Title
Driving Safely In Smart Cars

Authors
Puri, Anuj
Varaiya, Pravin

Publication Date
1995

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

Driving Safely in Smart Cars

Anuj Puri, Pravin Varaiya

California PATH Research Report
UCB-ITS-PRR-95-24

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

Report for MOU 135

July 1995

ISSN 1055-1425

Executive Summary

Automation of driving functions is central to proposals for the design of an Automated Vehicle/Highway System (AVHS). In the control architecture in [Var93], it is proposed that vehicles travel in platoons. Three maneuvers are needed: in *merge*, two platoons join together; in *split*, one platoon separates into two; and in *change lane*, a single vehicle changes lane. Using these maneuvers, a vehicle enters the system, becomes part of a platoon, travels to its destination, detaches itself from the rest of the platoon, and exits out of the system.

In this paper we consider the following problem. Consider an AVHS, for example the system proposed in [Var93, GL94, FAHL94, HESV93]. How do we know that such a system is safe? Of course we have to define what safety means. We say a system is *unsafe* if there is a possibility of a high relative velocity collision on the AVHS. We want to *prove* for a proposed design of an AVHS, that there is no such possibility. We can simulate the system. But that only checks safety for a finite number of simulation paths or trajectories of the system. We want to prove safety for every trajectory of the system. In this paper we develop an approach for proving that a system is safe. We consider the design for an AVHS proposed in [Var93] and show that if the physical controllers in the vehicles satisfy a set of constraints then the AVHS is safe. The problem of checking whether the controllers satisfy the constraints is equivalent to solving an optimal control problem.

Driving Safely in Smart Cars*

Anuj Puri and Pravin Varaiya

Department of Electrical Engineering and Computer Science,
University of California, Berkeley, CA-94720

Abstract

We address the following question: how do we know a proposed design for an Automated Vehicle/Highway System (AVHS) is safe? In particular, can we prove that there can be no high relative velocity collision on the AVHS? We show that if the controllers in the vehicles satisfy a set of constraints, then the AVHS is safe. The problem of checking whether the controllers satisfy the constraints is equivalent to solving an optimal control problem.

Keywords: Automated Vehicle/Highway System

1 Introduction

Automation of driving functions is central to proposals for the design of an Automated Vehicle/Highway System (AVHS). In the control architecture in [Var93], it is proposed that vehicles travel in platoons. Three maneuvers are needed: in *merge*, two platoons join together; in *split*, one platoon separates into two; and in *change lane*, a single vehicle changes lane. Using these maneuvers, a vehicle enters the system, becomes part of a platoon, travels to its destination, detaches itself from the rest of the platoon, and exits out of the system.

*Research supported by the California PATH program and by the National Science Foundation under grant ECS9417370

The overall architecture is divided into five layers: network layer, link layer, coordination layer, regulation layer and physical layer. The physical layer describes the vehicle dynamics. The regulation layer comprises the set of control laws for acceleration, braking and steering. The control law that is applied depends upon whether the vehicle is a leader or a follower, and upon the commands from the coordination layer. The coordination layer contains protocols. These protocols exchange coordination messages with the other vehicles in order to determine which of three maneuvers to execute, and when to do so. The link layer manages a section of the highway, setting the recommended velocity and platoon size for vehicles in that section of the highway. The network layer determines the route for the vehicles.

Designs for the various layers of AVHS have been proposed. A design for the control laws in the regulation layer is proposed in [GL94] and [FAHL94]. In [GL94], control laws are proposed for the *leader* mode in which a platoon tracks the recommended velocity, or if there is a platoon in front, then it remains a safe distance behind that platoon. Control laws for the *merge* and *split* maneuver are proposed in [FAHL94]. A design for the coordination layer is proposed in [HESV93]. This consists of protocols modeled with finite state machines.

In this paper we consider the following problem. Consider an AVHS, for example the system proposed in [Var93, GL94, FAHL94, HESV93]. How do we know that such a system is safe? Of course we have to define what safety means. We say a system is *unsafe* if there is a possibility of a high relative velocity collision on the AVHS. We want to *prove* for a proposed design of an AVHS, that there is no such possibility. We can simulate the system. But that only checks safety for a finite number of simulation paths or trajectories of the system. We want to prove safety for every trajectory of the system. In this paper we develop an approach for proving that a system is safe. We consider a proposed design for an AVHS and show that if the physical controllers in the vehicles satisfy a set of constraints then the AVHS is safe. Faults in vehicle components are not considered in this paper.

