Semiautonomous Multivehicle Safety

A Hybrid Control Approach

By Rajeev Verma and Domitilla Del Vecchio

Intelligent transportation systems (ITS) for in-vehicle cooperative active safety continue to be examined worldwide by government and industry consortia. The role of these systems in everyday driving tasks will be to warn the driver about incoming collisions, suggest safe actions, and ultimately take control of the vehicle to prevent an otherwise certain collision. Several initiatives are taking place, including the Crash Avoidance Metrics Partnership (CAMP) [2] and Vehicle Infrastructure Integration Consortium (VIIC) [3], [4] in the United States, the Car-2-Car Communications Consortium in Europe [1], and the Advanced Safety Vehicle Project 3 (ASV3) in Japan (The CAMP Vehicle Safety Consortium formed between Toyota, General Motors, Ford, Daimler, and Honda works under collaborative agreement with the U.S. Department of Transportation Joint Program Office.). Specifically, reducing collisions at traffic intersections, mergings, and roundabouts is a central part of these initiatives [28]. Positioning (differential global positioning systems) and wireless communication (dedicated short-range communication 5.9 GHz in the United States) technologies are becoming more advanced, while their cost is declining to the point that ITS can be employed to improve in-vehicle production safety systems by the automotive industry. In the near future, ITS is expected

Digital Object Identifier 10.1109/MRA.2011.942114
Date of publication: 13 September 2011

1070-9932/11/$26.00©2011 IEEE

to become more comprehensive connecting vehicles with each other and with the surrounding road infrastructure through vehicle-to-vehicle and vehicle-to-infrastructure wireless communication.

In order for the in-vehicle cooperative active safety systems to be a realistic solution to decrease the number of accidents, they should be safe by design while adapting to the presence of human-driven vehicles. Hence, the control algorithms developed for guaranteeing safety must be able to operate in this semiautonomous real-world scenario as long as roadside infrastructure provides an approximate position of noncommunicating vehicles. An interesting challenge is that a conventional approach that accounts for the worst-case uncertainty due to human driving decisions would not be practical as too conservative solutions would result. Conservative solutions cannot be considered for deployment as they would cause false alarms, leading the users to loose trust in the safety system and to routinely neglect its warnings.

There is a rich literature about the classification through hybrid dynamical models of human behavior in structured tasks (see [15] and [16] and the references therein). These works show that human behavior can be recognized, provided certain identifiability assumptions are satisfied. In this article, we propose an approach in which human driving behavior is modeled as a hybrid automaton in which the mode is unknown and represents a primitive driving dynamics such as braking and acceleration. On the basis of this hybrid model, the vehicles equipped with the cooperative active safety system estimate in real time the current driving mode of noncommunicating human-driven vehicles and exploit this information to establish the least restrictive safe control actions. This type of solution leads to less conservative safety controllers than those that treat human-driven vehicles as enemies to be counteracted for the worst-case scenarios. This approach can be formulated as a safety control problem for hybrid automata with imperfect mode information [37]–[39]. Specifically, in [37] and [38], a mode estimator is constructed, which keeps track of the current mode uncertainty based on continuous state measurements. For each current mode uncertainty, a mode-dependent capture set is constructed, which determines the set of all continuous states that lead to an unsafe configuration for the given mode uncertainty. Then, a hybrid feedback map is computed for each mode uncertainty that keeps the continuous state outside of the current mode-dependent capture set. These algorithms are provably safe and least restrictive.

Related Work

Although the safety control problem for hybrid systems has been extensively considered when the state is measured [18], [22], [26], [31], [32], [34], [35], the same control problem has been receiving less attention when the mode is unknown. A number of works have addressed the control problem for special classes of hybrid systems with imperfect state information [12], [13], [20], [37]–[39], [41]. There has been a wealth of work on employing hybrid system models and formal methods to generate collision-free trajectories in multivehicle and multirobot systems. The automated highway system (AHS) by the California Partners for Advanced Transportation Technology in the 1990s is an early example. The objective of the AHS project was the development of fully autonomous highway systems, mainly based on the concept of platooning, to increase traffic throughput, safety, and fuel efficiency [21]. In the context of platooning, a number of papers have proposed a formal hybrid modeling and control approach based on the computation of a safe set of initial conditions (the complement of the static capture set), optimal control, and game theory [8], [19], [24], [25]. A decentralized cooperative policy for conflict resolution in multivehicle systems with guaranteed safety has been proposed in [29]. Since conflicts are resolved locally, the complexity of the control policy is independent of the number of vehicles. Other approaches have been focusing on formal methods for collision detection based on stochastic reachability analysis (see [7] and the references therein). Formal reasoning for both design and verification for autonomous vehicles driving in the presence of human drivers has been developed and implemented in the 2007 Defense Advanced Research Projects Agency (DARPA) Urban Challenge by many of the participating teams [11]. The behavior prediction for human drivers has also been widely investigated (see [23] and [30]). Yet, formally including these predictions into planning mostly remains an open question [11].

The human-driven vehicle follows the outer path, while the autonomous vehicle follows the inner path.

Safety Control Problem for Hidden-Mode Hybrid Systems

In this section, we formally introduce the safety control problem for hidden-mode hybrid systems (HMHSs) and provide the solution as it has been proposed in earlier works [37]–[39].

Definition 1

A hybrid automaton with uncontrolled-mode transitions H is a tuple $H = (Q, X, U, D, \Sigma, \mathrm{Inv}, R, f)$ in which Q is the set of modes; X is the continuous state space; U is the continuous set of control inputs; D is the continuous set of disturbance inputs; Σ is the set of disturbance events that trigger transitions among modes; $\mathrm{Inv} = \{\epsilon\}$ is the discrete set of silent events, which correspond to no transition; $R : Q \times \Sigma \to Q$ is the mode-update map, and $f : X \times Q \times U \times D \to X$ is the vector field, which is allowed to be piecewise continuous with its arguments.

The hybrid trajectories $(q(t), x(t))$ of H are piecewise continuous signals with transitions because of the occurrence of discrete events (see [26] for details).

Definition 2

A HMHS is a hybrid automaton with uncontrolled-mode transitions in which the discrete state $q(t)$ is not measured and the initial mode q_0 is only known to belong to a set $\bar{q}_0 \subseteq Q$.

Let Bad $\subseteq X$ be a bad set of states; the control task is to keep the continuous state $x(t)$ outside Bad for all time using the available information $(x(t), u(t), \bar{q}_0)$.

The objective of the AHS project was the development of fully autonomous highway systems to increase traffic throughput, safety, and fuel efficiency.

Application Scenario

Referring to Figure 1, we assume that the infrastructure measures the position and speed of Vehicle 2 through roadside sensors such as cameras and magnetic induction loops and transmits this information to the onboard controller of Vehicle 1. Vehicle 1 has to use this information to avoid a collision. Vehicle 1 longitudinal dynamics along its path is given by the second-order system $\dot{p}_1 = v_1,\ \dot{v}_1 = a\,u + b - cv_1^2$, in which p_1 is the longitudinal displacement of the vehicle along its path and v_1 is the longitudinal speed (see Figure 1), $u \in [u_L, u_H]$ is the control input (positive when the vehicle accelerates and negative when the vehicle brakes), $b < 0$ represents the static friction term, and $c > 0$ with the cv_1^2 term modeling air drag (see [40] for more details on the model). Vehicle 2 is controlled by a driver. There has been a wealth of work on modeling human driving behavior through hybrid systems, wherein each mode corresponds to a primitive behavior such as braking, acceleration, steering, run out, and lane change maneuver [6], [33].

