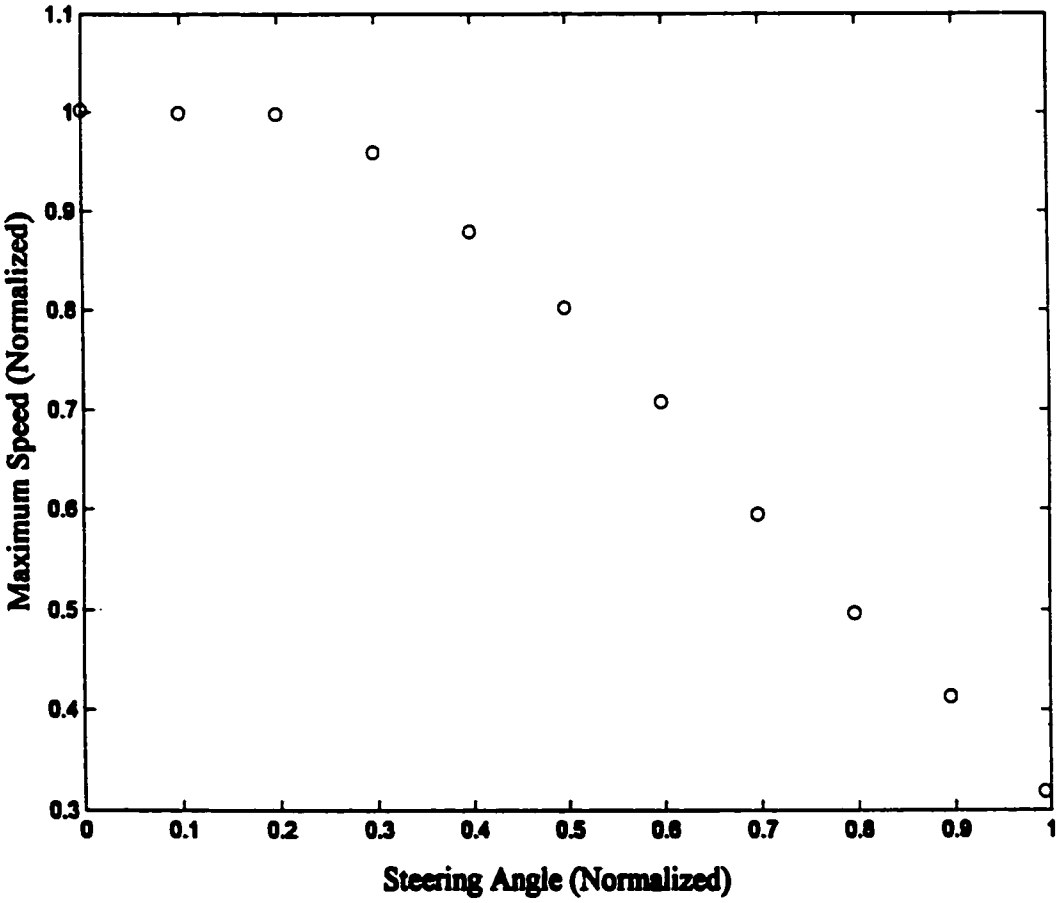
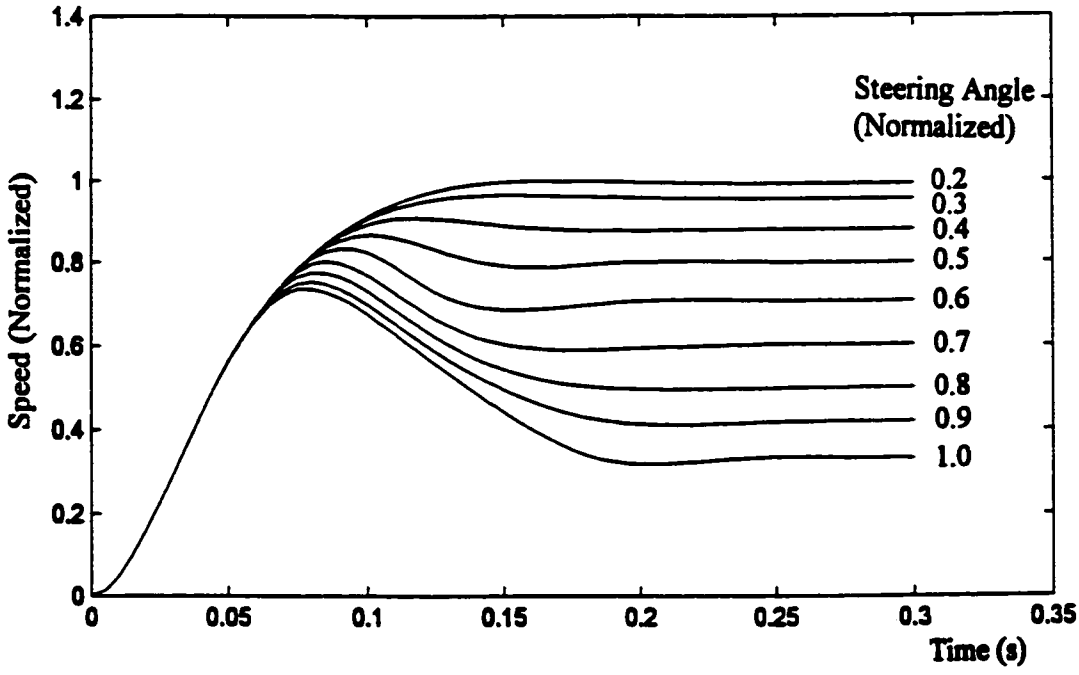
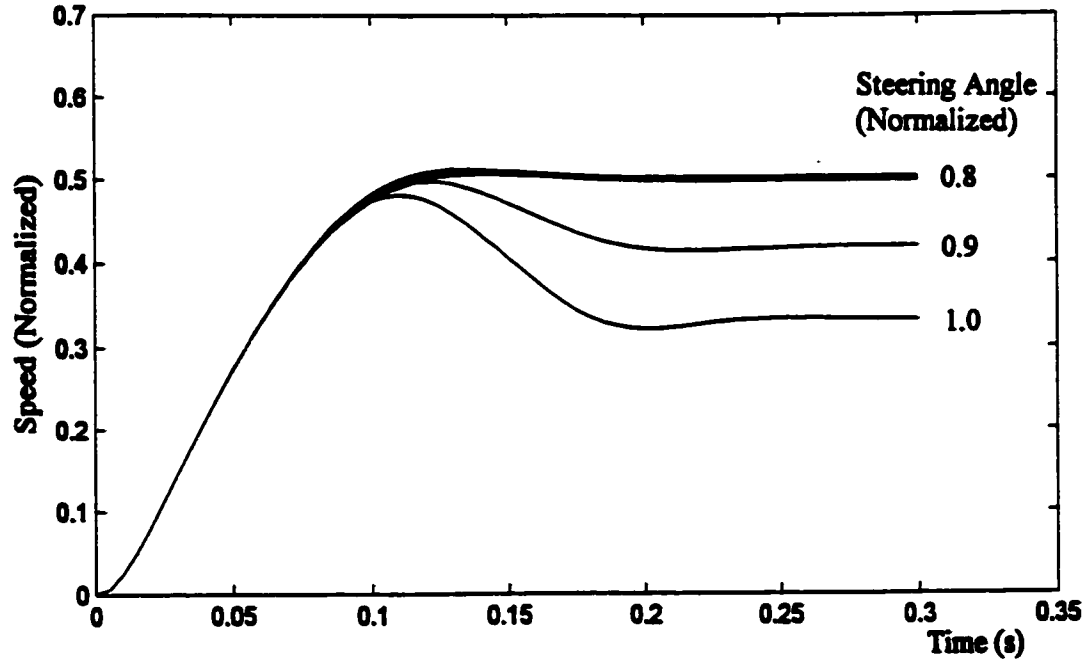


Figure 5.2. Maximum speed vs. steering angle for steady-state conditions.

values are given exactly as shown in Figure 5.2, the transient shows initial overshoot which is progressively worse as steering angle increases. What this indicates is the situation where the vehicle is stopped, the steering angle of the wheels is set at some angle, and then the speed is commanded to maximum as a step-function. If the wheels are initially turned to their maximum value, the speed will overshoot to over 0.7 before settling to its steady-state value of about 0.32. It must be noted that the system is multivariable and (nonlinear) fuzzy control is used in both the speed and steering control systems. This is the reason for the complexity of the response. Overshoot is not a desirable feature in a control system but no major effort has been undertaken to improve this response. The operational significance is for the operator to either reduce the steering angle before applying full speed or, more logically, increase speed more slowly. This should not be a problem in a practical system since the speed control can avoid step functions. Also, by reducing the speed set command to a lower value than



(a)



(b)

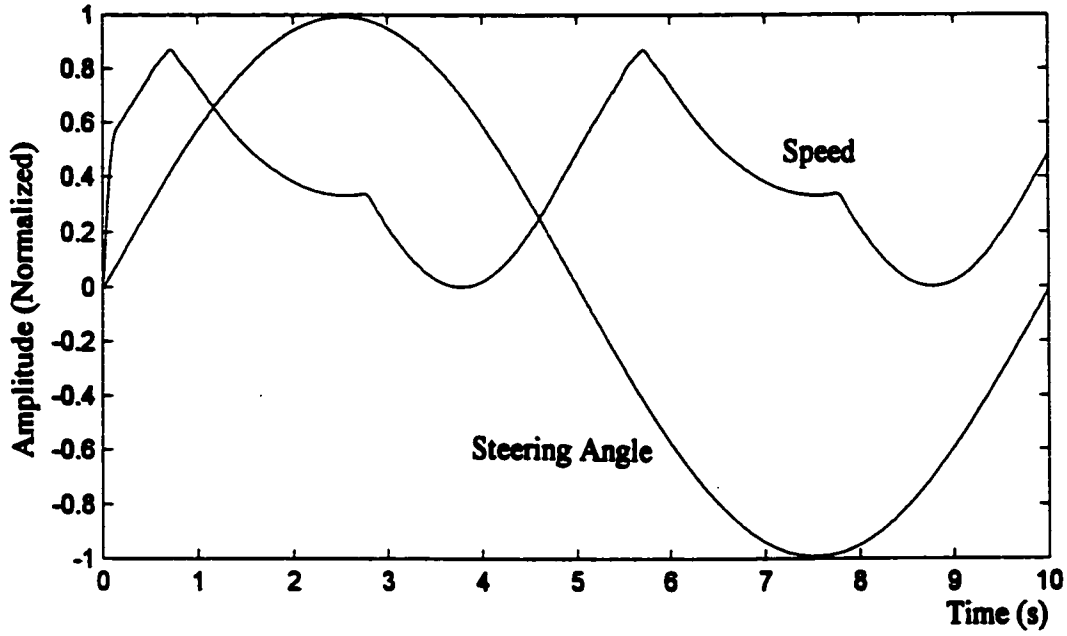
Figure 5.3. Speed response to step-function speed set inputs for various values of steering angle with no time delay and no human operator in the system: (a) unit step input, and (b) step input of 0.5.

1 (ie. full speed), the effect of the steering angle on the speed is lessened. Figure 5.3(b) shows the response of a step-function speed set command of 0.5, indicating dependency on the steering angle only at steering angles greater than 0.8.

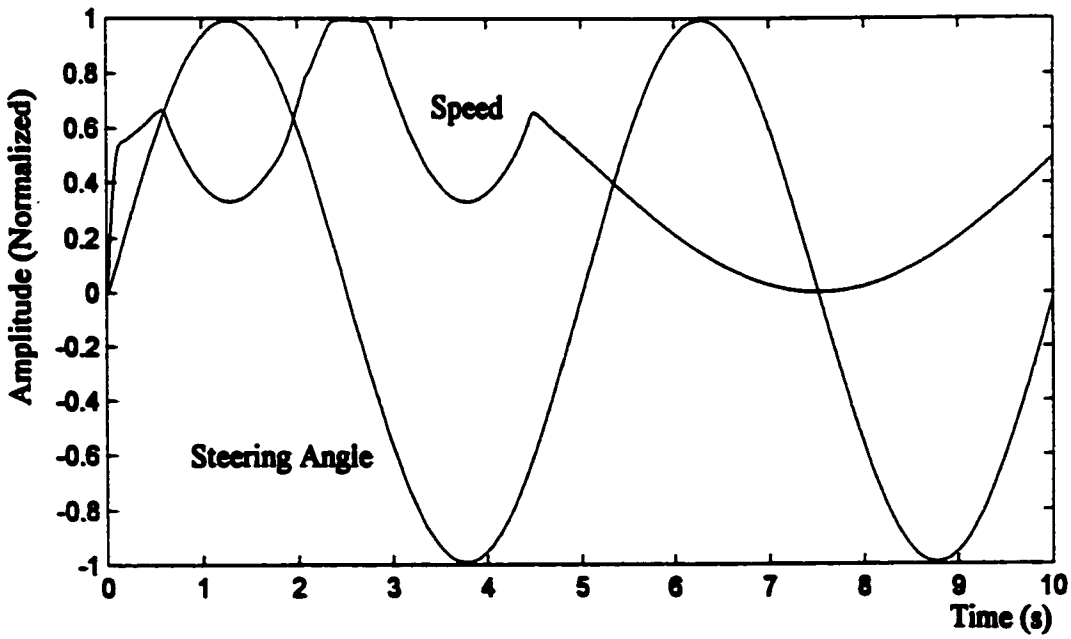
The dependence of the speed on the steering angle in a dynamic environment is illustrated in Figure 5.4. While this appears to be rather unusual at first sight, it is just intended to show the required dependence of the speed on the steering angle. These two examples illustrate the speed and steering angle responses when both the speed set and steering set inputs are varied from minimum to maximum sinusoidally. This would be an example of extreme operating condition. The frequencies of both signals have been chosen to be sufficiently low to not incur any decrease in amplitudes due to frequency response effects. The desired amplitudes, in the absence of speed dependency on steering angle, would occupy the full normalized ranges of [0 1] and [-1 1] for speed and steering angle, respectively. The dependency, however, requires that the speed be decreased with increased steering angle, and this is shown. The specific value that the speed is reduced to is not important, only that it be reduced sufficiently to safely accomplish the vehicle's turn. It is interesting to envision the resulting motion of the robot vehicle from these curves, showing the speed being reduced as the steering angle increases in both the negative and positive directions, and reaching high values only when the steering angle is at relatively low values.

5.2.2 Human Operator in the Loop

Teleoperation/telerobotics requires a human operator in the control loop. The human operator in the system monitors feedback status signals using an appropriate display device. Displays are generally categorized as being either compensatory-type or pursuit-type. The configuration used in the system simulations in this project (as shown in Appendix A) is essentially compensatory-type since the model of thehuman operator acts on the error between the desired response and the delayed actual system output fed back. It has been discussed earlier that the specific type of display is not of major importance because of the complexity of the human information-processing system. The model of the human operator dynamics chosen for this study has also been described earlier. This model includes a normal reaction delay time. However, it has also been shown earlier that given sufficient preview of the intended input signal as well as of the road ahead, an operator can essentially use precognitive control



(a)



(b)

Figure 5.4. Speed and steering angle responses to sinusoidal inputs of speed set with amplitude range [0 1] and to steering set with amplitude range [–1 1]: (a) speed set at 0.2 Hz and steering set at 0.1 Hz, and (b) speed set at 0.1 Hz and steering set at 0.2 Hz.

capability to cancel the reaction delay. For this reason, reaction delay has not been included in the system simulations even though it is easy to compensate for in the simplified Smith controller. The Matlab/Simulink models of the human operator in the Appendix include the reaction delays but they have been switched out in the simulations.

Initial results of system response tests with the model of the human operator in the loop but without compensation were disastrous. Response was ragged and totally unacceptable. Since simplified Smith control was to be included in the system to compensate for time delays, this concept was tested to overcome the problem caused by the human operator dynamics. The technique proved to be totally successful. By including the same dynamics in the Smith control loop as the loop delay, complete compensation was accomplished. The simplified Smith control system used in this project includes a model of the human (operator) dynamics.

Speed and steering angle responses have been repeated with the human operator in the loop and human dynamics compensation in the simplified Smith controller. Results are shown in Figures 5.5 and 5.6, respectively. The speed response is very similar to the results shown in Figure 5.3(a). The overall transient response is now controlled by the neuromuscular time constant of the human operator, taking approximately 1.5 seconds to reach steady state vs. about 0.2 seconds for the case without the operator in the loop. The speed/steering angle dependency is still apparent as is the overshoot. Likewise, the steering angle response is similar to that in Figure 5.1 except for the longer time to reach steady-state. Since the time constant of the human operator is controlling, the transient responses are essentially identical for both systems. It is apparent that there is no deleterious effect to the system using a human operator other than a change of time constant if the dynamics are compensated for, as they are by using the simplified Smith controller.

5.3 System Performance with Time Delay in the Loop

5.3.1 No Human Operator in the Loop

The problem of control whenever time delay is present in the system has been discussed throughout this thesis. For the remote-control system described herein, time delay is catastrophic. As an example, Figure 5.7 shows the step-function

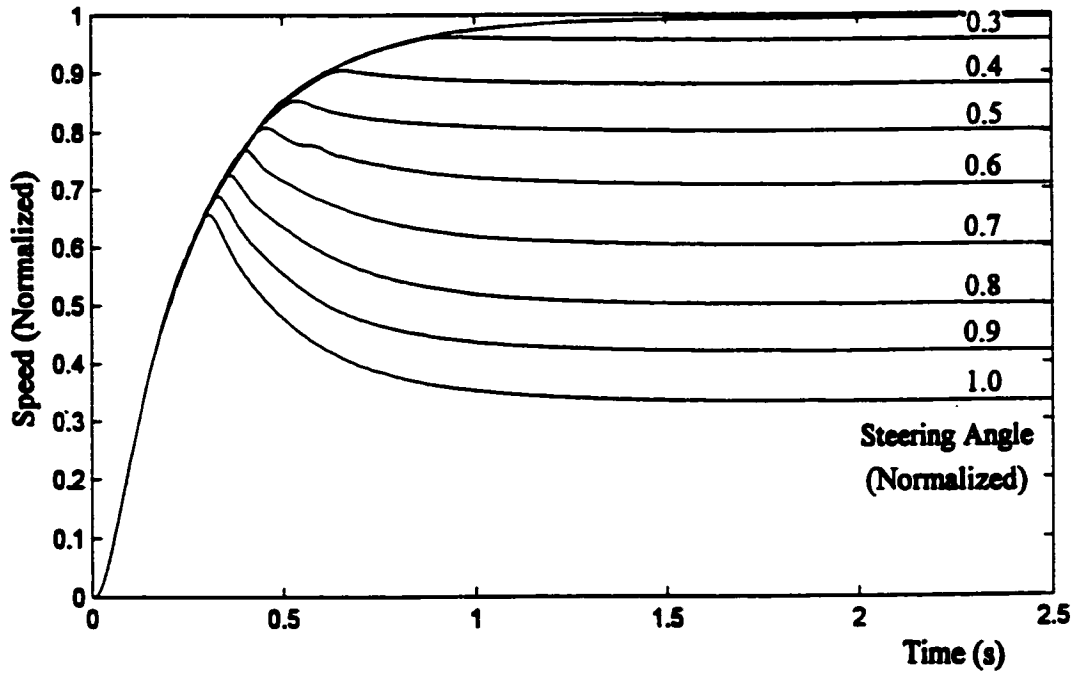


Figure 5.5. Speed response to unit step-function speed set inputs for various values of steering angle with human operator but no time delay in the system.

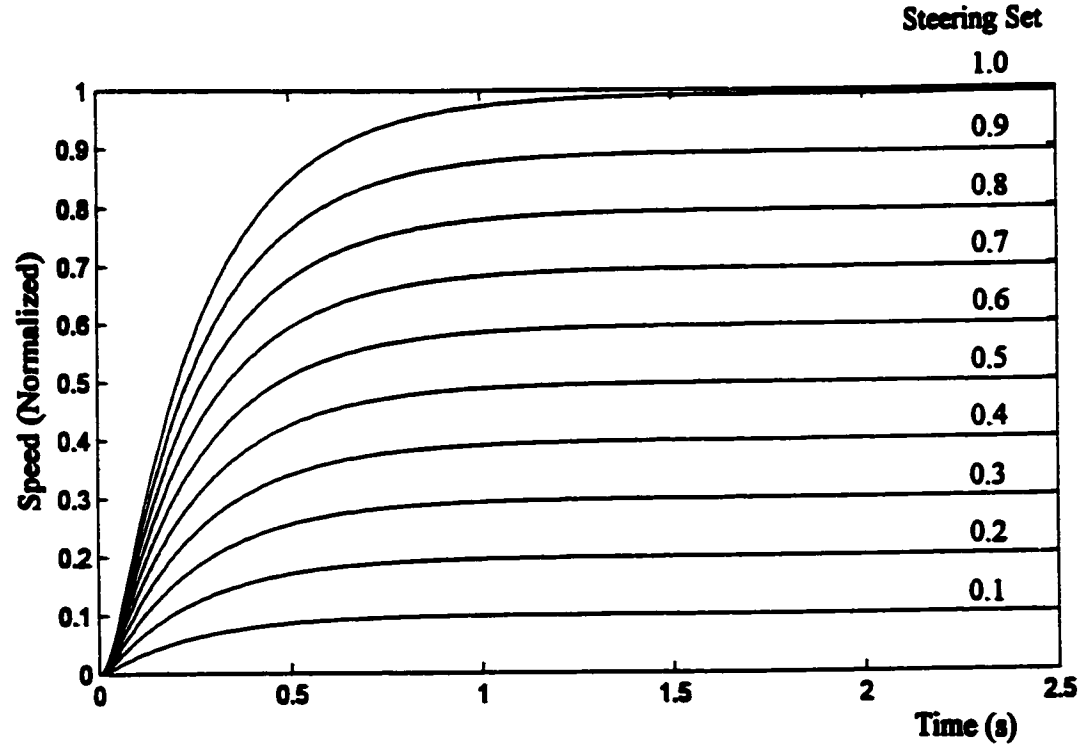
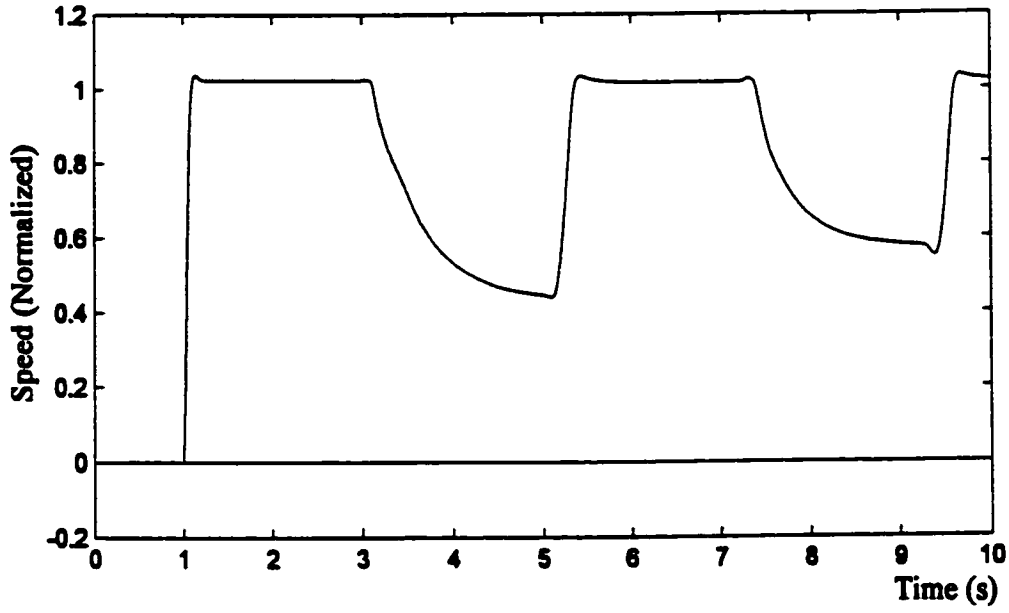
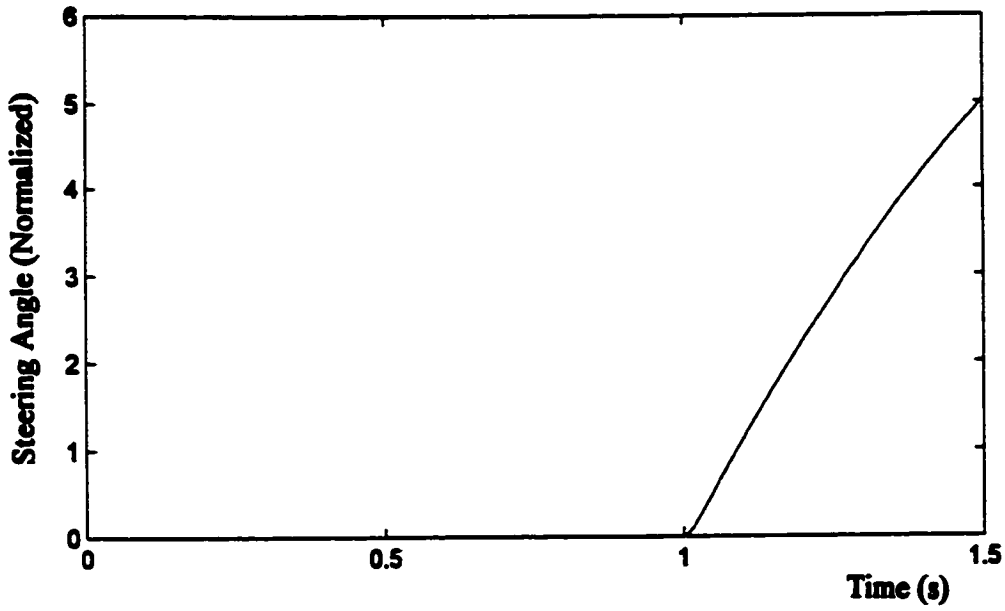


Figure 5.6. Steering angle response to step-function steering set inputs in increments of 0.1 with human operator but no time delay in the system.



(a)

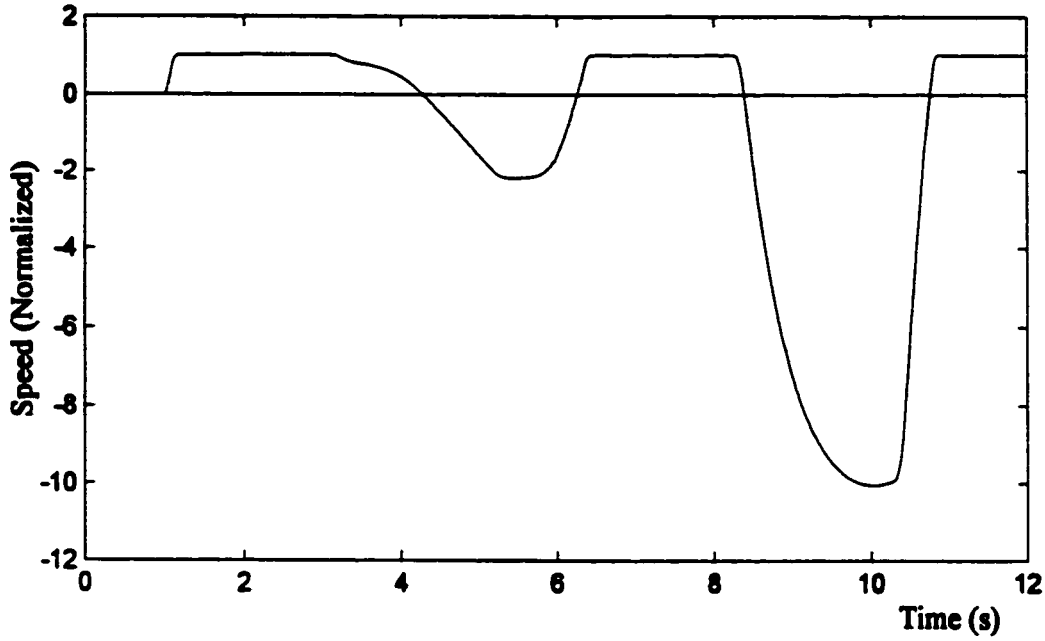


(b)

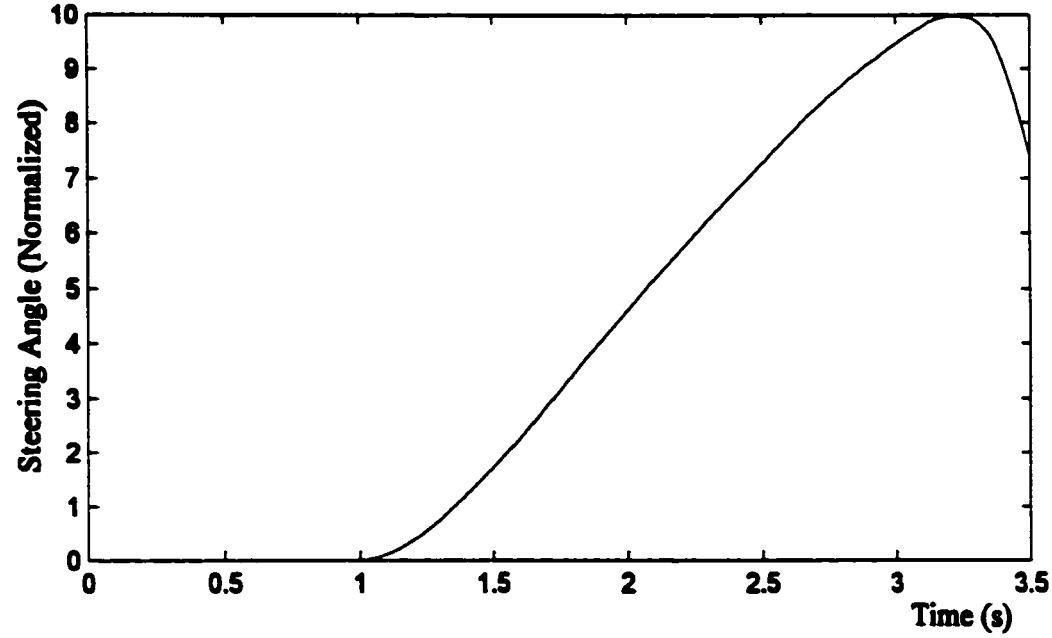
Figure 5 7. Step response with no human operator but with 1 second time delay in the system and no time-delay compensation: (a) speed response to unit step speed set input with steering set of zero, and (b) steering angle response to unit step steering set input with speed set of zero.

responses of both the speed and steering control systems, respectively, when a delay of 1 second is introduced into the system with no Smith control compensation. (All references to time delays in the remote-control system refer to the one-way system delay in both forward and reverse paths, unless total loop delay is specified). The speed response in Figure 5.7(a) seems fine until the 3 second point. This is when the output signal which is fed back to the input appears at the output again. From here on, the response deteriorates. The desired output (normalized) value is unity. Figure 5.7(b) shows the steering angle response rising rapidly beyond the desired output (normalized) maximum value of 1. The simulation was stopped manually at 1.5 seconds because the Matlab/Simulink simulation began issuing warnings due to going out of range. Figures 5.8 is included to show speed and steering angle responses to sinusoidal inputs at 0.1 Hz. The actual responses show little evidence of the expected sinusoidal output responses. Once again, the simulations were stopped manually when warnings began appearing on the screen. Clearly, time delay is a totally catastrophic problem for systems like this, making compensation absolutely mandatory.

By including the simplified Smith controller in each of the speed and steering control systems, time delay compensation is provided. Figure 5.9 shows the step-function responses of both speed and steering angle together for time delays of 0 seconds and 2 seconds, respectively. Due to technical difficulties encountered with simulations when using Matlab/Simulink, a non-zero delay of 0.001 s was found to be necessary in the system delay blocks for the case of no delay in the system. This has negligible effect on the results. It is obvious from these two plots that, in spite of the time delay, the responses are identical. The non-delayed speed response is identical to the response shown in Figure 5.3 with the different time scale and clearly illustrates the dependency of the speed on steering angle for the maximum normalized angle of (+/-) unity. The sinusoidal responses were also tested using the same values of time delay. The response of the speed control system to speed set input of 1 Hz is shown in Figure 5.10. These plots show that the responses are also identical except for the time delay. Similarly, the response of the steering control system to steering set inputs of 1 Hz shown in Figure 5.11 is unchanged regardless of the time delay. The horizontal lines at an amplitude of zero in Figures 5.11(a) and (b) is the speed response to the speed set input of zero. These plots all verify the excellent performance of the Smith control scheme to compensate for time delay in remote-control systems. How well this will work when a human operator is included in the loop will be examined next.

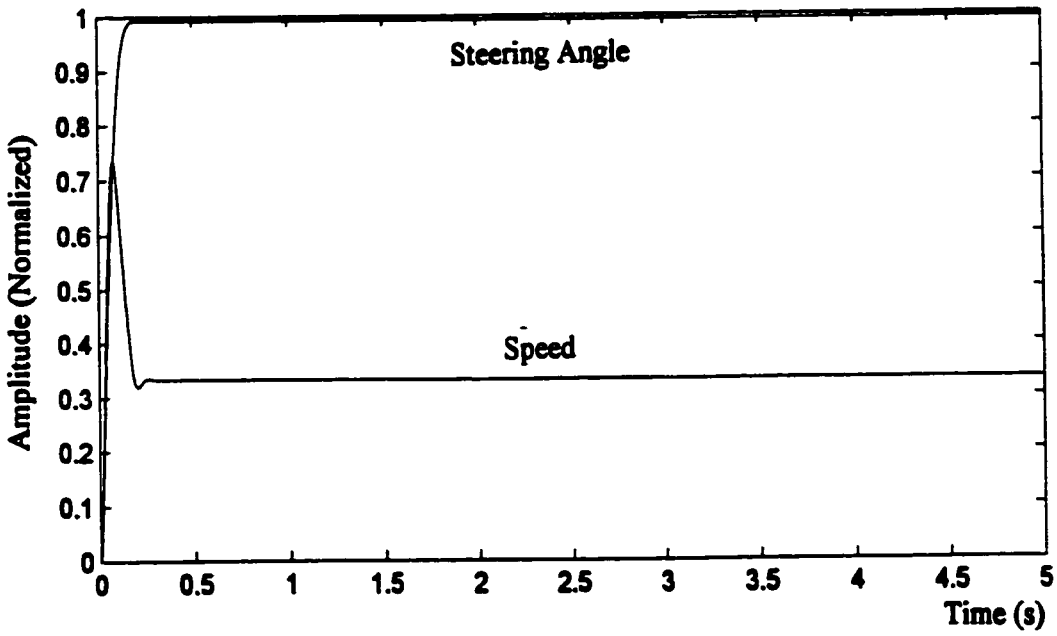


(a)

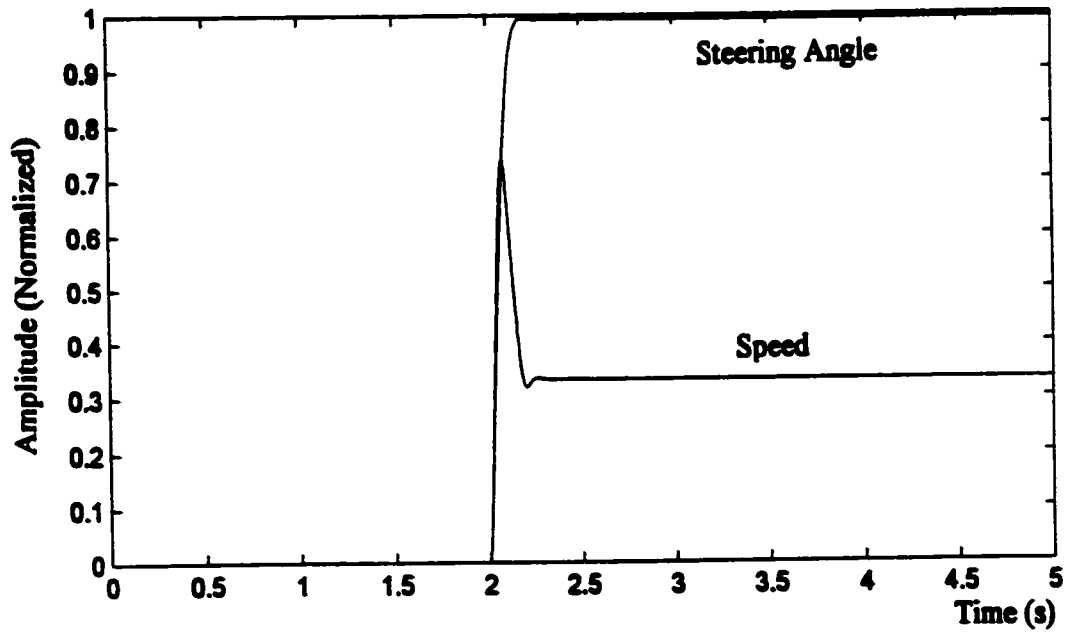


(b)

Figure 5.8. Sinusoidal response with no human operator but with 1 second time delay in the system and no time-delay compensation: (a) speed response to speed set input of 0.1 Hz and amplitude range [0 1] with steering angle of zero, and (b) steering angle response to steering set input of 0.1 Hz and amplitude range [–1 1] with speed of zero.