In Section 2, we describe relevant parts of the AVHS design proposed by [Var93, GL94, FAHL94, HESV93]. In Section 3, we show that a single lane AVHS is safe when the controllers satisfy a set of constraints. In Section 4, we extend the design to include the change lane maneuver, and prove that the new design is also safe. In Section 5, we conclude with some open problems.

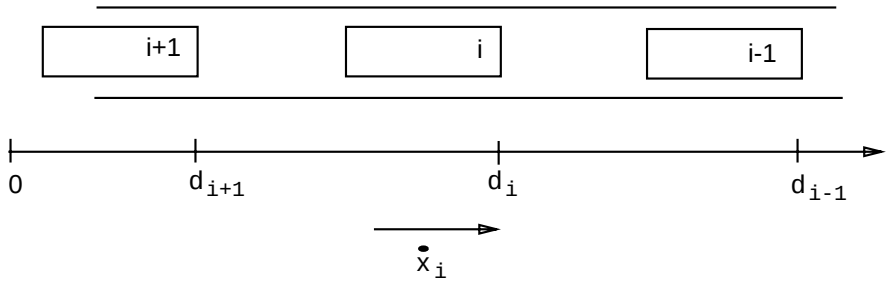

Figure 1: A Lane of the Highway

2 Single Lane AVHS

In this section we describe the merge and split maneuvers, and the relevant aspects of the coordination and the regulation layer that are important from a safety viewpoint.

2.1 Manuevers and Architecture

Figure 1 shows a lane of the highway with different platoons. The vehicle at the head of the platoon is called the leader, and it is normally under the *leader* control law. Under the leader control law, it follows the platoon in front at a safe distance and speed. The other vehicles in the platoon are under the *follow* control which tracks the leader of the platoon. At certain times, the leader of the platoon may decide to merge with the platoon in front of it. To do this, it communicates with the platoon in front, and if permitted, it orders the regulation layer to follow the *merge* control. This causes it to merge with the platoon in front. On the other hand, if during the merge maneuver, the platoon in front suddenly deccelerates or behaves erratically, the merge maneuver is aborted and a switch is made to the *abort* control. The *abort* control law steers the leader to a state from which it is safe to switch back to *leader* control. When the merge is successful, the platoon which was merging becomes part of the platoon in front, and the leader switches to the *follow* control. Similarly, a vehicle in a platoon may decide that it wants to split from the platoon. In this case, it communicates with the leader of the platoon, and if the leader permits, the vehicle orders its regulation layer to follow the *split* control law, which causes the vehicle and all vehicles behind it to split and become a separate platoon.

As shown in figure 1, the distance of platoon i from the origin is d_i. The contin-

uous state of the leader of platoon i is x_i. In [GL94], it is assumed $x_i = (d_i, \dot{d}_i, \ddot{d}_i)$.

The Regulation Layer has five control laws: leader ($\dot{x}_i = L(x_i, x_{i-1})$), merge ($\dot{x}_i = M(x_i, x_{i-1})$), follow, split ($\dot{x}_i = S(x_i, x_{i-1})$), and abort ($\dot{x}_i = A(x_i, x_{i-1})$). The regulation layer of a vehicle is in the *follow* mode when the vehicle is not the leader of the platoon. We assume that a vehicle in the *follow* mode exactly tracks the leader of the platoon. Hence, the continuous state of platoon i is x_i, the same as its leader's state. The continuous state of the AVHS is $x = (x_0, x_1, x_2, \ldots)$. Notice, the control for a platoon depends on its own state, and the state of the platoon in front.

2.2 Safety Criterion and Control Table

In a lane of the highway, the vehicles will be going through a sequence of modes such as *merge, split, follow, abort*, or *leader*. We want to prove that at no point in time is there a high relative velocity collision between any two vehicles. A high relative velocity collision is defined as a collision in which $\dot{d}_i - \dot{d}_{i-1} \geq c$ m/s, where c is a design parameter.