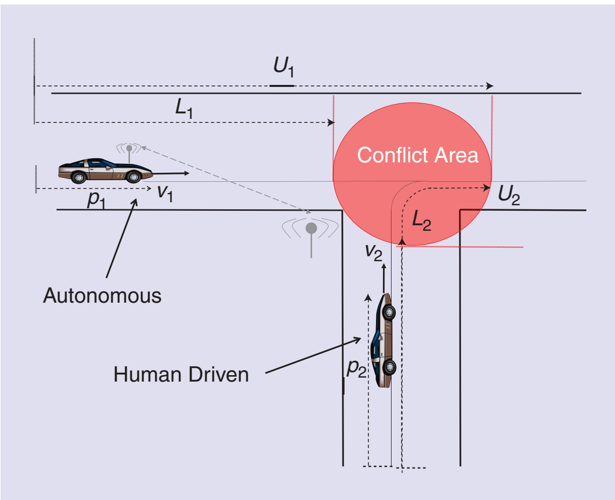

Figure 1. Two-vehicle conflict scenario. Vehicle 1, whose longitudinal displacement and speed are denoted as p_1 and v_1, respectively, is autonomous and communicates with the infrastructure via wireless. Vehicle 2, whose longitudinal displacement and speed are denoted as p_2 and v_2, respectively, is human-driven and does not communicate with the infrastructure. A collision occurs when more than one vehicle occupies the conflict area at the same time.

We model human driving behavior in the proximity of an intersection through a hybrid system with two modes: braking and acceleration, i.e., $\dot{p}_2 = v_2,\ \dot{v}_2 = \beta_q + \gamma_q d$, with $q \in \{A, B\},\ d \in [-\bar{d}, \bar{d}]$, in which p_2 is the longitudinal displacement of the vehicle along its path and v_2 is the longitudinal speed (see Figure 1), $\bar{d} > 0$, q is the mode with $q = B$ corresponding to braking mode and $q = A$ corresponding to acceleration mode, and $\gamma_q > 0$. The value of β_q corresponds to the nominal dynamics of mode q; thus, we have $\beta_B < 0$ and $\beta_A > 0$. The disturbance d models the error with respect to the nominal model. This implies that if $\dot{v}_2 \in \beta_q + \gamma_q[-\bar{d}, \bar{d}]$, the current mode can be mode q. This allowed error in each mode captures the fact that there are several ways in which modes A or B can be realized (e.g., having harder braking or softer braking and harder acceleration or softer acceleration). It also captures variability among drivers. Finally, we assume that there is no transition between modes, i.e., the driver cannot change his/her mind. This is a reasonable assumption when one models the behavior of vehicles that are close enough to the intersection. Models considering transitions from acceleration to coasting and to braking have been considered in [39]. More complex models involving arbitrary transitions among modes will be considered in future work. Since the vehicles do not go in the reverse direction, there is a lower nonnegative speed limit denoted as $v_{\min}$. Note that a strictly positive $v_{\min}$ also guarantees the liveness of the system preventing vehicles to stop. Similarly, we allow an upper speed limit (which could be infinity), denoted as $v_{\max}$, with respect to speed limitation regulations in the proximity of intersection.

The intersection system is a hybrid automaton with uncontrolled mode transitions H, in which $Q = \{A, B\}$; $X = \mathbb{R}^4$, and $x \in X$ is such that $x = (p_1, v_1, p_2, v_2)$; $U = [u_L, u_H] \subset \mathbb{R}$; $D = [-\bar{d}, \bar{d}] \subset \mathbb{R}$; $\Sigma = \emptyset$ as there is no transition allowed between the modes; $R : Q \times \Sigma \to Q$ is the mode update map, which is trivial as $\Sigma = \emptyset$, and $f : X \times Q \times U \times D \to X$ is the vector field, which is piecewise continuous, and it is given by $f(x, q, u, d) = (f_1(p_1, v_1, u), f_2(p_2, v_2, q, d))$ in which

$$f_1(p_1, v_1, u) = \left(\begin{array}{c} v_1 \\ \begin{cases} 0 & \text{if } (v_1 = v_{\min} \text{ and } \alpha_1 < 0) \text{ or} \\ & (v_1 = v_{\max} \text{ and } \alpha_1 > 0) \\ \alpha_1 & \text{otherwise} \end{cases} \end{array} \right), \quad (1)$$

with $\alpha_1 = au + b - cv_1^2$, and

$$f_2(p_2, v_2, q, d) = \left(\begin{array}{c} v_2 \\ \begin{cases} 0 & \text{if } (v_2 = v_{\min} \text{ and } \alpha_2 < 0) \text{ or} \\ & (v_1 = v_{\max} \text{ and } \alpha_2 > 0) \\ \alpha_2 & \text{otherwise} \end{cases} \end{array} \right), \quad (2)$$

with $\alpha_2 = \beta_q + \gamma_q d$. Referring to Figure 1, the set of bad states for system H models collision configurations, and it is given by Bad $:= \{(p_1, v_1, p_2, v_2) \in \mathbb{R}^4 \mid (p_1, p_2) \in [L_1, U_1] \times [L_2, U_2]\}$.

Problem Solution

The control problem can be interpreted as a game between u and d in which d has full information about the environment state (the mode) while u is uninformed. In the theory of games, such problems with imperfect information have been elegantly solved by first translating them into equivalent problems with full-state information and then leveraging the available techniques for solving games of perfect information [36]. To formulate an equivalent problem with full-state information, an estimator is introduced. For details on the conditions for equivalence, the reader is referred to [37]–[39].

Definition 3

An estimator is a hybrid automaton with uncontrolled mode transitions $\hat{H} = (\hat{Q}, X, U, D, Y, \hat{\text{Inv}}, \hat{R}, \hat{f})$, in which $\hat{Q} \subseteq 2^Q$, $\hat{\text{Inv}} = \{\epsilon\}$, $\hat{f} : X \times \hat{Q} \times U \times D \to 2^X$ is a set-valued map such that $\hat{f}(x, \hat{q}, u, d) := \bigcup_{q \in \hat{q}} f(x, q, u, d)$, $\hat{q}(t)$ is such that $q(t) \in \hat{q}(t)$ for all $t \geq 0$, and $\dot{x}(t) \in \hat{f}(\hat{x}(t), \hat{q}(t), u(t), d(t))$ while $\hat{q}(t)$ is constant.

Here, 2^Q denotes the set of all subsets of Q. The estimator keeps track of a set of possible modes compatible with the measurements and with the system dynamics (see [10] and [14] and the references therein). Here, we show how to construct a suitable estimator for the application example.

Application Scenario

We have $\hat{H} = (\hat{Q}, X, U, D, Y, \hat{\text{Inv}}, \hat{R}, \hat{f})$, in which $\hat{Q} = \{\hat{q}_1, \hat{q}_2, \hat{q}_3\}$ with $\hat{q}_1 = \{A, B\}$, $\hat{q}_2 = \{A\}$, $\hat{q}_3 = \{B\}$, and $\hat{q}(0) = \hat{q}_1$. We define $Y = \{y_A, y_B\}$. Starting in $\hat{q}_1$, event y_A occurs as soon as B is not currently possible given the measurement x, and event y_B occurs as soon as A is not currently possible given the measurement x. This results in the map $\hat{R}$ defined as $\hat{R}(\hat{q}_1, y_A) := \hat{q}_2$ and $\hat{R}(\hat{q}_1, y_B) := \hat{q}_3$ leads to the automaton given in Figure 2.

To establish when A or B is ruled out given the measurement of x, we consider the estimate $\hat{\beta}(t) = (1/t) \int_0^t \dot{v}_2(\tau) d\tau, t \geq T$, where $T > 0$ is a time window. Note that, in practice, we will not require measurement of acceleration, as we will consider discrete time models where derivative is replaced by time anticipation. If the mode is q, then we necessarily have that $|\hat{\beta}(t) - \beta_q| \leq \gamma_q \bar{d}$. Thus, for $t > T$, define $y(t) = y_A$ if $|\hat{\beta}(t) - \beta_B| > \gamma_B \bar{d}$, $y(t) = y_B$ if $|\hat{\beta}(t) - \beta_A| > \gamma_A \bar{d}$, and $y(t) = \epsilon$ otherwise.