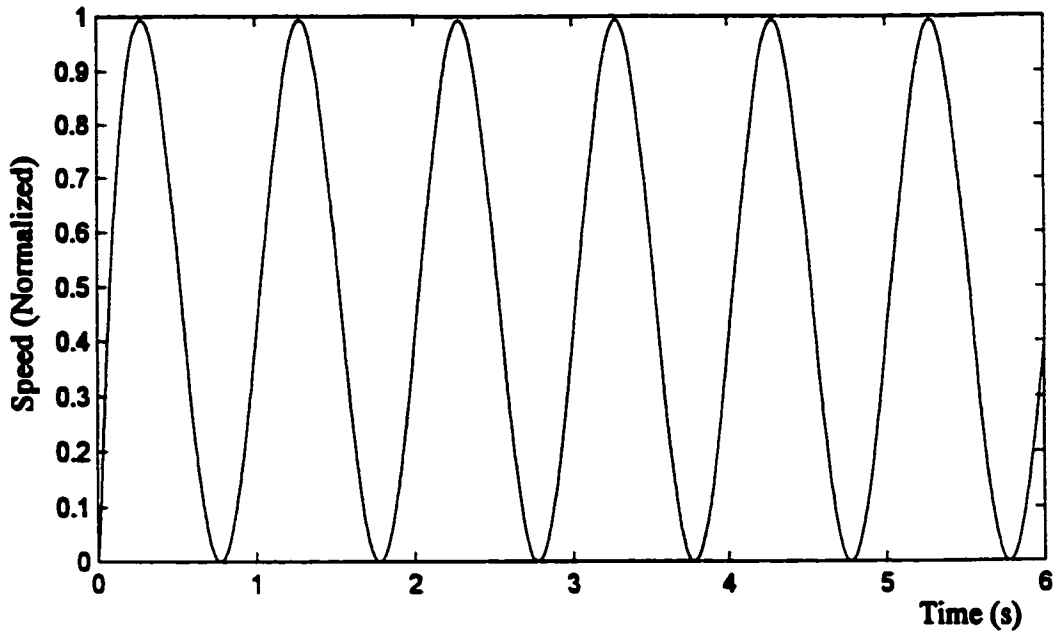


(a)

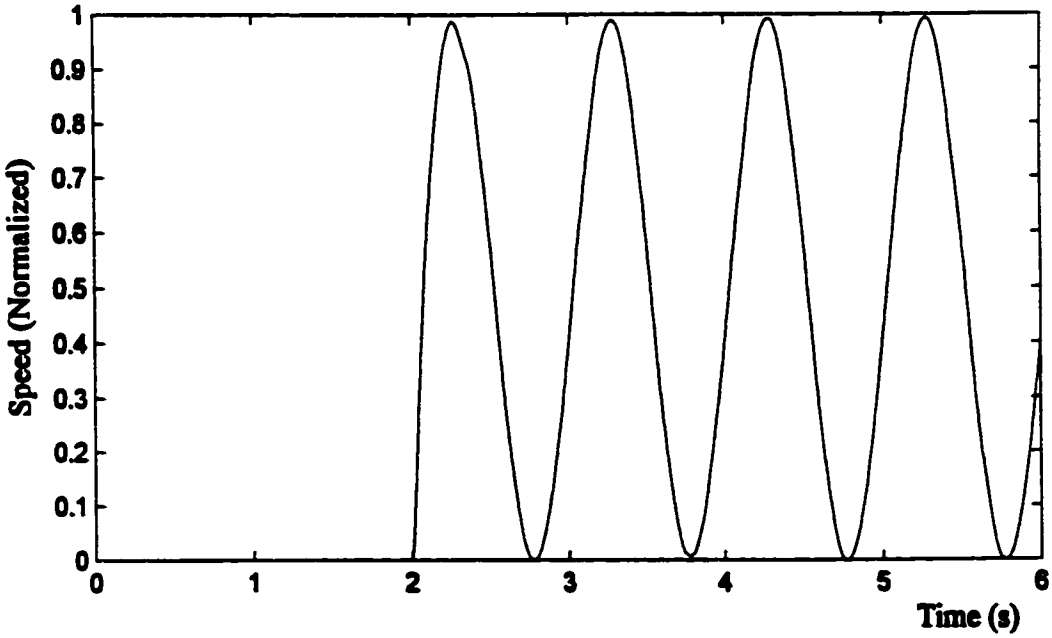


(b)

Figure 5.9. Speed and steering angle responses to unit step speed set and steering set inputs with no human operator: (a) zero time delay, and (b) 2 second time delay with time-delay compensation.

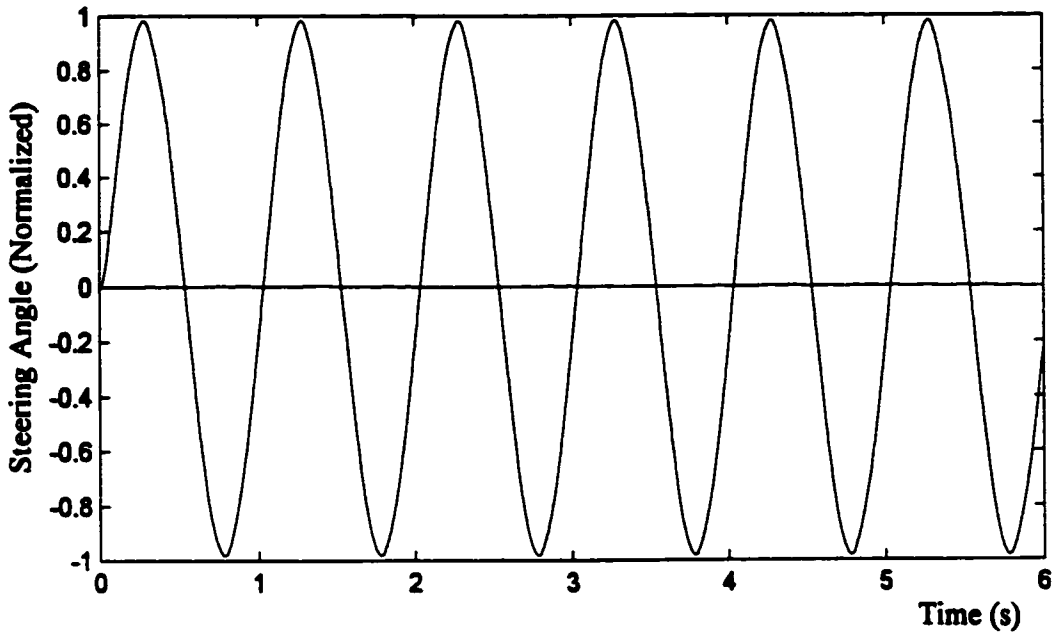


(a)

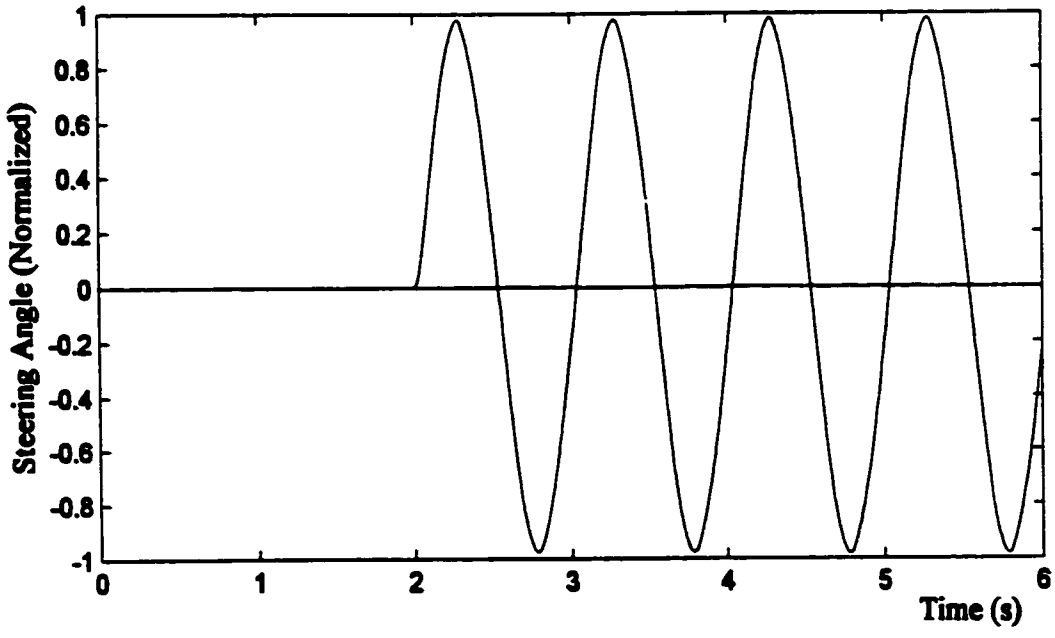


(b)

Figure 5.10. Speed response to 1 Hz sinusoidal speed set input of amplitude range [0 1] and with no human operator: (a) zero time delay, and (b) 2 second time delay with time-delay compensation.



(a)



(b)

Figure 5.11. Steering angle response to 1 Hz sinusoidal steering set input with amplitude range [-1 1] and with no human operator: (a) zero time delay, and (b) 2 second time delay with time-delay compensation.

5.3.2 Human Operator in the Loop

Before investigating system performance including the compensation of human operator dynamics, it is important to consider the case of having no compensation. The unit step-input responses of both the speed and steering control systems with a time delay of 1 second are shown in Figure 5.12. The ideal (or desired) response for each is a unit step. The neuromuscular time constant of the human operator model is evident in the transient response. The speed response curve looks excellent until the 3 second point, similar to Figure 5.7. However, just as in that case, the response deteriorates after 3 seconds. It does remain essentially at the right level but it is very irregular. The steering angle response is dramatically irregular. We can see the effect of the feedback as it repeats every 2 seconds (ie. the loop delay time). It should be noted that this response was allowed to finish despite Matlab/Simulink warnings after the 3 second mark. This is obviously not acceptable performance. It is time to include the compensation.

As soon as the model of the human dynamics is included in the Smith controller the responses clear up. Step-function responses of both speed and steering angle are shown in Figure 5.13 for time delays of 0 seconds and 2 seconds. These responses show the dependency of speed on steering angle in exactly the same manner and the same steady-state values as shown in Figure 5.9. The only difference is the time scale due to the human operator's dynamics. Figure 5.13(a) is the same response as shown in Figures 5.5 and 5.6 for the case when both speed set and steering set are unit step-functions. The speed and steering angle responses are again seen to be identical to one another despite any time delay in the system. Responses to sinusoidal inputs are shown in Figures 5.14 and 5.15 for the speed and steering angle, respectively, with time delays of 0 seconds and 2 seconds. Since the ideal speed and steering angle responses would occupy the full normalized ranges of [0 1] and [-1 1], respectively, a small amount of attenuation due to frequency response effects can be seen in the actual responses. This is due to the human operator's dynamics. Again, the responses are identical from one to another for each parameter. Finally, in Figures 5.16 and 5.17, speed and steering angle responses were plotted concurrently to show the speed dependency on the steering angle by using sinusoidal speed set and steering set inputs. Once again, as in Figure 5.4, the speed is seen to be reduced to a safe value as the steering angle increases in each direction. The small amount of attenuation due to frequency response effects can be seen again in the responses as in Figures 5.14 and 5.15. Two combinations of frequencies were used: speed set and steering set inputs each at 0.15 Hz and then with speed set at 0.1 Hz with steering set at 0.15 Hz. These

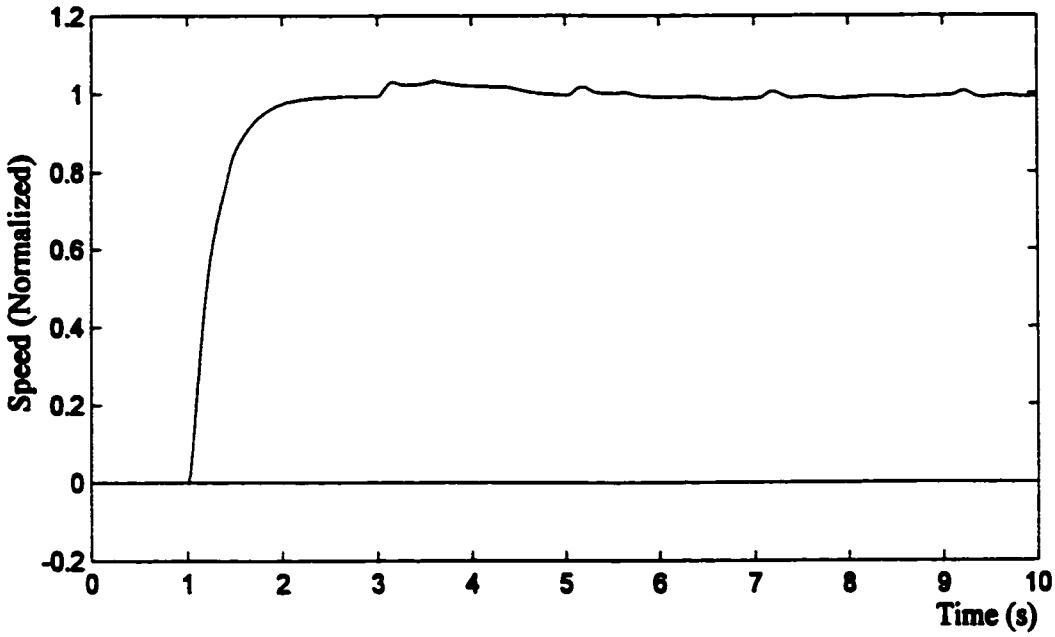
two tests were plotted at time delays of zero seconds and 5 seconds in Figure 5.16 and zero seconds and 2 seconds in Figure 5.17. Once again, the time delay is seen to have no effect on the system responses. All simulations involving time delay include delay compensation using Smith control unless otherwise indicated.

5.4 System Performance Using Predictor

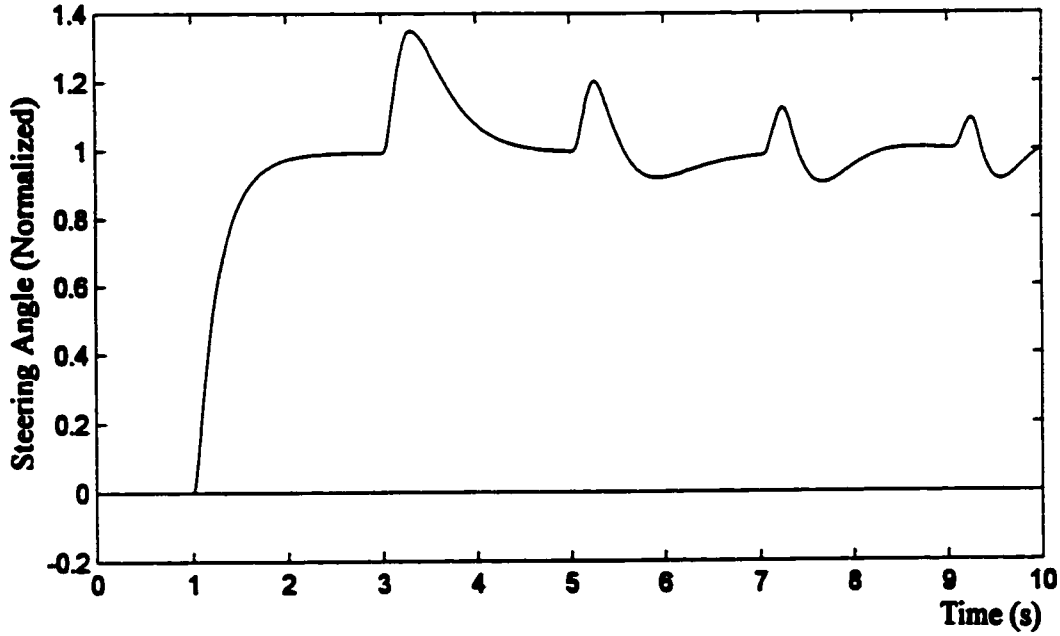
5.4.1 General

The predictor is designed to provide the human operator with an immediate estimate of the output of the system during teleoperation mode as if there was no time delay in the loop. This should allow the operator to drive the robot vehicle in simulation while the same control signals are transmitted to the remote terminal to drive the actual vehicle. (Alternatively, the output of the predictor may be used as control signals to the remote terminal.as discussed in Chapter 3.) Obviously, this process will work perfectly only with perfect modeling of the vehicle in its environment. Nevertheless, the usefulness of predictors in time-delay control applications has been proven beyond question as discussed in Chapter 2.

The predictor used in this project is a 2-input, 2-output neural network. The inputs are speed and steering command signals (ie. speed set and steering set); the outputs are the predicted speed and steering angle of the robot vehicle. Ideally, the predictor should behave exactly as the real (non-delayed) system, representing the input-output mapping perfectly, regardless of initial conditions or combination of input values and including the complete transient response to steady state. This was the goal for the neural network. This ideal was not realized. After a considerable effort of testing various neural network models using various configurations, the transient behavior of the predictor could not be realized. This prompted another approach. Since the time constants of this system are controlled by the dynamics of the human operator, it was decided to model the neural network using the steady-state conditions of the system. It was already known that training the neural network in this capacity was feasible. Tests would prove the soundness of this concept.

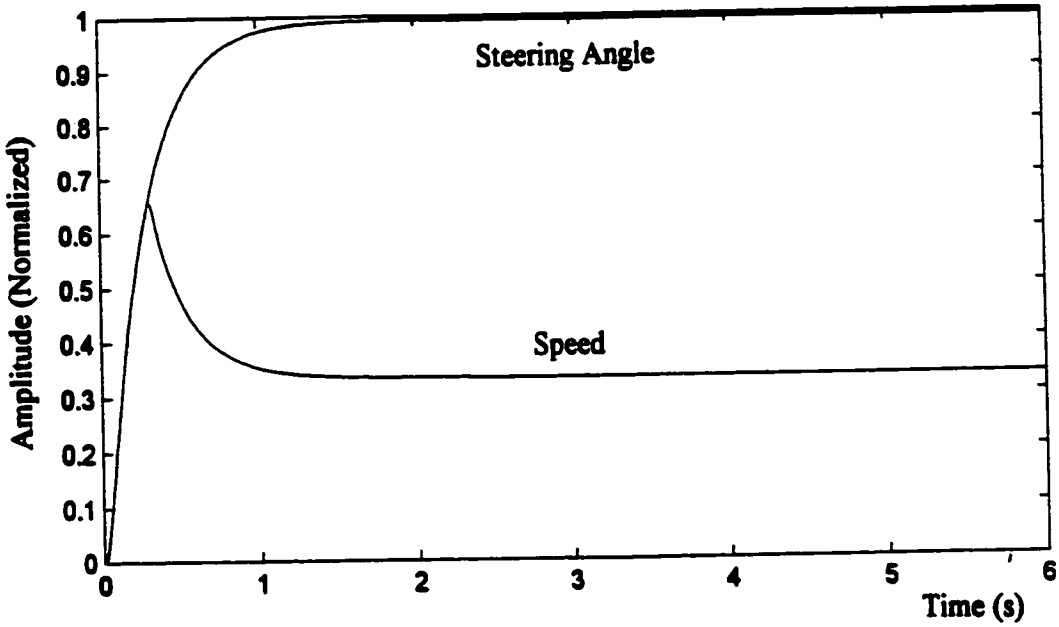


(a)

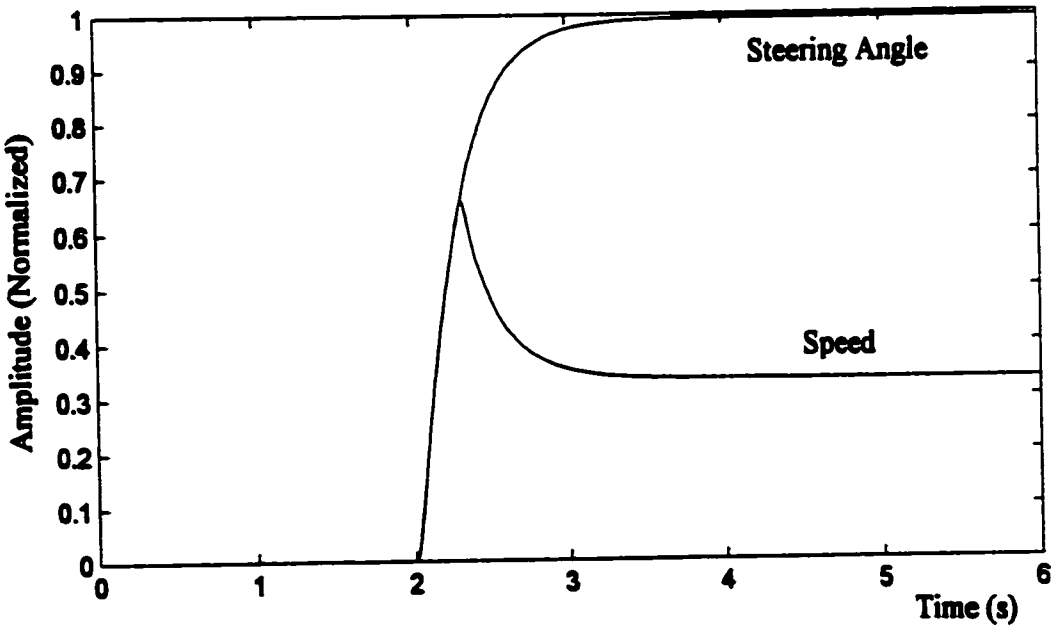


(b)

Figure 5.12. Step response with human operator and 1 second time delay in the system but with no human dynamics compensation: (a) speed response to unit step speed set input with steering set of zero, and (b) steering angle response to unit step steering set input with speed set of zero.

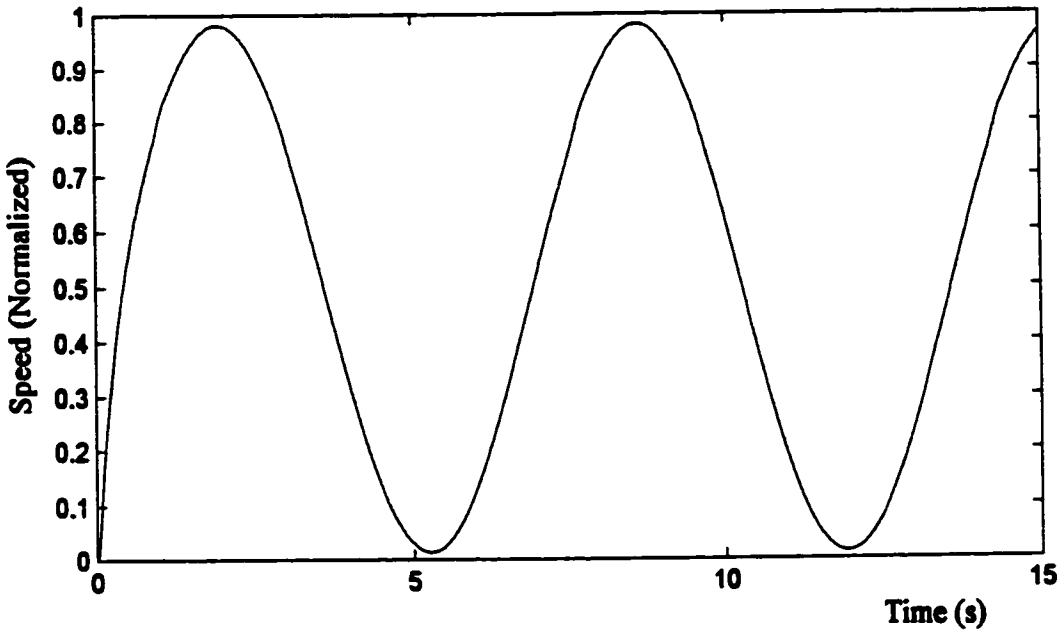


(a)

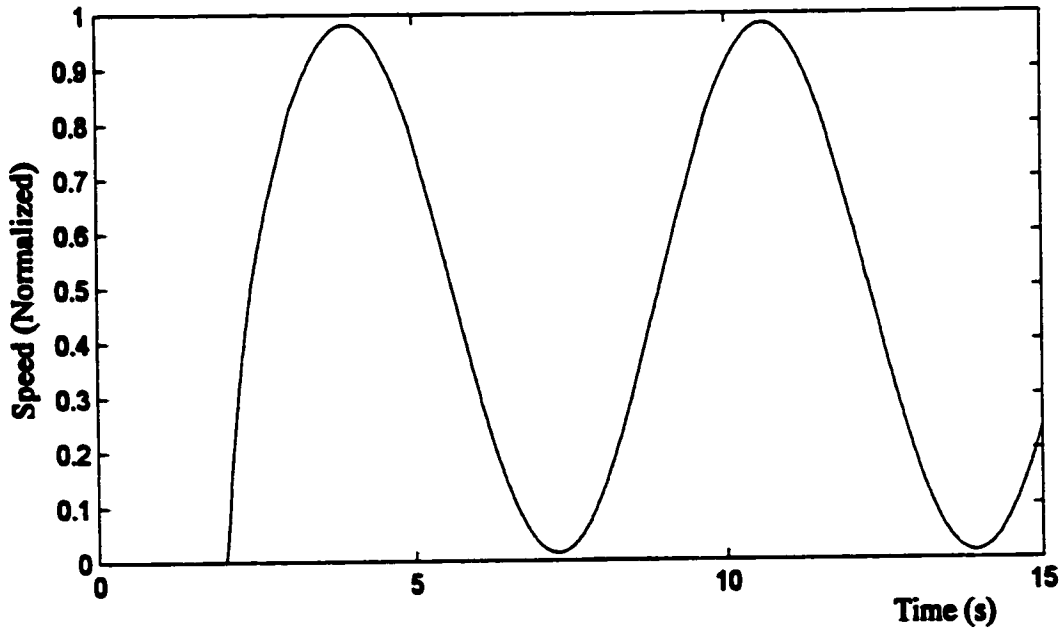


(b)

Figure 5.13. Responses of speed and steering angle to unit step speed set and steering set inputs with human operator in the loop, and with human dynamics compensation: (a) zero time delay, and (b) 2 second time delay.

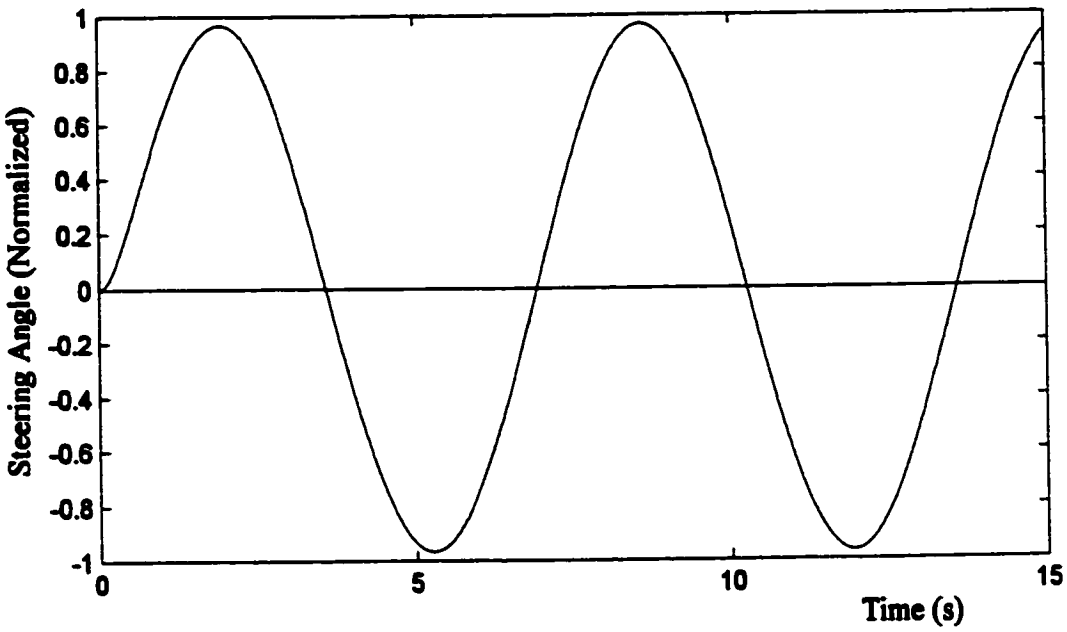


(a)

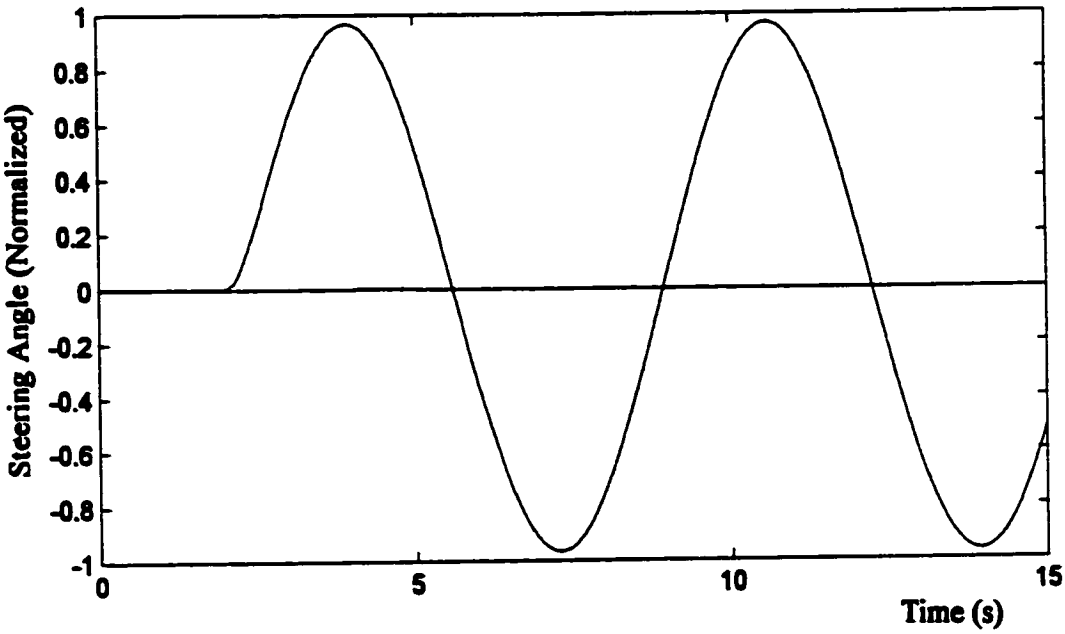


(b)

Figure 5.14. Speed response to 0.15 Hz sinusoidal speed set input of amplitude range [0 1], with human operator in the loop, and human dynamics compensation: (a) zero time delay, and (b) 2 second time delay.

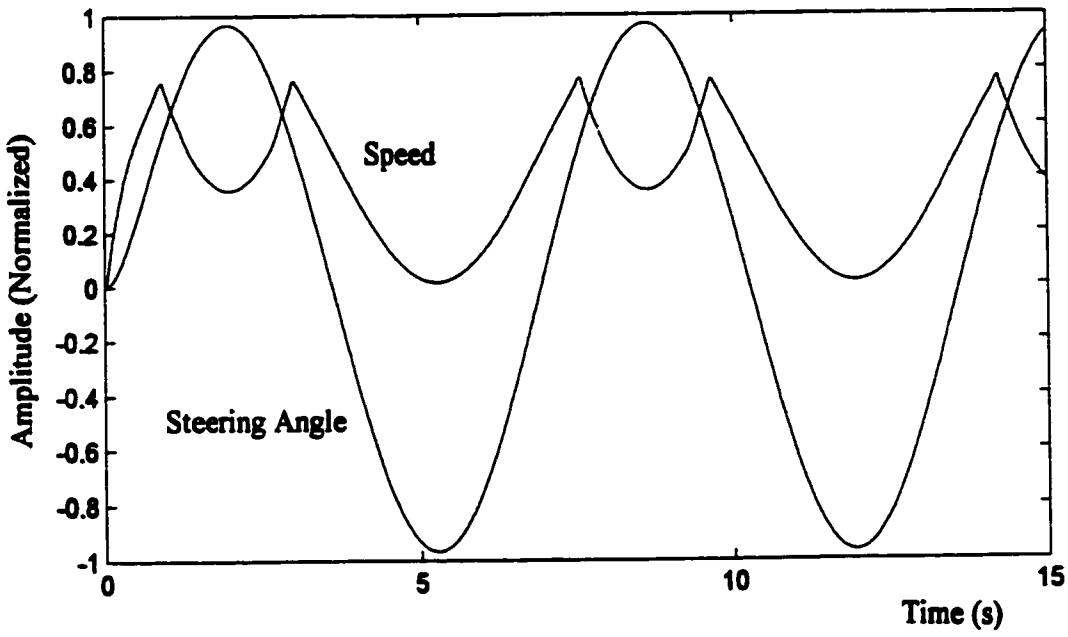


(a)

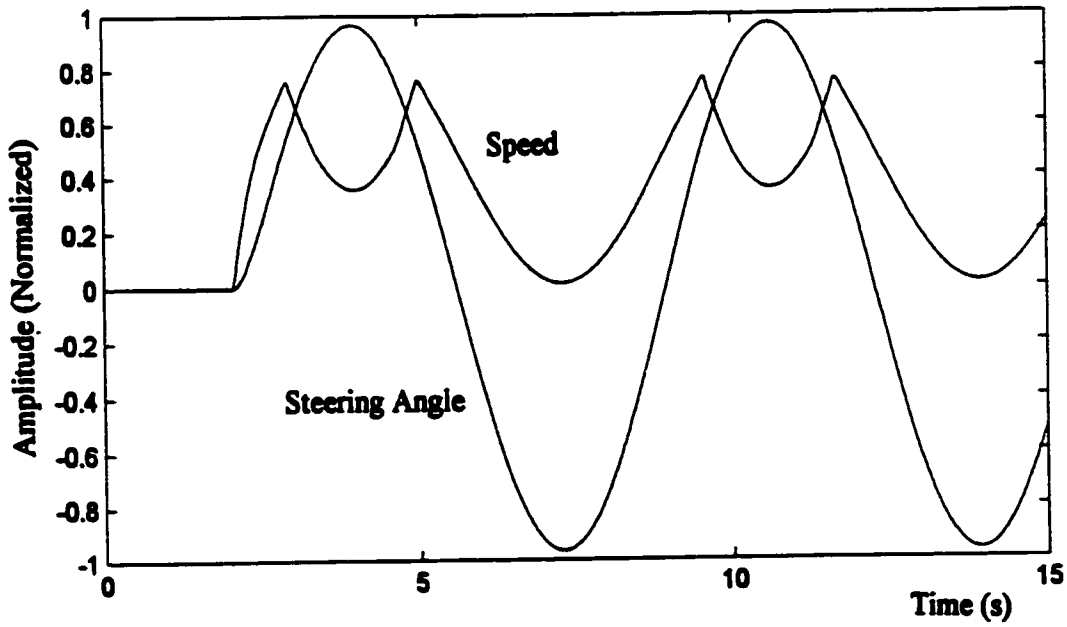


(b)

Figure 5.15. Steering angle response to 0.15 Hz sinusoidal steering set input with amplitude range [-1 1], with human operator in the loop, and human dynamics compensation: (a) zero time delay, and (b) 2 second time delay.