Associated with each control law f (where f could be the *leader, merge* or the *split* control) are two sets: an initial set $\mathcal{S}_f$ and an unsafe set U_f. The control f starts from an initial condition $(x_i(0), x_{i-1}(0)) \in \mathcal{S}_f$. The unsafe set U_f is the set of undesirable states which should be unreachable; for example, U_f is the set of states representing collision between vehicles. The initial set $\mathcal{S}_f$ is chosen so that starting from an initial condition $(x_i(0), x_{i-1}(0)) \in \mathcal{S}_f$, the unsafe set U_f is unreachable. Before the control $\dot{x}_i = f(x_i, x_{i-1})$ is applied, it is checked that the initial state $(x_i, x_{i-1}) \in \mathcal{S}_f$.

The set U_f can depend on control f. For the merge control, we require that there be no high relative velocity collision; for the leader control, we impose the stronger condition that there be no collision. We summarize this information in the *Control Table* (table 1). The control table shows when a particular control is permissible. The control law $\dot{x}_i = f(x_i, x_{i-1})$ can be applied provided $(x_i, x_{i-1}) \in \mathcal{S}_f$. Before a control is applied, a check is made in the Control Table to see whether the initial state $(x_i, x_{i-1}) \in \mathcal{S}_f$.

In the abort control law, a vehicle applies full brakes. Therefore, it is safe to switch to the abort control law at any time from any of the other controls.

Control Table			
	Control	Initial Set	Unsafe Set
Leader	$\dot{x}_i = L(x_i, x_{i-1})$	$\mathcal{S}_L$	$U_L = \{x \mid d_i = d_{i-1}\}$
Merge:	$\dot{x}_i = M(x_i, x_{i-1})$	$\mathcal{S}_M$	$U_M = \{x \mid d_i = d_{i-1}$ and $\dot{d}_i - \dot{d}_{i-1} \geq c\}$
Split:	$\dot{x}_i = S(x_i, x_{i-1})$	$\mathcal{S}_S$	U_S
Abort:	$\dot{x}_i = A(x_i, x_{i-1})$		

Table 1: Control Table to Check Safety

3 Safe Driving, Abstractions and Optimal Control

Consider the single lane system of figure 1. Each vehicle follows the control

$$\dot{x}_i = f(x_i, x_{i-1}), \tag{1}$$

where $(x_i(0), x_{i-1}(0)) \in \mathcal{S}_f$. The control f could be the leader, merge, split, or the abort control law. We must show that the unsafe set U_f is unreachable in each case.

We face the problem that the control for vehicle $i-1$ depends on vehicle $i-2$, which depends on vehicle $i-3$, and so on. To avoid working with an infinite dimensional system, we use a conservative abstraction. We look at the dynamics between two vehicles, vehicle i and vehicle $i-1$, and abstract the differential equation for vehicle $i-1$,

$$\dot{x}_{i-1} = f(x_{i-1}, x_{i-2}), \tag{2}$$

with the differential inclusion

$$\ddot{d}_{i-1} \in [A_{min}, A_{max}]. \tag{3}$$

We choose A_{min} to be the maximum deceleration (full brakes), and A_{max} to be the maximum acceleration (full throttle). This implies that for any law f in equation 2, the solution for equation 2 is contained in the set of solutions for equation 3. In this sense, equation 3 is a (conservative) abstraction of equation 2.

We now prove safety for the abstracted system

$$\dot{x}_i = f(x_i, x_{i-1}), \tag{4}$$

$$\ddot{d}_{i-1} \in [A_{min}, A_{max}],$$
$$(x_i(0), x_{i-1}(0)) \in \mathcal{S}_f,$$

by showing that U_f is unreachable. This will imply that the system of equation 1 is safe, since the reachable set of equation 4 is larger than the reachable set of equation 1. Notice that proving safety for equation 4 is equivalent to proving that despite any erratic behavior on part of vehicle $i-1$, the control law for vehicle i prevents it from having a high speed collision with vehicle $i-1$. Furthermore, equation 4 is independent of vehicles $i-2$ and beyond.

We show that equation 4 is safe for each control f and initial set $\mathcal{S}_f$. From this it follows that there can be no high relative velocity collision involving vehicle i and vehicle $i-1$. Since i is arbitrary, it follows that for *every* i, there is no high relative velocity collision involving vehicle i and vehicle $i-1$.[1] That is, the system of figure 1 is safe when the initial state is such that $(x_{i-1}(0), x_i(0)) \in \mathcal{S}_f$ for each i.