Basically, the continuous dynamics of $\hat{H}$ describes a set of dynamics of x that are compatible with the current discrete state estimate. Let $\hat{\pi} : \hat{Q} \times X \to U$ be a feedback map. We denote the x trajectories of the closed-loop system by $\phi_{\hat{x}}^{\hat{\pi}}(t, (\bar{q}_0, x_0), \mathbf{d}, \mathbf{y})$, which are given by the system $\hat{H}$ in which we have set $u(t) = \hat{\pi}(\hat{q}(t), \hat{x}(t))$. The capture set for system $\hat{H}$ is given by $\hat{C} := \bigcup_{\hat{q} \in \hat{Q}} (\hat{q} \times \hat{C}_{\hat{q}})$, in which $\hat{C}_{\hat{q}} := \{x_0 \in X \mid \forall \hat{\pi}, \exists \mathbf{d}, \mathbf{y}, t \geq 0 \text{ s.t. some } \phi_{\hat{x}}^{\hat{\pi}}(t, (\hat{q}, x_0), \mathbf{d}, \mathbf{y}) \in \text{Bad}\}$ is called a mode-dependent capture set. It is the set of all continuous states that are taken to Bad for all feedback maps when the initial mode estimate is equal to $\hat{q}$.

Problem 1

Determine the set $\hat{C}$ and a feedback map $\hat{\pi}$ that keeps any trajectory starting outside $\hat{C}$ outside it.

We briefly describe the solution as it appears in [37]–[39]. For this purpose, for any $\hat{q} \in \hat{Q}$ and $F \subseteq X$, define the operator Pre as $\text{Pre}(\hat{q}, F) := \{x \in X \mid \forall \hat{\pi}, \exists \mathbf{d}, t \geq 0 \text{ s.t.some } \phi_{\hat{x}}^{\hat{\pi}}(t, (\hat{q}, x), \mathbf{d}, \epsilon) \in F\}$, in which $\phi_{\hat{x}}^{\hat{\pi}}(t, (\hat{q}, x), \mathbf{d}, \epsilon)$ is the continuous trajectory of $\hat{H}$ when the mode $\hat{q}(t)$ stays constant. Hence, $\text{Pre}(\hat{q}, F)$ is a set of all continuous states that are taken to F for all feedback maps when the mode estimate is kept constant to $\hat{q}$. The sets $\hat{C}_{\hat{q}}$ for $\hat{q} \in \hat{Q}$ can be obtained as a fixed point of the following algorithmic procedure. Let $\hat{Q} = \{\hat{q}_1, \ldots, \hat{q}_M\}$, $S_i \subseteq X$ for $i \in \{1, \ldots, M\}$, and define $S = (S_1, \ldots, S_M)$. We define the map $G : (2^X)^M \to (2^X)^M$ as

$$G(S) := \begin{bmatrix} \text{Pre}\Big(\hat{q}_1, \bigcup_{\{j \mid \hat{q}_j \in \hat{R}(\hat{q}_1, Y)\}} S_j \cup \text{Bad}\Big) \\ \vdots \\ \text{Pre}\Big(\hat{q}_M, \bigcup_{\{j \mid \hat{q}_j \in \hat{R}(\hat{q}_M, Y)\}} S_j \cup \text{Bad}\Big) \end{bmatrix}.$$

Algorithm 1

$S^0 := (S_1^0, S_2^0, \ldots S_M^0) := (\emptyset, \ldots, \emptyset)$
$S^1 = G(S^0)$
while $S^{k-1} \neq S^k$ **do**
  $S^{k+1} = G(S^k)$
end while.

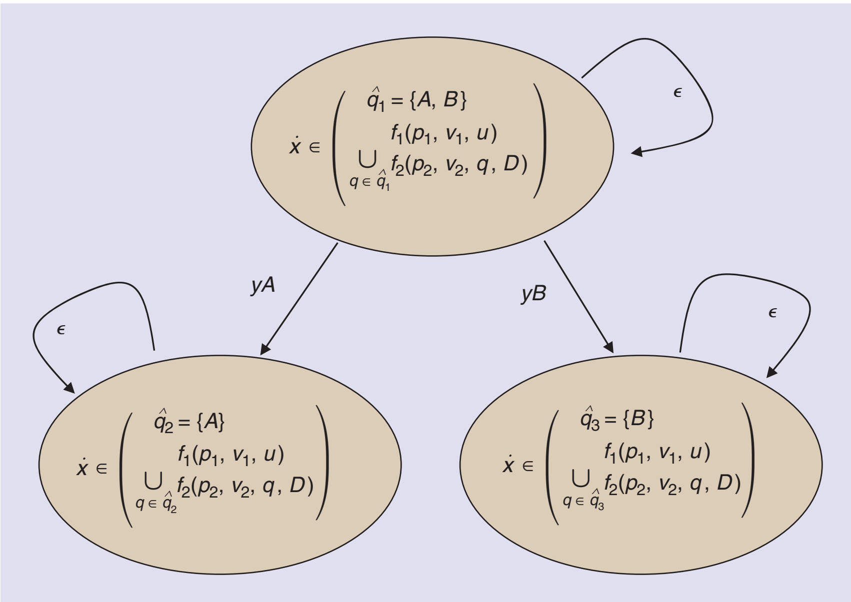

Figure 2. Hybrid automaton $\hat{H}$.

If Algorithm 1 terminates, the fixed point is equal to the tuple of sets $(\hat{C}_{\hat{q}_1}, \ldots, \hat{C}_{\hat{q}_M})$ (see [38] for details). We next show how to calculate the steps of this algorithm for the hybrid automaton of Figure 2.

Application Scenario

Referring to Figure 2, we have the system $\hat{H}$ such that $\hat{Q} = \{\hat{q}_1, \hat{q}_2, \hat{q}_3\}$ with $\hat{q}_1 = \{A, B\}$, $\hat{q}_2 = \{A\}$, and $\hat{q}_3 = \{B\}$. As a consequence, Algorithm 1 leads to

$$G(S) = \begin{bmatrix} \mathrm{Pre}(\hat{q}_1, S_2 \cup S_3 \cup \mathrm{Bad}) \\ \mathrm{Pre}(\hat{q}_2, \ \mathrm{Bad}) \\ \mathrm{Pre}(\hat{q}_3, \ \mathrm{Bad}) \end{bmatrix}$$

so that

$$S^1 = \begin{bmatrix} \mathrm{Pre}(\hat{q}_1, \ \mathrm{Bad}) \\ \mathrm{Pre}(\hat{q}_2, \ \mathrm{Bad}) \\ \mathrm{Pre}(\hat{q}_3, \ \mathrm{Bad}) \end{bmatrix}$$

and

$$S^2 = \begin{bmatrix} \mathrm{Pre}(\hat{q}_1, \mathrm{Pre}(\hat{q}_2, \ \mathrm{Bad}) \cup \mathrm{Pre}(\hat{q}_3, \ \mathrm{Bad}) \cup \mathrm{Bad}) \\ \mathrm{Pre}(\hat{q}_2, \ \mathrm{Bad}) \\ \mathrm{Pre}(\hat{q}_3, \ \mathrm{Bad}) \end{bmatrix}.$$

The first component of this expression means that, when the system starts in mode $\hat{q}_1$, the trajectory can enter Bad by flowing in $\hat{q}_1$ or by first transitioning to $\hat{q}_2$ or $\hat{q}_3$ and then by flowing in either of these modes. By the properties of the Pre operator (refer to [37] and [38]), since $\hat{q}_2, \hat{q}_3 \subseteq \hat{q}_1$, it can be shown that $\mathrm{Pre}(\hat{q}_1, \mathrm{Pre}(\hat{q}_2, \mathrm{Bad}) \cup \mathrm{Pre}(\hat{q}_3, \mathrm{Bad}) \cup \mathrm{Bad}) = \mathrm{Pre}(\hat{q}_1, \mathrm{Bad})$ so that Algorithm 1 terminates at the second step. Therefore, we have that $\hat{C}_{\hat{q}_1} = \mathrm{Pre}(\hat{q}_1, \mathrm{Bad})$, $\hat{C}_{\hat{q}_2} = \mathrm{Pre}(\hat{q}_2, \mathrm{Bad})$, and $\hat{C}_{\hat{q}_3} = \mathrm{Pre}(\hat{q}_3, \mathrm{Bad})$.