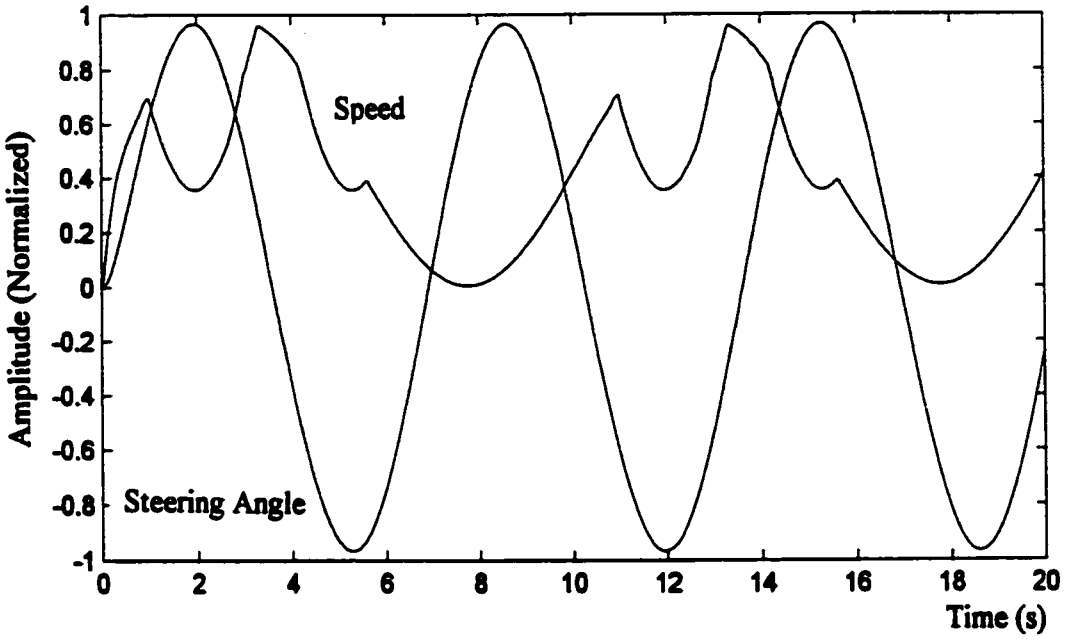


(a)

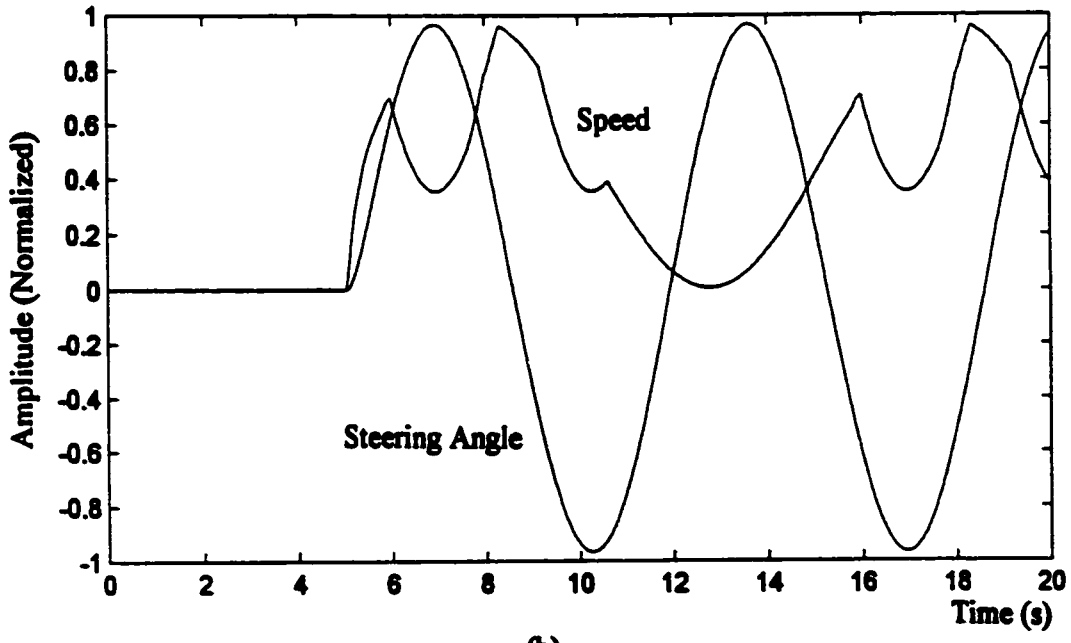


(b)

Figure 5.16. Responses of speed and steering angle to 0.15 Hz sinusoidal speed set and steering set inputs of amplitude ranges [0 1] and [-1 1], respectively, with human operator in the loop, and human dynamics compensation: (a) zero time delay, and (b) 2 second time delay.



(a)



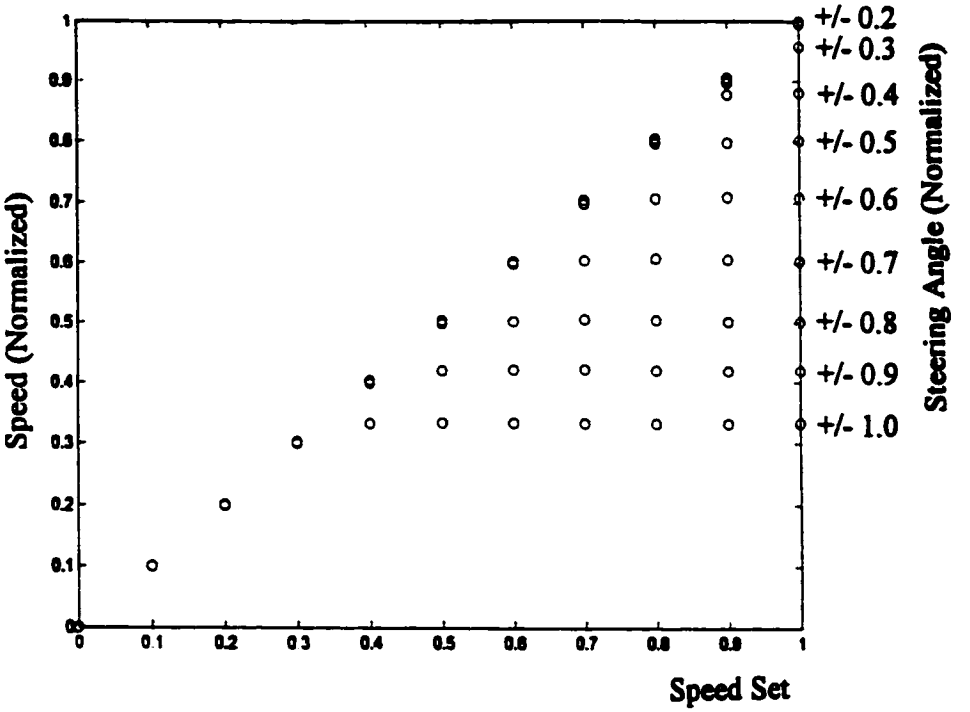
(b)

Figure 5.17. Responses of speed and steering angle to sinusoidal speed set input of 0.1 Hz with amplitude range [0 1] and to sinusoidal steering set input of 0.15 Hz with amplitude range [-1 1] with human operator in the loop and with human dynamics compensation: (a) zero time delay, (b) 5 sec time delay.

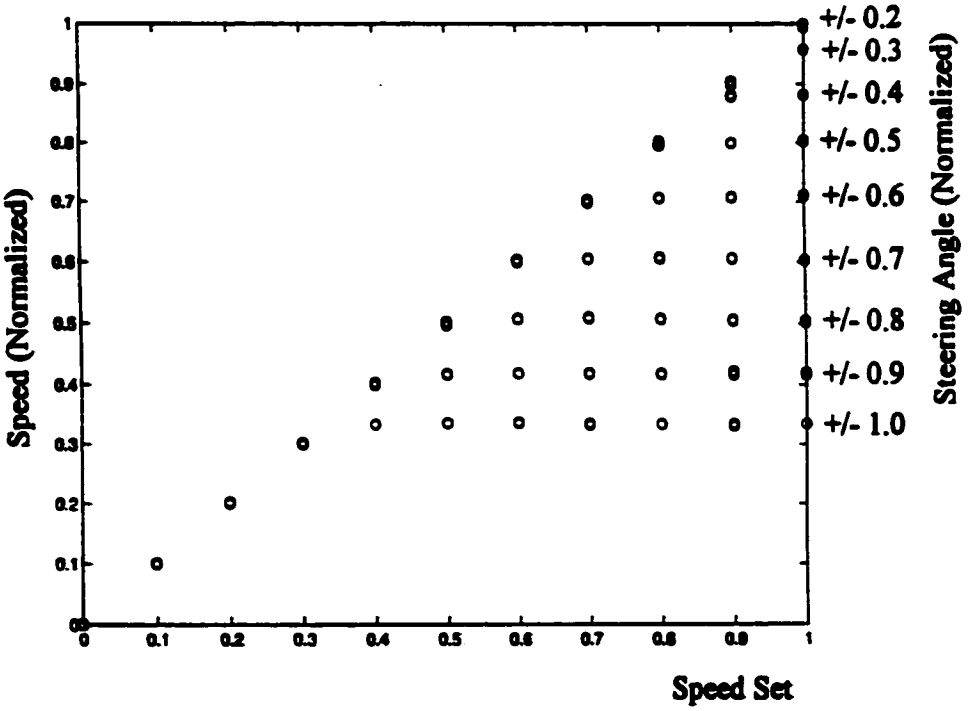
Input pattern and output target data for training the neural network were generated using the actual system with input speed set and steering set commands over their full ranges of [0 1] and [-1 1], respectively, at increments of 0.1. This resulted in a speed/steering angle matrix of [11 x 21] or 231 points forming the base of the 3-dimensional control surface. The neural network provides the interpolation (or generalization) between the output data points to create a continuous control surface. The predictor is a model of the complete input-output system. As test results of the predictor will verify, these points are sufficient to train the network to a high degree of accuracy over the entire surface.

A feed-forward backpropagation network was chosen using an 8,8,2 neuron, fully-connected, three-layer configuration. Matlab's "newff" simulation software from The Mathworks' Neural Network Toolbox was used. The first two layers used tan-sigmoid activation functions with the third layer being linear. The network was trained for 1000 epochs and the accuracy attained was a mean-squared-error of 1.26×10^{-6}. The resulting speed and steering angle target responses are shown in Figures 5.18 and 5.19, respectively, compared with their training data patterns. The match is virtually perfect.

Two system configurations using the predictor were tested. In the first, or basic configuration, as shown in Appendix A, the predictor was driven in parallel with the actual system by the speed set and steering set inputs. Models of human operator dynamics were used to provide the necessary transient system responses since the predictor is based on steady-state values. The second, or alternate configuration, shown in Appendix B, is based on Figure 3.15 in which the output of the predictor was used in series to drive the actual robot vehicle. This has been discussed in section 3.3. Appendix B includes only those portions of the system which are changed from the system in Appendix A. The scaling gain of 0.4 for the speed set input is not required in the local terminal since its effect has been included in the neural network model. The predicted output value from the model is then correct. However, when using the predictor to drive the remote terminal, the gain is still required in the speed set command path, since the predicted output is now effectively an input to the original system, which requires the gain. For convenience, it is included in the block diagram of the remote teleautonomous controller. This configuration is open-loop on an end-to-end basis, so Smith control is not required.

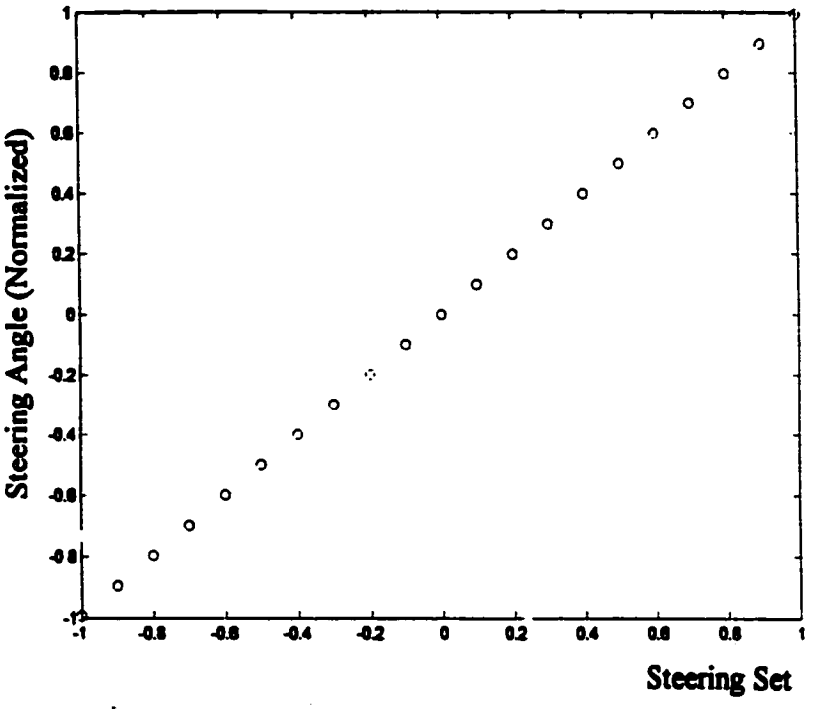


(a)

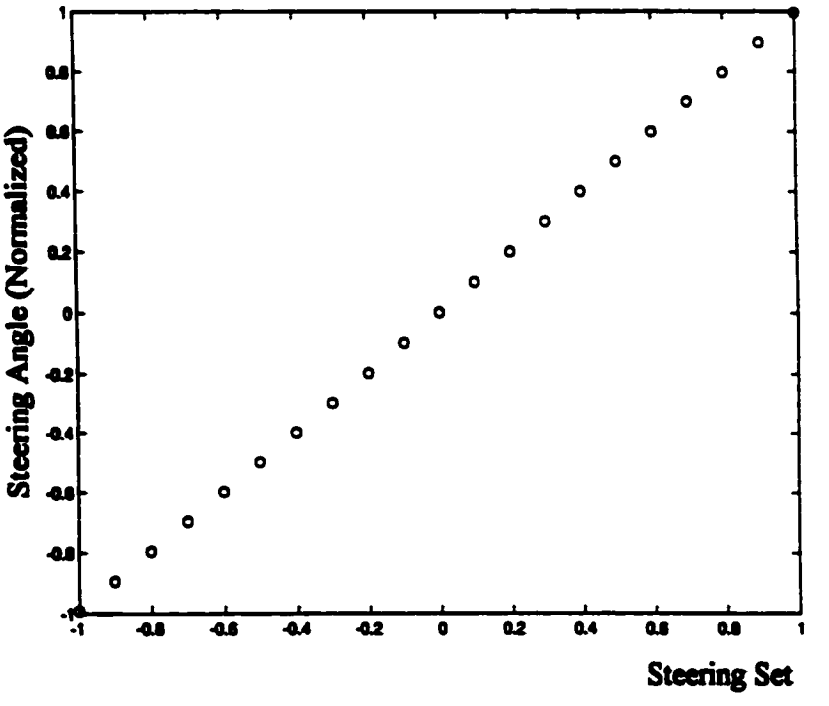


(b)

Figure 5.18. Steady-state speed vs. speed set for step increments of 0.1 and for fixed steering angles from zero to +/- 1.0: (a) actual system, and (b) predictor.



(a)



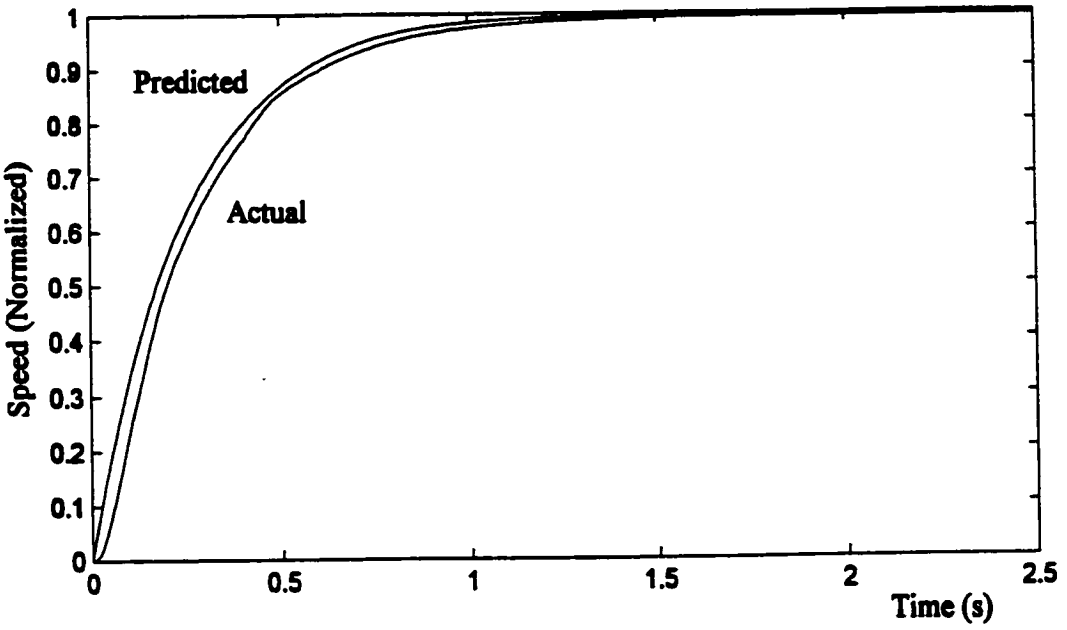
(b)

Figure 5.19. Steady-state steering angle vs. steering set for step inputs at increments of 0.1: (a) actual system, and (b) predictor.

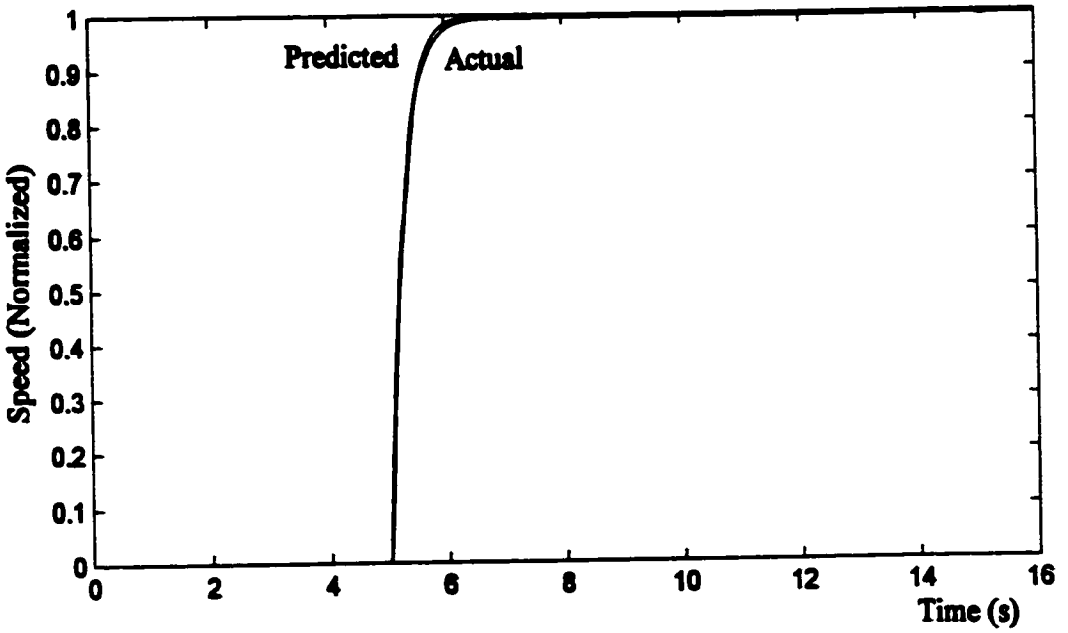
5.4.2 No Predictor in the Loop

The predictor was installed in the local teleautonomous controller as shown in Appendix A for the first (ie. parallel) configuration and system response tests were performed. Step input response for speed is shown in Figure 5.20 for the cases of zero delay and 5 seconds delay. The expanded scale for the zero delay response shows the very small discrepancy between the actual system response and the predicted. This appears to be less than 0.05 seconds, obviously of no concern at all in a practical teleoperation system. This discrepancy is due to the transient response of the actual system; the predicted response is based on steady-state outputs only. Both actual and predicted responses include the same human operator dynamics. For the case of 5 seconds delay, the compressed scale does not show the 0.05 second difference. There is only a slightly discernible discrepancy as the response nears the steady-state value. Figure 5.21 shows the step input response for steering angle for zero delay and 5 second delay. Again, only the expanded scale for the zero delay response shows the small discrepancy of less than 0.05 seconds between the actual system response and the predicted. At 5 seconds delay, there is no discernible difference. These results verify the exceptional accuracy of the predictor.

Sinusoidal responses were also plotted at zero delay and 5 second delay for both the speed and steering angle. These are shown in Figures 5.22 and 5.23, respectively. Once again, the accuracy of the predictor reveals only a slight difference between the actual and predicted responses for the speed. There is no discernible difference between the actual and predicted responses for the steering angle, at least at the scale shown. Two final tests were run to test the dependency of the speed on the steering angle using the predictor with a 5 second delay in the system. Figure 5.24(a) shows the results of the speed response when both speed set and steering set inputs use a 0.15 Hz sine wave. Figure 5.24(b) shows the same response using 0.1 Hz for speed set and 0.15 Hz for steering set. Again, the discrepancy between actual system and predicted is barely discernible.

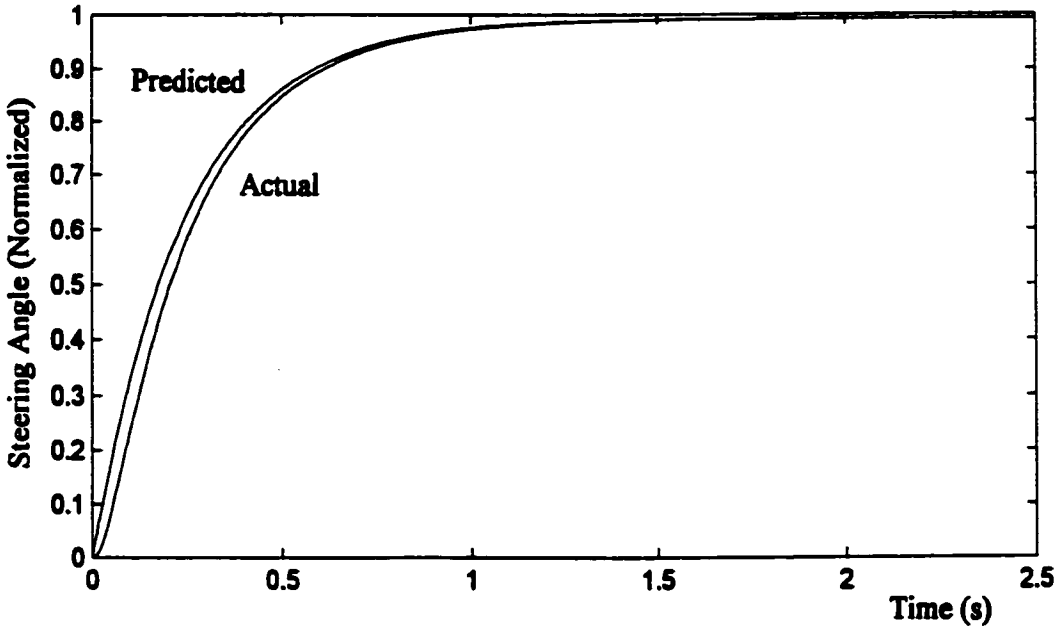


(a)

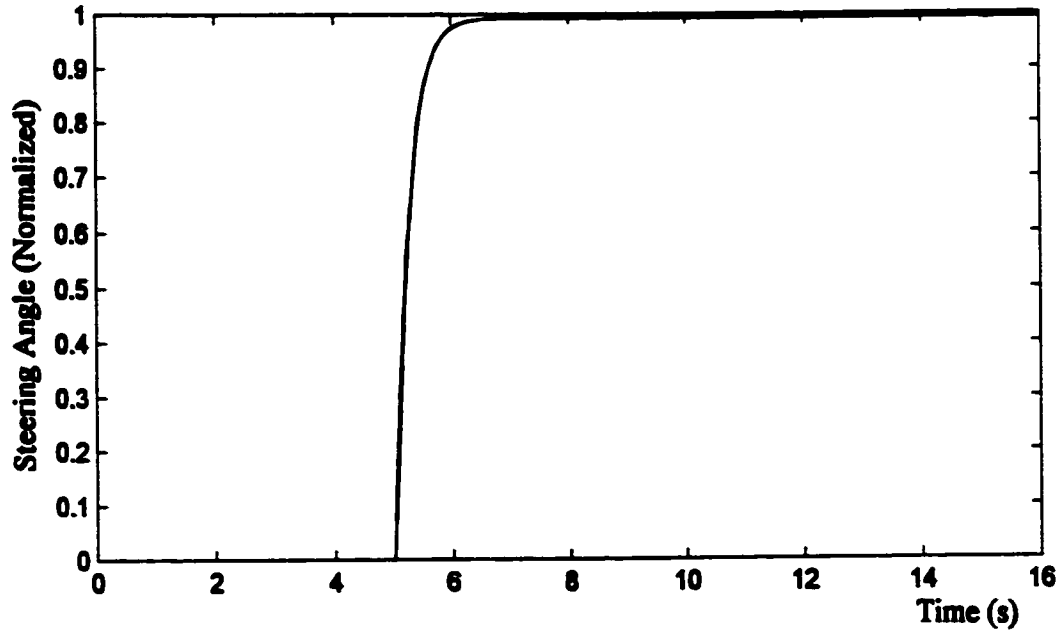


(b)

Figure 5.20. Speed response to unit step input for actual system and predicted: (a) zero time delay, and (b) 5 second time delay.

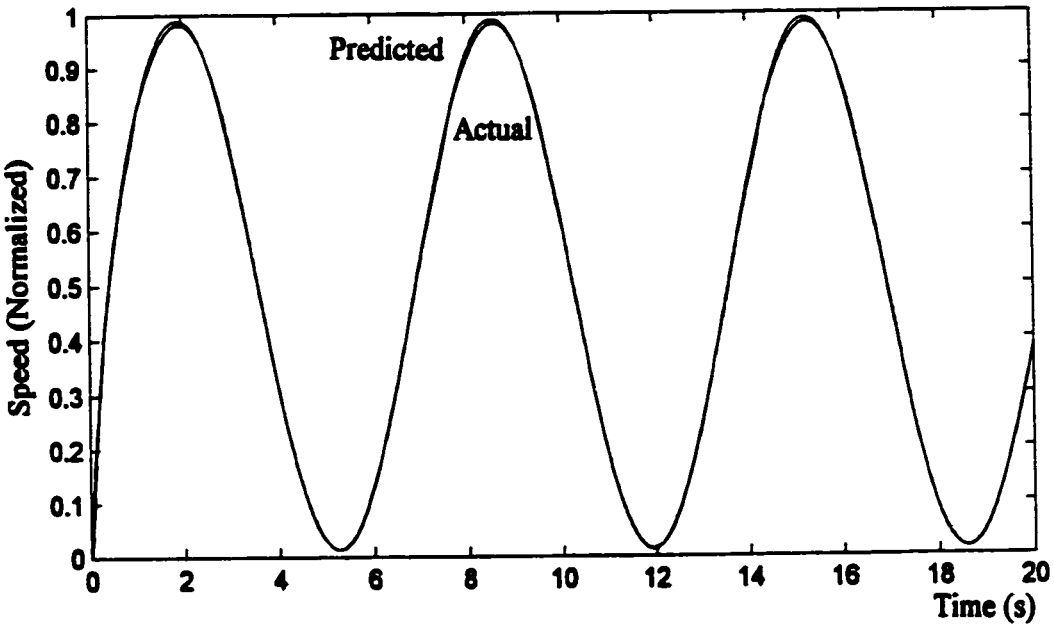


(a)

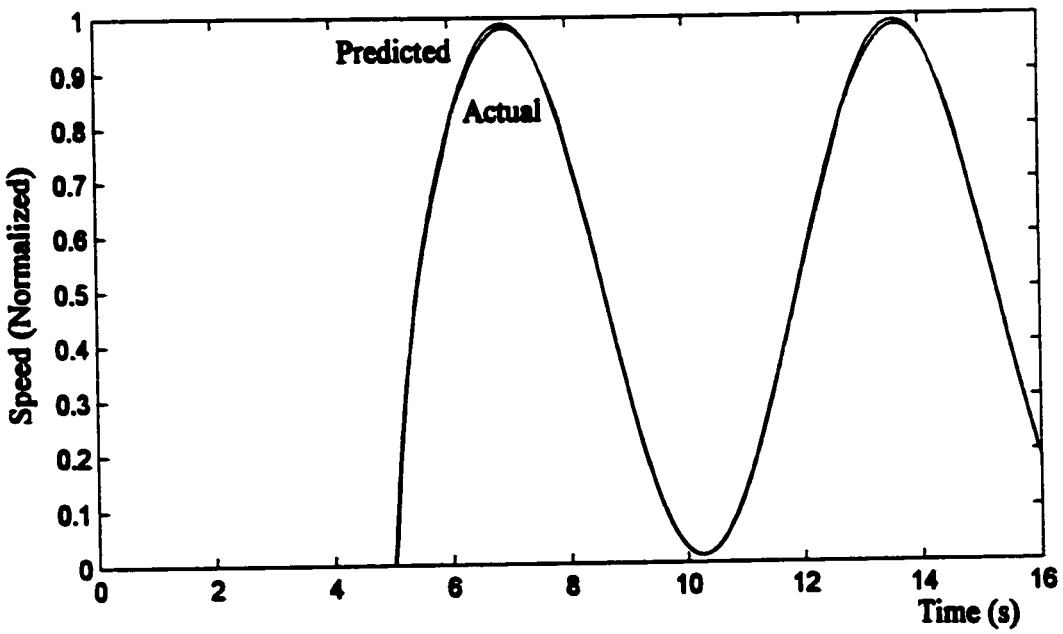


(b)

Figure 5.21. Steering angle response to unit step input for actual system and predicted: (a) zero time delay, and (b) 5 second time delay.

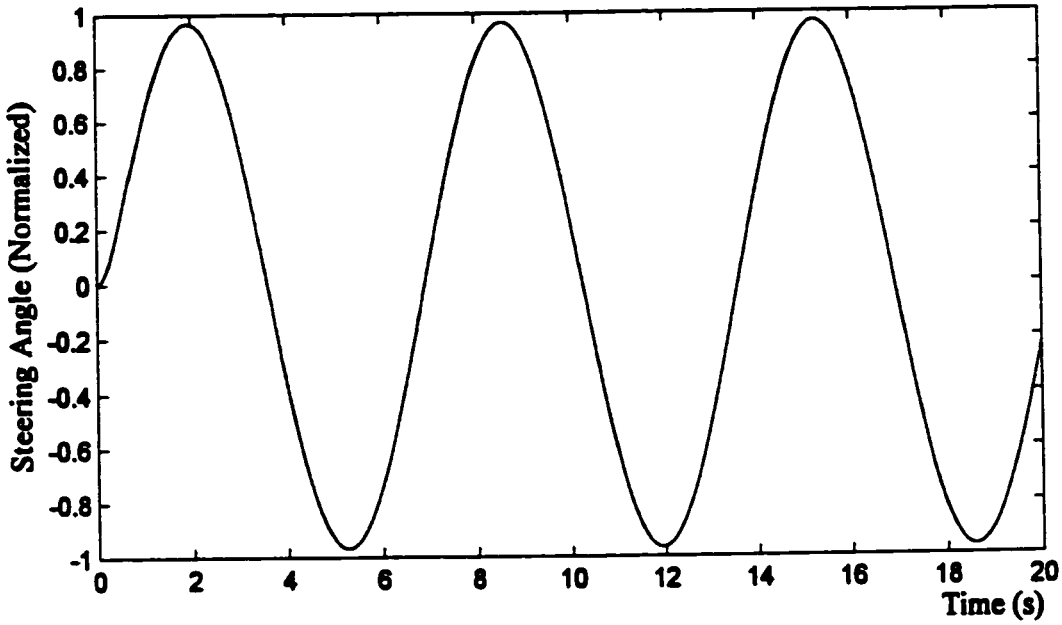


(a)

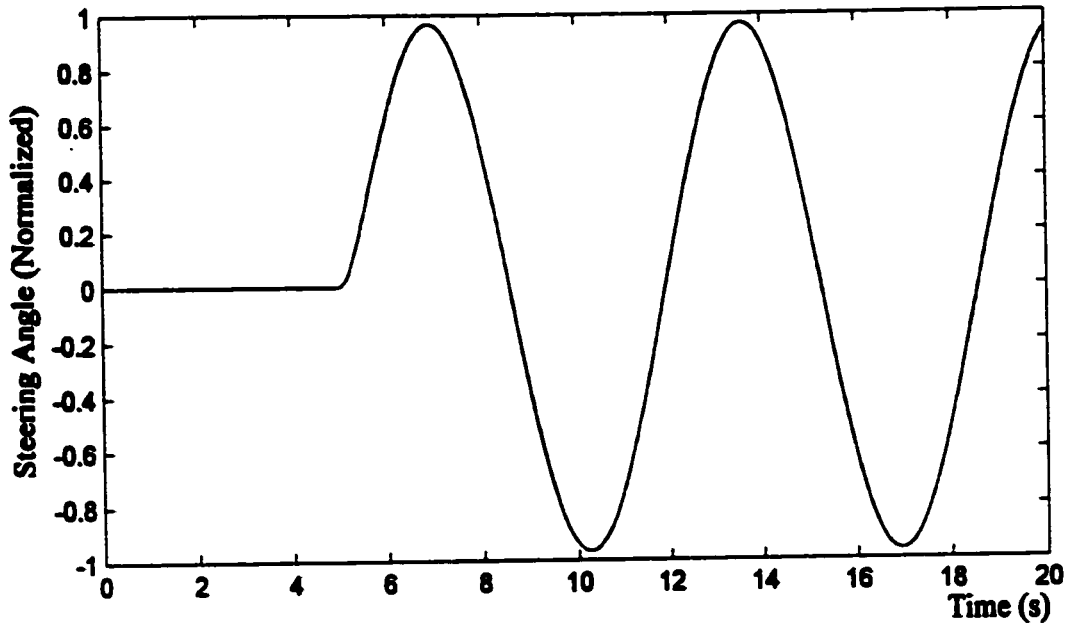


(b)

Figure 5.22. Speed response to 0.15 Hz sinusoidal input of amplitude range [0 1] for actual system and predicted: (a) zero time delay, and (b) 5 second time delay.

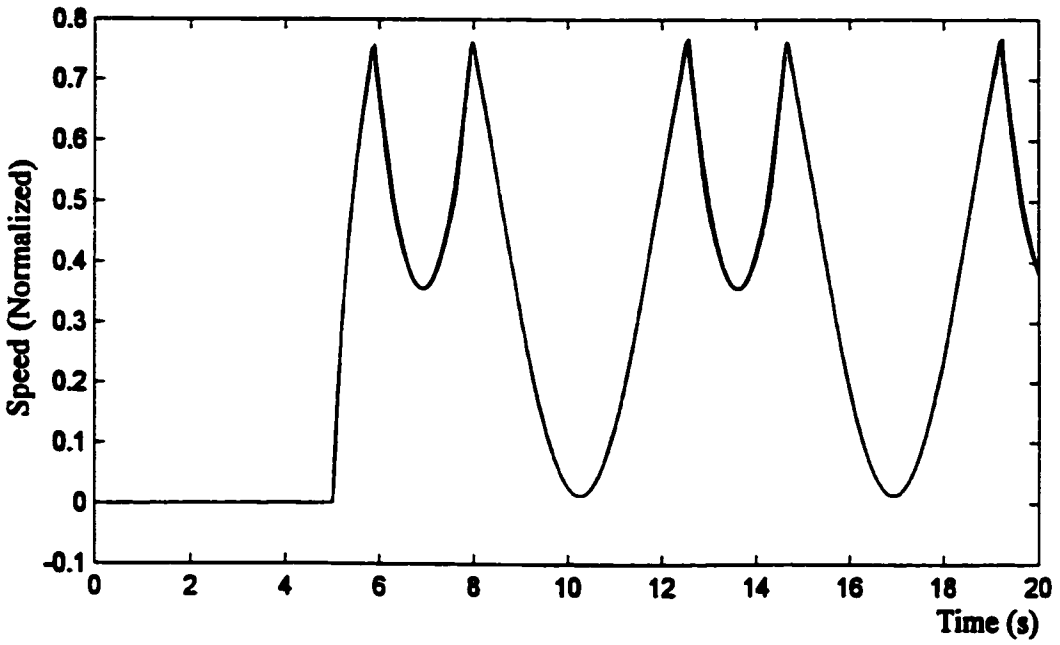


(a)

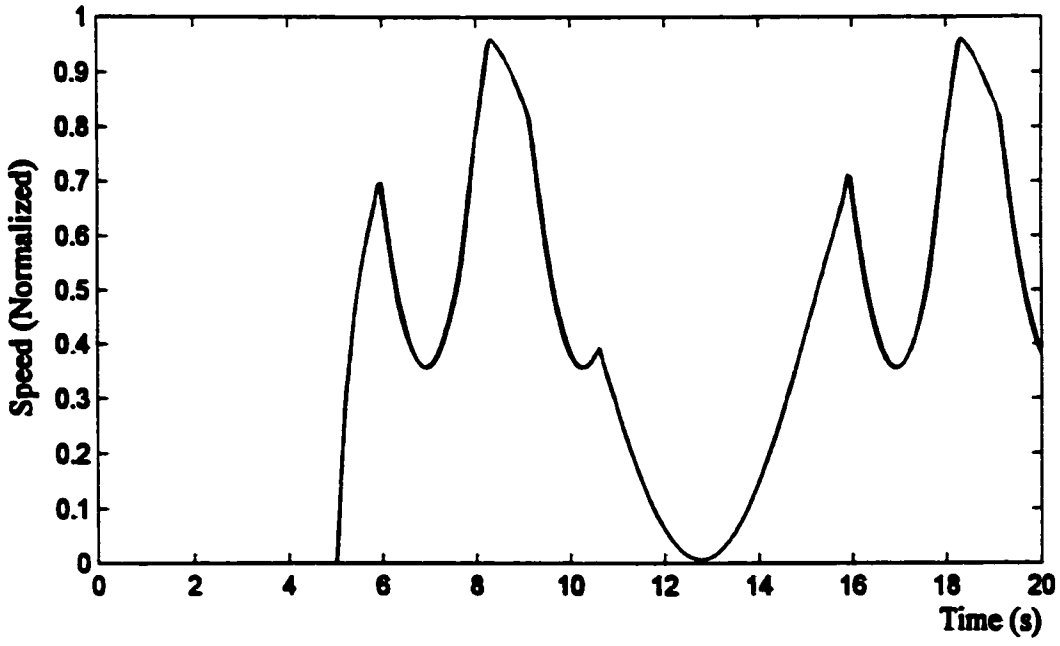


(b)

Figure 5.23. Steering angle response to 0.15 Hz sinusoidal input of amplitude range [-1 1] for actual system and predicted: (a) zero time delay, and (b) 5 second time delay.