To show that U_f is unreachable in equation 4, we need to compute the reach set $Reach_f(\mathcal{S}_f)$ (i.e., all states reachable from $\mathcal{S}_f$ under the law f), and check whether $Reach_f(\mathcal{S}_f) \bigcap U_f = \emptyset$. For a control f, there is also a largest set of safe initial states $\mathcal{S}_f^* = (Reach_{-f}(U_f))^c$ (i.e., complement of all states which can reach U_f). It is clear that for a set $\mathcal{S}_f$, $Reach_f(\mathcal{S}_f)$ and $\mathcal{S}_f^*$ are invariant sets, and $\mathcal{S}_f \subset Reach(\mathcal{S}_f) \subset \mathcal{S}_f^*$. At present, techniques for computing reach sets of differential equations and inclusions are not available. Instead of explicitly computing the reach set, we turn our problem into an equivalent optimal control problem.

3.1 Optimal Control Problem

To determine whether $U_f = \{x | g(x) \leq 0\}$ is unreachable from $\mathcal{S}_f$ in equation 4, we solve the following optimal control problem, with control $u = \ddot{d}_{i-1}$:

$$\begin{aligned}
Cost: \quad & J = \min_t \; g(x(t)), \\
Differential\ equation: \quad & \dot{x}_i = f(x_i, x_{i-1}), \\
Initial\ condition\ constraint: \quad & (x_i(0), x_{i-1}(0)) \in \mathcal{S}_f, \\
State\ constraint: \quad & \dot{d}_i \geq 0, \; \dot{d}_{i-1} \geq 0, \\
Control\ constraint: \quad & u = \ddot{d}_{i-1} \in [A_{min}, A_{max}].
\end{aligned} \tag{5}$$

[1] A vehicle that is not engaged in the merge maneuver should have no collision.

The optimal control finds the choice of the initial condition $(x_i(0), x_{i-1}(0)) \in \mathcal{S}_f$ and control which minimize the cost while remaining within the constraints. Notice that the state constraints require the velocities of both vehicles to be positive. If the optimal cost $J > 0$, then we know that for every initial condition $(x_i, x_{i-1}) \in \mathcal{S}_f$, the set U_f is unreachable in equation 4. On the other hand, if $J \leq 0$, then the trajectory which minimizes J also takes equation 4 into U_f. Therefore the system of equation 4 is safe if and only if the optimal cost $J > 0$ in equation 5.

3.2 A Leader Control Example

Equation 6 shows part of the leader control developed in [GL94]. The control is applied during safety-critical situations when the inter-vehicle distance is small, or the relative velocity between vehicles is large.

$$\dddot{d}_i = -3\ddot{d}_i - 3(\dot{d}_i - \dot{d}_{i-1}) + ((d_{i-1} - d_i) - (\dot{d}_i + 10)) \tag{6}$$

The state of the system is $x = ((d_{i-1} - d_i), \dot{d}_i, \dot{d}_{i-1}, \ddot{d}_i)$. The maximum braking capacity of a vehicle is $A_{min} = -5\frac{m}{s^2}$, and the maximum acceleration is $A_{max} = 2\frac{m}{s^2}$. We choose the initial set $\mathcal{S}_L$ where

$$\mathcal{S}_L = \{(d_{i-1} - d_i) + \frac{(\dot{d}_i^2 - \dot{d}_{i-1}^2)}{2A_{Min}} - 10 - (\dot{d}_i - \dot{d}_{i-1}) \geq 0,$$
$$d_{i-1} - d_i \geq 5, \quad -5 \leq \ddot{d}_i \leq 2,$$
$$0 \leq \dot{d}_i \leq 30, \quad 0 \leq \dot{d}_{i-1} \leq 30\}.$$

We want to determine if a collision between vehicle i and vehicle $i-1$ is possible when vehicle i starts from an initial condition $x(0) \in \mathcal{S}_L$ and follows the control in equation 6. To determine this, we solve the following equivalent optimal control problem:

$$\begin{aligned}
Cost: \quad & J = \min_t (d_{i-1} - d_i), \\
Differential\ Eqn: \quad & \dddot{d}_i = -3\ddot{d}_i - 3(\dot{d}_i - \dot{d}_{i-1}) + ((d_{i-1} - d_i) - (\dot{d}_i + 10)), \\
& \ddot{d}_{i-1} = u, \\
Initial\ Condition: \quad & x(0) \in \mathcal{S}_L, \\
State\ Constraint: \quad & \dot{d}_i \geq 0,\ \dot{d}_{i-1} \geq 0, \\
Control\ Constraint: \quad & u \in [A_{Min}, A_{Max}]
\end{aligned} \tag{7}$$