Computational Tools

The sets $\mathrm{Pre}(\hat{q}, \mathrm{Bad})$ can be computed by linear complexity algorithms. This is because for every mode estimate $\hat{q}$ the continuous dynamics is the parallel composition of two order-preserving systems, and the bad set is convex [13], [20]. Specifically, for the application example, define the restricted Pre operators for $i \in \{1, 2, 3\}$ $\mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_L} := \{x \in X \mid \exists \mathbf{d}, t \geq 0 \text{ s.t. some } \phi_{\hat{x}}(t, (\hat{q}_i, x), u_L, \mathbf{d}, \epsilon) \in \mathrm{Bad}\}$ and $\mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_H} := \{x \in X \mid \exists \mathbf{d}, t \geq 0 \text{ s.t. some } \phi_{\hat{x}}(t, (\hat{q}_i, x), u_H, \mathbf{d}, \epsilon) \in \mathrm{Bad}\}$. Then, we have that (refer to [20]) $\mathrm{Pre}(\hat{q}_i, \mathrm{Bad}) = \mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_L} \cap \mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_H}$ for $i \in \{1, 2, 3\}$. Each of the sets $\mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_L}$ and $\mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_H}$ can be computed by linear complexity discrete time algorithms (see the "Experimental Setup" section).

For each mode $\hat{q}_i$ for $i \in \{1, 2, 3\}$, a safe control map $\hat{\pi}(\hat{q}_i, x)$ acts in such a way to maintain the state outside the current mode-dependent capture set $\hat{C}_{\hat{q}}$. This results in a map $\hat{\pi}(\hat{q}_i, x)$ that makes the vector field point outside set $\hat{C}_{\hat{q}_i}$ when x is on the boundary of $\hat{C}_{\hat{q}_i}$. One can show (refer to [20]) that a control map $\hat{\pi}(\hat{q}_i, x)$ that maintains the state x outside $\mathrm{Pre}(\hat{q}_i, \mathrm{Bad})$, which is equal to $\hat{C}_{\hat{q}_i}$ for the application, is given by

$$\begin{cases} u_H & \text{if} x \in \mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_L} \cap \partial \mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_H} \\ u_L & \text{if} x \in \mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_H} \cap \partial \mathrm{Pre}(\hat{q}_i, Bad)_{u_L} \\ \{u_H, u_L\} & \text{if} x \in \partial \mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_H} \cap \partial \mathrm{Pre}(\hat{q}_i, \mathrm{Bad})_{u_L} \\ U & \text{otherwise.} \end{cases}.$$

Since we have that $\mathrm{Pre}(\hat{q}_i, \mathrm{Bad}) \subseteq \mathrm{Pre}(\hat{q}_1, \mathrm{Bad})$ for $i \in \{2, 3\}$, when the mode switches from $\hat{q}_1$ to $\hat{q}_2$ or from $\hat{q}_1$ to $\hat{q}_3$, the continuous state x being outside $\mathrm{Pre}(\hat{q}_1, \mathrm{Bad})$ implies that it is also outside $\mathrm{Pre}(\hat{q}_2, \mathrm{Bad})$ and $\mathrm{Pre}(\hat{q}_3, \mathrm{Bad})$. Therefore, the above feedback map guarantees that the state never enters the capture set.

Experimental Setup

The two-vehicle conflict scenario of Figure 1 was implemented in an in-scale multivehicle lab. The laboratory is equipped with an overhead camera-based positioning system, a control station, a human–driver interface, the roundabout system, and six scaled vehicles (https://wikis.mit.edu/confluence/display/DelVecchioLab).

A car chassis (length 0.375 m, width 0.185 m, and wheelbase 0.257 m) is used as the hardware platform for the scaled vehicle. The vehicles are equipped with an onboard computer (Mini ITX) and a motion controller. The longitudinal dynamics is dynamically similar to that of a high-mobility multipurpose wheeled vehicle (HMMWV) [40]. One of the scaled vehicles is configured to be an autonomous vehicle that can follow a predefined path and control its throttle/brake input while another acts as a human-driven vehicle that can be driven by a human driver using a human–driver interface. The human–driver interface comprises a steering wheel and two pedals for throttle and brake commands (see Figure 3). The hardware used is a Logitech MOMO force feedback racing wheel and pedal set. The hardware is connected to the control station via a Universal Serial Bus (USB) cable, and the input command from the hardware is transmitted to the vehicle via the wireless connection.

Figure 3 shows the roundabout system. There are two circular paths that share a common section on a 6 m × 6 m arena. The human-driven vehicle follows the outer path while the autonomous vehicle follows the inner path. Both vehicles travel in an anticlockwise direction. A collision is possible at the intersection when both vehicles are in the area shaded red (Figure 3) at the same time. This area corresponds to the set $\{(p_1, p_2) \mid (p_1, p_2) \in [L_1, U_1] \times [L_2, U_2]\}$. The maximum vehicle speed is 1,100 mm/s, and the minimum speed is 350 mm/s. A software module on all the vehicles maintains the speed between the specified bounds. When the two vehicles are simultaneously present in the shared path (between points Pt_1 and Pt_2), another software module prevents rear-end collision by appropriately accelerating or decelerating the autonomous vehicle when the two vehicles are too close. The maintain speed and rear-end collision prevention modules are based on a simple proportional-integral differential (PID) control scheme. The positioning system transmits the position information to the vehicles over the wireless network.

Learning Human Driving Model

A set of experiments were performed in which five human subjects drove a vehicle on the outer path in the roundabout system in ten acceleration and ten braking trials each. In these experiments, the subjects were directed to either brake or accelerate at the human-decision point D^P in Figure 3 while also avoiding a moving target on the inner path. The data collected in these braking and acceleration trials were then analyzed to estimate the parameters β_q and γ_q and presented in the "Safety Control Problem for Hidden-Mode Hybrid Systems" section. We denote the position measurement at time step k as $p(k)$ with $dT = 0.1$ s as the time lapsed between two consecutive steps. The acceleration/deceleration at time step k is denoted as $a(k)$ and is calculated as $a(k) = p(k) - 2p(k-1) + p(k-2)/dT^2$. The average acceleration/deceleration is calculated for the trial as $\bar{a} = 1/N - 1\sum_{k=2}^{N} a(k)$. A total of 99 trial runs were obtained. These trials were divided into a training set and a test set. The model of the driver behavior was then obtained by fitting two Gaussian distributions to the training data for braking and acceleration trials and then using the test data to verify the model. More than 1,000 randomly chosen training and test sets were considered. The average training and test errors are 0.56% and 0.96%, respectively. As the final model, we chose one with zero training and test errors, in which 79 trials were used as the training set (40 braking and 39 acceleration trials) and 20 trials were used as the test set (ten braking and ten acceleration trials). The resulting values of the model parameters in (2) are given by $\beta_B = -282.7$ mm/s^2 and $\beta_A = 350.5$ mm/s^2. The values of γ_B and γ_A are given by $\gamma_A = 139.6$ mm/s^2 and $\gamma_B = 106.6$ mm/s^2. We set $\bar{d} = 3$ corresponding to three standard deviations.

Trials Experimental Conditions

A total of eight human subjects participated in the study. This set of subjects is different from the set used to generate the human driving model. To start the experiment, the subjects were given an introduction about the setup. This was followed by a practice session in which the subject drove the vehicle on an outer path. The autonomous vehicle was run on the inner path at a constant speed of 500 mm/s. Subjects were free to drive the human-driven vehicle at any speed between the points Pt_1 and Pt_2. Between points Pt_2 and D^P, the speed module keeps the vehicle speed at 600 mm/s. This ensures that the human-driven vehicle does not cross the decision point with minimum or maximum speed. Thus, we instructed the human subjects to either accelerate or decelerate as soon as they crossed the decision point D^P to force the two vehicles in the bad set at the same time.