(a)



(b)

Figure 5.24. Speed response for actual and predicted system with 5 second delay and speed set and steering set amplitude ranges of [0 1] and [–1 1], respectively: (a) speed set and steering set inputs at 0.15 Hz, and (b) speed set input at 0.1 Hz and steering set input at 0.15 Hz.

5.4.3 Predictor in the Loop

Another set of simulations was run using the second (ie.series) configuration as shown in Appendix B with the operator driving the predictor and the output of the predictor driving the robot vehicle. Since the test results mirror those of the previous configuration virtually identically, only a small number of results for the case of 2 second time delay are included here. These include step-function and sinusoidal responses in Figures 5.25, 5.26 and 5.27. The slight discrepancies in the step responses of Figures 5.25 and 5.26 are due to the actual system transient responses as was described in Section 5.4.2. There is also a small drop in the steady-state amplitude of the steering angle step response, probably a frequency response effect. Otherwise, the performance of the system using this configuration is obviously excellent, validating both the accuracy of the predictor and its use in driving the remote system.

5.5 System Performance with Mismatches

5.5.1 General

The excellent performance results described in the previous sections are based on a number of specific assumptions:

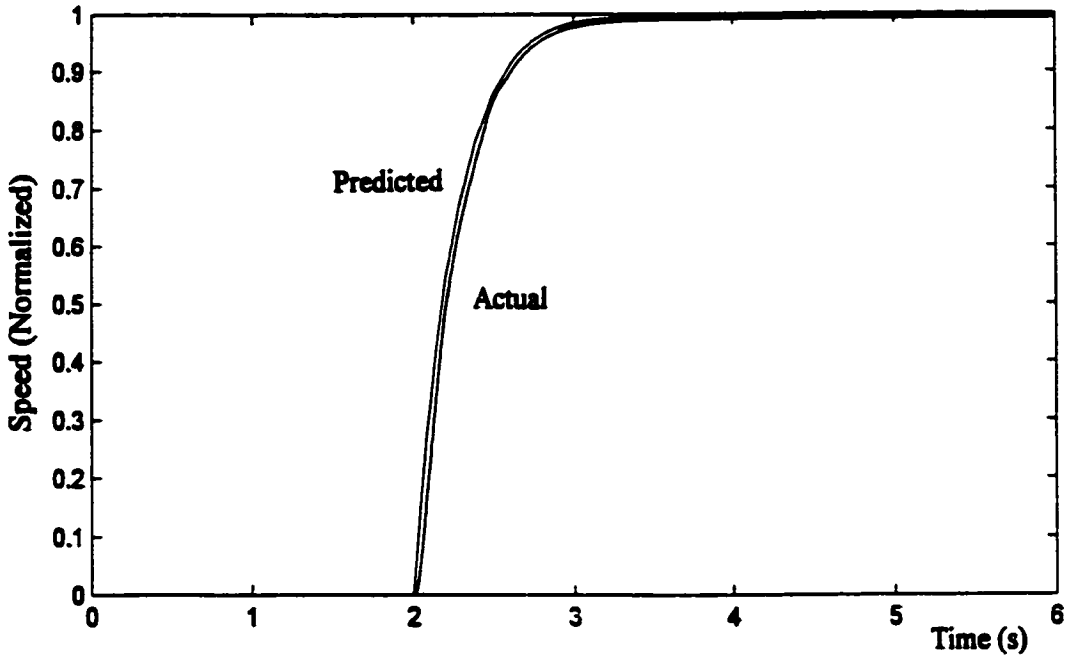
- the loop time delay used in the simplified Smith controller is the exact time delay in the loop

- the model of the human dynamics used in the simplified Smith controller is a perfect replica of the human operator's dynamics

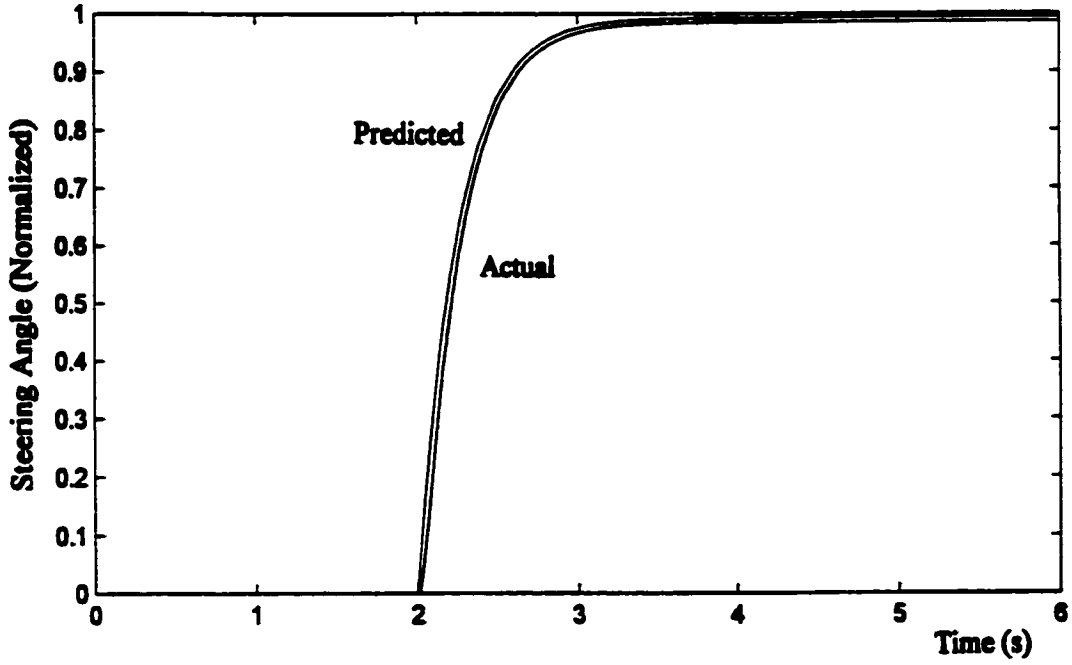
- the model of the human dynamics used in the predictor path is a perfect replica of the human operator's dynamics

- in teleoperation mode, the human operator acts as an ideal error detector

- there are no external disturbances of any significance



a)



(b)

Figure 5.25. Actual and predicted response of system with predictor in the loop to unit step-function with 2 second time delay in the system: (a) speed response, and (b) steering angle response.

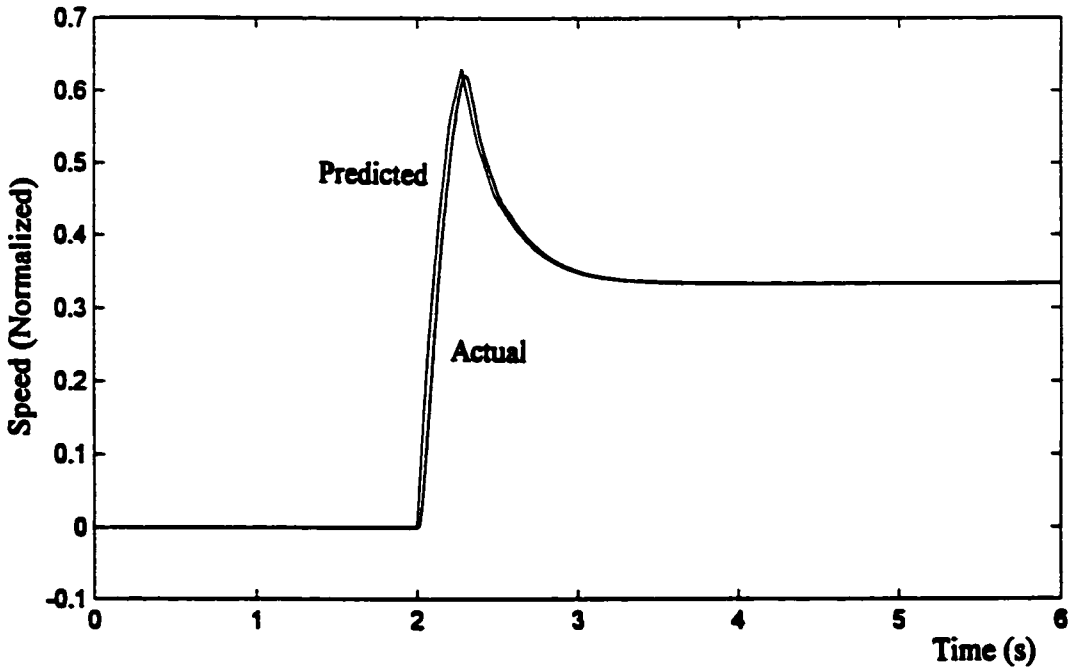
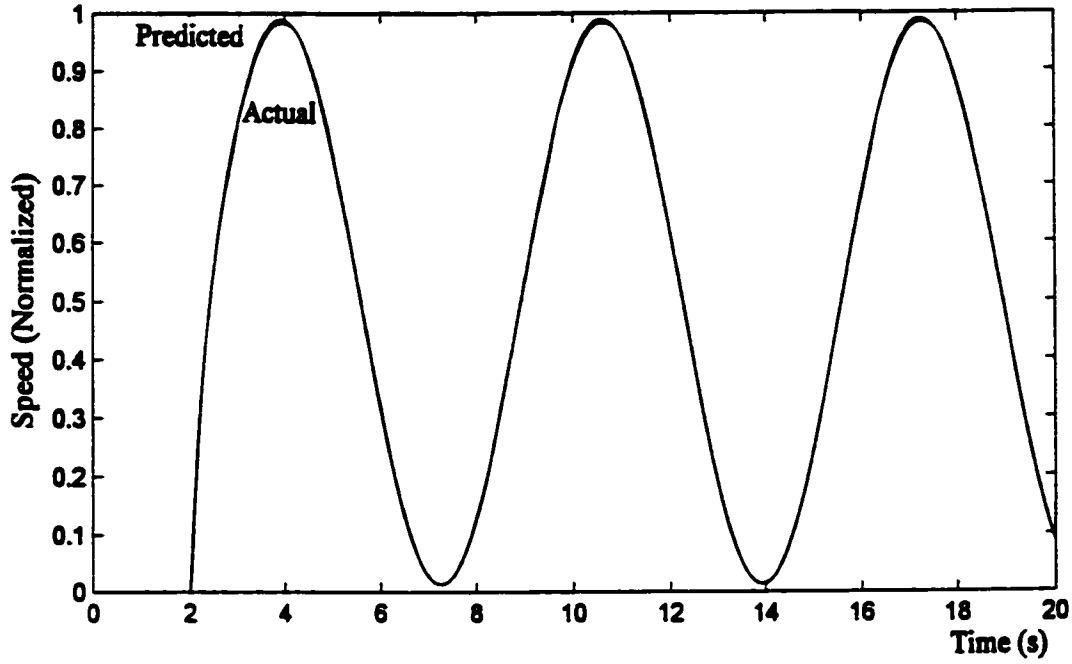


Figure 5.26. Actual and predicted speed response to unit step-functions of speed set and steering set with 2 second time delay in the system and with the predictor in the loop.



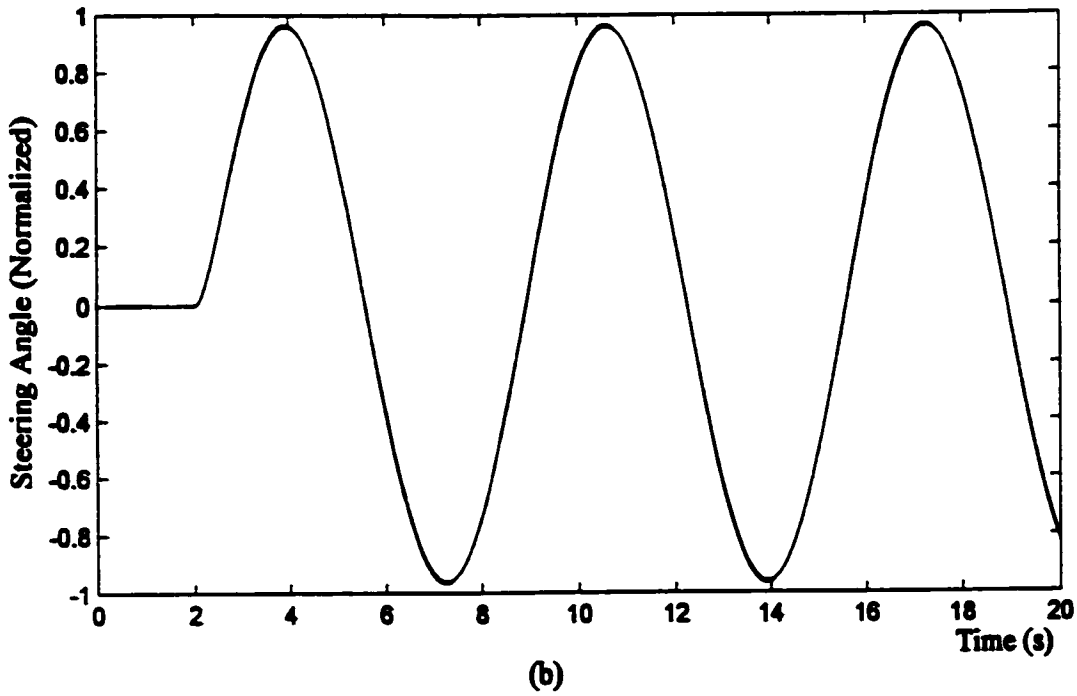


Figure 5.27. Actual and predicted response of system with predictor in the loop to 0.15 Hz sinusoidal inputs and amplitude ranges [0 1] and [-1 1] for speed and steering set, respectively and with 2 second time delay in the system: (a) speed response, and (b) steering angle response.

It has been stated earlier that the loop time delay should be relatively easy to measure with great accuracy, so the simplified Smith controller should be valid. Nevertheless, mismatch in the (estimated) time delay is a known significant problem in traditional Smith control so this will be examined. Modeling a human operator's response dynamics is a well-known problem. The basic model used in this project represents the neuromuscular response characteristics of a typical human operator. It should be a good example. Reaction delay time has been omitted since sufficient preview of the path in front of the robot vehicle is expected to be available via the video display. Providing a perfect model of the human dynamics for the simplified Smith controller and for the path of the predictor obviously can only be attempted. However, it is well known that, in non-delayed control environments, a human operator will adopt the necessary equalization characteristics required to accomplish a particular control action. Therefore, the use of models which are ideal replicas of the human operator's dynamics should be reasonable. The expectation that the operator will be able to perform the error detection function in a near-ideal capacity given appropriate training has already been discussed in section 4.6.

All real-world systems experience disturbances of one form or another. Depending on the type of communication link used, there would likely be some signal variation. Assuming adequate digital transmission, noise and distortion should not be a factor, but there could be some timing jitter. This has not been modeled, nor has any other specific type of disturbance. Disturbances should be considered in any future work on this subject.

5.5.2 Time Delay Mismatch

The sensitivity of the Smith controller to mismatch error in estimates of non-delayed plant dynamics and internal time delay has been identified [9,10,31]. The simplified Smith controller used in this project requires only the loop time delay to be provided. For remote-control systems with non-delayed plants, this time delay can be measured accurately. Nevertheless, it is of interest to examine the performance of the system with mismatch in the loop time delay used in the simplified Smith controller.

Following are system response plots made by using system delays of 1 second and loop delay mismatches in the Smith controller of 2.02 seconds and 2.002 seconds, representing errors of 1% and 0.1%, respectively. It must also be recognized that

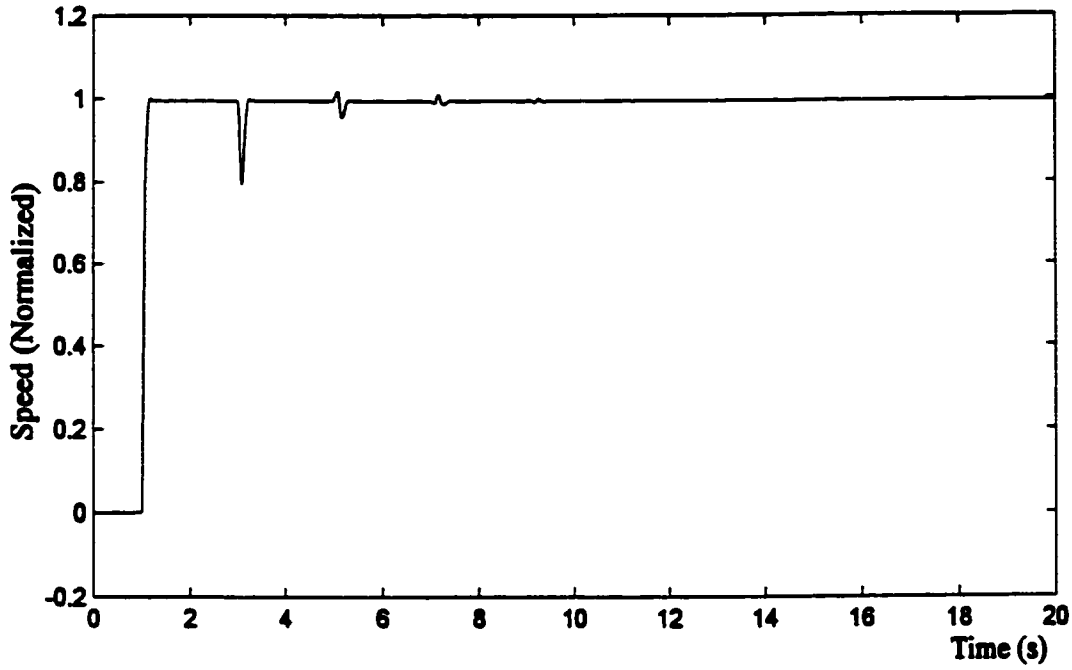
these errors represent 20 ms and 2 ms, respectively. From the point of view of using communication ranging techniques to measure loop time delay, 20 ms or 2 ms should be no problem at all. In 1983, time accuracy of 1μs was achievable with ranging (for deep-space communications) using a 1MHz reference signal [101].

The basic system performance criterion used throughout this thesis is the comparison with a non-delayed system and with no human operator. Those results have been presented earlier. The following system performance plots present a direct comparison of time-delay mismatch vs. no mismatch.

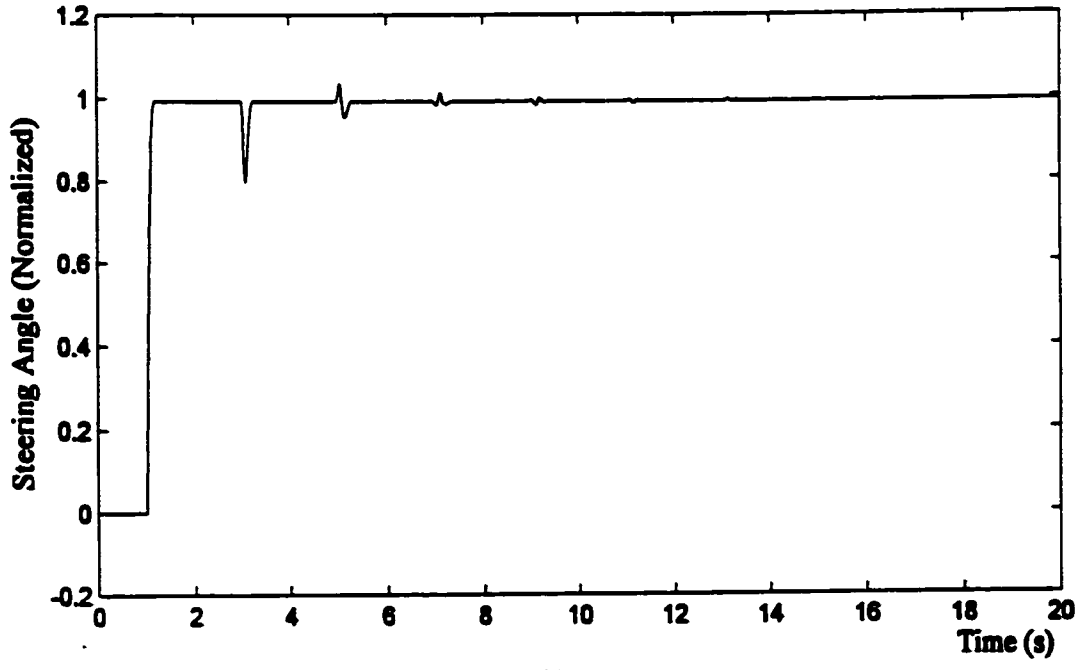
5.5.2.1 No Human Operator in the Loop

Results of loop delay mismatch of 1% (2 02 seconds) with no human operator in the loop are shown in Figures 5.28 and 5.29. The speed and steering angle unit step responses in Figure 5.28 are very similar, showing significant glitches at the 3 second point, but decaying rapidly to zero. This is caused by the output signals being fed back around the 2 second loop and not being totally compensated for by the Smith controllers due to the time mismatch. Increasing the mismatch results in larger glitches, but these results are not included. For example, at 2% error there is small ringing in the speed response and significant, non-decayed ringing in the steering angle response every two seconds beyond the 19 second point. The sine wave responses in Figure 5.29 reveal the same glitches as the unit step responses for speed but the steering response shows only the slightest effect at the 3 second point. Once again, increasing the mismatch to 2% reveals larger glitches and ringing that increases every 2 seconds. These results are also not shown but it is clear that the system (both speed and steering) is unstable at 2% mismatch.

Results of loop delay mismatch of 0.1% (2 002 seconds) with no human operator in the loop are shown in Figures 5.30 and 5.31. Again, the speed and steering angle responses in Figure 5.30 are very similar to each other, with somewhat larger glitches for steering, but the effect of the improved mismatch is apparent with the much smaller glitches as compared with Figure 5.28. The sine wave responses show a very slight degrading effect only for the speed response at the 3 second point. There is obviously negligible degradation in basic system performance with loop time mismatch of 0.1%.



(a)



(b)

Figure 5.28. Response to unit step-function input with 1 second system time delays and 2.02 second loop delays (ie.1% error) in the Smith controllers, and with no human operator in the loop: (a) speed response, and (b) steering angle response.

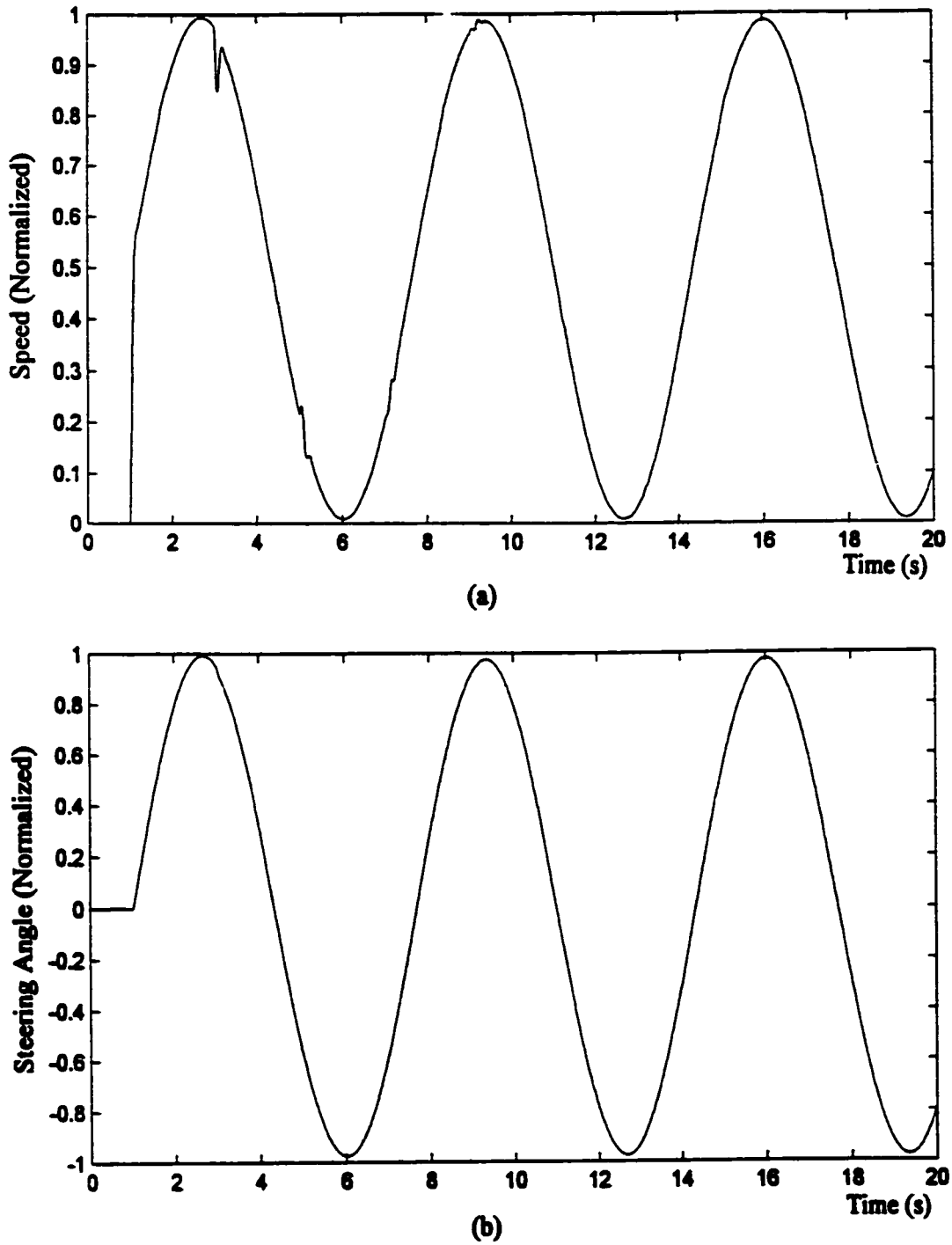
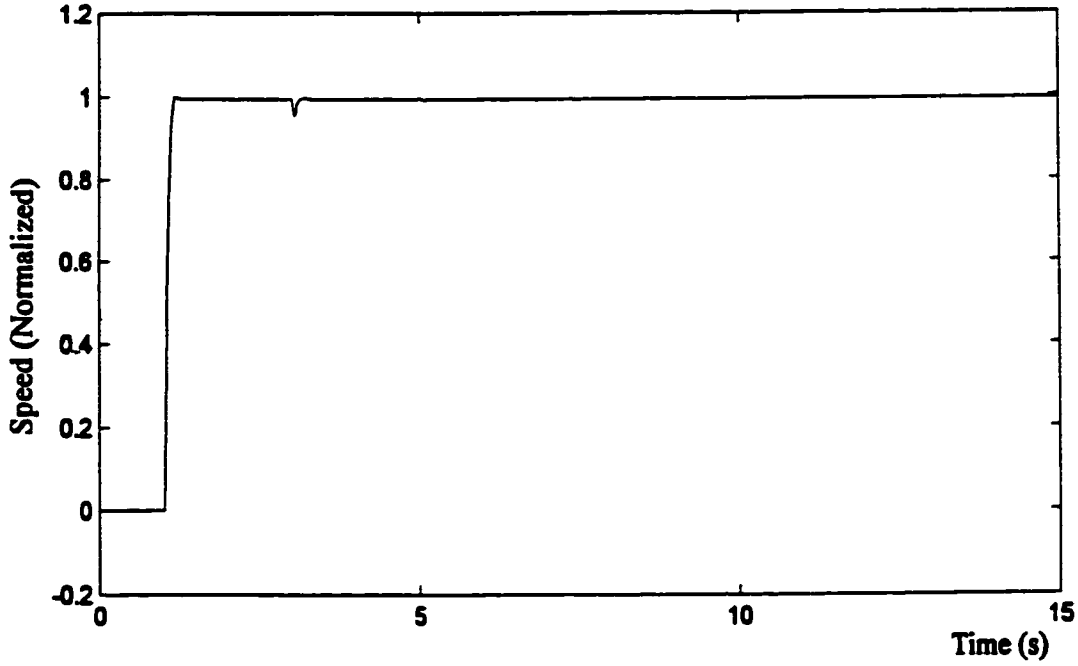
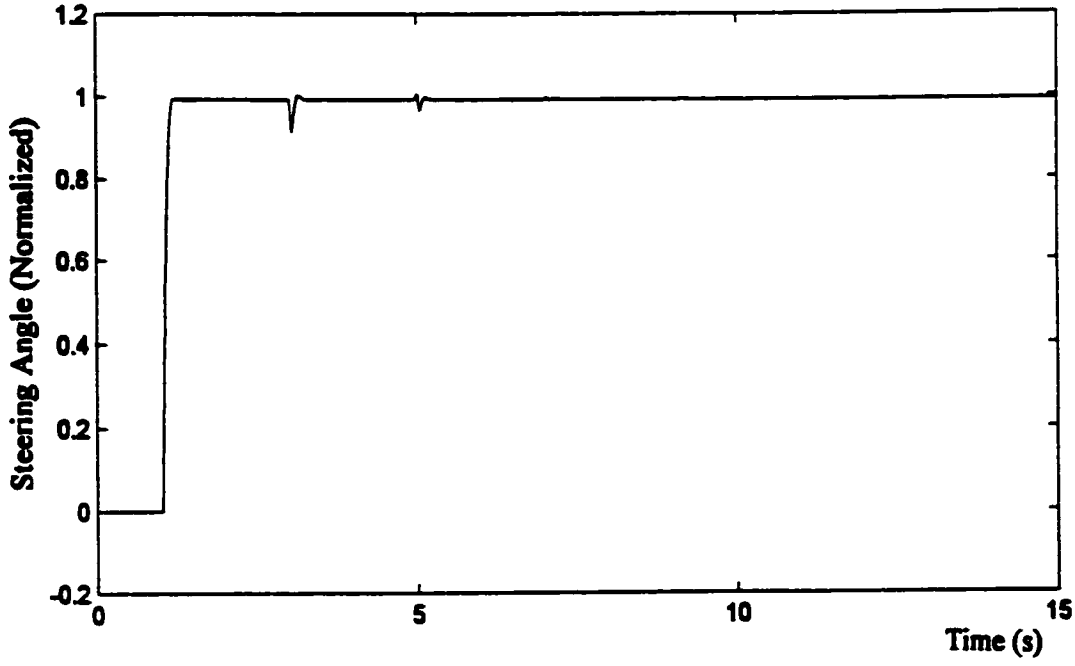


Figure 5.29. Response to 0.15 Hz sinusoidal input with 1 second system time delays and 2.02 second loop delays (ie.1% error) in the Smith controllers, with no human operator in the loop (speed set amplitude range [0 1] and steering set amplitude range [-1 1]): (a) speed response, and (b) steering angle response.

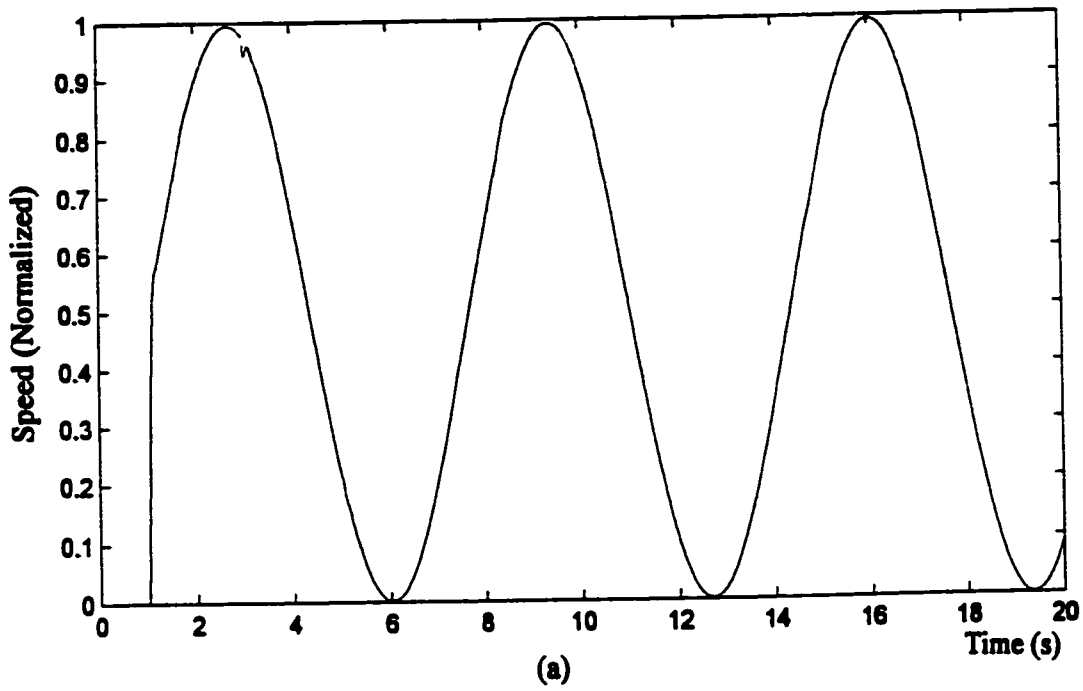


(a)



(b)

Figure 5.30. Response to unit step-function input with 1 second system time delays and 2.002 second loop delays (ie. 0.1% error) in the Smith controllers, and with no human operator in the loop: (a) speed response, and (b) steering angle response.