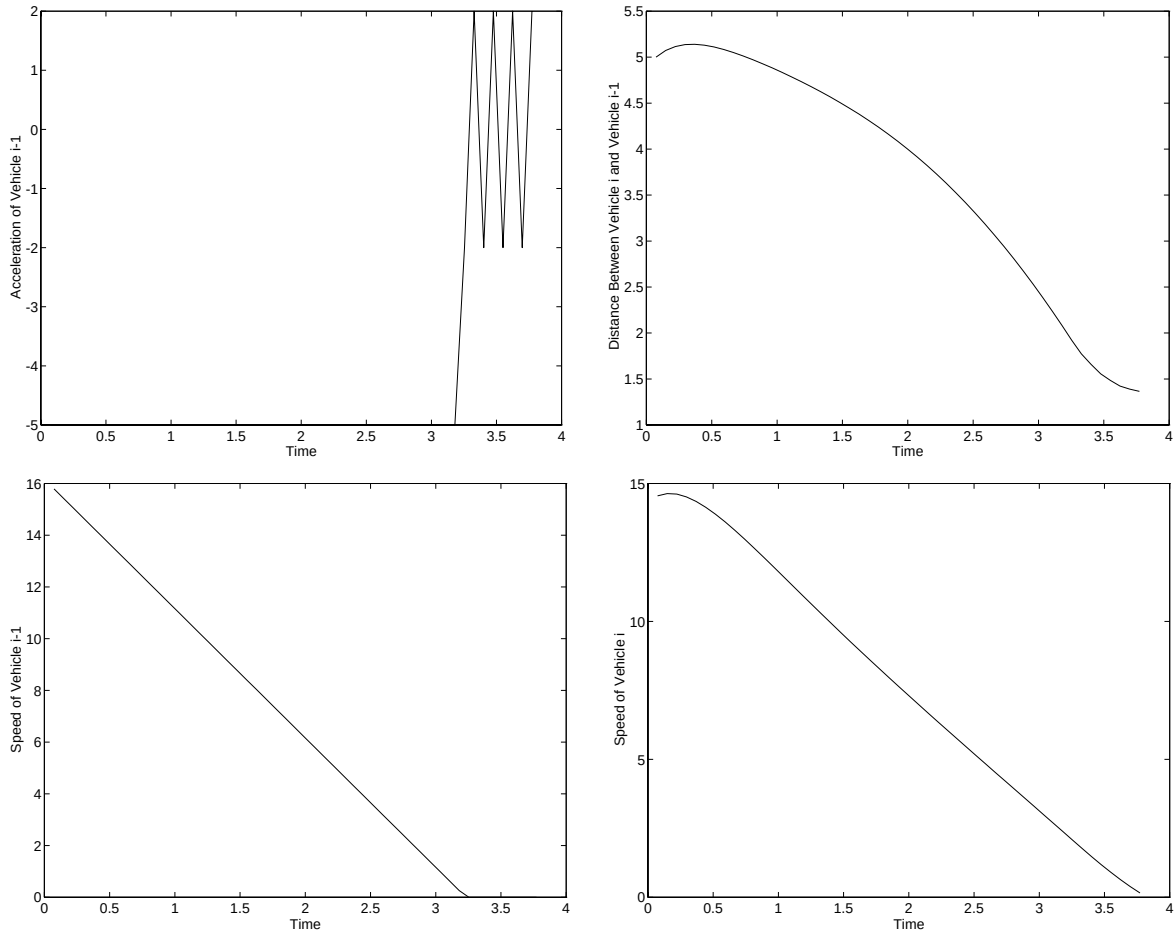

Figure 2: Solution of the Optimal Control Problem

The optimal control problem is a free end time problem. The solution is obtained using mathematical programming techniques with the optimal cost $J = 1.4$, and $x^*(0) = (5.0, 14.6, 15.8, 2.0)$.[2] Figure 2 shows the optimal control (acceleration of vehicle $i-1$), the distance between vehicle i and vehicle $i-1$, and the speed of vehicle $i-1$ and vehicle i respectively. The optimal control corresponds to vehicle $i-1$ applying full brakes for the first 3 seconds. Since the optimal cost $J = d_{i-1} - d_i$ is slightly above 1 meter, we conclude that starting from any initial condition in $\mathcal{S}_L$, despite any behaviour of vehicle $i-1$, the distance between the vehicles never falls below 1 metre. Therefore, vehicle i can safely switch to the control of equation 6 when $x(0) \in \mathcal{S}_L$.