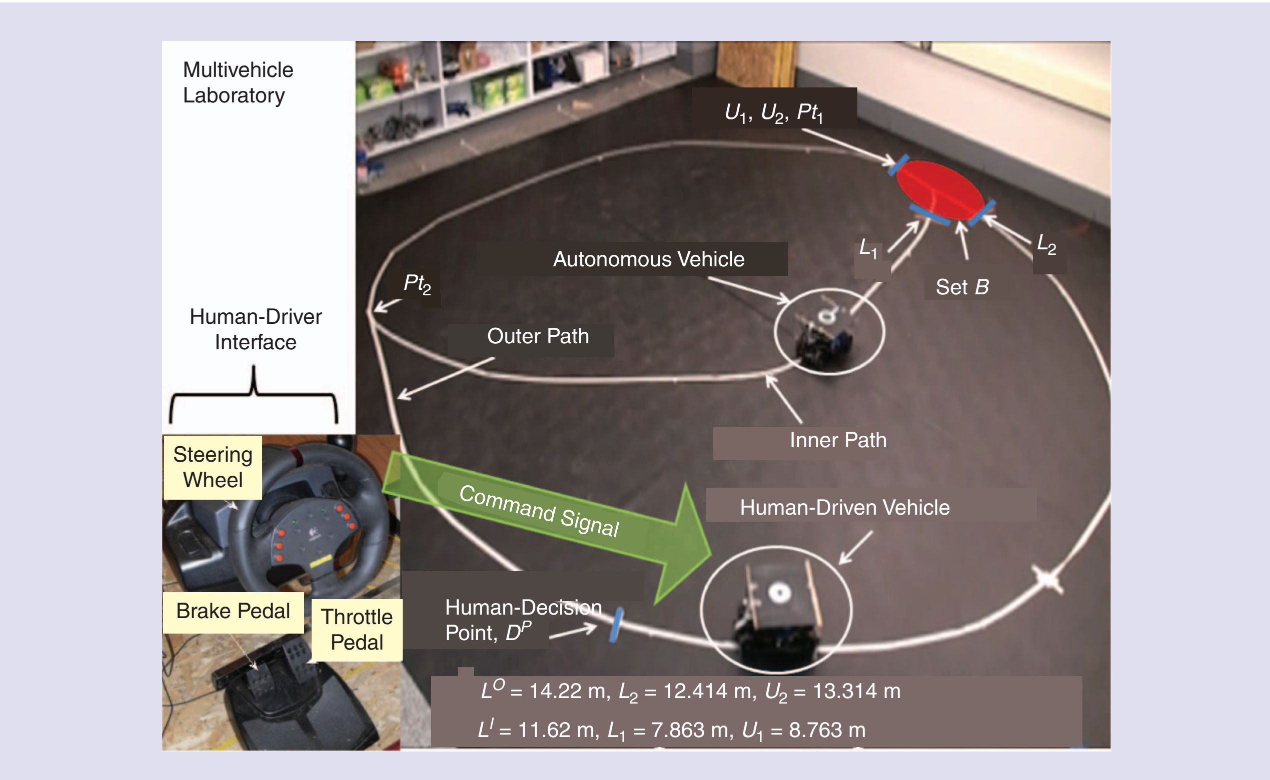

Figure 3. Human–driver interface and roundabout system. L^O is the length of the outer path, and L^I is the length of the inner path.

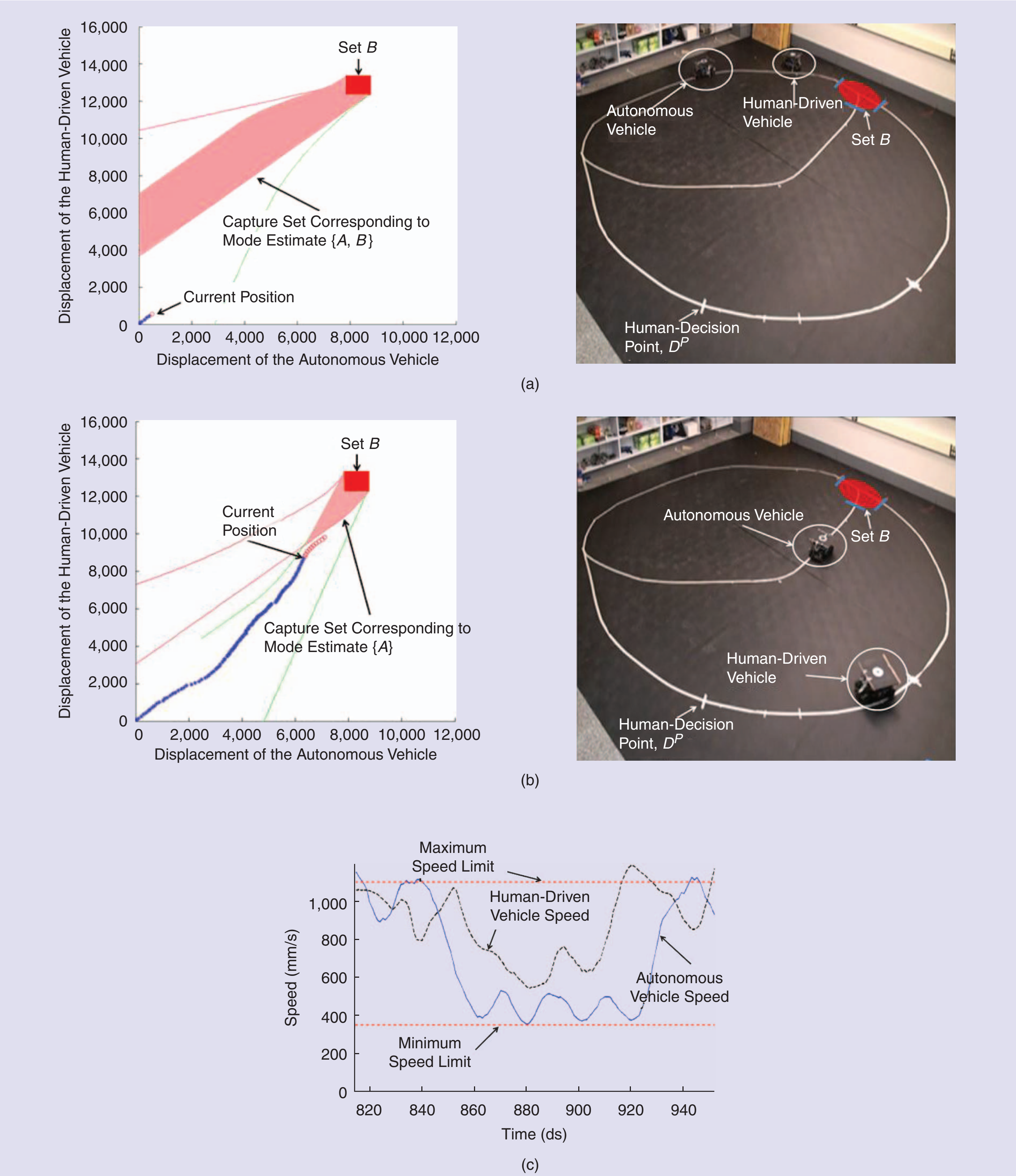

Figure 4. (a), (b), (d), and (e) show the displacement of autonomous and human-driven vehicles along their paths on the *x* and *y* axis, respectively, along with the corresponding snapshots from the experiment. The slice of the current mode-dependent capture set, corresponding to the current velocity of the two vehicles, is shown as the area shaded in red. In the case when the hidden mode is not known, both braking and acceleration are taken as possible modes resulting in a larger capture set (a). *(Continued on next page)*