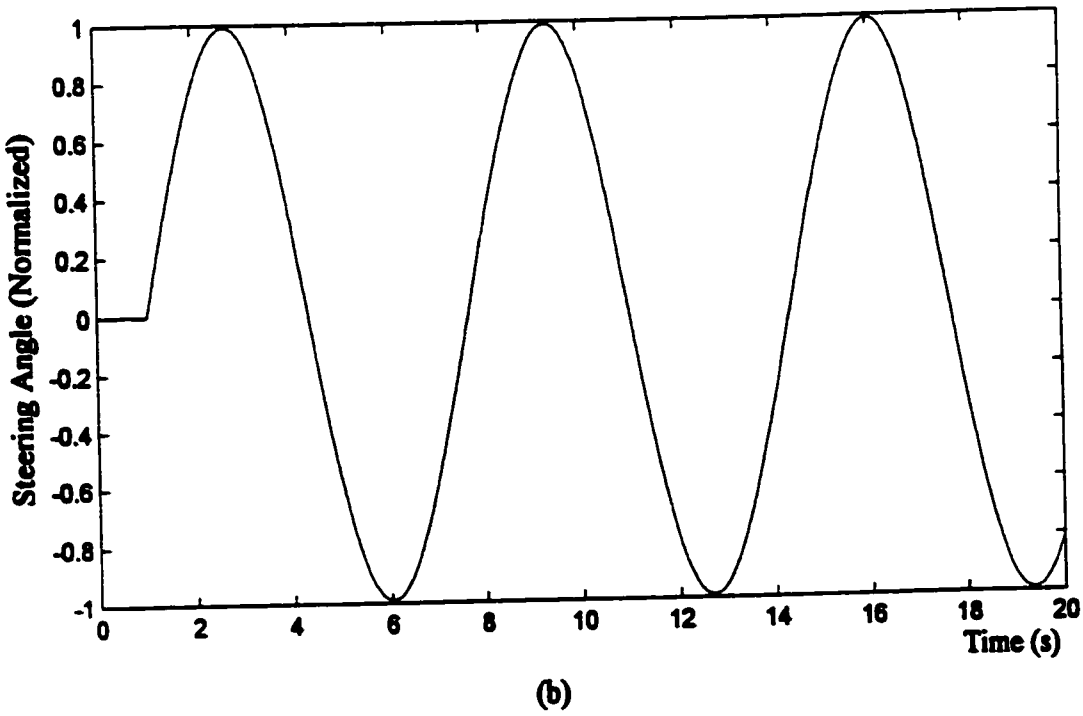
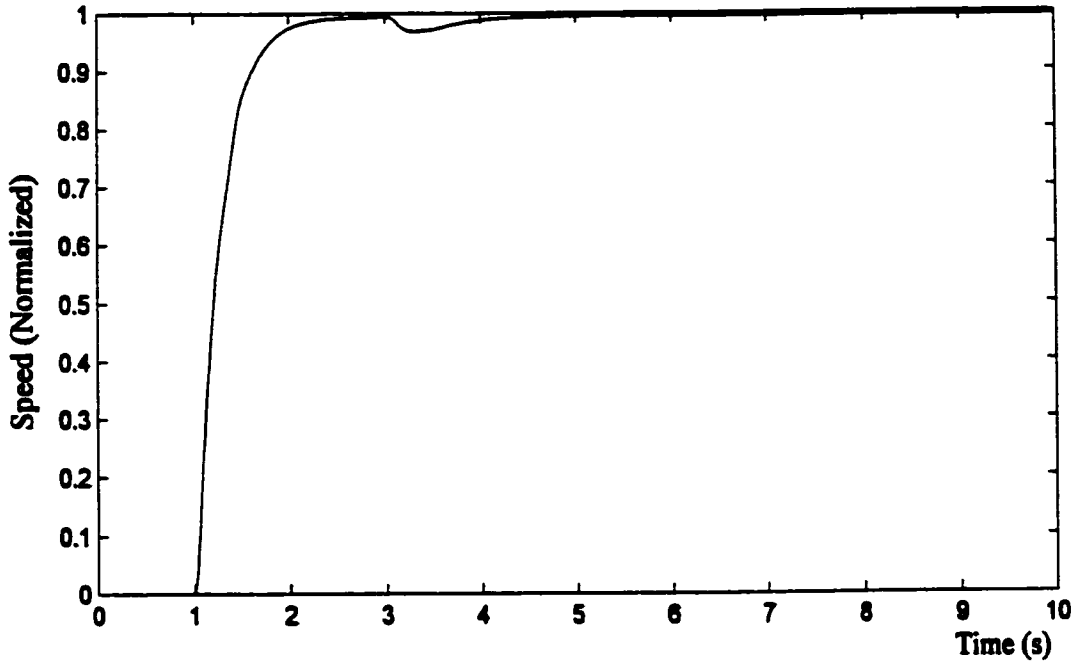


Figure 5.31. Response to 0.15 Hz sinusoidal input with 1 second system time delays and 2.002 second loop delays (ie. 0.1% error) in the Smith controllers, and with no human operator in the loop (speed set and steering set amplitude ranges [0 1] and [-1 1], respectively): (a) speed response, and (b) steering angle response.

5.5.2.2 Human Operator in the Loop

Results of loop delay mismatch of 1% (2.02 seconds) and including the human operator in the loop are shown in Figures 5.32 and 5.33. The speed and steering angle unit step responses in Figure 5.32 are virtually identical, showing a small dip starting at the 3 second point. While not included here, increasing the mismatch to 2% results in an increase in the severity of the dip. The sine wave responses in Figure 5.29 reveal a very slight degrading effect only at the 3 second point for the speed response. This effect is more pronounced with larger mismatches, but those results are not included here.

Results of loop delay mismatch of 0.1% (2 002 seconds) and including the human operator in the loop are shown in Figure 5.34. Again, the speed and steering angle responses are very similar to each other, but the effect of the improved mismatch is apparent with the very minimal glitches as compared with Figure 5.32. The sine wave responses are not included since they do not show any noticeable degradation due to the small time mismatch.



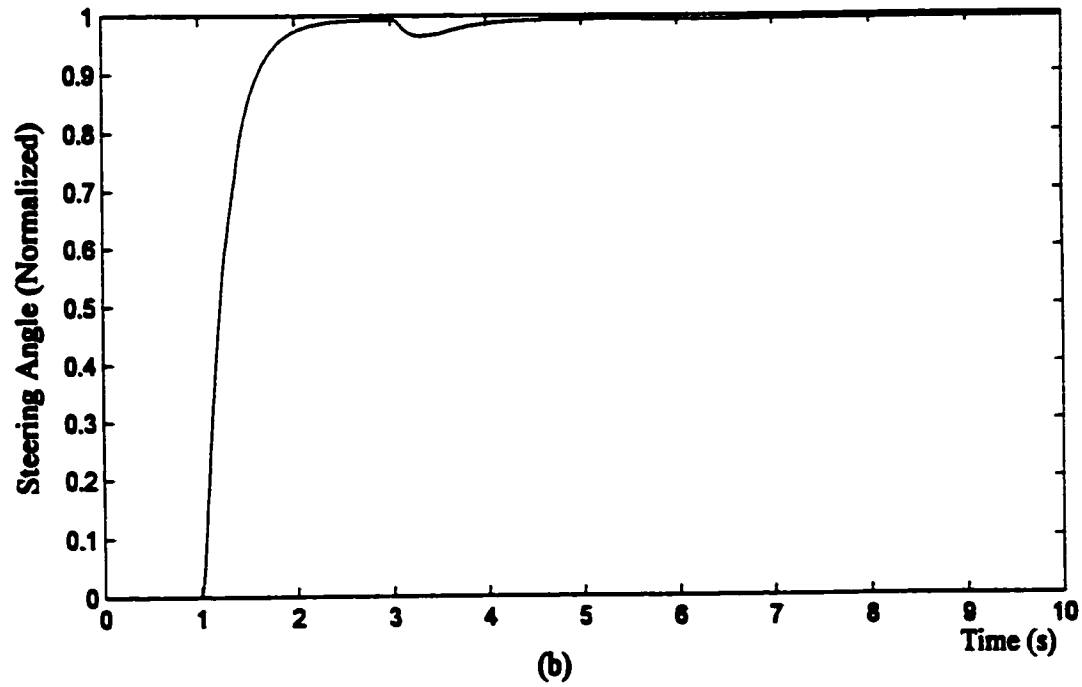


Figure 5.32. Response to unit step-function input with 1 second system time delays and 2.02 second loop delays (ie.1% error) in the Smith controllers, including the human operator in the loop: (a) speed response, and (b) steering angle response.

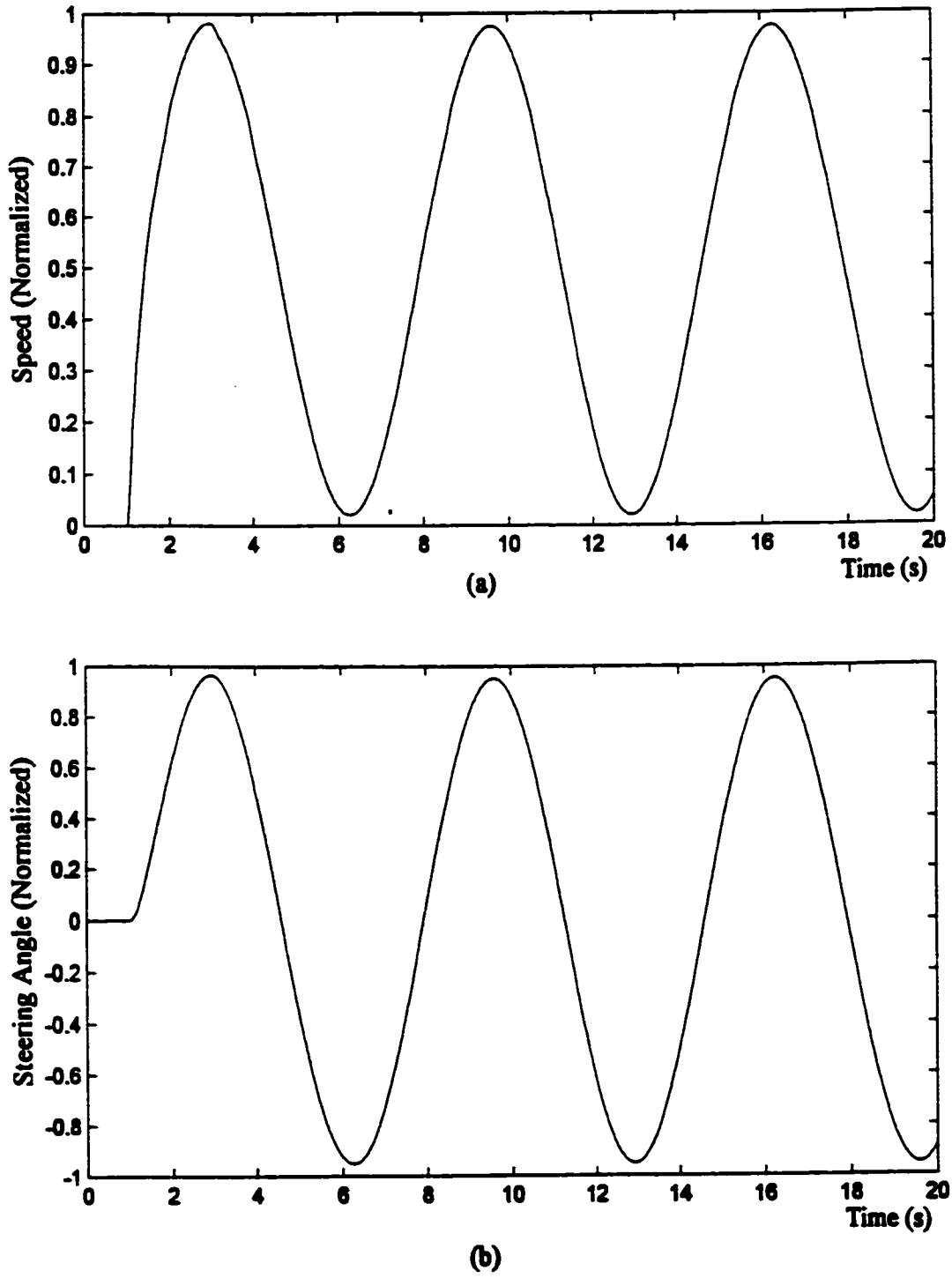
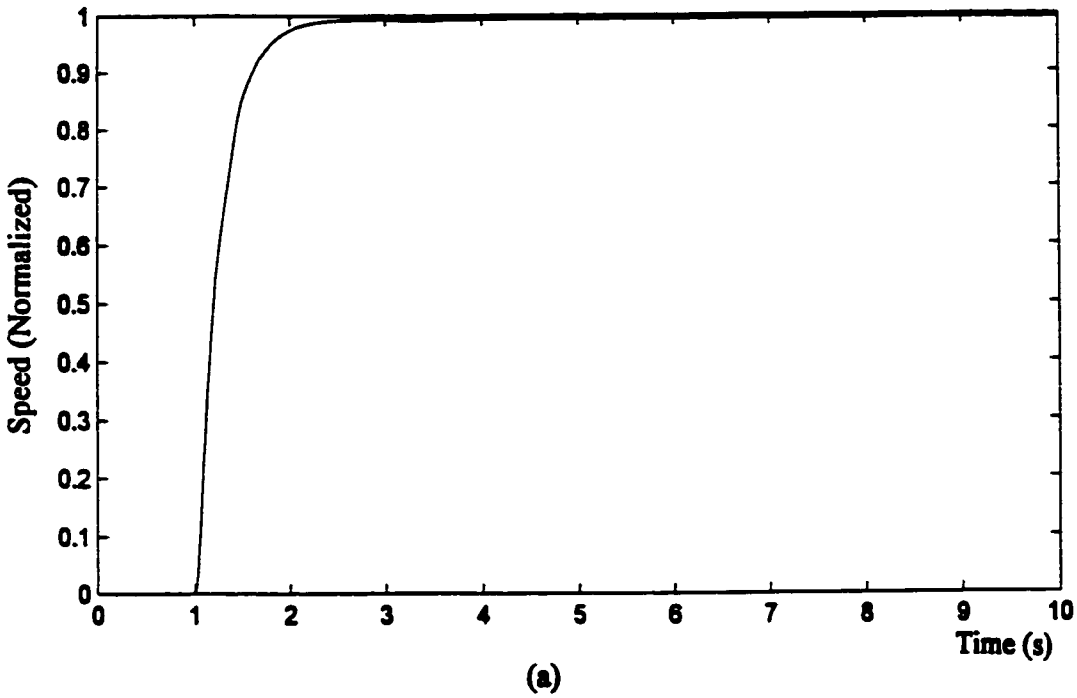


Figure 5.33. Response to 0.15 Hz sinusoidal input with 1 second system time delays and 2.02 second loop delays (ie. 1% error) in the Smith controllers, including the human operator in the loop (speed set and steering set amplitude ranges [0 1] and [-1 1], respectively): (a) speed response, and (b) steering angle response.



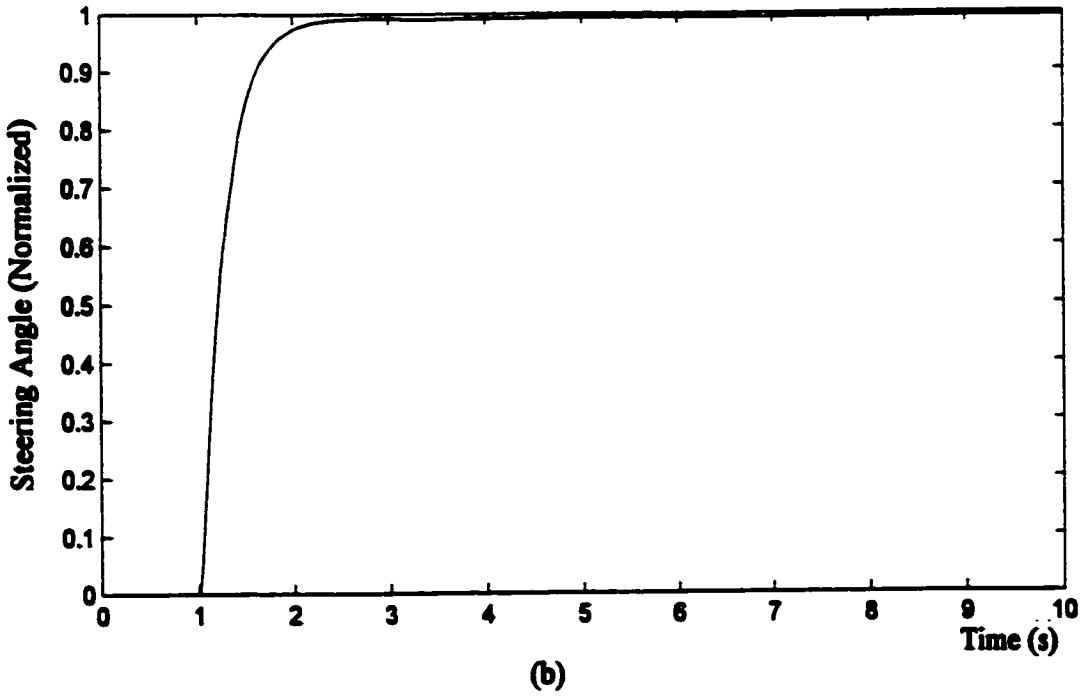


Figure 5.34. Response to unit step-function input with 1 second system time delays and 2.002 second loop delays (ie. 0.1% error) in the Smith controllers, including the human operator in the loop: (a) speed response, and (b) steering angle response.

5.5.3 Predictor Mismatch

The neural network predictor used in the previous simulations is obviously highly accurate over the total input-output space judging by the results in section 5.4.3. In order to see the effect of a less accurate predictor, another neural network was trained by using the same input pattern and target data as for the original network. The new network is a 5,5,2 neuron network using the same activation functions (ie. tansig, tansig, linear) as the original. Using fewer neurons should reduce the accuracy of the network, particularly when using fewer training epochs. By restricting the training of this 5,5,2 network to 100 epochs, the resulting mean-squared-error (MSE) was 5.7×10^{-5} as compared to MSE of 1.26×10^{-6} for the original 8,8,2 network. This may not seem significant but the resulting speed vs. speed set response shown in Figure 5.35 obviously shows that the network is not well trained at all, particularly compared with the results of the 8,8,2 network in Figure 5.18(b). The steering angle vs. steering set response turned out to be essentially as perfect as for the 8,8,2 network in Figure 5.19(b) and is not included here. In retrospect, a predictor with even worse results should have been used for comparison purposes.

By including this new predictor in the basic system shown in Appendix A, the speed and steering angle responses were again plotted. Step-function responses are shown in Figure 5.36. Sinusoidal responses are shown in Figure 5.37. The speed response showing the dependency on the steering angle is included in Figure 5.38. These plots were all made using the same scales as used for the original 8,8,2 neuron predictor in Figures 5.20 through 5.24 for ease of comparison.

The only significant difference between the unit-step responses in Figure 5.36 and those in Figures 5.20(a) and 5.21(a) is that the steady-state amplitudes are lower when using the 5,5,2 neuron predictor. The reduction in the speed response is understandable from the poorly-trained network's speed response in Figure 5.35, but the network's steering angle response looked virtually perfect, so one would not expect the steering angle's unit step response to have a reduced steady-state value. Rechecking the neural network and the predictor response has verified that these results are accurate. The predictor was generated from the trained neural network using the same "gensim" function in Matlab as was used for the original predictor. The reason for the steering angle's reduced steady-state value is not known. At any

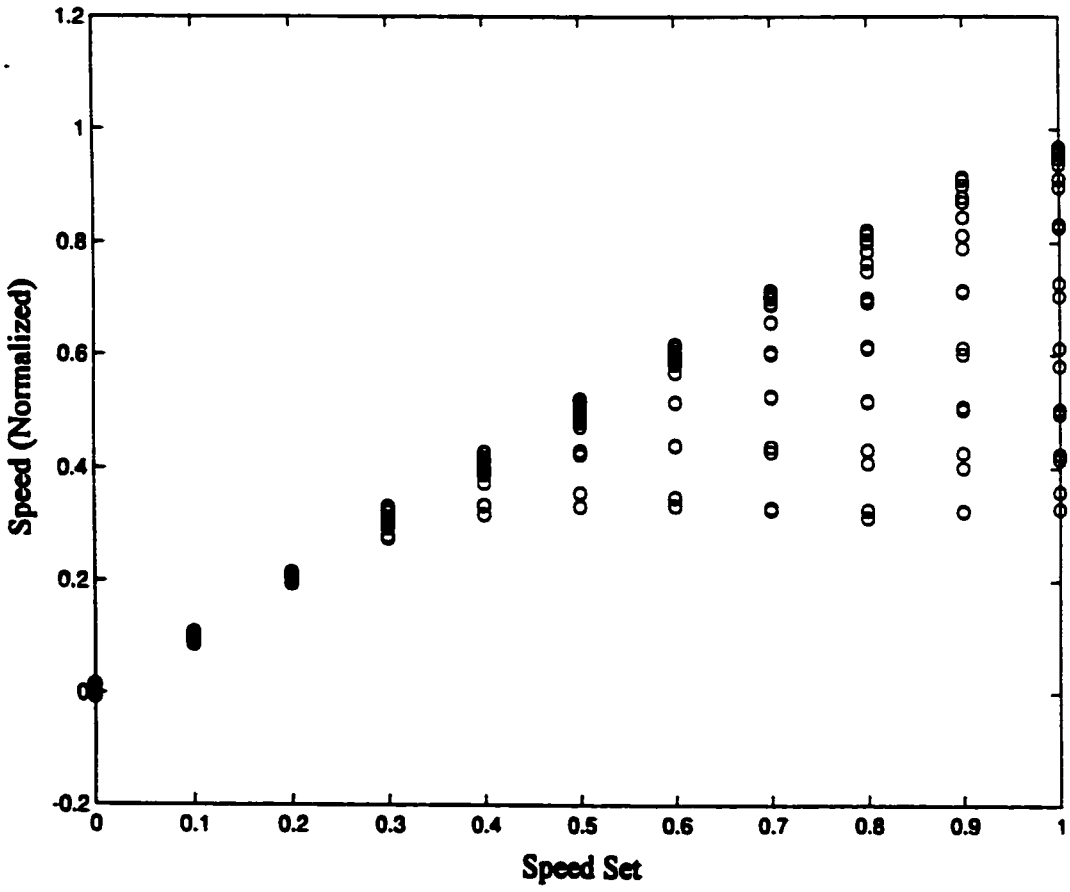
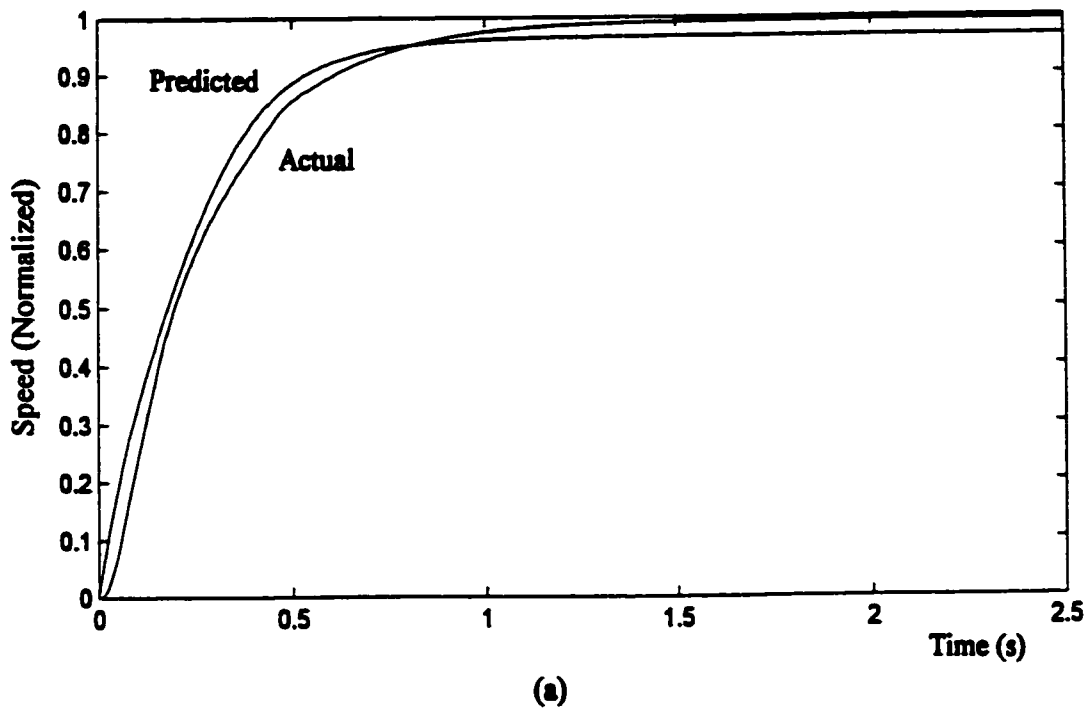
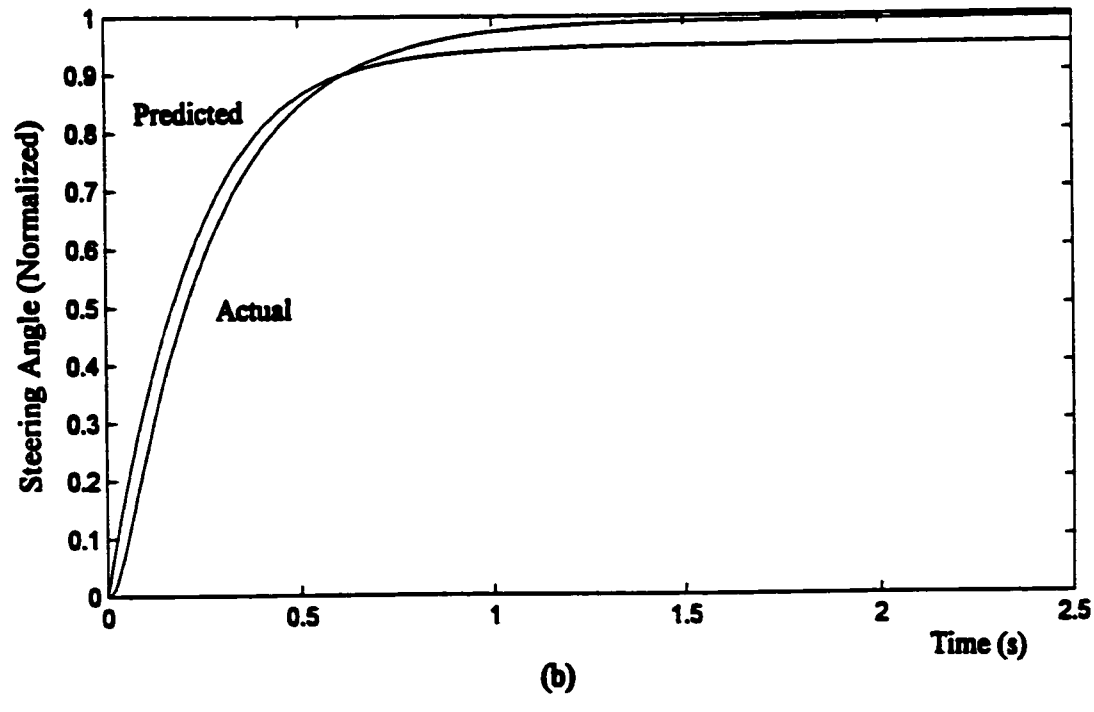


Figure 5.35. Steady-state speed vs. speed set for step increments of 0.1 and for fixed steering angles from zero to +/- 1.0 using the 5,5,2 neuron predictor.

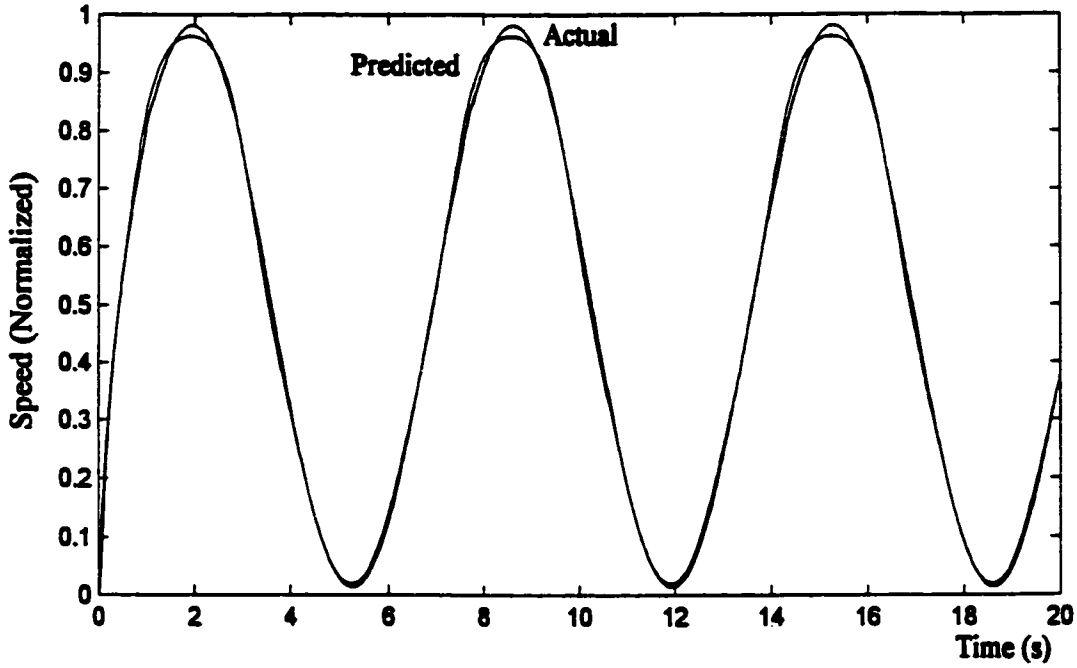


(a)

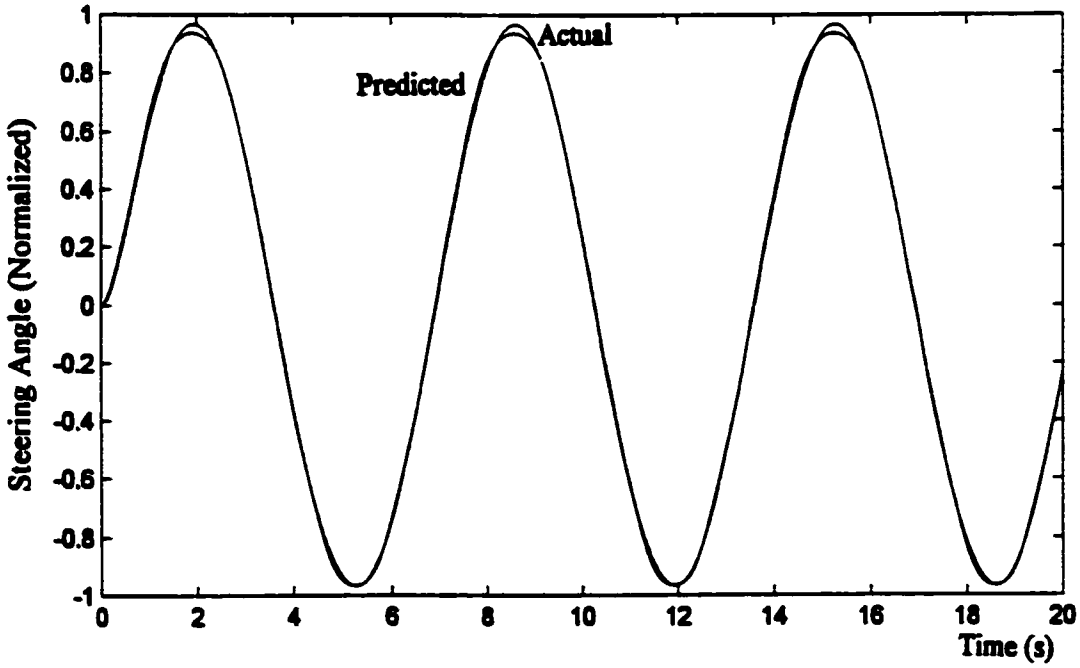


(b)

Figure 5.36. Unit step response of actual system and 5,5,2 neuron predictor: (a) speed response, and (b) steering angle response.



(a)



(b)

Figure 5.37. Response of actual system and 5,5,2 neuron predictor to a 0.15 Hz sinusoidal input with amplitude ranges [0 1] for speed set and [-1 1] for steering set: (a) speed response, and (b) steering angle response.

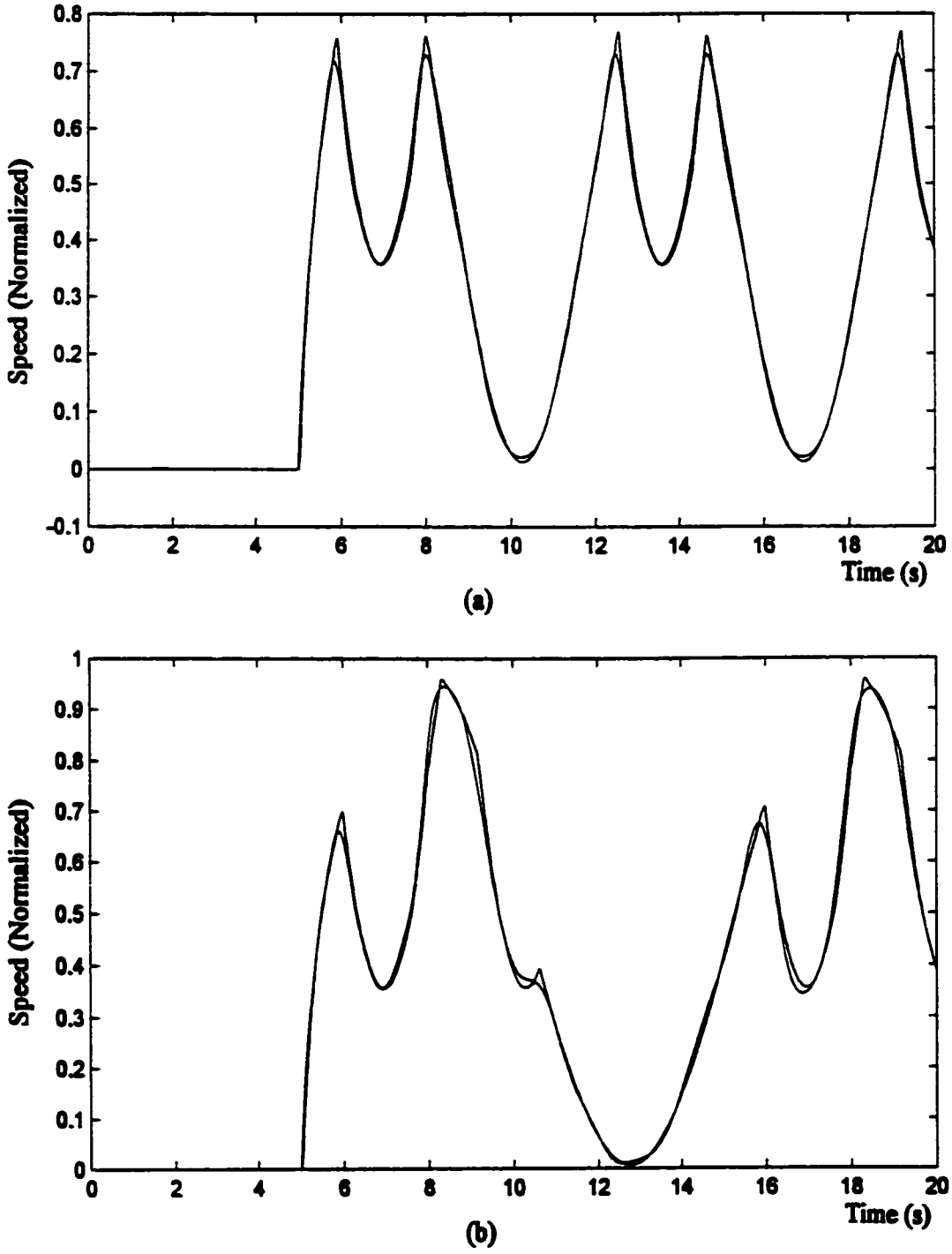


Figure 5.38. Speed response of actual system and 5,5,2 neuron predictor: (a) speed set and steering set inputs at 0.15 Hz with amplitude ranges [0 1] for speed set and [-1 1] for steering set, and (b) speed set input at 0.1 Hz and steering set input at 0.15 Hz with amplitude ranges [0 1] for speed set and [-1 1] for steering set.

rate, it shows the reduced accuracy of this predictor. Comparing the sinusoidal responses in Figure 5.37 with their counterparts in Figure 5.22(a) and Figure 5.23(a), there is some degradation, most prominently at the positive maximum values on the speed response, although the steering angle response also shows reduction at these points. Figure 5.37(b) also reveals a degree of non-symmetry since the negative response is not the same as the positive. Figure 5.38(a) shows the reduction in amplitude for speed compared to Figure 5.24(a). Figure 5.38(b) also shows the amplitude reduction for speed, but there is also more noticeable distortion between the actual system and predicted responses, where there was virtually none in Figure 5.24(b). These results are sufficient to demonstrate the inadequacy of the 5,5,2 neuron predictor tested and to validate the 8,8,2 neuron predictor used.

5.6 Obstacle-Avoidance

The obstacle-avoidance subsystem design has been described in section 4.4.2. Proper performance of this subsystem cannot easily be verified without taking this project to the next logical stage, which would be animation in a dynamic environment. The robot vehicle needs to be simulated in a mine environment on a video screen where various obstacles can be placed and where the human operator can drive the vehicle over a time-delayed link. A few simulations were run to give some examples of the functioning of this subsystem in the overall system. Since the obstacle factor can directly influence the speed, this dependency is shown in Figure 5.39. Obstacle factors ranging from 0 to 1 were applied directly to the RTC speed control system (see Figure A.6) while the speed had a unit step input (maximum speed) and the steering angle was fixed at an angle of zero. Figure 5.39 shows how the obstacle factor reduces the speed as the obstacle factor decreases (increasing likelihood of hitting an obstacle).