Although in this example vehicle $i-1$ applies full brakes, in general, the trajectory which vehicle $i-1$ executes to minimize the separation between the two vehicles can be much more complicated. For example, vehicle $i-1$ can accelerate, causing vehicle i to accelerate, and then vehicle $i-1$ applies full brakes. The form of the optimal solution (the trajectory which vehicle $i-1$ executes to minimize the separation between the two vehicles) will depend on the control used by vehicle i.

To solve our problem we require a global optimum. This is in general difficult using mathematical programming techniques unless the problem is convex. But we can find an approximation to the global optimal by repeatedly running the optimization procedure with different start values.

4 Changing Lane with Abstract Vehicles

In section 2, we described the design for a single lane. The basic maneuvers were the *merge* and the *split* maneuvers. In this section, we extend the design with the *change lane* maneuver that single vehicles execute to move from one lane to the next.

The basic idea we use to show that the change lane maneuver is safe is the same as in section 3: a vehicle follows the vehicles in front at a safe distance and speed. Its control law prevents a high-speed collision with the vehicles in front, despite any erratic behavior on their part. In the case of a single lane, the meaning of "front" is well-defined. In the case of a multilane highway, this is not so clear. Consider vehicle A changing from lane k to lane $k+1$ in figure 3. The process of changing lane takes a certain amount of time, and is a continuous phenomenon. It is not clear in figure 3,

[2] The solution is due to Adam Schwartz.

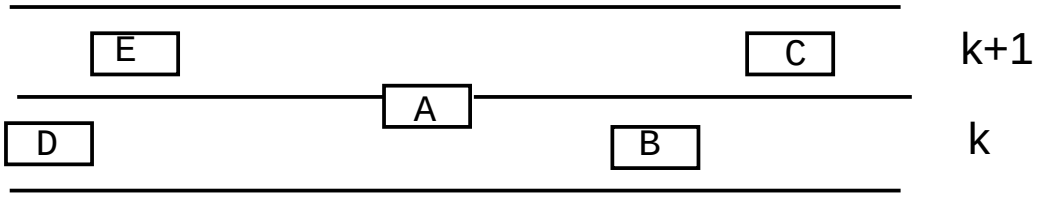

Figure 3: Change Lane Manuever

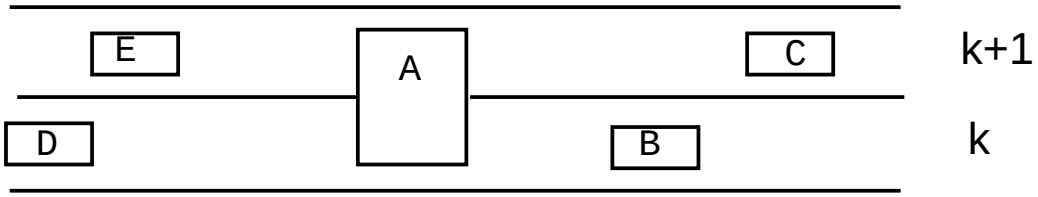

Figure 4: Changing Lane with Abstract Vehicles

which vehicle is in front of vehicle A, and what should be the longitudinal control for vehicle A. Or when should the longitudinal control for D take B into account, rather than A.

Since keeping safely behind the vehicle in *front* was the key to proving safety in a single lane system, we extend this idea to multilane system by using the concept of an *abstract vehicle*. Abstract vehicles will be conceptual devices used to represent real vehicles. For example, a vehicle changing from lane k to lane $k+1$ will be represented by an abstract vehicle occupying both lane k and lane $k+1$. We will design the multilane highway system so that the abstract vehicles remain safe. Since a real vehicle is within the space occupied by the abstract vehicle, this will also guarantee the safety of real vehicles.