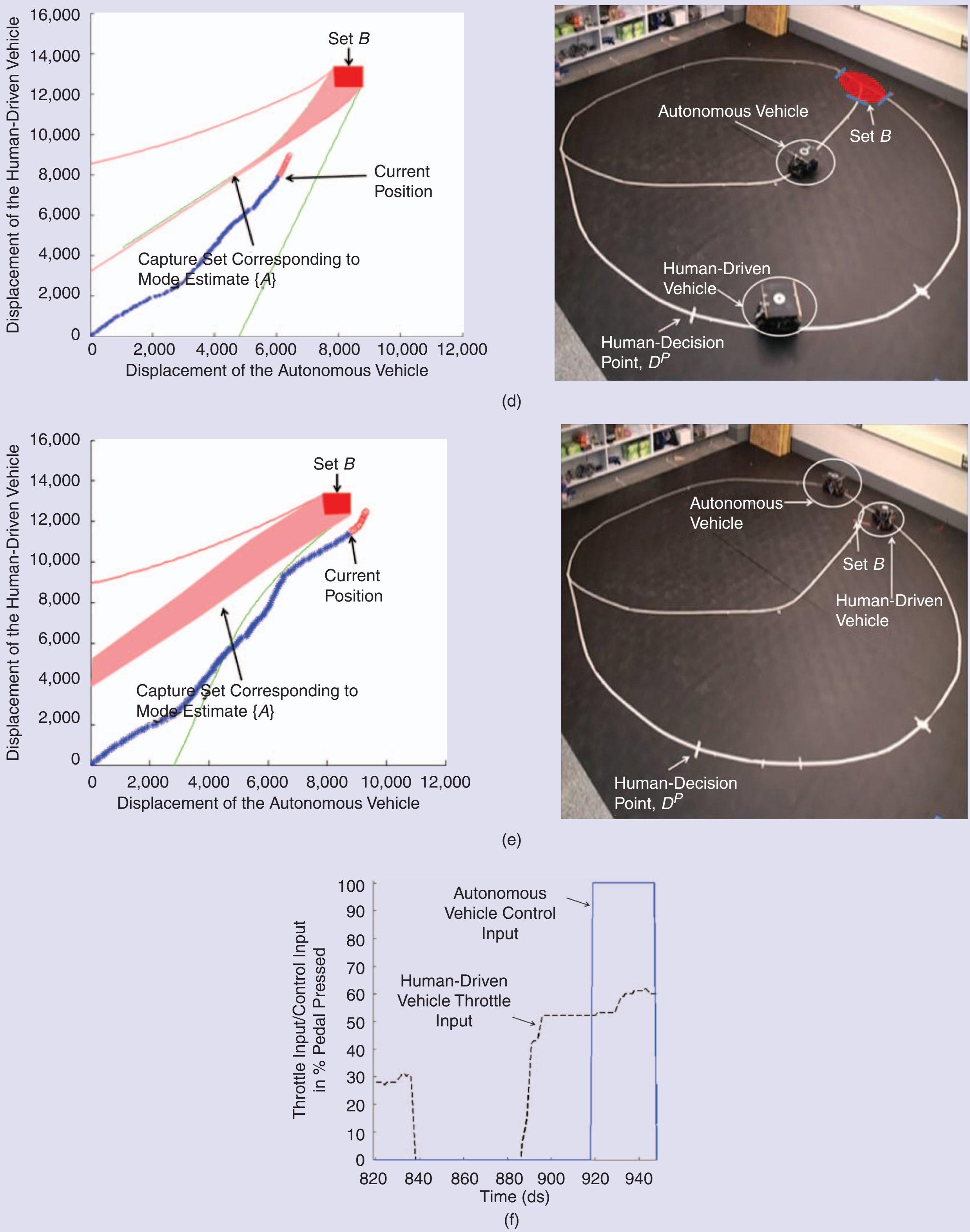

Figure 4. (Continued) With more data, the estimator identifies the mode as acceleration and thus the capture set shrinks (d). The control input is applied in (b) since the predicted state (denoted by red circles) enters the capture set. The applied control keeps the two vehicles from entering the bad set as shown in (e). The velocity is shown in (c), and the control input is shown in (f).

Mode Estimator Implementation

We use a discrete time form of the estimator proposed in the "Problem Solution" section. Since the driver decides to switch the mode to brake or accelerate once the human-driven vehicle crosses D^P, the mode estimator running on the autonomous vehicle uses the continuous state measurements of the human-driven vehicle after it crosses D^P. The instance $n = 0$ corresponds to the time step when the human-driven vehicle crosses this decision point. We take $N = 20$ and consider $n > N$. At the nth time step after the human-driven vehicle crosses the human-decision point, the estimate is calculated by using the formula: $\hat{\beta}(n) = (1/n-1)\Sigma_{k=2}^{n} a(k)$. Hence, n time steps after the human-driven vehicle crosses the decision point, $y(n)$ is given by $y(n) = y_A$ if $|\hat{\beta}(n) - \beta_B| > \gamma_B \bar{d}$, $y(n) = y_B$ if $|\hat{\beta}(n) - \beta_A| > \gamma_A \bar{d}$, and $y(n) = \epsilon$ otherwise.

The human–driver interface comprises a steering wheel and two pedals for throttle and brake commands.

Control Map Implementation

We introduce the following discretization of system H given in (1) and (2) (employing forward Euler approximation) with step size $dT > 0$, $i \in \{1, 2\}$, and index j: $p_i[j+1] = p_i[j] + F_1^i(v_i[j], \alpha_i[j])$ and $v_i[j+1] = \bar{F}^i(v_i[j], \alpha_i[j])$, where $F_1^i = dT\, v_i[j]$, $\bar{F}^i(v_i[j], \alpha_i[j]) = v_i[j] + dT\gamma(v_i[j], \alpha_i[j])$, $\gamma(v_i, \alpha_i) := \alpha_i$ if $v_i + \alpha_i dT < v_{\max}$ and $v_i + \alpha_i dT > v_{\min}$, $\gamma(v_i, \alpha_i) := (v_{\max} - v_i)/dT$ if $v_i + \alpha_i dT > v_{\max}$, and $\gamma(v_i, \alpha_i) := (v_{\min} - v_i)/dT$ if $v_i + \alpha_i dT < v_{\min}$. We define the notation for a sequence of constant inputs α_i for $i \in \{1, 2\}$: $\bar{F}^{i,0}(v_i, \alpha_i) := v_i$ and $\bar{F}^{i,k+1}(v_i, \alpha_i) := \bar{F}^i(\bar{F}^{i,k}(v_i, \alpha_i), \alpha_i)$ with $k \in \mathbb{N}$. The value of $p_i[k]$ starting from initial conditions (p_i, v_i) can be calculated as $p_i[k] = p_i + \sum_{j=0}^{k-1} F_1^i(\bar{F}^{i,j}(v_i, \alpha_i), \alpha_i)$. Since $\text{Bad} = [L_1, U_1] \times \mathbb{R} \times [L_2, U_2] \times \mathbb{R}$, define for $i \in \{1, 2\}$ the sequences $L_1^k(v_1, \alpha_1) := L_1 - \sum_{j=0}^{k-1} F_1^i(\bar{F}^{i,j}(v_1, \alpha_1), \alpha_1)$, $U_1^k(v_1, \alpha_1) := U_1 - \sum_{j=0}^{k-1} F_1^i(\bar{F}^{i,j}(v_1, \alpha_1), \alpha_1)$, $L_2^k(v_2, \max(\alpha_2)) := L_2 - \sum_{j=0}^{k-1} F_1^i(\bar{F}^{i,j}(v_2, \max(\alpha_2)), \max(\alpha_2))$, $U_2^k(v_2, \min(\alpha_2)) := U_2 - \sum_{j=0}^{k-1} F_1^i(\bar{F}^{i,j}(v_2, \min(\alpha_2)), \min(\alpha_2))$, where $\max(\alpha_2) = \beta_q + \gamma_q \bar{d}$ and $\min(\alpha_2) = \beta_q - \gamma_q \bar{d}$ when $\hat{q} = q$, while $\max(\alpha_2) = \beta_A + \gamma_A \bar{d}$ and $\min(\alpha_2) = \beta_B - \gamma_B \bar{d}$ when $\hat{q} = \{A, B\}$. Then, one can show that $\text{Pre}(\hat{q}, \text{Bad})_u = \{x \in X \mid \exists k \geq 0 \text{ s.t. } L_1^k(v_1, \alpha_1) < p_1 < U_1^k(v_1, \alpha_1) \text{ and } L_2^k(v_2, \max(\alpha_2)) < p_2 < U_2^k(v_2, \min(\alpha_2))\}$. Hence, given mode estimate $\hat{q}$, $\text{Pre}(\hat{q}, \text{Bad})_{u_L}$ and $\text{Pre}(\hat{q}, \text{Bad})_{u_H}$ are computed for the given pair of speeds (v_1, v_2) as a union of rectangles in the position plane. Checking whether a point $x = (p_1, v_1, p_2, v_2)$ is in $\text{Pre}(\hat{q}, \text{Bad})_{u_L} \cap \text{Pre}(\hat{q}, \text{Bad})_{u_H}$ is performed by comparing (p_1, p_2) against the upper and lower bounds L_1^k, U_1^k, L_2^k, and U_2^k. Moreover, to check whether $p_1 \in [L_1^k, U_1^k]$, it is enough to compute such intervals only while $U_1^k > p_1$, since the sequences $\{L_1^k\}_{k\geq 0}, \{U_1^k\}_{k \geq 0}$, $\{L_2^k\}_{k \geq 0}$, and $\{U_2^k\}_{k \geq 0}$ are strictly decreasing [20]. Thus, we only need to make a finite number of computations.