Automatic steering to avoid obstacles is also designed into the system. For the case when the steering set input is zero (ie. straight ahead), a unit step obstacle avoid input was applied to the steering control system (see Figure A.6). The resulting steering angle response is shown in Figure 5.40(a). The maximum value attainable with the multivariable system is 0.785. This should be sufficient for most obstacle-avoidance scenarios. If not, the operator is always ultimately available for intervention via teleoperation over the time-delayed link. The response of the steering

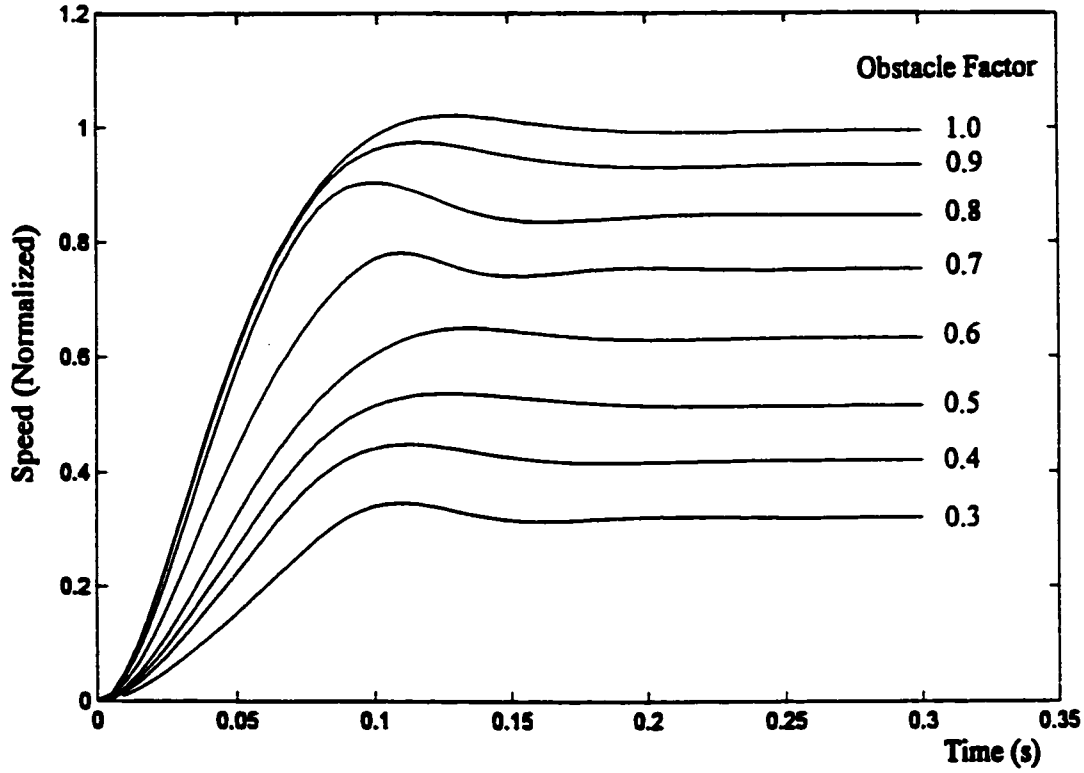
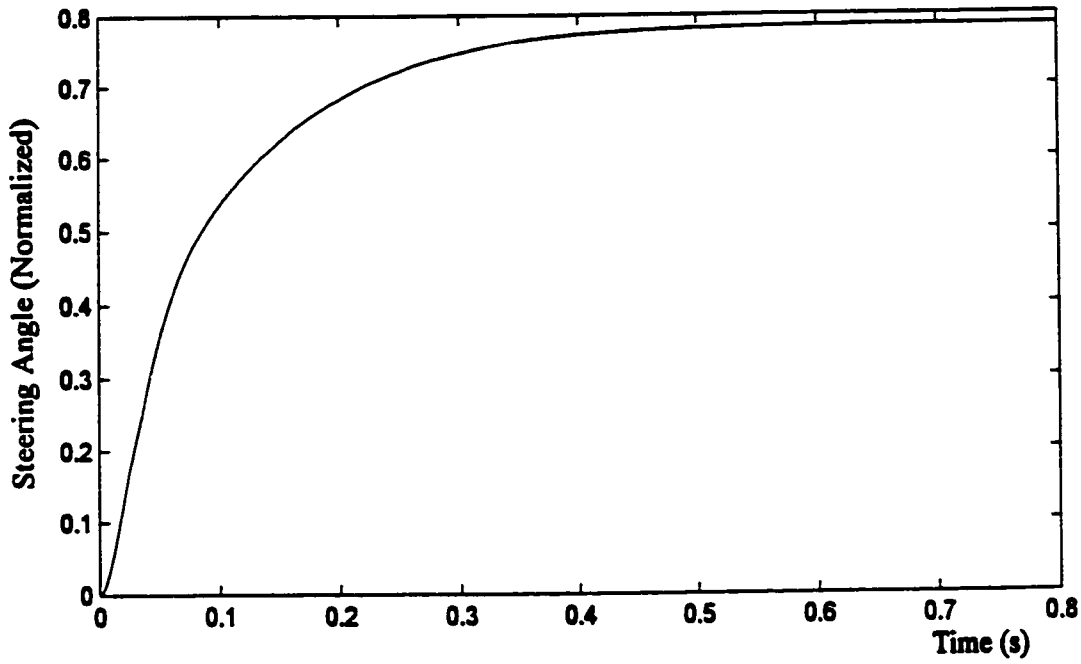
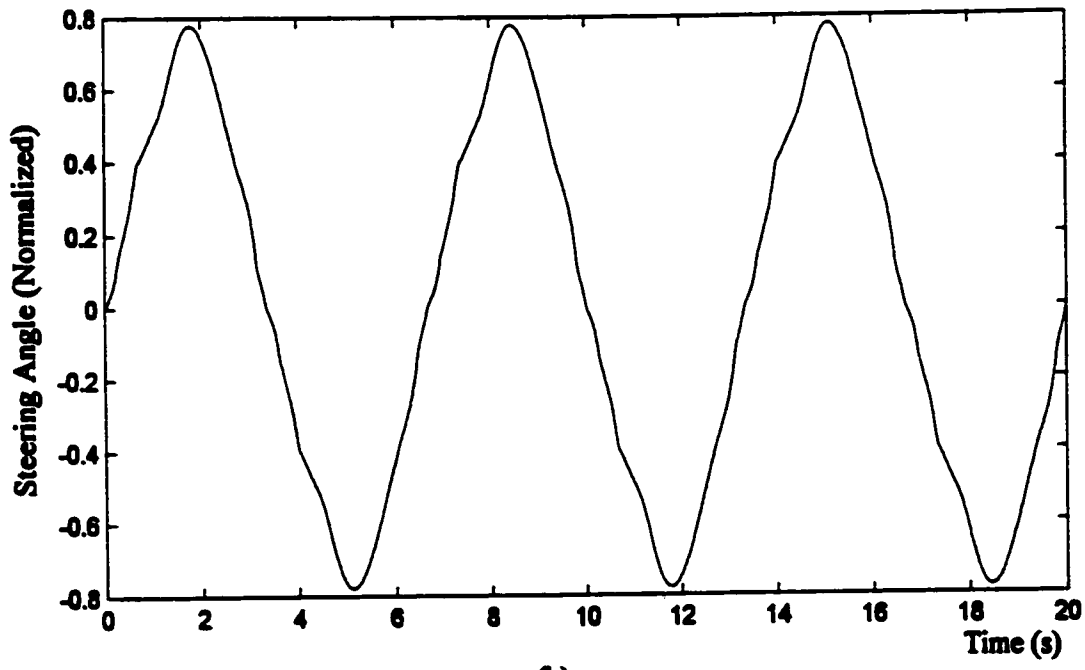


Figure 5.39. Speed response as a function of obstacle factor for unit step inputs of speed set.

angle to a sinusoidal obstacle avoid input was also plotted and included in Figure 5.40(b) for the case when the steering set input is zero. Nonlinearities in the system are evident. Nevertheless, it should be sufficiently linear for adequate obstacle-avoidance. Finally, two more runs were made to test the effect of the obstacle avoid input on steering command inputs from the operator. Using a sinusoidal steering set input, the steering angle response was plotted for positive fixed obstacle avoid inputs of 0.25 and 0.75 shown in Figure 5.41. The general effect is seen to provide a positive offset (the effect is symmetrical for negative inputs). A limiter was installed at the input to the vehicle to prevent the speed from exceeding unity. While it is not shown, an obstacle avoid input of 1.0 was sufficient to ensure that the steering angle remained positive throughout the cycle, even when the steering set input reached –1.0. There is little doubt that the obstacle avoid control signal can steer the vehicle independently of an operator.

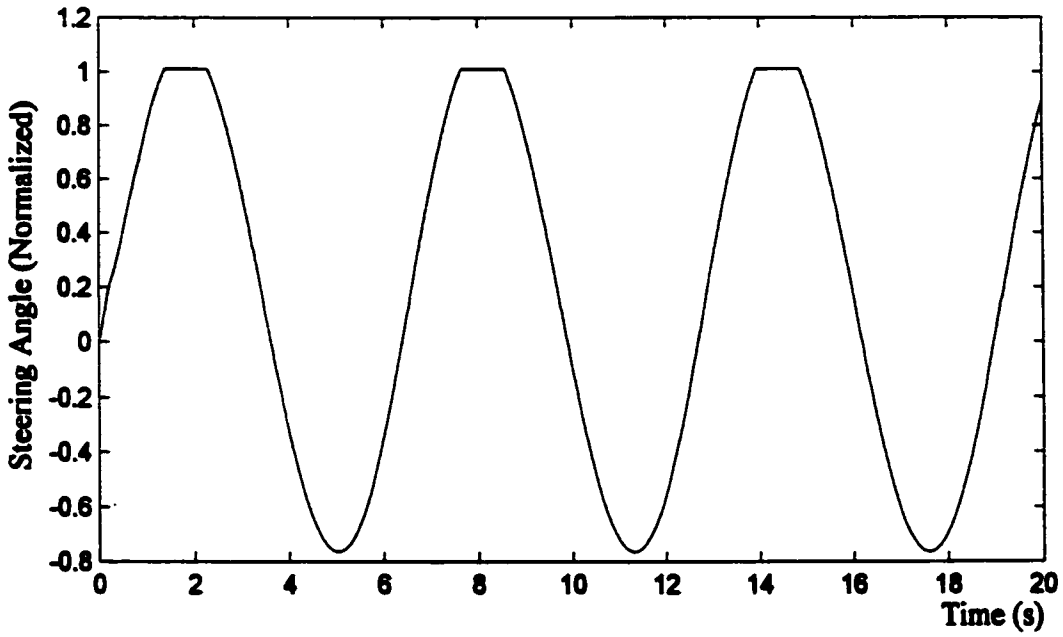


(a)

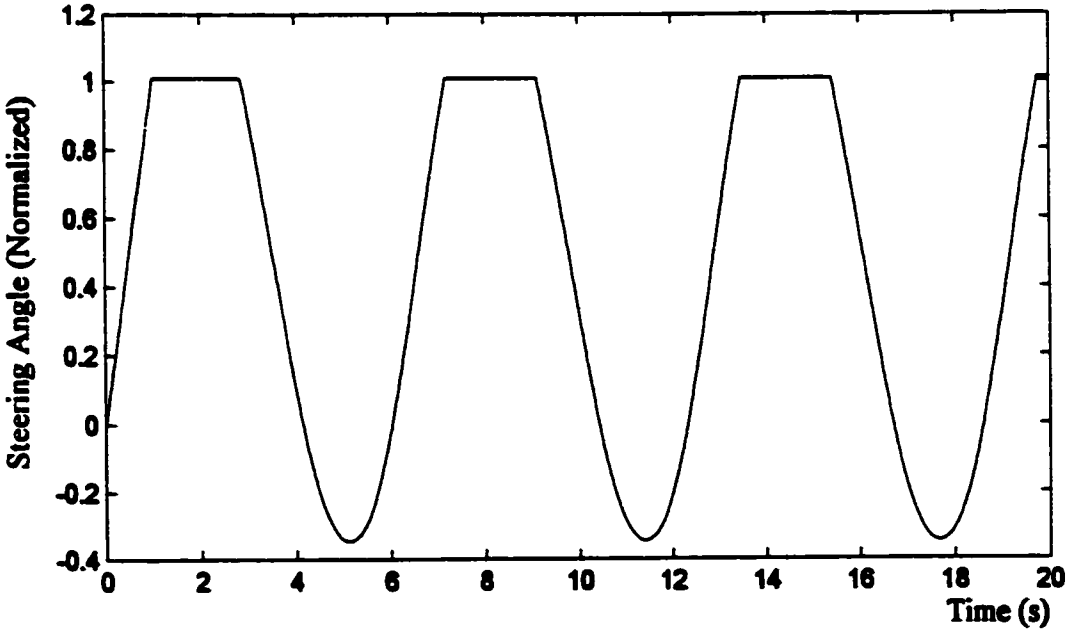


(b)

Figure 5.40. Steering angle response to obstacle avoid inputs with steering set input of zero: (a) step-function response, and (b) sinusoidal response to 0.15 Hz input of amplitude range [-1 1].



(a)



(b)

Figure 5.41. Steering angle response to sinusoidal steering set input of 0.15 Hz and amplitude range [-1 1] with obstacle avoid fixed inputs of: (a) 0.25, and (b) 0.75.

Tests conducted on the obstacle-avoidance subsystem alone do not provide a great deal of information. There are too many variables for a meaningful appraisal without getting swamped with data. Two examples are included here. The obstacle factor vs. range to obstacle has been plotted as a function of the angle to obstacle for the case when the steering angle is zero (straight ahead) and the speed is 1 (full speed). This is shown in Figure 5.42. The increments are 0.1 so the curves are very angular. We can see that at very close range, the obstacle factor is roughly proportional to the angle to obstacle, which is as one would expect. Small angle to obstacle inputs gives a small obstacle factor signifying high risk of collision. As the range to obstacle increases, the obstacle factor increases, signifying less risk of collision.

Figure 5.43 includes results of another test showing the obstacle avoid steering adjustment vs. range to obstacle as a function of the angle to obstacle for the same case of zero steering angle and maximum speed. At very close range the steering adjustment is maximum for all angles to obstacle and decreases with increasing range as expected. At large angles, the adjustment is very small as expected. The specific case of zero angle to obstacle is not shown since it is strongly negative. Note that the adjustment angles are positive for negative angles to obstacle, as we would expect. Positive angles to obstacle will result in negative adjustment angles by symmetry.

5.7 Tunnel-Tracking

The tunnel-tracking mode is activated by simply closing the teleoperation/tracking mode switch in the fuzzy steering control block (see Figure A.11) using a supervisory control command from the local terminal. The tunnel error signal is presumed to come from an externally-supplied sensor signal-processing system. This signal is the error between the robot vehicle's heading and the angle to the estimated centreline of the tunnel. A single simulation was run using a sinusoidal input varying over the full range of [-1 1] which represented the location of the tunnel centreline (ie. swinging back and forth between hard left to hard right). The difference between this signal and the steering angle was used as the tunnel error signal. The steering angle was used since the heading would only be available when system animation is provided. Figure 5.44 is the steering angle response. Just as the response to the obstacle avoid signal was less than the available range [-1 1], so this response also is less, spanning [-0.6 0.6]. Extensive trial and error in varying the available parameters of the multivariable fuzzy steering control system could not increase this

range without adversely affecting other parameters. Once again, however, we can presume that this range in steering angle should be sufficient for tunnel tracking, even in the presence of obstacles. Fortunately, in times of difficulty, the operator is there.

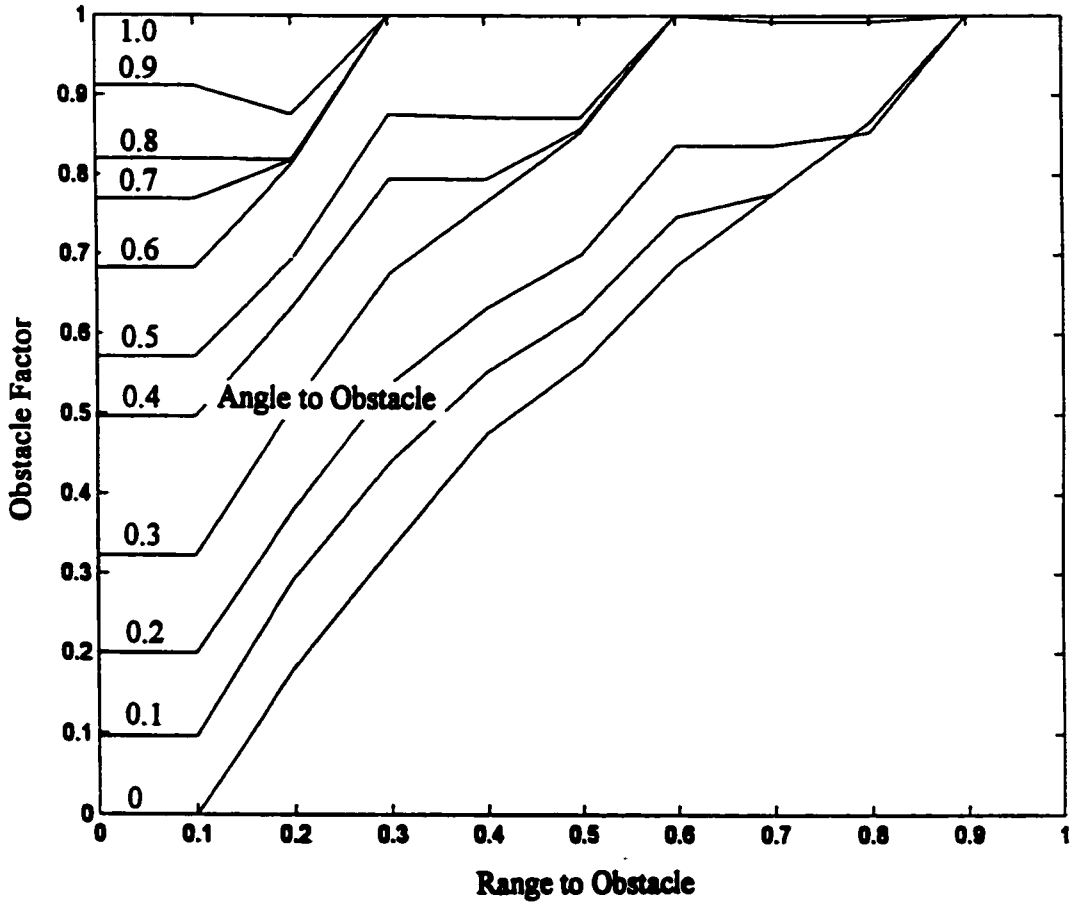


Figure 5 42. Obstacle factor vs. range to obstacle as a function of angle to obstacle with steering angle of zero and speed of 1.

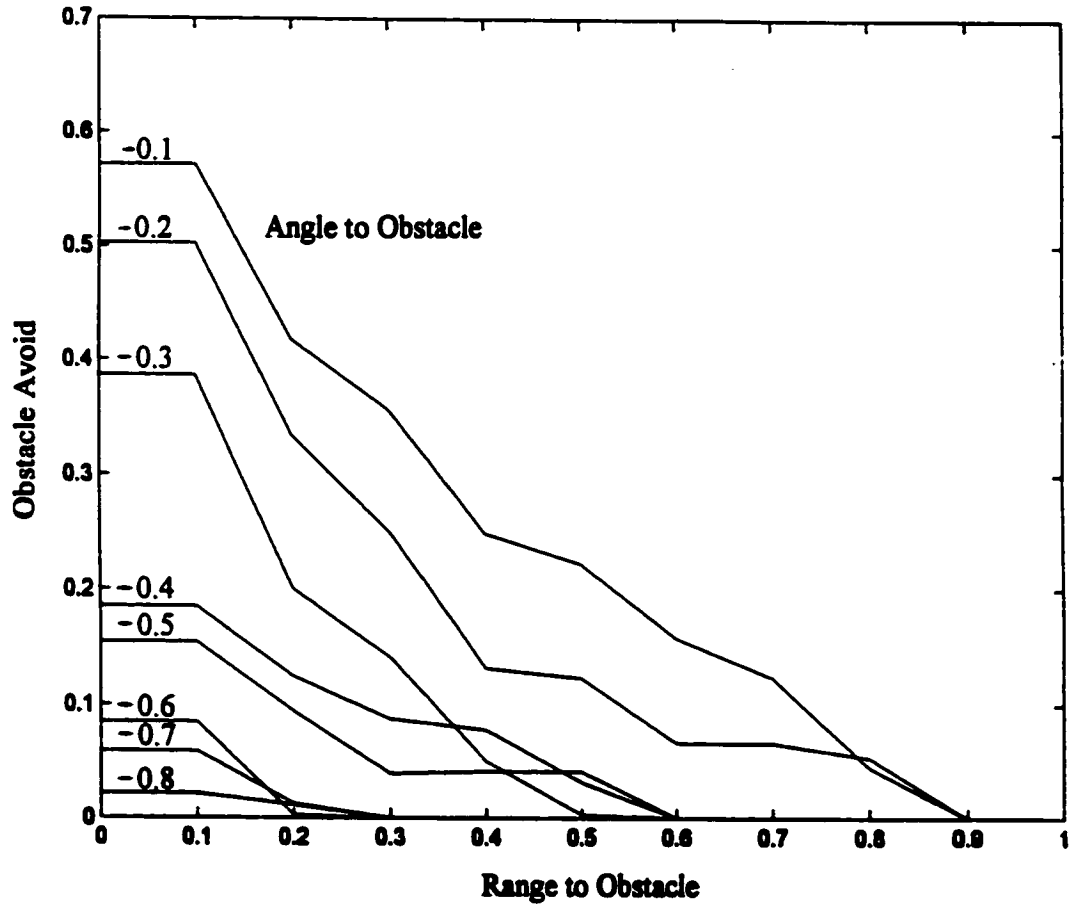


Figure 5 43. Obstacle avoid steering adjustment vs range to obstacle as a function of angle to obstacle with steering angle of zero and speed of 1.

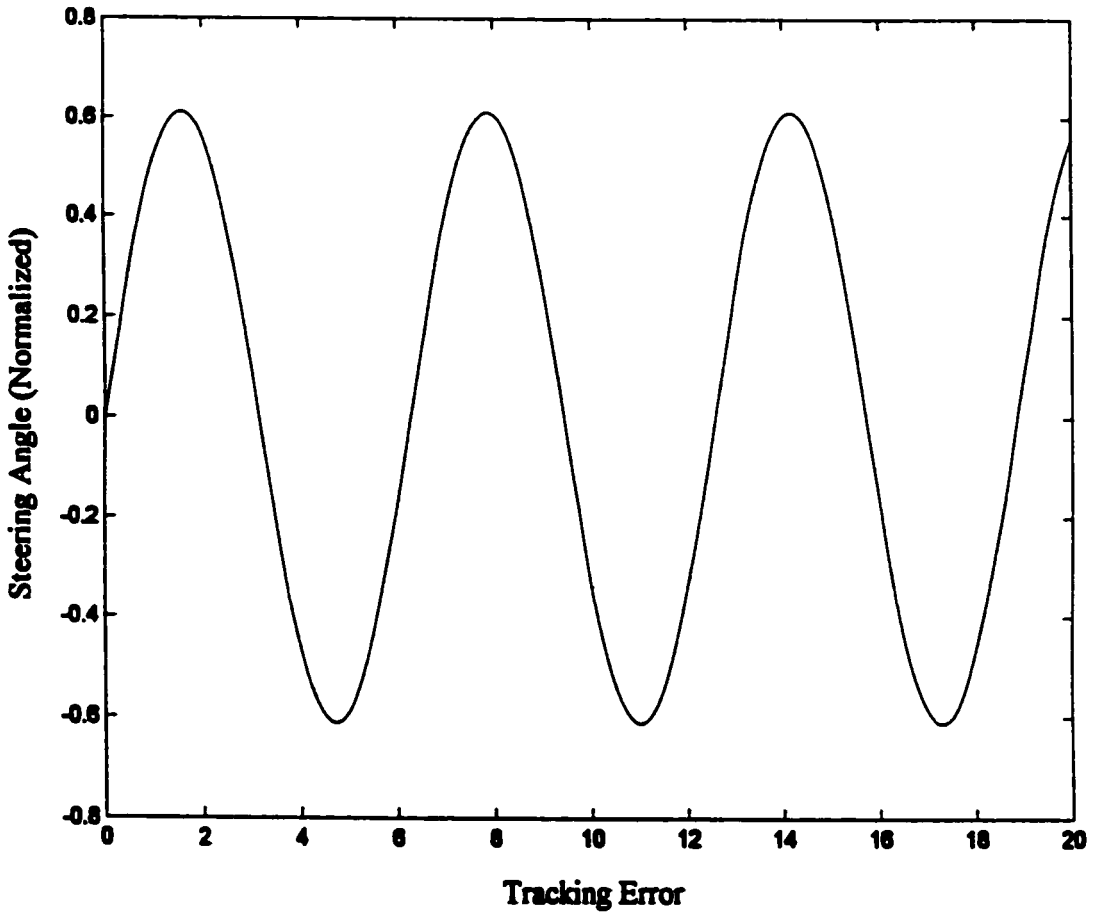


Figure 5.44. Steering angle response to tunnel-tracking error.

6. CONCLUSIONS

6.1 Summary and Conclusions

This thesis presents the results of a research project to develop a novel design for remotely controlling a robot vehicle over a time-delayed communication link. A fundamental requirement was to incorporate methods of intelligent control [28] wherever these concepts seemed proper to offer their unique approaches and solutions. This has been accomplished. A multivariable fuzzy control system is the basis for the system. A neural network predictor is included to provide the human operator with a non-delayed estimate of the system output. The fundamental technique used in compensating for the time delay is Smith control. While this technique was originated in the 1950's for process control problems, the evolution of technology (primarily digital computers and digital processing) enhances its viability in various applications of control involving time delay in the loop. It has been shown that Smith control can be simplified in remote-control applications [11]. Its capability of accomplishing time-delay compensation is very obvious from the results obtained and presented in this thesis. Simulations using a linear plant with no disturbances show that end-to-end closed-loop control is essentially independent of the time delay.

Since the primary mode of control in this system is teleoperation, the additional complication of incorporating a human operator in the system has been a significant extra challenge. Modeling human operators is not a trivial task. Since a human operator introduces additional time-response dynamics in teleoperated systems, it has been discussed herein and shown how detrimental this can be for control, particularly if time delay is present. These additional dynamics must be compensated for if the system is to be able to function with time delay in the loop. It has been shown that it is particularly convenient to accomplish this by using the same simplified Smith control system as used for the time delay. This does require a good model of the operator's dynamics for Smith control compensation. This is easy in simulations. How well it would work in reality is a question that can only really be answered by testing human subjects in controlled tests. It is known that in relatively straightforward manual

control tasks (with no delay or very small delay in the loop) that a human operator will compensate for the system dynamics to a high degree. All trained human operators such as automobile drivers or airplane pilots do this automatically. The resulting response is approximately modeled by the crossover model [70]. This is essentially the frequency response of the human's neuromuscular dynamics in the crossover region (ie.0 dB gain). This frequency is approximately 0.4 Hz on average for most people. Therefore, the transfer function model of the human operator used in this system should be valid. The actual time constant used may need some tuning depending on the individual operator, but the vast capability of the human information-processing system can most likely provide the necessary adjustment. It has already been discussed that a human operator will tend to adopt the necessary equalization required to accomplish a particular control task as long as it is within a human's capability. This cannot be modeled easily. This suggests an area for further work involving human operator testing to determine this capability and the degree of sensitivity to specific models. Another issue with human operators is reaction delay. This could be compensated for with the system delays in the simplified Smith controller, but it has been shown that with preview ability, such as seeing the road (or tunnel) ahead, an operator can compensate for reaction delay by using precognitive control. On this basis, reaction delay was not considered to be particularly significant and was not included in tests.

Multivariable fuzzy control was used in the speed and steering control systems described. The speed control system uses two parallel, error-based loops to provide the necessary feedback control. The speed set primary control loop uses conventional error-detection and operates on an end-to-end basis over the time delay, compensated by using Smith control. The speed factor control loop uses fuzzy error detection since the three speed factor inputs are best described in terms of fuzzy linguistic variables. The steering control system is also a closed-loop multivariable system using fuzzy control. The steering set primary control loop uses conventional error-detection and operates on an end-to-end basis over the time delay, also compensated by using Smith control. The other loops are used for obstacle-avoidance and semi-autonomous modes including tunnel-tracking and automatic steering into intersecting tunnels. In general, the speed and steering control systems performance is excellent.

A system predictor using a neural network has been included at the local terminal to provide the operator with a non-delayed estimate of the system output. This allows the operator to drive the robot vehicle over the time delay by driving the

predictor in real time. This configuration provides an alternate method of remotely controlling a robot vehicle over a time-delayed link. Without feedback information from the remote terminal, this configuration is essentially open-loop and would not require Smith control. Of course, actual (delayed) system output would be fed back to the local terminal to keep the display updated. The operator would then provide commands based on the error between the desired position, heading and motion of the predicted vehicle and the actual updated parameters shown on the display, based on the delayed feedback from the remote terminal. The success of this method of remote control obviously depends on a number of factors including the accuracy of the displayed environment, the length of the time delay and the speed of the vehicle relative to the size of the display. This is another area for further research.

While a complete system has been described in this thesis, it is basically theoretical since its function has primarily been to study the control aspects of the system. It is recognized that additional work is necessary before all of these features could be implemented in an actual system. Certainly, the fundamental concept of using simplified Smith control and the general structure of the basic system should be sound. Similarly, using the alternate configuration, the operator should be able to drive the robot vehicle over the time delay by driving the predictor in real time on the console display. The "ancillary" features including obstacle-avoidance, tunnel-tracking and automatic intersection-turning have only been tested in a rudimentary manner since they really require taking this system to the next level of simulation for final tuning and development. That level would be an animated simulation environment. Without this level of simulation, system and subsystem tests can go only so far.

6.2 Unique Contributions

The contributions presented by the research and development in this thesis are outlined below.

1. This thesis has demonstrated the successful use of Smith control in compensating for time delay in remote-control applications. System simulations have shown that the specific length of the delay is irrelevant (subject to adequate computer memory) in maintaining stable, closed-loop control. A simplified method of Smith control has been used which requires only the loop time delay. Based on a 2 second

loop delay, this delay needs to be measured to an accuracy in the order of ~+/-0.1% to avoid any significant visual degradation in system response.

2. Multivariable fuzzy control has been successfully used in the speed and steering control systems. The fuzzy speed controller uses two parallel closed loops: a conventional error-based loop operating over the time delay (using Smith control compensation) and a fuzzy error-based loop operating at the remote terminal. The fuzzy steering controller includes a conventional error-based loop operating end-to-end across the time delay (using Smith control compensation) and semi-autonomous intersection-turning, tunnel-tracking and obstacle-avoidance in a hierarchical structure.

3. A fuzzy logic obstacle-avoidance subsystem based on the use of obstacle factors has been introduced. These obstacle factors are based not on the present heading of the vehicle but on the steering angle of the wheels, the angle to obstacle and the angles adjacent to the steering angle. This method has been designed to control both the speed and steering angle of the vehicle to avoid collisions with obstacles.

4. Human operator dynamics in the time-delayed control loop can be catastrophic. Compensating for these performance-degrading effects has been successfully accomplished using the same simplified Smith control system as used for the time delay.

6.3 Future Work

A number of areas can be identified for future work:

1. A major disappointment is the failure to develop a neural network predictor that can accommodate the transient response of the system. A true system predictor would accept any initial conditions of the inputs and respond exactly the same as the actual system. This appears to be a significant challenge that must be investigated.

2. The system described in this thesis is based on the use of vehicle speed and steering dynamics models having time constants shorter than that of a human operator. The effect(s) of models with longer time constants should be tested.

3. The next logical stage in development of this system before actual implementation in hardware is to develop a suitable simulation environment where a complete animation of the system can be tested. This project has taken the design as far as reasonably possible using the level of simulation described herein. A dynamic simulation environment is necessary to verify and tune the obstacle-avoidance subsystem as well as the semi-autonomous modes of tunnel-tracking and automatic intersection-turning.

4. With a total system and environment available in animation, this would be an appropriate time to begin testing a human operator in teleoperation trials over time delay using a suitable control device such as a joystick and a video display. This would also be the ideal time and place to observe how the operator prefers to drive the system. This level of system research and development could be very informative and useful for enhancing remote-control system design.

5. Having the system animation available will also allow testing sensitivity of human operator dynamics models with actual operators in the system. This may lead to various unique improvements in the design of teleoperated systems.

REFERENCES

[1] J.L. Adams, "An Investigation of the Effects of the Time Lag Due to Long Transmission Distances Upon Remote Control, Phase II-Vehicle Experiments & Phase III-Conclusions", NASA Technical Note D-1351, National Aeronautics & Space Administration, April 1962.

[2] L. Conway, R.A. Volz and M.W. Walker, "Teleautonomous Systems: Projecting and Co-ordinating Intelligent Action at a Distance", IEEE Transactions on Robotics and Automation: 6(2), 146-158, April 1990.

[3] IEEE Standard Dictionary of Electrical and Electronic Terms, The Institute of Electrical and Electronic Engineers, Inc., New York, 1984.

[4] J.G. Truxal, M.J. Shooman, W.R. Blesser and J.W. Clark, Remote Control, in the Handbook of Telemetry and Remote Control, E.L. Gruenberg, Editor-in-Chief, McGraw-Hill, New York, pp.15-1 to 15-169, 1967.

[5] R.A. Hess, Human-in-the-Loop Control, in The Control Handbook, W.S. Levine, ed., CRC Press & IEEE Press, Boca Raton, FL, pp. 1497-1505, 1996.

[6] W.M. Smith, "Delayed Visual Feedback and Behavior", Science, 132, 1013-1014, October 1960.

[7] O.J.M. Smith, "Closer Control of Loops With Dead Time", Chemical Engineering Progress, 53(5), 217-219, 1957.

[8] K.J. Astrom, "Frequency-Domain Properties of Otto Smith Regulators", International Journal of Control, 26(2), 307-314, 1977.

[9] J.E. Marshall, Control of Time-Delay Systems, Stevenage United Kingdom: Peter Peregrinus Ltd., 1979.

[10] Z.J. Palmor, Time-Delay Compensation – Smith Predictor and its Modifications, in The Control Handbook, W.S. Levine, ed., CRC Press & IEEE Press, Boca Raton, FL, 224-237, 1996.

[11] H.C. Wood and M.R Olson, "Time-Delayed Remote-Control Systems Using Smith Controllers", Proceedings of the IASTED International Conference on Intelligent Systems and Control, Halifax, N.S., 36-39, June 1-3, 1998.

[12] A. Kheddar, R. Chellali and P. Coiffet, Virtual Reality Assisted Teleoperation, in VE Handbook, K.M. Stanney, ed., Lawrence Erlbaum Associates, Inc., Chapter 56, February 2002 (expected).

[13] T.B. Sheridan, Telerobotics, Automation and Human Supervisory Control, The MIT Press, Cambridge, Mass., 1992.

[14] A. Hemami, "Robotics and Automation in Mining", IEEE Canadian Review, 11-15, Fall 1993.

[15] P. Corke, J. Roberts and G. Winstanley, "Vision-Based Control for Mining Automation", IEEE Robotics and Automation Magazine: 5(4), 44-49, December 1998.

[16] S. Scheding, G. Dissanayake, E.M. Nebot and H. Durrant-Whyte, "An Experiment in Autonomous Navigation of an Underground Mining Vehicle", IEEE Transactions on Robotics & Automation: 15(1), 85-95, February 1999.

[17] J. Yen, R. Langari and L.A. Zadeh, eds., Industrial Applications of Fuzzy Logic and Intelligent Systems, IEEE Press, New York, 1995.

[18] P.J. Antsaklis, M. Lemmon and J.A. Stiver, Learning to be Autonomous, in Intelligent Control Systems, M.M. Gupta and N.K. Sinha, eds., IEEE Press, Picataway, NJ, 28-34, 1996.

[19] L.A. Zadeh, Foreward, in Intelligent Control Systems, M.M. Gupta and N.K. Sinha, eds., IEEE Press, Picataway, NJ, xix-xx, 1996.

[20] T.J.Ross, Fuzzy Logic With Engineering Applications, McGraw-Hill, New York, 1995.

[21] S. Haykin, Neural Networks, IEEE Press, Macmillan College Publishing Company, New York, 1994.

[22] N. Wakami, H. Nomura, S. Araki, "Fuzzy Logic for Home Appliances", in Fuzzy Logic and Neural Network Handbook, C.H. Chen, ed., McGraw-Hill, New York, pp. 21.1-21.23, 1996.

[23] B. Kosko, Neural Networks and Fuzzy Systems, Prentice-Hall, Englewood Cliffs, N.J., 1992.

[24] T.J. Nelson, M.R. Olson and H.C. Wood, "Long-Delay Telecontrol of Lunar Mining Equipment", Space 98, Proceedings of the Sixth Inernational Conference and Exposition on Engineering, Construction and Operations in Space, Albuquerque, N.M., 477-484, April 26-30, 1998.

[25] T.B. Sheridan and W.R. Ferrell, "Remote Manipulative Control With Transmission Delay", IEEE Transactions on Human Factors in Engineering, 25-29, September 1963.