Consider figure 4. Vehicle A is changing from lane k to lane $k+1$, but it is represented by an abstract vehicle. The meaning of "front" is clear in this figure. Vehicles B and C both are in front of vehicle A, and vehicle A is in front of vehicles D and E. Vehicle A must remain a safe distance and at a safe speed behind vehicles B and C. Despite any erratic behaviour on the part of vehicles B and C, the control law for vehicle A should prevent a collision with B or C. Of course if the abstract vehicle is safe, then so is the real vehicle. The controls for vehicles D and E are identical to those in section 2, with vehicle A in front.

When a vehicle is ready to change lane, it turns on its change lane signal. At

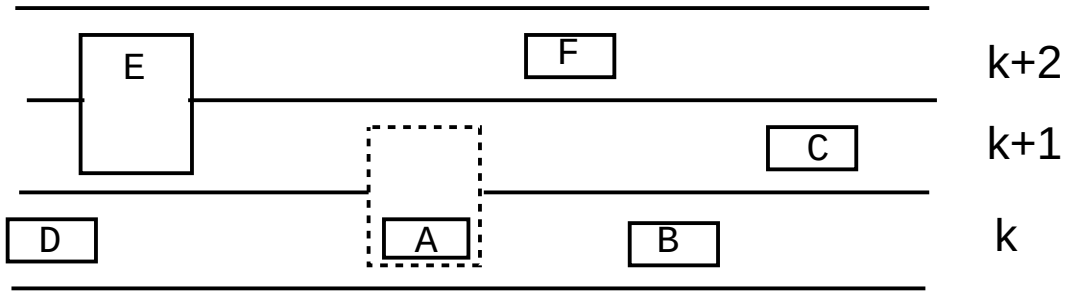

Figure 5: Change Lane Manuever

this time, it also becomes an abstract vehicle occupying two lanes. It switches to a longitudinal control which keeps the abstract vehicle safe from the vehicles in the front, and a lateral control which causes the vehicle to move from one lane to the next. The change lane signal also indicates to the other vehicles that this vehicle is an abstract vehicle occupying two lanes. The vehicles in the neighborhood take this into account when they figure out which vehicle is in front of them.

4.1 Longitudinal Control for an Abstract Vehicle

An abstract vehicle can have two vehicles in front of it—one in each lane— the objective of the longitudinal control should be to prevent a collision with either vehicle. Consider vehicles A,B and C in figure 4. The states of vehicles A,B and C are $x_A = (d_A, \dot{d}_A, \ddot{d}_A)$, $x_B = (d_B, \dot{d}_B, \ddot{d}_B)$ and $x_C = (d_C, \dot{d}_C, \ddot{d}_C)$ respectively, where d_A, d_B and d_C are the distances of the vehicles from the origin. The longitudinal control for vehicle A is

$$\dot{x}_A = C(x_A, x_B, x_C) \tag{8}$$

The unsafe set is $U_C = \{(x_A, x_B, x_C) | d_A = d_B \text{ or } d_A = d_C\}$ (i.e., A has a collision with B or C). An initial set $\mathcal{S}_C$ is specified such that U_C is unreachable when equation 8 starts from an initial state $(x_A, x_B, x_C) \in \mathcal{S}_C$. To check that the control C satisfies this safety criterion, an optimal control problem can be solved as in section 3.

4.2 Changle Lane Manuever

Suppose vehicle A wants to change from lane k to lane $k+1$. Let $[A]$ be the abstract vehicle which occupies both lane k and lane $k+1$. Since $[A]$ occupies two lanes, there

may be a new vehicle in front of it in lane $k+1$. Furthermore, $[A]$ itself may be in front of another vehicle in lane $k+1$. Before A becomes an abstract vehicle, it must check that $[A]$ starts from a safe initial condition. And if $[A]$ is in front of a new vehicle in lane $k+1$, then that vehicle must also start from a safe initial condition. Consider figure 5. Before A becomes an abstract vehicle, it must check that $(x_A, x_B, x_C) \in \mathcal{S}_C$. And since $[A]$ will be in front of E, it must also check that $(x_E, x_A, x_F) \in \mathcal{S}_C$. When these two conditions are true, A turns on its change lane signal, becomes an abstract vehicle, and begins to change lane. After A has finished changing lane, it turns off the change lane signal and resumes leader control.