To implement the feedback map $\hat{\pi}(\hat{q}, x)$ of the "Problem Solution: Computational Tools" section, we need to track when the continuous flow hits the boundary of the relevant set Pre(.,.). In discrete time, we consider the continuous state to be on the boundary of Pre(.,.) when it is outside it while its prediction forward in time is inside it. To make this procedure robust to both communication and actuator delays, we consider ten forward predictions in time instead of only one.

Experimental Results

The cumulative time for which the trials were conducted is 3,479 s, resulting in a total of 97 instances of collision avoidance in which the autonomous vehicle applied control to avoid a collision. In doing so, the autonomous vehicle entered the capture set in three such instances and resulted in a collision in one such instance, resulting in an overall success rate of 96.9%. During the total duration of the experiments, the mode was estimated as A (acceleration) 102 times, as B (braking) 45 times, and remained at $\{A, B\}$ (acceleration or braking) nine times. These results are presented in Table 1. All mode estimations are correct. Figure 4 shows a collision-avoidance instance when the human-driven vehicle mode was identified as A.

Discussion and Conclusions

In this article, we have illustrated the application of a formal hybrid control approach to design semiautonomous multivehicle systems that are guaranteed to be safe. Our experimental results illustrate that, in a structured task, such as driving, simple human-decision models can be effectively learned and employed in a feedback control system that enforces a safety specification. They also highlight how the incorporation of these models in a safety control system makes the control actions required for safety less conservative. In fact, by virtue of the mode estimate, the current (mode-dependent) capture set to avoid guaranteeing safety is considerably smaller than the capture set to be avoided when the mode estimate is not available. This is essential for the practical applicability of cooperative active safety systems. In our data set, the flow entered the capture set only 3% times. These failures are mainly due to communication delays between the vehicles and the workstation. These delays, when significant, cause the calculated capture set to be different from the actual one and hence may cause to enforce control too late. These delays, in future work, should be formally accounted for in the models and in the safety control algorithm.

More complex models of human decisions in the proximity of an intersection and the incorporation of additional details, such as weather conditions and road geometry, offer the potential for reducing the conservatism of safe control actions even further. Future work will also consider

Table 1. Mode estimation for various subjects.

Subject Number	Duration (s)	Mode A	Mode B	Mode $\{A, B\}$	Number of CA Instances	Times Entered $\hat{C}$	Times Entered Bad
1	374.8	9	6	1	14	1	0
2	265	8	5	0	8	1	0
3	258	5	3	1	5	1	1
4	670	18	6	2	19	0	0
5	560	17	7	3	6	0	0
6	230	11	2	0	7	0	0
7	522	16	10	0	16	0	0
8	600	18	6	2	22	0	0

The first column shows the subject number, the second column presents the total trial time, the third, fourth, and fifth columns show the number of times the mode was identified as acceleration $\{A\}$, braking $\{B\}$, or remained at $\{A, B\}$, respectively. The sixth column shows the number of collision-avoidance instances generated by the subject. The seventh column shows the times the flow entered the capture set. The last column shows the number of times the flow entered the bad set Bad.

the extension to the case in which vehicles are not known to evolve on a fixed route. This case will be handled by keeping track of routes that are compatible with the position and speed of the vehicle and by progressively eliminating those that become incompatible. The models considered here are deterministic because most of the tools currently available to perform safety control have assumed deterministic models, wherein uncertainty is bounded. However, human decision models are more naturally captured by stochastic frameworks, in which uncertainty due to variability in both subjects and realizations of the same decision is probabilistic (see [27] for a review on the topic). As results in stochastic safety verification and design become available [5], [9], it will be important to extend the proposed techniques of this article to safety control of stochastic hybrid automata in which the mode estimate is constructed probabilistically.

By virtue of the order-preserving dynamics of the vehicles and the fact that the bad set is convex, the complexity of the algorithm that calculates the capture set (Algorithm 1) is linear with the number of continuous variables and inputs (see [13] and [20]). Hence, the algorithm can be efficiently implemented in real time. When there are more than two vehicles, the bad set is not convex, and in general, determining an exact solution is harder. However, one can perform modular synthesis in which a two-vehicle collision avoidance routine is employed as a control primitive [17], or exploit the order-preserving structure of the system to obtain suitable abstractions for which the problem is computationally simpler. This is subject of current research.

Finally, in any real-life implementation of cooperative active safety systems, the algorithms implemented by the autonomous vehicle should be capable of interacting with a human driver. In other words, they should first warn the driver, suggest actions, and take control of the vehicle only when the driver is incapable of preventing a collision. Hence, future work will consider the incorporation of human response time to warnings in the algorithms and the problem of establishing when it is absolutely necessary to override a human driver for maintaining safety.

Acknowledgment

The authors thank Matt McCullough and Mads Almassalkhi for their help in the experiment trials. This work was in part supported by NSF CAREER Award Number CNS-0642719.

References

[1] Car-2-Car Communication Consortium [Online]. Available: http://www.car-to-car.org

[2] Cooperative intersection collision avoidance systems (CICAS) [Online]. Available: http://www.its.dot.gov/cicas

[3] Vehicle Infrastructure Integration Consortium (VIIC) [Online]. Available: http://www.vehicle-infrastructure.org

[4] Vehicle infrastructure integration (VII) [Online]. Available: http://www.its.dot.gov/vii

[5] A. Abate, J.-P. Katoen, J. Lygeros, and M. Prandini, "Approximate model checking of stochastic hybrid systems," *Eur. J. Control*, vol. 16, no. 6, pp. 624–641, 2010.

[6] T. Akita, S. Inagaki, T. Suzuki, S. Hayakawa, and N. Tsuchida, "Hybrid system modeling of human driver in the vehicle following task," in *Proc. SICE, 2007 Annu. Conf.*, pp. 1122–1127.

[7] M. Althoff, O. Stursberg, and M. Buss, "Model-based probabilistic collision detection in autonomous driving," *IEEE Trans. Intell. Transport. Syst.*, vol. 10, no. 2, pp. 299–310, 2009.

[8] L. Alvarez and R. Horowitz, "Hybrid controller design for safe maneuvering in the PATH AHS architecture," in *Proc. American Control Conf.*, Albuquerque, NM, 1997, pp. 2454–2459.

[9] S. Amin, A. Abate, M. Prandini, J. Lygeros, and S. Sastry, "Reachability analysis for controlled discrete time stochastic hybrid systems," in *Hybrid Systems: Computation and Control.* (Lecture Notes in Computer Science 3927), J. Hespanha and A. Tiwari, Eds. Berlin: Germany: Springer-Verlag, 2006, pp. 49–63.