[26] W.R. Ferrell, "Remote Manipulation With Transmission Delay", IEEE Transactions on Human Factors in Engineering: HFE-6, 24-32, September 1965.

[27] R. Ferrell, "Remote Manipulation With Transmission Delay", NASA Technical Note D-2665, National Aeronautics & Space Administration, February 1965.

[28] L. Adams, "An Investigation of the Effects of the Time Lag Due to Long Transmission Distances Upon Remote Control, Phase I-Tracking Experiments", NASA Technical Note D-1211, National Aeronautics & Space Administration, December 1961.

[29] R.H. Lewis, "Preliminary Lunar Teleoperations Investigations Using a Low-Cost Mobile Robot", Proceedings of the Ninth Princeton/AIAA/Space Studies Institute Conference, 138-143, May 10-13, 1989.

[30] Robert J. Anderson and Mark W. Spong, "Bilateral Control of Teleoperators With Time Delay", IEEE Transactions on Automatic Control: 34(5), 494-501, May 1989.

[31] G. Niemeyer and J-J.E. Slotine, "Stable Adaptive Teleoperation", IEEE Transactions of Oceanic Engineering: 16(1), 152-162, January 1991.

[32] W.S. Kim, B. Hannaford and A.K. Bejczy, "Force-Reflection and Shared Compliant Control in Operator Telemanipulators With Time Delay", IEEE Transactions on Robotics & Automation: 8(2), 176-185, April 1992.

[33] S. Lee and H.S. Lee, "Modeling, Design & Evaluation of Advanced Teleoperator Control Systems With Short Time Delay", IEEE Transactions on Robotics and Automation: 9(5), 607-623, October 1993.

[34] D.A. Lawrence, "Stability and Transparency in Bilateral Teleoperation", IEEE Transactions on Robotics and Automation: 9(5), 624-637, October 1993.

[35] G.M.H. Leung, B.A. Francis and J. Apkarian, "Bilateral Controller for Teleoperators With Time Delay via Mu-Synthesis", IEEE Transactions on Robotics and Automation: 11(1), 105-116, February 1995.

[36] M.S. Triantafyllou & M.A. Grosenbaugh, "Robust Control for Underwater Vehicle Systems With Time Delays", IEEE Journal of Oceanic Engineering: 16(1), 146-151, January 1991.

[37] C.R. Kelly, "Predictor Instruments Look Into The Future", Control Engineering, 86-90, March 1962.

[38] L.C. Fargel and E.A. Ulbrich, "Predictor Displays Extend Manual Operation", Control Engineering, 57-60, August 1963.

[39] C.R. Kelly, "Predictor Displays – Better Control for Complex Manual Systems", Control Engineering, 86-90, August 1967.

[40] J.E. Arnold and Paul W. Braisted, "Design and Evaluation of a Predictor For Remote Control Systems Operating With Signal Transmission Delays", NASA Technical Note D-2229, National Aeronautics & Space Administration, December 1963.

[41] L. Stark, W.S. Kim, F. Tendick, B. Hannaford, S. Ellis, M. Denome, M. Duffy, T. Hayes, T. Jordan, M. Lawton, T. Mills, R. Peterson, K. Sanders, M. Tyler and S. Van Dyke, "Telerobotics: Display, Control & Communication Problems', IEEE Transactions of Robotics & Automation: 3(1), 67-75, February 1987.

[42] J.M.E. van de Vegte, P. Milgram and R.H. Kwong, "Teleoperator Control Models: Effects of Time Delay and Imperfect System Knowledge", IEEE Transactions on Systems, Man and Cybernetics: 20(6) 1258-1272, November/December 1990.

[43] T.B. Sheridan, "Space Teleoperation Through Time Delay: Review & Prognosis", IEEE Transactions on Robotics & Automation: 9(5), 592-606, October 1993.

[44] G. Hirzinger, B. Brunner, J. Dietrich and J. Heindl, "Sensor-Based Space Robotics-ROTEX and its Telerobotic Features", IEEE Transactions on Robotics and Automation: 9(5), 649-663, October 1993 .

[45] W.S. Kim and A.K. Bejczy, "Demonstration of a High-Fidelity Predictive/Preview Display Technique for Telerobotic Servicing in Space", IEEE Transactions on Robotics & Automation: 9(5), 698-702, October 1993.

[46] S.G. Lanteigne, "Control of Telemanipulators With Large Time Delays", M.Sc.E.E. Thesis, University of New Brunswick, May 1994.

[47] W.R. Ferrell and T.B. Sheridan, "Supervisory Control of Remote Manipulation", IEEE Spectrum: 4(10), 81-88, October 1967.

[48] D. Pivirotto,"Rovers! Using Mobile Robots as Planetary Explorers", The Planetary Report, XI(4), 8-13, July/August 1991.

[49] T.S. Lindsay, Teleprogramming: Remote Site Robot Task Execution, PhD Dissertation, University of Pennsylvania, 1992.

[50] M.R. Stein, Behavior-Based Control for Time-Delayed Teleoperation, PhD Dissertation, University of Pennsylvania, 1994.

[51] C.M. Anderson, "Advancing Our Ambitions: The 1994 Mars Rover Test", The Planetary Report, XIV(5), 16-17, 1994.

[52] B.S. Graves, A Generalized Teleautonomous Architecture Using Situation-Based Action Selection, PhD Dissertation, Texas A & M University, 1995.

[53] G. Meinsma and H. Zwart, "On H_∞ Control for Dead-Time Systems", IEEE Transactions on Automatic Control: 45(2), 272-285, February 2000.

[54] R.R. Šelmić and F.L. Lewis, "Deadzone Compensation in Motion Control Systems Using Neural Networks", IEEE Transactions on Automatic Control: 45(4), 602-613, April 2000.

[55] H.-X. Li and S.K. Tso, "Higher-Order Fuzzy Control Structure for Higher Order or Time-Delay Systems", IEEE Transactions on Fuzzy Systems, 7(5), 540-552, October 1999.

[56] F.L. Lewis, W.K. Tim, L.-Z. Wang and Z.X. Li, "Deadzone Compensation in Motion Control Systems Using Adaptive Fuzzy Logic Control", IEEE Transactions on Control Systems Technology, 7(6), 731-742, November 1999.

[57] K. Taylor and B. Dalton, "Internet Robots: A New Robotics Niche", IEEE Robotics and Automation Magazine, 7(1), 27-34, March 2000.

[58] K. Goldberg, S. Gentner, C. Sutter and J. Wiegley, "The Mercury Project: A Feasibility Study for Internet Robots", IEEE Robotics and Automation Magazine, 7(1), 35-40, March 2000.

[59] P. Saucy and F. Mondada, "KhepOnTheWeb: Open Access to a Mobile Robot on the Internet", IEEE Robotics and Automation Magazine, 7(1), 41-47, March 2000.

[60] D. Schulz, W. Burgard, D. Fox, S. Thrun and A. Cremers, "Web Interfaces for Mobile Robots in Public Places", IEEE Robotics and Automation Magazine, 7(1), 48-56, March 2000.

[61] R.C. Luo and T.M. Chen, "Development of a Multibehavior-Based Mobile Robot for Remote Supervisory Control Through the Internet", IEEE Transactions on Mechatronics, 5(4), 376-385, December 2000.

[62] J.M. Leslie, "Effects of Time Delay in the Visual Feedback Loop of a Man-Machine System", NASA Contractor Report, NASA CR-560, National Aeronautics & Space Administration, September 1966.

[63] D. McRuer and E. Krendel, Dynamic Response of Human Operators, Technical Report WADC-TR-56-524, U.S. Air Force,1957.

[64] E. Gabay and S.J. Merhav, "Identification of a Parametric Model of the Human Operator in Closed-Loop Control Tasks", IEEE Transactions on Systems, Man and Cybernetics: 7(4), 284-292, April 1977.

[65] M. Tomizuka and M. Fujimura, "Extended Signal Quickening for Manual Control", IEEE Transactions on Systems, Man and Cybernetics: 9(10), 668-676, October 1979.

[66] F. Osafo-Charles, G.C. Agarwal, W.D. O'Neill and G.L. Gottlieb, "Application of Time-Series Modeling to Human Operator Dynamics", IEEE Transactions on Systems, Man and Cybernetics: 10(12), 849-860, December 1980.

[67] P.W. Koken, H.J.J. Jonker and C.J. Erkelens, "A Model of the Human Smooth Pursuit System Based on an Unsupervised Adaptive Controller", IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans: 26(2), 275-280, March 1996.

[68] G.O.A. Zapata, R.K.H. Galvao and T. Yoneyama,"Extracting Fuzzy Control Rules From Experimental Human Operator Data", IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics: 29(3), 398-406, June 1999.

[69] M. Pachter and C.H. Houpis, "Flight Control of Piloted Aircraft", in The Control Handbook, W.S. Levine, ed., CRC Press & IEEE Press, Boca Raton, FL, 1301-1302, 1996.

[70] D. McRuer, "Human Dynamics in Man-Machine Systems", Automatica, 16, 237-253, 1980.

[71] D.T. McRuer, R.W.Allen, D.H.Weir and R.H.Klein, "New Results in Driver Steering Control Models", Human Factors, 19(4), 381-397, 1977.

[72] C.C. MacAdam, "Application of an Optimal Preview Control for Simulation of Closed-Loop Automobile Driving", IEEE Transactions on Systems, Man and Cybernetics: 11(6), 393-399, June 1981.

[73] D.L. Kleinman, S. Baron and W.H. Levison, "An Optimal Control Model of Human Response, Part 1: Theory and Validation", Automatica, 6, 357-369, 1970.

[74] G.O. Burnham and G.A. Bekey, "A Heuristic Finite-State Model of the Human Driver in a Car-Following Situation", IEEE Transactions on Systems, Man and Cybernetics: 6(8), 554-562, August 1976.

[75] K.D. Enstrom and W.B. Rouse, "Real-Time Determination of how a Human Has Allocated His Attention Between Control and Monitoring Tasks", IEEE Transactions on Systems, Man and Cybernetics: 7(3), 153-161, March 1977.

[76] D.H. Lee and K.W. Han, "Analysis of Human Operator's Behavior in High-Order Dynamic Systems", IEEE Transactions on Systems, Man and Cybernetics: 10(4), 207-213, April, 1980.

[77] R.A. Hess and B.D. McNally, "Automation Effects in a Multiloop Manual Control System", IEEE Transactions on Systems, Man and Cybernetics: 16(1), 111-121, January/February 1986.

[78] A. Abdel-Malek and V.Z. Marmarelis, "Modeling of Task-Dependent Characteristics of Human Operator Dynamics During Pursuit Manual Tracking", IEEE Transactions on Systems, Man and Cybernetics: 18(1), 163-172, January/February 1988.

[79] R.A. Hess, "Pursuit Tracking and Higher Levels of Skill Development in the Human Pilot", IEEE Transactions on Systems, Man and Cybernetics: 11(4), 262-273, April 1981.

[80] M. Yoshizawa and H. Takeda, "An Output Regulation Model for Human Input Adaptibility in the Manual Control System", IEEE Transactions on Systems, Man and Cybernetics: 18(2), 193-203, March/April 1988.

[81] D.E. Greene, R.E. Barr, C. Fulcher, L. Hwang and S.G.K. Rao, "A Stochastic Sequential Model for Man-Machine Tracking Systems", IEEE Transactions on Systems, Man and Cybernetics: 18(2), 316-326, March/April 1988.

[82] R.A. Hess,"Modeling the Effects of Display Quality upon Human Pilot Dynamics and Perceived Vehicle Handling Qualities", IEEE Transactions on Systems, Man and Cybernetics: 25(2), 338-344, February 1995.

[83] E.R. Boer and R.V. Kenyon, "Estimation of Time-Varying Delay Time in Nonstationary Linear Systems: An Approach to Monitor Human Operator Adaptation in Manual Tracking Tasks", IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans: 28(1), 89-99, January 1998.

[84] E. Donges, "A Two-Level Model of Driver Steering Behavior", Human Factors, 20(6), 691-707, 1978.

[85] A.T. Bahill, "A Simple Adaptive Smith Predictor for Controlling Time-Delay Systems", IEEE Control Systems Magazine, 3(2), 16-22, May 1983.

[86] T. Furukawa and E. Shimemura, "Predictive Control For Systems With Time Delay", International Journal of Control, 37(2), 399-412, 1983.

[87] K. Watanabe and M. Ito, "A Process-Model Control for Linear Systems With Delay", IEEE Transactions on Automatic Control, 26(6), 1261-1269, December 1981.

[88] T. Hagglund, "A Predictive PI Controller for Processes With Long Dead Times", IEEE Control Systems Magazine, 12(1), 57-60, February 1992.

[89] K.J. Astrom, C.C. Hang and B.C. Lim, "A New Smith Predictor for Controlling a Process With an Integrator and Long Dead-Time", IEEE Transactions on Automatic Control, 39(2), 343-345, February 1994.

[90] M.R. Matausek and A.D. Micic, "A Modified Smith Predictor for Controlling a Process With an Integrator and Long Dead-Time", IEEE Transactions on Automatic Control, 41(8), 1199-1203, August 1996.

[91] P. Blaha and P. Pivonka, "Intelligent Corrector for the Systems with Time Delay", 42nd International Scientific Colloquium, Ilmenau, 637-641, 1997.

[92] M.R. Matausek and A.D. Micic, "On the Modified Smith Predictor for Controlling a Process With an Integrator and Long Dead-Time", IEEE Transactions on Automatic Control, 44(8), 1603-1606, August 1999.

[93] J.E. Normey-Rico and E.F. Camacho, "Robust Tuning of Dead-Time Compensators for Processes with an Integrator and Long Dead Time", IEEE Transactions on Automatic Control, 44(8), 1597-1603, August 1999.

[94] Z.-Q. Wang and S. Skogestad, "Robust Control of Time-Delay Systems Using the Smith Predictor", Int. J. of Control, 57(6), 1405-1420, 1993.

[95] E. Cox, The Fuzzy Systems Handbook, AP Professional, Cambridge, MA, 97-98, 1994.

[96] M.K. Liubakka, D.S. Rhode, J.R. Winkelman, P.V. Kokotovic, "Adaptive Automotive Speed Control", in The Control Handbook, W.S. Levine, ed., CRC Press & IEEE Press, Boca Raton, FL, 1274-1285, 1996.

[97] T.D. Gillespie, Fundamentals of Vehicle Dynamics, Society of Automotive Engineers, Inc., Warrendale, PA, 21-42, 1992.

[98] J.Y. Wong, Theory of Ground Vehicles, John Wiley and Sons, New York, pp. 149-153, 1978.

[99] J.Y. Wong, Theory of Ground Vehicles, John Wiley and Sons, New York, pp. 210-239, 1978.

[100] J.R. Ellis, Venicle Dynamics, Business Books, London, England, pp. 60-92, 1969.

[101] J.C. Breidenthal and T.A. Komarek, "Radio Tracking System", in Deep Space Telecommunications Systems Engineering, J.H. Yuen, ed., Plenum Press, New York, p.144, 1983.

APPENDIX

The general test configuration of the basic remote-control system modeled and simulated using Matlab/Simulink is presented in this Appendix. Each subsystem diagram in the figures uses the same nomenclature as used in the next higher level. For example, the fuzzy steering control diagram in Figure A.14 is the expanded view of the fuzzy steering control block in the RTC steering control diagram shown in Figure A.13. This latter diagram, in turn, is the expanded view of the RTC steering control block in the remote teleautonomous controller diagram shown in Figure A.8 and so on up to the complete system block diagram shown in Figure A.1.

The system description has been covered in Chapter 4. The simulation diagrams here show all the detailed components required or included in the various simulations. These components include scaling amplifiers, constants, limiters, switches and blocks for performing algebraic and logical operations. Scaling amplifiers are used to provide adequate ranges for the particular parameters by amplifying or attenuating the signals as necessary. Constants are included to provide the necessary DC shift in level for the same reason. This will be described in each relevant section.

A. Matlab/Simulink Basic System Configuration

A.1 Overall Test Configuration

The top-level diagram used in performing system simulations is shown in Figure A.1. Step-function or sinusoidal test signals for speed set and steering set inputs are selected by the input switches and monitored using the input oscilloscopes. The actual system speed and steering angle outputs are multiplied by the scaling amplifiers of gain 2.6 and 1.01, respectively, to provide their full ranges of [0 1] and [-1 1], respectively. The predicted system outputs do not require scaling gains since they are included in the neural network predictor. These output signals are passed through offset test delays that are set to the one-way system delays in order to perform comparisons of the predicted outputs with the actual outputs. The various multiplexers, oscilloscopes and Simulink output ports allow viewing and plotting the four output signals in appropriate combinations.

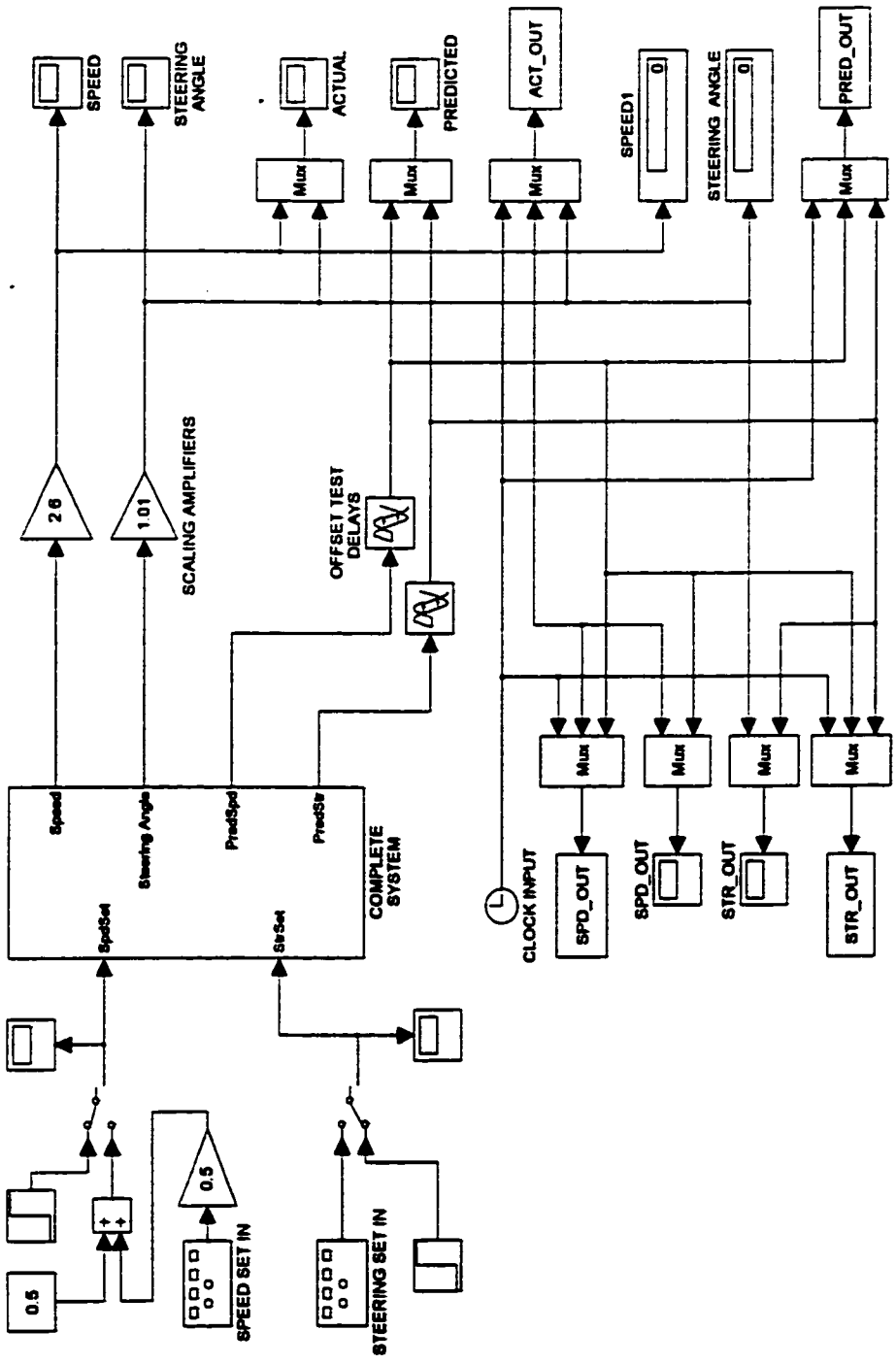


Figure A.1. Overall system test configuration

A.2. Complete System

The major system blocks are shown in Figure A.2. This shows the general layout of the system with the forward and reverse signal paths between the various blocks and over the system (time) delays. The constant block inputs to the RTC represent the signals expected to be provided by an externally-provided sensor-processing system located on-board. These signals are the angle to intersection (IntAng), range to intersection (IntRan), angle to obstacle (ObAng), range to obstacle (ObRan) and tunnel-tracking error (TrackErr). For simulations, these inputs have been connected to appropriate step-function or sine wave generators.

A.3. Operator/Console

The operator/console shown in Figure A.3 includes the human operator dynamics for both the actual and predicted paths. An input scaling gain of 0.4 is included in the speed path to provide the proper scaling for the correct range (ie. [-2 2]) for the speed set error signal from the operator.

A.4. Operator Speed/Steering Error Response

Figure A.4 shows the transfer function of the human operator used in this project. The switches have been used to test the various configurations, primarily human operator in and out of the loop. The reaction delay is trivial to compensate for with the Smith controller, but as discussed previously, it is expected that the operator can provide the necessary compensation given a video view of the path in front of the robot vehicle. The reaction delay has not been included in any of the simulations. The error-detector is included since the operator is expected to provide this function.

A.5. Operator Speed/Steering Response

Figure A.5 provides the same human operator dynamics as the previous figure for the predicted output without the error-detector.

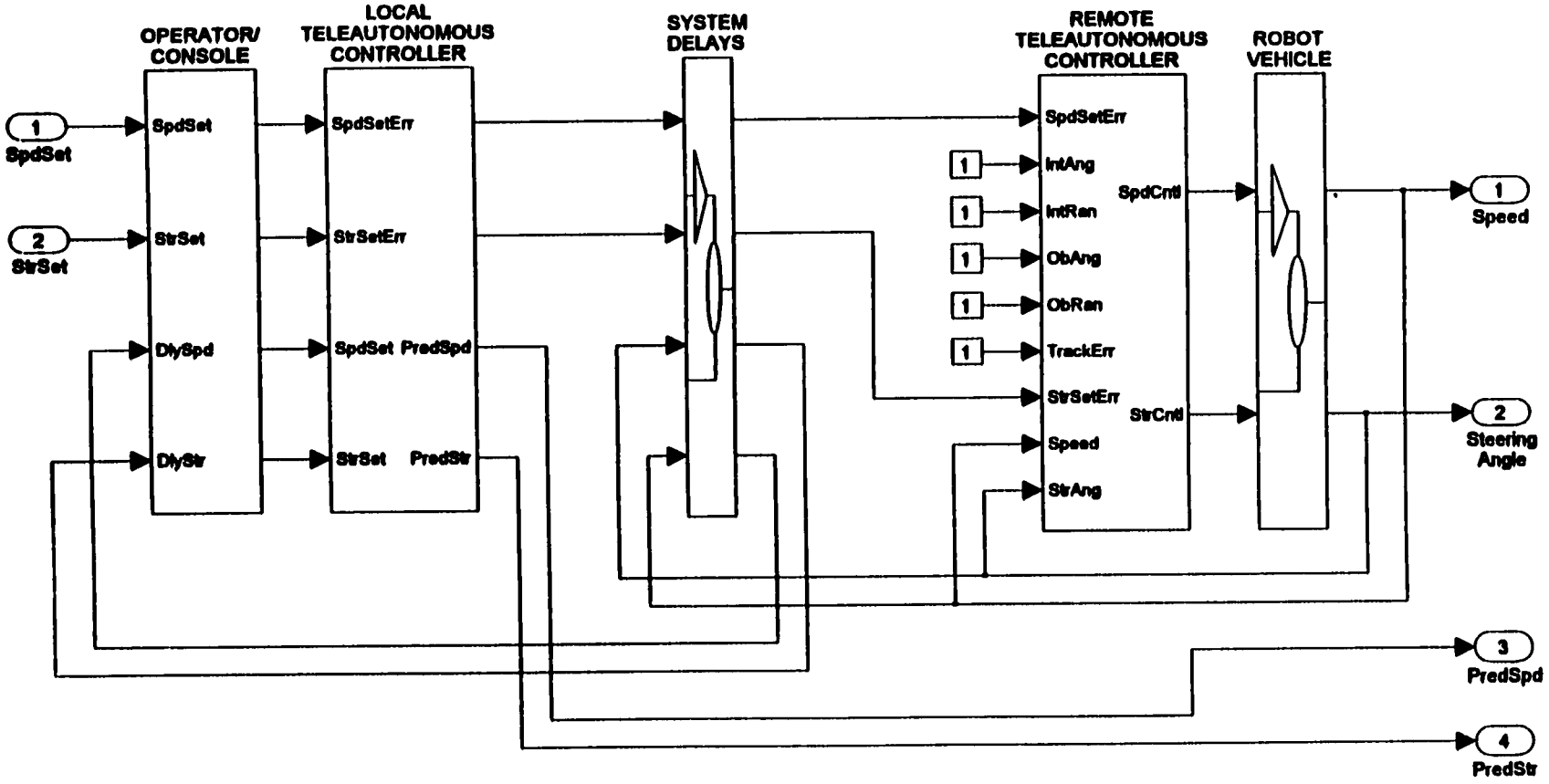


Figure A.2. Complete system.

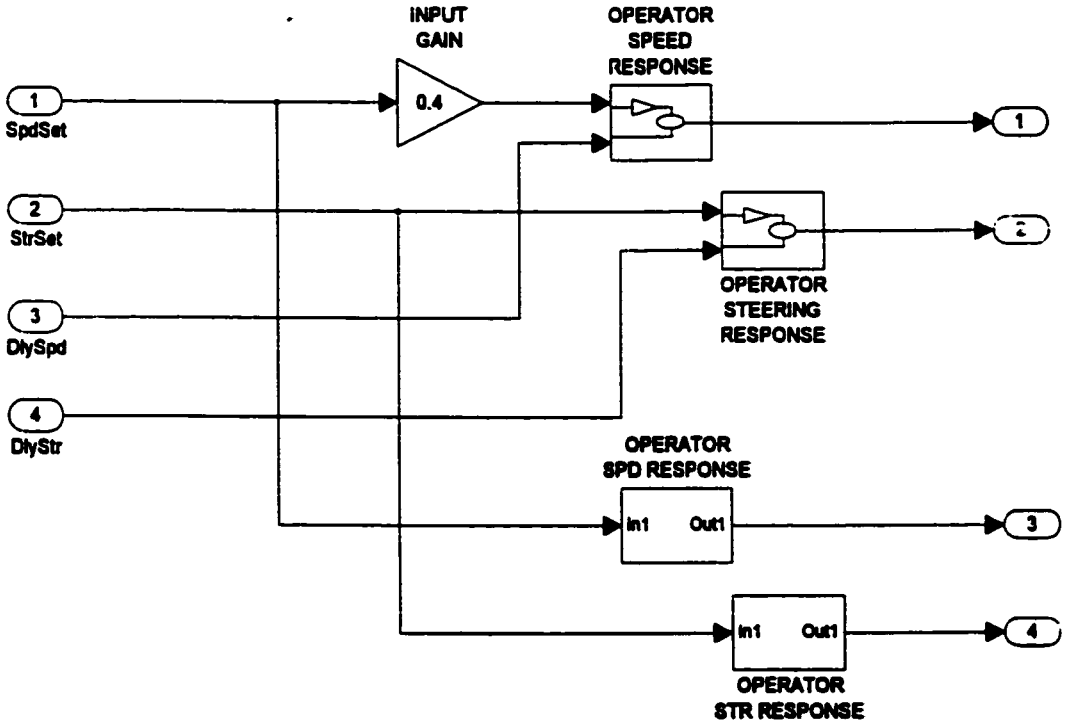


Figure A.3. Operator/console.

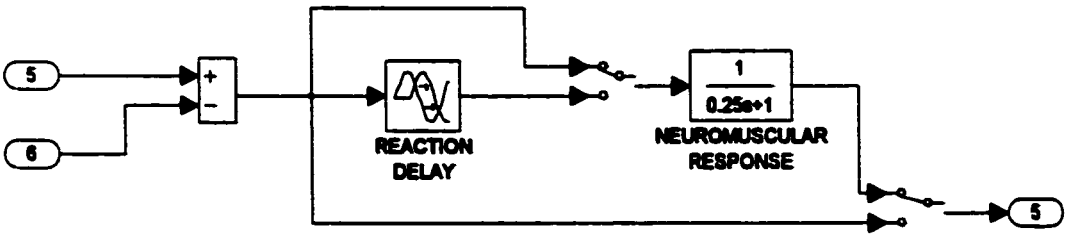


Figure A.4. Operator speed/steering error response.

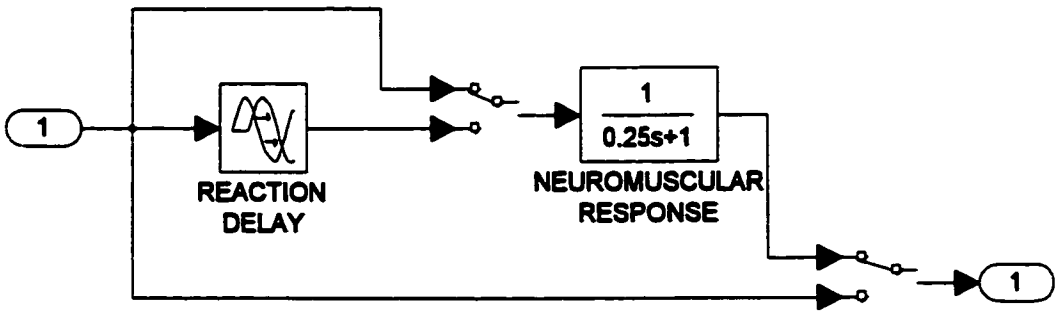


Figure A.5. Operator speed/steering response.

A.6. Local Teleautonomous Controller

The only component in the LTC as shown in Figure A.6 that is relevant to this thesis is the predictor. This is the neural network-based subsystem that provides the (steady-state) model of the subsequent system without the time delay so the operator can see the immediate effects of the control commands without waiting for the delayed feedback from the actual vehicle. The operator's command inputs, speed set error (SpdSetErr) and steering set error (StrSetErr) pass directly through the LTC.

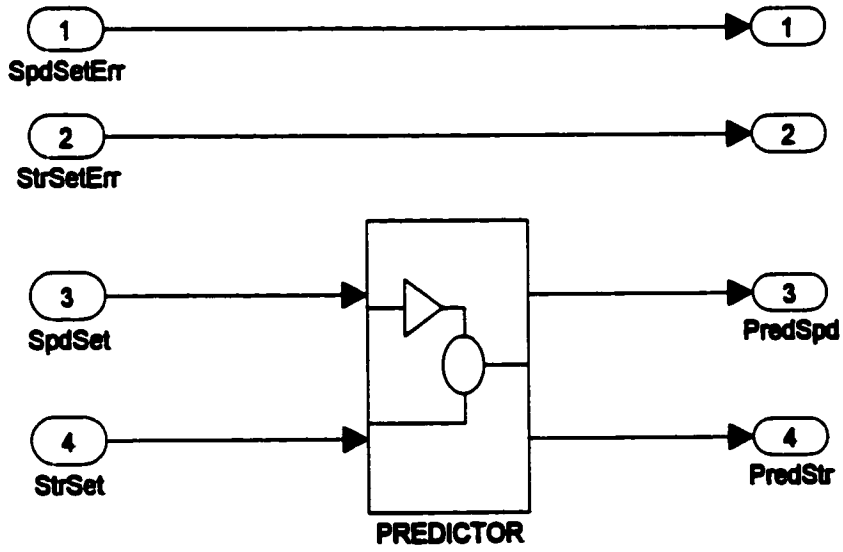


Figure A.6. Local teleautonomous controller.

A.7. System Delays

The forward and reverse system time delays are shown in Figure A.7.

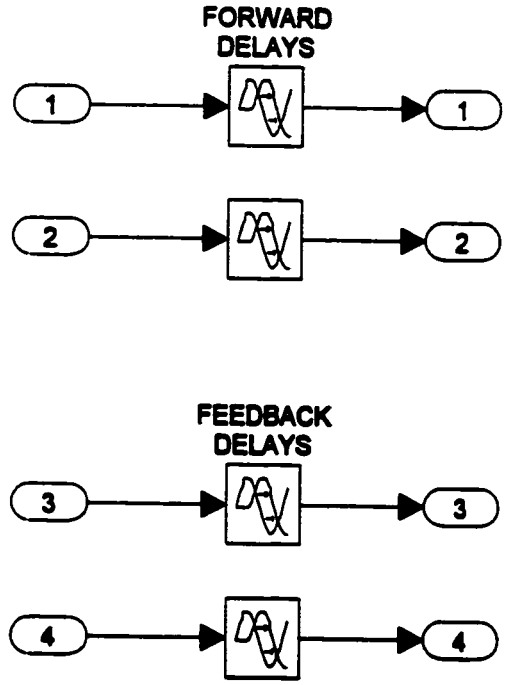


Figure A.7. System delays.

A.8. Remote Teleautonomous Controller

Figure A.8 shows the major subsystem blocks in the RTC. These are the obstacle avoidance, RTC speed control and RTC steering control blocks along with their various input and output signals. The speed set error (SpdSetErr) and steering set error (StrSetErr) inputs are the control commands from the LTC. Speed and steering angle (StrAng) inputs are the feedback outputs from the vehicle. The other inputs are from the on-board signal-processing system. The outputs of the obstacle avoidance block, obstacle factor (ObFac) and obstacle avoid (ObstAv) are shown connected to the other two blocks. The two outputs of the RTC are the speed control (SpdCntl) and steering control (StrCntl) signals which are used to drive the speed and steering actuators, respectively, on the robot vehicle.

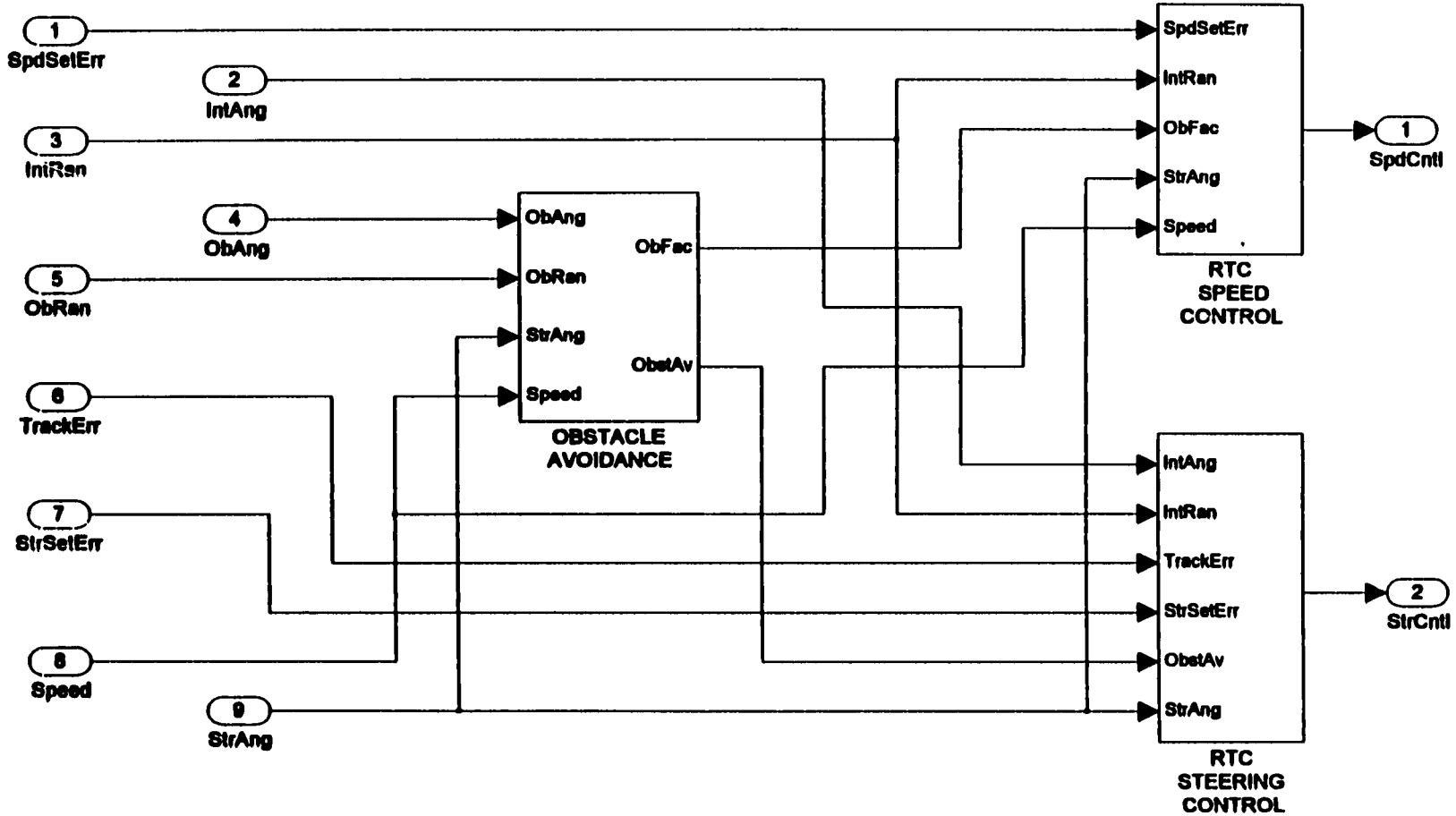


Figure A.8. Remote teleautonomous controller.