Notice a subtle point in the design. Before A resumes leader control, it needs to check that the new initial condition is safe. Also, in figure 5, $[A]$ is safe from B, and D is safe from $[A]$. But when A finishes changing into lane $k+1$, D finds B in front of it, and may not be safe from it. We assume the controls satisfy the transitivity property (i.e., if A is safe from B, and D is safe from A, then D is safe from B) which prevents this possibility.

The possibility of two vehicles *simultaneously* turning on their change lane signals has to be avoided or resolved by using some coordination or contention resolution mechanism.

There is no high-speed collision involving a vehicle and the vehicle in front. This is also true for the abstract vehicles. Therefore, vehicles that are changing lane are also safe. Furthermore, a change in the highway configuration due to the beginning or ending of the change lane maneuver also keeps the system safe. Since *every* vehicle on the multilane AVHS is safe at all times, we conclude that the multilane AVHS is safe.[3] It is interesting that the proof of safety in a multilane AVHS is independent of the lateral control law for the change lane maneuver.

5 Conclusion and Open Problems

We considered the problem of safety on an AVHS. We presented two main techniques: conservatively abstracting the dynamics of a vehicle by a simple differential inclusion; and representing a vehicle changing lane by an abstract vehicle occupying two lanes. Using these methods, it becomes possible to determine the safety of a vehicle by

[3]Under the assumption that a low relative velocity collision does not push a vehicle into a different lane.

considering only its own controllers. When the controllers satisfy a set of constraints, the vehicle is safe. We showed that checking whether the controllers satisfy the set of constraints is equivalent to solving an optimal control problem. Since we prove that each vehicle is safe, we conclude the multilane AVHS operates safely.

Several problems need to be studied in more detail. Computing the initial set $\mathcal{S}_f$, and determining whether the unsafe set U_f is reachable from an initial condition in $\mathcal{S}_f$ is a key problem. Although the problem is equivalent to an optimal control problem, solving for the global optimal is difficult. In Section 3, we get an approximation to the global optimal by using different start values in the optimization procedure. Other approaches for solving the problem should also be studied. Alternatively, a simpler class of controls with more desirable properties may be used. For example, the class of controls in which $u^*(t) = A_{Min}$ (i.e., the optimal solution corresponds to the front car applying full brakes). The affect of modeling and measurement errors should also be considered. The initial set $\mathcal{S}_f$ should be conservatively designed so that despite errors, U_f is unreachable.

Staying in a lane safely behind the vehicles in front is the main safety requirement. The change lane maneuver is needed to exit out of the AVHS. Failure of these capabilities due to faults of vehicle components should be investigated. Also strategies should be devised to avoid high-speed collision in case of failure of these functions.

Advanced adaptive cruise control (AACC) is likely to be introduced by several vehicle manufacturers. The setup is the same as a single lane, with no platoons (i.e., platoon size of one), and with a single law, namely the leader law. The approach proposed here can be used to determine if a proposed AACC law will be safe if it is activated from certain initial states. Note that in practice the parameters such as maximum acceleration and deceleration will depend upon road surface and tire characteristics and other parameters. Thus the safe initial conditions will change with those parameters.

Acknowledgement

We thank Adam Schwartz for making available his software for solving optimal control problems, and for solving the optimal control example of Section 3. We also thank Luis Alvarez, Mireille Broucke, Dr. Akash Deshpande, Jonathan Frankel, Dr. Datta Godbole, Prof. Roberto Horowitz, Dr. Michael Kourjanski, Perry Li, John Lygeros, Prof. Shankar Sastry and other members of the PATH community for their helpful

comments and encouragement. The contents of the paper reflect the views of the authors. They do not necessarily reflect the official views or policies of the State of California.

References

[FAHL94] J.Frankel, L.Alvarez, R.Horowitz, and P.Li. "Robust Platoon Manuevers for AVHS," UCB-PATH TECH NOTE 94-09, University of California.

[GL94] D.Godbole and J.Lygeros, Longitudinal Control of the Lead Car of a Platoon. *IEEE Transactions on Vehicular Technology*, 43(4):1125-35, November 1994.

[HESV93] A.Hsu, F.Eskafi, S.Sachs, and P.Varaiya. Protocol Design for an Automated Highway System. *Discrete Event Dynamic Systems: Theory and Applications*, vol.2,(no.3-4):183-206, February 1993.

[Var93] P.Varaiya. Smart Cars on Smart Roads: Problems of Control. *IEEE Transactions on Automatic Control*, 38(2):195-207, February 1993.