[10] A. Balluchi, L. Benvenuti, M. D. Di Benedetto, and A. Sangiovanni-Vincentelli, "Design of observers for hybrid systems," in *Hybrid Systems: Computation and Control.* (Lecture Notes in Computer Science, vol. 2289),

C. J. Tomlin and M. R. Greensreet, Eds. Berlin: Germany: Springer-Verlag, 2002, pp. 76–89.
[11] M. Campbell, M. Egerstedt, J. P. How, and R. M. Murray, "Autonomous driving in urban environments: Approaches, lessons and challenges," *Philos. Trans. R. Soc.*, vol. 368, no. 1928, pp. 4649–4672, 2010.
[12] D. Del Vecchio, "Observer-based control of block triangular discrete time hybrid automata on a partial order," *Int. J. Robust Nonlinear Control*, vol. 19, no. 14, pp. 1581–1602, 2009.
[13] D. Del Vecchio, M. Malisoff, and R. Verma, "A separation principle for a class of hybrid automata on a partial order," in *Proc. American Control Conf.*, 2009.
[14] D. Del Vecchio, R. M. Murray, and E. Klavins, "Discrete state estimators for systems on a lattice," *Automatica*, vol. 42, no. 2, pp. 271–285, 2006.
[15] D. Del Vecchio, R. M. Murray, and P. Perona, "Primitives for human motion: A dynamical approach," in *Proc. IFAC World Congr.*, Barcelona, 2002.
[16] D. Del Vecchio, R. M. Murray, and P. Perona, "Decomposition of human motion into dynamics-based primitives with application to drawing tasks," *Automatica*, vol. 39, no. 12, pp. 2085–2098, 2003.
[17] J. Duperret, M. Hafner, and D. Del Vecchio, "Formal design of a provably safe robotic roundabout system," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2010, pp. 2006–2011.
[18] O. Maler, E. Asarin, and A. Pnueli, "Symbolic controller synthesis for discrete and timed systems," in *Hybrid Systems II* (Lecture Notes in Computer Science, vol. 999), P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry, Eds. Berlin, Germany: Springer-Verlag, 1995, pp. 1–20.
[19] J. A. Haddon, D. N. Godbole, A. Deshpande, and J. Lygeros, "Verification of hybrid systems: Monotonicity in the AHS control system," in *Hybrid Systems III* (Lecture Notes in Computer Science, vol. 1066), R. Alur, T. A. Henzinger, and E. D. Sontag, Eds. Berlin: Springer-Verlag, 1996, pp. 161–172.
[20] M. Hafner and D. Del Vecchio, "Computation of safety control for uncertain piecewise continuous systems on a partial order," in *Proc. Conf. Decision and Control*, 2009, pp. 1671–1677.
[21] R. Horowitz and P. Varaiya, "Control design of an automated highway system," *Proc. IEEE*, vol. 88, no. 7, pp. 913–925, July 2000.
[22] A. B. Kurzhanski and P. Varaiya, "Ellipsoidal techniques for hybrid dynamics: The reachability problem," in *New Directions and Applications in Control Theory* (Lecture Notes in Control and Information Sciences, vol. 321), W. P. Dayawansa, A. Lindquist, and Y. Zhou, Eds. Berlin: Springer, 2005, pp. 193–205.
[23] C. F. Lin, A. G. Ulsoy, and D. J. LeBlanc, "Vehicle dynamics and external disturbance estimation for vehicle path prediction," *IEEE Trans. Control Syst. Technol.*, vol. 8, no. 3, pp. 508–518, 2000.
[24] J. Lygeros, D. N. Godbole, and S. Sastry, "A verified hybrid controller for automated vehicles," in *Proc. Conf. Decision and Control*, Kobe, Japan, 1996, pp. 2289–2294.
[25] J. Lygeros and N. Lynch, "Strings of vehicles: Modeling and safety conditions," in *Hybrid Systems: Computation and Control* (Lecture Notes in Computer Science 1386), T. Henzinger and S. Sastry, Eds. Berlin: Springer, 1998, pp. 273–288.
[26] J. Lygeros, C. J. Tomlin, and S. Sastry, "Controllers for reachability specifications for hybrid systems," *Automatica*, vol. 35, no. 3, pp. 349–370, 1999.
[27] T. B. Moeslunda, A. Hiltonb, and V. Krügerc, "A survey of advances in vision-based human motion capture and analysis," *Comput. Vis. Image Understanding*, vol. 104, no. 2–3, pp. 90–126, 2006.
[28] U.S. DOT National Highway Traffic Administration (NHTSA), "Analysis of fatal crashes due to signal and stop sign violations," NHTSA, Cambridge, MA, DOT HS 809 779, 2004.
[29] L. Pallottino, V. G. Scordio, A. Bicchi, and E. Frazzoli, "Decentralized cooperative policy for conflict resolution in multivehicle systems," *IEEE Trans. Robot.*, vol. 23, no. 6, pp. 1170–1183, 2007.
[30] A. Polychronopoulos, M. Tsogas, A. J. Amditis, and L. Andreone, "Sensor fusion for predicting vehicles' path for collision avoidance systems," *IEEE Trans. Intell. Transport. Syst.*, vol. 8, no. 2, pp. 549–562, 2007.
[31] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *Hybrid Systems: Computation and Control* (Lecture Notes in Computer Science, vol. 2993), R. Alur and G. Pappas, Eds. Berlin: Germany: Springer-Verlag, 2004, pp. 477–492.
[32] O. Shakernia, G. J. Pappas, and S. Sastry, "Semi-decidable synthesis for triangular hybrid systems," in *Hybrid Systems: Computation and Control* (Lecture Notes in Computer Science, vol. 2034), M. D. Di Benedetto and A. Sangiovanni-Vincentelli, Eds. Berlin: Springer-Verlag, 2001, pp. 487–500.
[33] T. Suzuki, "Advanced motion as a hybrid system," *Electron. Commun. Japan*, vol. 93, no. 12, pp. 35–43, 2010.
[34] C. J. Tomlin, J. Lygeros, and S. Sastry, "A game theoretic approach to controller design for hybrid systems," *Proc. IEEE*, vol. 88, no. 7, pp. 949–970, 2000.
[35] C. J. Tomlin, I. Mitchell, A. M. Bayen, and M. Oishi, "Computational techniques for the verification of hybrid systems," *Proc. IEEE*, vol. 91, no. 7, pp. 986–1001, 2003.
[36] B. Tovar and S. M. LaValle, "Visibility-based pursuit-evasion with bounded speed," in *Proc. Workshop Algorithmic Foundations of Robotics*, 2006.
[37] R. Verma and D. Del Vecchio, "Continuous control of hybrid automata with imperfect mode information assuming separation between state estimation and control," in *Proc. Conf. Decision and Control*, 2009, pp. 3175–3181.
[38] R. Verma and D. Del Vecchio, "Control of hybrid automata with hidden modes: Translation to a perfect state information problem," in *Proc. Conf. Decision and Control*, 2010, pp. 5768–5774.
[39] R. Verma and D. Del Vecchio, "Safety control of hidden mode hybrid systems," *IEEE Trans. Automat. Contr.*, to be published.
[40] R. Verma, D. Del Vecchio, and H. Fathy, "Development of a scaled vehicle with longitudinal dynamics of a HMMWV for an ITS testbed," *IEEE/ASME Trans. Mechatronics*, vol. 13, no. 1, pp. 46–57, 2008.
[41] M. De Wulf, L. Doyen, and J.-F. Raskin, "A lattice theory for solving games of imperfect information," *Hybrid Systems: Computation and Control* (Lecture Notes in Computer Science, vol. 3927), J. Hespanha and A. Tiwari, Eds. Berlin, Germany: Springer-Verlag, 2006, pp. 153–168.

Rajeev Verma, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA. E-mail: rajverma@umich.edu.

Domitilla Del Vecchio, Department of Mechanical Engineering, MIT, Cambridge, MA 02139, USA. E-mail: ddv@mit.edu.