A.9. Obstacle-Avoidance

Figure A.9 shows the detailed obstacle-avoidance subsystem. The top row is used for calculating the primary obstacle factor (which is simply referred to as the obstacle factor outside of this subsytem). The two fuzzy controllers are shown along with scaling amplifiers and DC offset. The first scaling amplifier with gain 1.12 was used to ensure the full range [-1 1] of the hazard situation input. The constant 0.1056 was provided to shift the obstacle factor signal level downward to ensure that it would reach a minimum value of zero. The amplification by 1.27 immediately following was necessary to realize a full "dynamic" range from 0 to1. A number of limiters were found to be necessary to ensure that signals beyond a certain range would not cause problems in the simulations. It is quite probable that many of them could be removed in practice, but this is not known with certainty at this time so they have been left in. A number of other blocks, such as comparators and multipliers are used for calculating parameters such as left and right steering adjustments and autosteering control output in Figures A.9 and A.15, respectively.

The second and third rows are used to calculate the left and right obstacle factors, respectively, in an identical manner to the first, except that these values are now based on views that are shifted by 20% on either side of the current steering angle. The (arbitrarily chosen) 20% shifts in the steering angle are seen as DC offsets of +/- 0.2. The presence of the saturation blocks at the inputs is to ensure that the inputs to the first fuzzy controllers do not exceed the range [-1 1] which, of course, would occur when the steering angle exceeds +/- 0.8. The angle to obstacle (ObAng) and range to obstacle (ObRan) inputs represent the obstacle having the lowest obstacle factor (ie. highest danger level). The steering angle (StrAng) and speed inputs are the outputs from the robot vehicle. The time to collision input is calculated by multiplying the range to obstacle by the reciprocal of the speed. Since the speed varies between 0 and 1, a non-zero DC offset of 0.001 has been chosen to allow the reciprocal to vary between 0.999 (~1) and 1000. This is then truncated to the range [1 10] before being multiplied by the range to obstacle. This effectively covers a speed range of [0.1 1], which should be sufficient.

All three of the obstacle factors are used to determine if any steering adjustment is to be made. The left and right obstacle factors are compared to see if one is larger.

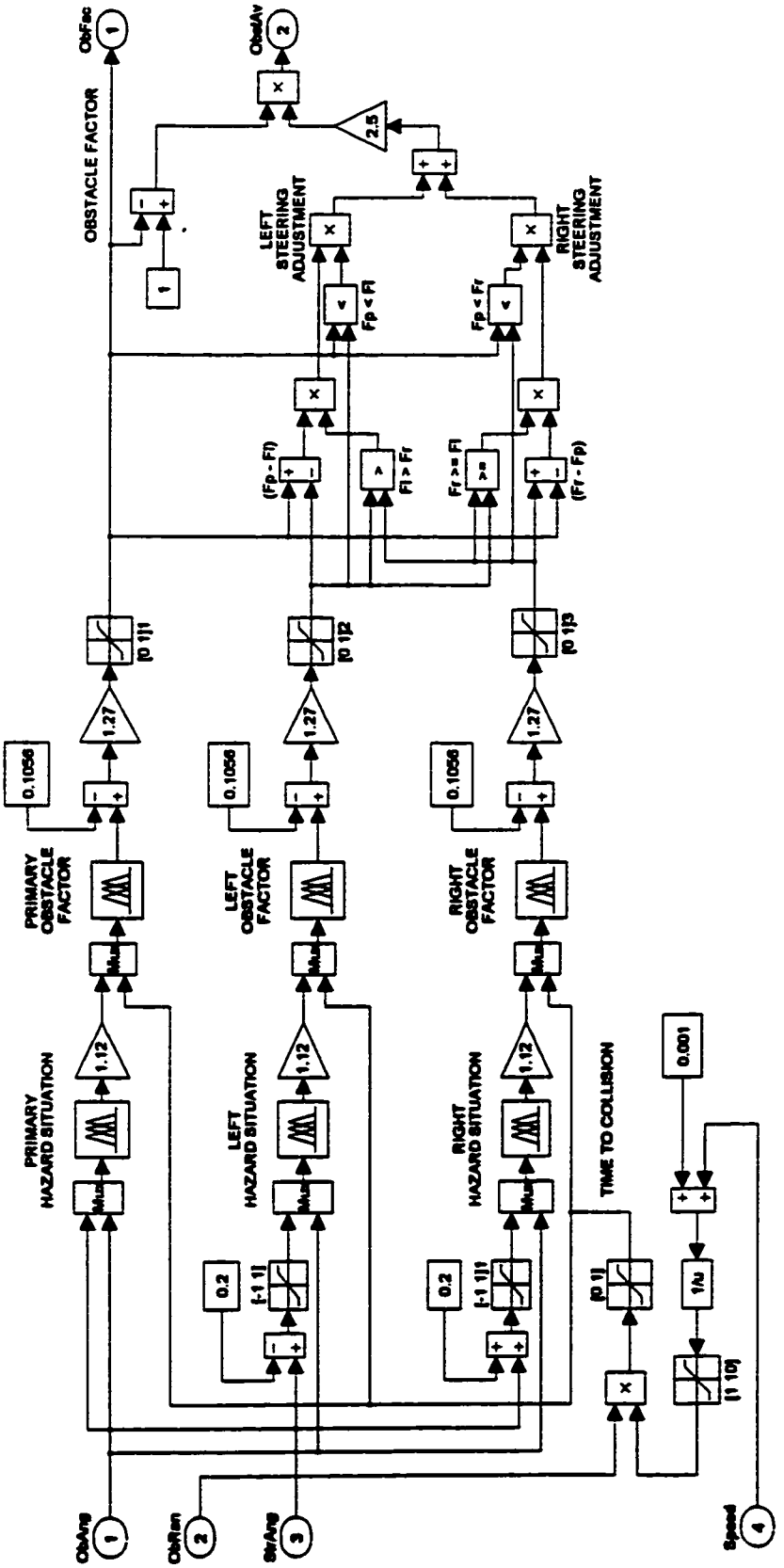


Figure A.9. Obstacle-avoidance.

That value is then compared with the primary obstacle factor to see which of those two is larger, and an appropriate steering adjustment for obstacle avoidance is generated as the output obstacle avoid (ObstAv) signal.

A.10. RTC Speed Control

The block diagram in Figure A.10 shows the speed control subsystem in the RTC. This includes the Smith control block and the fuzzy speed control block with their respective inputs and outputs. The speed set error command from the LTC is first processed in the Smith controller before going to the fuzzy speed control block. The range to intersection (IntRan) is used to restrict the speed of the vehicle to ~ 0.25 (ie. 25%) of maximum as it approaches and passes through intersections without stopping. The range to intersection is disabled as shown in Figure 4.9 during teleoperation mode by supervisory control command. The obstacle factor is designed to slow the vehicle in the presence of obstacles and to stop it if its numerical value reaches a minimum of ~ 0.25. The steering angle is the angle of the wheels referenced to zero (ie. "straight ahead") and measured directly. The dependency on the speed by the steering angle is such that maximum speed will be maintained for steering angles within about +/- 0.25 of zero. Beyond this range, the speed will be progressively reduced with increasing steering angle down to about 0.33 (ie. 33%) of maximum. The final input, speed, is the actual vehicle output speed. A loop gain of 10 has also been included. This was determined empirically during system development to to provide satisfactory system response.

A.11. Fuzzy Speed Control

The fuzzy speed control block contains the two fuzzy speed controllers and the speed factor selector as shown in Figure A.11. This latter block simply takes the minimum value of the three speed factor inputs, angle to intersection (IntAng), obstacle factor (ObFac) and steering angle (StrAng) to ensure that the output speed of the vehicle is appropriately constrained, if necessary. The limiter blocks were added to ensure that the inputs to the fuzzy controllers do not exceed the ranges of the respective universes of discourse.

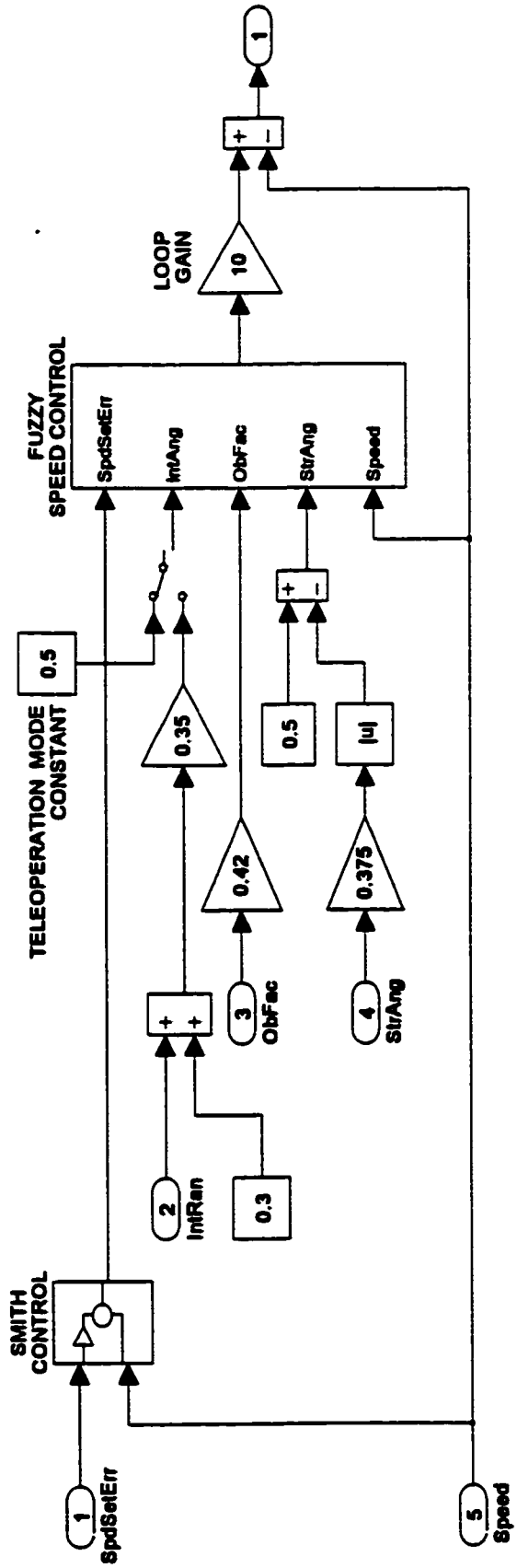


Figure A.10. RTC speed control.

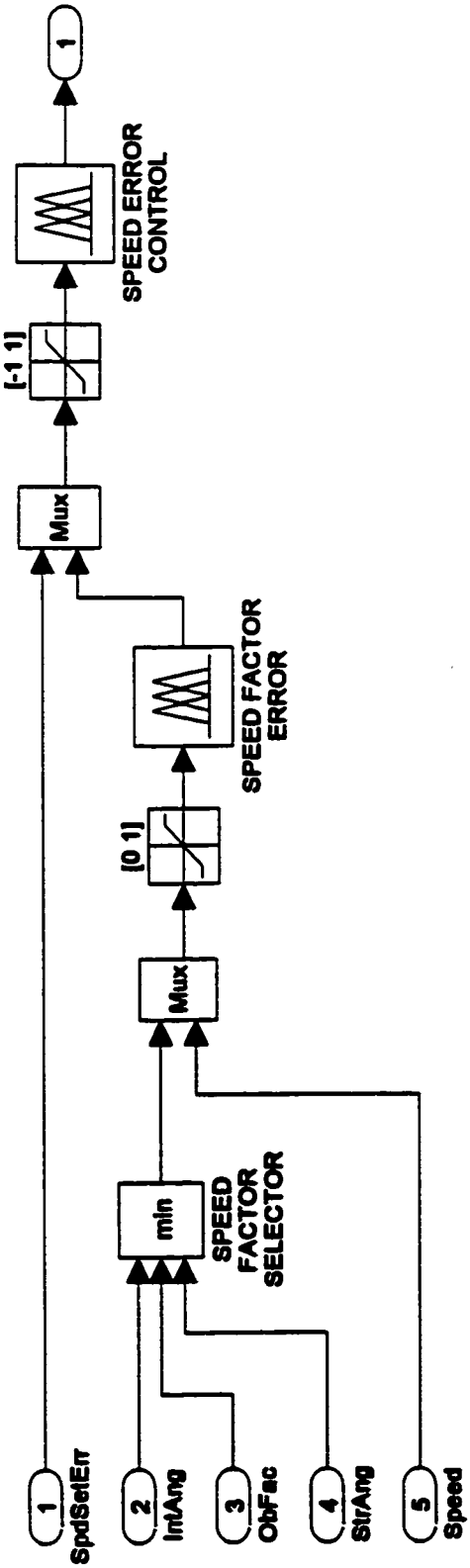


Figure A.11. Fuzzy speed control.

A.12. Smith Control

The Smith control block shown in Figure A.12 includes both the loop delay time and the human operator dynamics. This is essentially the operator's neuromuscular response since the operator's reaction delay has not been included in the simulations. The loop delay switch accommodates testing the system with and without the human operator in the loop. The same block is used for the steering control subsystem.

A.13. RTC Steering Control

This block contains the fuzzy steering control block along with its Smith controller and loop gain as shown in Figure A.13. The steering set error (StrSetErr) command from the LTC is processed by the Smith controller before going to the fuzzy steering control block. The other steering control inputs to the latter block include angle to intersection (IntAng), range to intersection (IntRan), tunnel tracking error (TrackErr) and obstacle avoid (ObstAv). The steering angle (StrAng) is the output from the robot vehicle fed back. The loop gain of 15 was determined empirically to provide satisfactory system response.

A.14. Fuzzy Steering Control

This block contains the fuzzy steering controllers as well as the autosteering control block as shown in Figure A.14. The switches, activated under supervisory control and used for semi-autonomous operation are included. In teleoperation mode, only the steering set error (StrSetErr) command input and the obstacle avoid (ObstAv) input are in the loop. Once again, scaling amplifiers and limiters are included to ensure that the full ranges of inputs and outputs are provided. The fuzzy controller for intersection control is included here.

A.15. Autosteering Control

The autosteering control block shown in Figure A.15 is the implementation of the autosteering control scheme shown in Figure 4.17. The three inputs, as shown in Figure A.14 are intersection control (IntCntl), range to intersection (IntRan) and

tracking error (TrackErr). Depending on the mode selected, these inputs are combined to provide the required autosteering control signal.

A.16. Robot Vehicle

This block includes the speed and steering dynamics models of the robot vehicle as shown in Figure A.16.

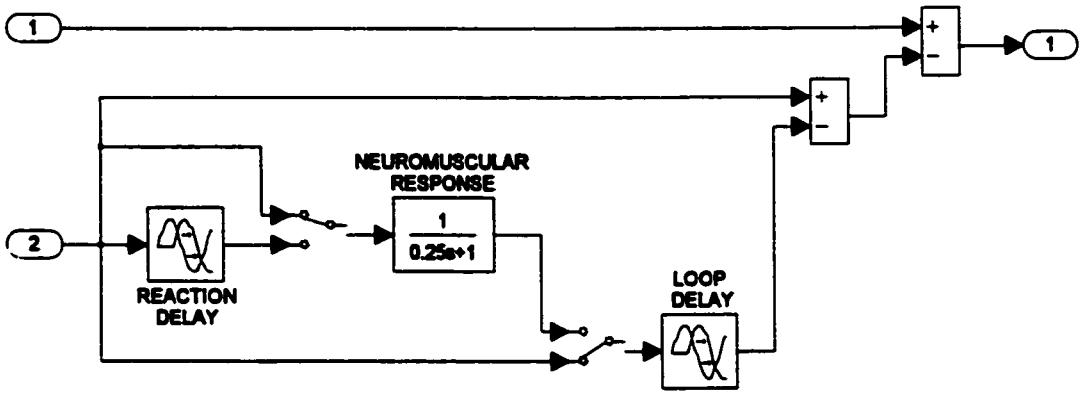


Figure A.12. Smith control.

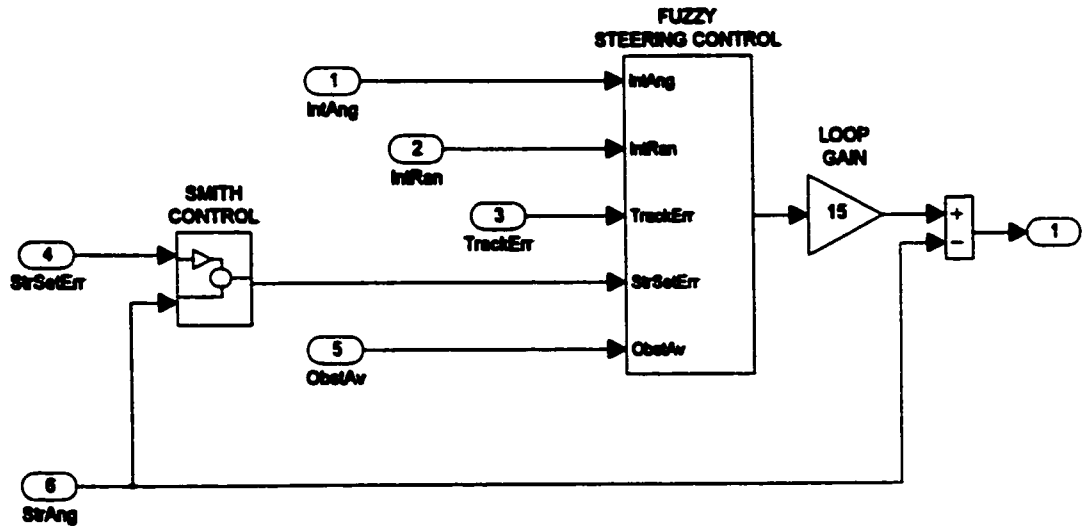


Figure A.13. RTC steering control.

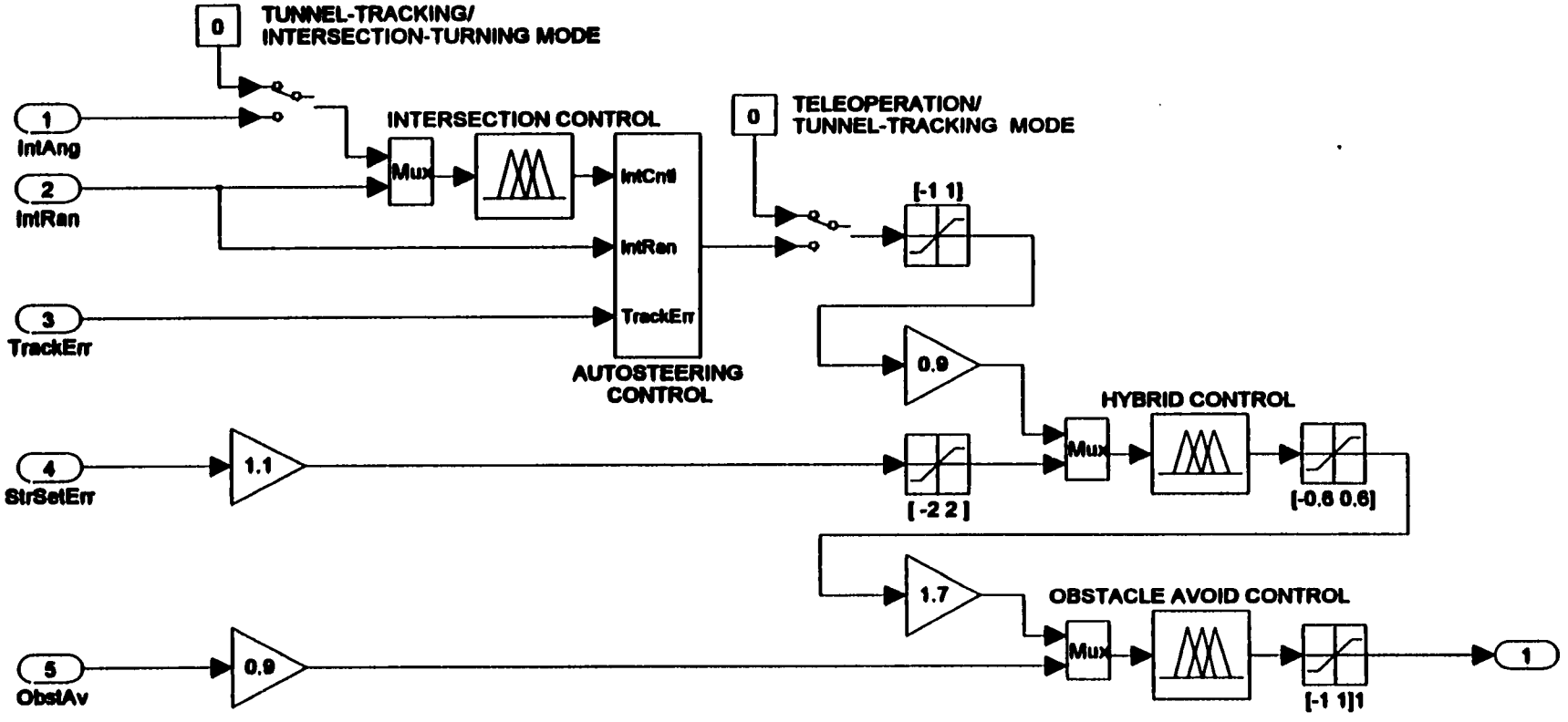


Figure A.14. Fuzzy steering control.

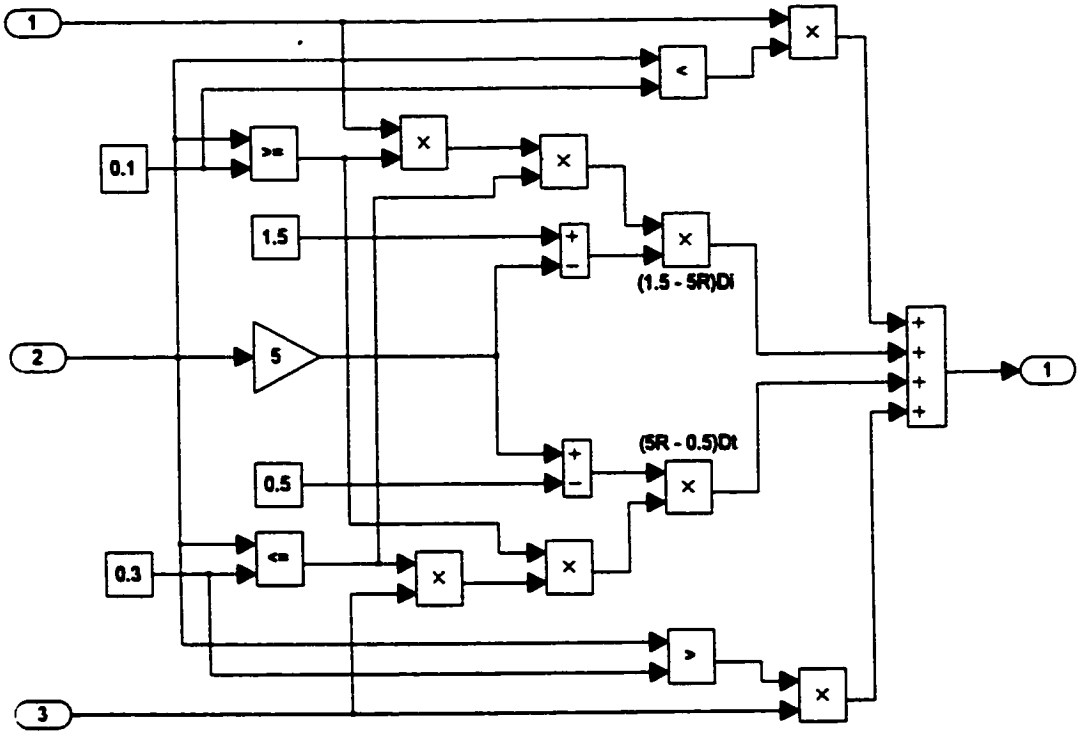


Figure A.15. Autosteering control.

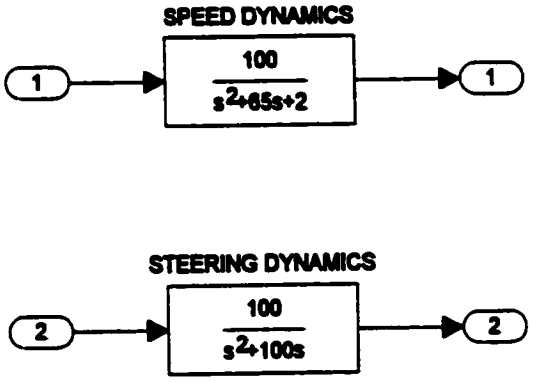


Figure A.16. Robot vehicle.

B. Matlab/Simulink Alternate System Configuration

This Appendix provides the system block diagrams for the alternate configuration described in Chapter 3 and shown in Figure 3.15. Since this configuration is a modification of the basic remote-control system, only those blocks that have been changed from those shown in Appendix A are included. Since this is an open-loop system on an end-to-end basis, the system delays are only in the forward direction and the simplified Smith controllers in the RTC have been bypassed. These changes are not explicitly shown.

B.1. Alternate Complete System

The major system blocks, shown in Figure A.2 are again shown in Figure B.1. This shows the general layout of the modified system with the revised signal paths between the various blocks and over the one-way system delays. The command signals are now the outputs from the predictor.

B.2. Alternate Operator/Console

The alternative operator/console block shown in Figure B.2 includes only the human operator's neuromuscular dynamics.

B.3. Alternate Local Teleautonomous Controller

This block includes only the predictor as shown in Figure B.3.

B.4. Alternate Remote Teleautonomous Controller

The only changes required in the RTC for this alternate configuration shown in Figure B.4 is the inclusion of the input scaling gain of 0.4 and the bypassing of the simplified Smith controllers. This has been discussed in Section 5.4.1.

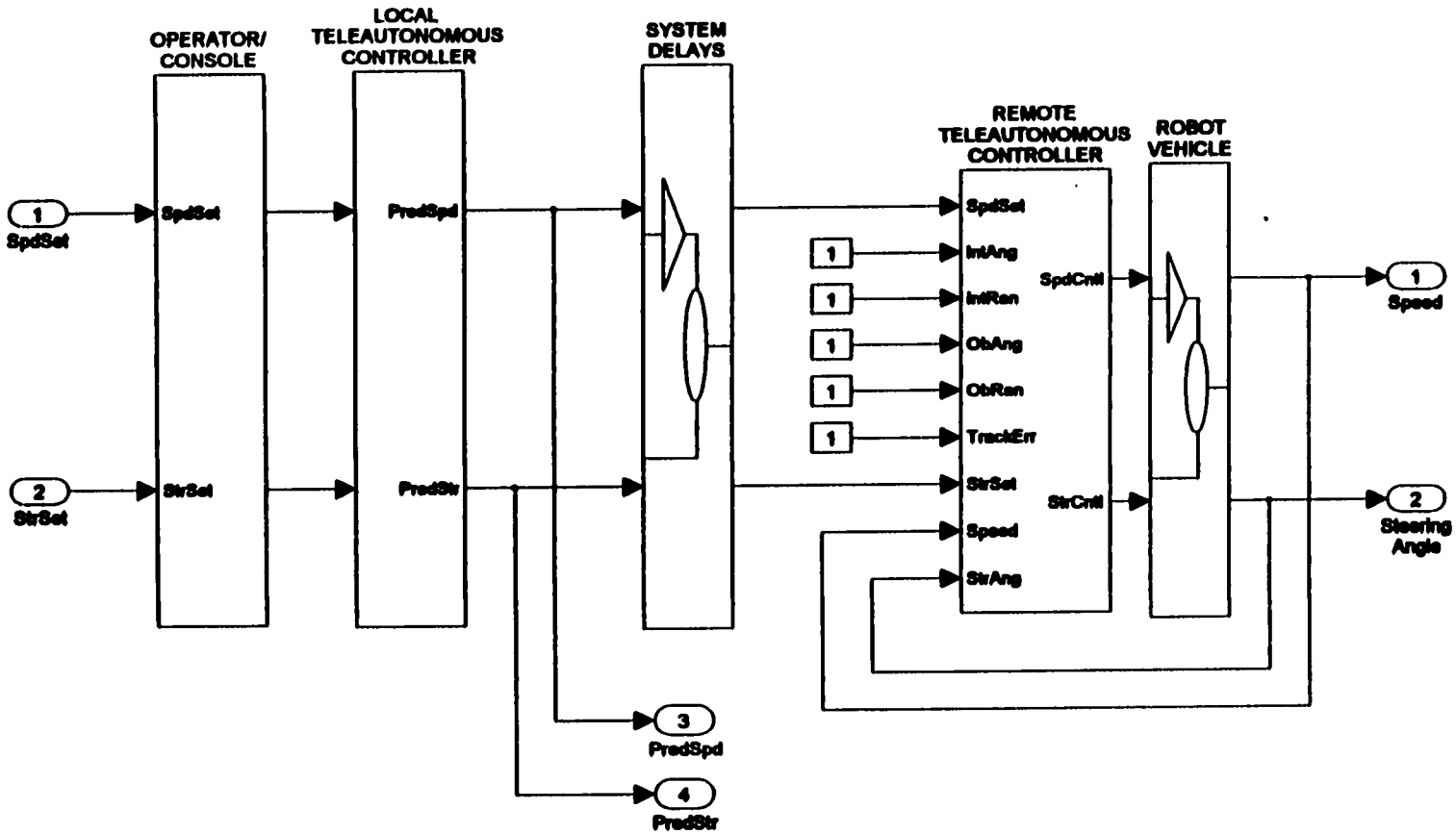


Figure B.1. Alternate complete system.

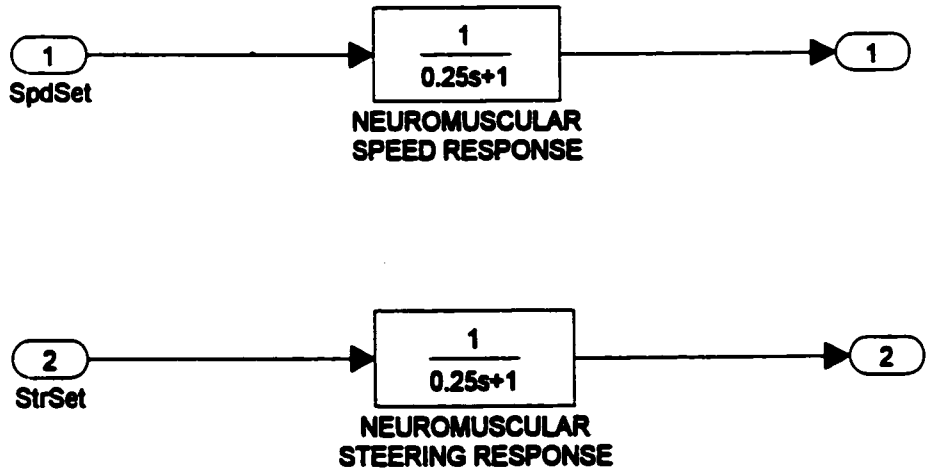


Figure B.2. Alternate operator/console.

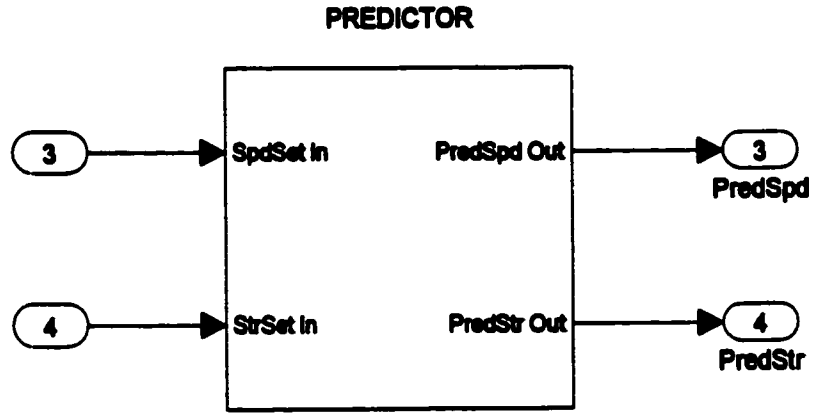


Figure B.3. Alternate local teleautonomous controller.

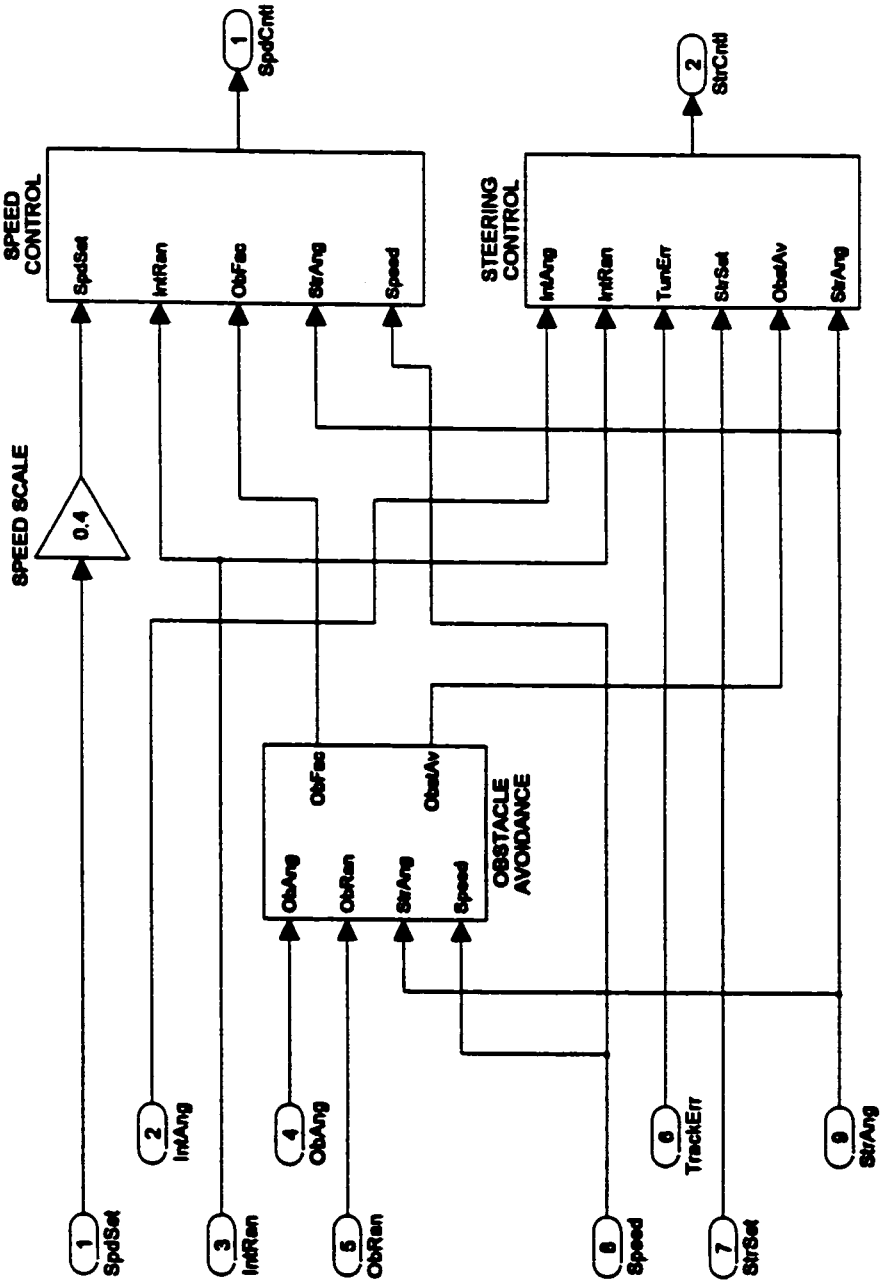


Figure B.4. Alternate remote teleautonomous controller.