2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Development and experimental validation of a semi-autonomous cooperative active safety system

Rajeev Verma and Domitilla Del Vecchio

Abstract— **In this paper, the problem of collision avoidance between two vehicles is considered, in which one vehicle is autonomous and the other one is human-driven. This problem arises in cooperative active safety systems at traffic intersections, mergings, and roundabouts, in which some vehicles are equipped with on-board communication and automatic control, while others are not capable of communicating and are human-driven. We model the human driving behavior through a hybrid automaton, whose current mode is determined by the driver's decisions, and solve the problem as a safety control problem for hybrid systems with imperfect state information. The experimental results demonstrate that our solution is substantially less conservative than solutions employing worst-case design.**

I. INTRODUCTION

Recent technological advancements in embedded computing and communication have made systems that assist drivers to maintain safety a reality. These systems are currently an important part of initiatives by government and industry such as the Crash Avoidance Metrics Partnership (CAMP) [1] and the Vehicle Infrastructure Integration Consortium (VIIC) [2] in the U.S. In these systems, road-side infrastructure will be equipped with sensors that obtain information about the surrounding vehicles and environment. This information will be transmitted to equipped vehicles through vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V) wireless communication. Based on this information, in principle, a coordinated control strategy among vehicles to guarantee collision free systems can be devised. However, safety must be guaranteed in the presence of human-driven vehicles that are not equipped with on-board control and communication capabilities. In [14, 15], this problem is theoretically formulated as a safety control problem for hybrid automata with imperfect mode information. Within this approach, the non-communicating human-driven vehicle is modeled as a hybrid system with unknown discrete state, called Hidden Mode Hybrid System (HMHS). The discrete mode of the HMHS represents the unknown driving intension of the human driver, such as braking, acceleration, or coasting. The vehicle with on-board controller estimates this mode in real time and establishes a control action that maintains safety.

R. Verma is with Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, USA {rajverma}@umich.edu

D. Del Vecchio is with the Department of Mechanical Engineering, MIT, Cambridge, USA ddv@mit.edu

This work was supported by NSF CAREER CNS 0642719

The safety control problem for hybrid systems has been extensively considered in the literature when the state is available for measurement [8–10, 12]. A number of works have addressed the control problem for special classes of hybrid systems with imperfect state information [4, 17]. A controller that utilizes a state estimator for systems with finite number of states is considered in [17]. These results are leveraged to control a class of rectangular hybrid automata with imperfect state information, which can be abstracted by a finite state system. In [4, 7], computationally efficient state estimation and control algorithms were proposed for special classes of hybrid system with order preserving dynamics.

In this paper, we employ the approach of [14, 15] to a semi-autonomous intersection system scenario realized in a multi-vehicle test-bed. Within this test-bed, a scaled vehicle driven by a human through a steering and throttle/brake pedal setup is used along with an autonomous vehicle that can drive on a pre-determined path containing a conflict point with the human-driven vehicle. Human behavior near the intersection is modeled by a hybrid automaton that can be in either of two modes: acceleration or braking. The human-driving behavior parameters are estimated through a process of supervised learning. The dynamic feedback map is composed of a mode estimator and a static feedback map. The mode estimator, based on position measurements, determines the current driving mode of the human-driven vehicle. The autonomous vehicle, on the basis of the current mode uncertainty, determines the control map that guarantees that the current system configuration is kept outside of a current mode-dependent capture set. This results in safe throttle/brake commands applied to the autonomous vehicle.

This paper is organized as follows: in Section II, we review the problem definition and solution as taken from [14, 15]; Section III presents the application scenario; we discuss the experimental setup and results in Sections IV and V.

II. SAFETY CONTROL PROBLEM FOR HIDDEN MODE HYBRID SYSTEMS

We now formally introduce the problem by first defining the general hybrid automaton model.

Definition 1. A *Hybrid Automaton with Uncontrolled Mode Transitions* H is a tuple $H = (Q, X, U, D,$

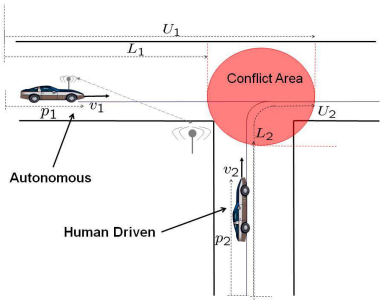


Fig. 1. **Two-vehicle Conflict Scenario**. Vehicle 1 is autonomous and communicates with the infrastructure via wireless, while vehicle 2 is human-driven and does not communicate with the infrastructure. The longitudinal displacement and speed of i^{th} vehicle is denoted by p_i and v_i, $i \in \{1, 2\}$. A collision occurs when more than one vehicle occupies the conflict area at the same time.

$\Sigma, Inv, R, f)$, 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; $Inv = \{\epsilon\}$ is the discrete set of silent events, which correspond to no transition occurring; $R : Q \times \Sigma \to Q$ 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.

For a hybrid automaton H, we denote by $\mathcal{T} = \bigcup_{i=0}^{N}[\tau_i, \tau_i')]$ a hybrid time trajectory [9] such that $\sigma(\tau_i') \in \Sigma$ and $\sigma(t) \in Inv$ for $t \in [\tau_i, \tau_i')$ for all i such that $\tau_i < \tau_i'$. The "$)$]" parenthesis denotes that the last interval (if $N < \infty$) may be open or closed. We thus represent H by $q(\tau_{i+1}) = R(q(\tau_i'), \sigma(\tau_i'))$, $\sigma(\tau_i') \in \Sigma$ and $\dot{x}(t) = f(x(t), q(t), u(t), d(t)), d(t) \in D, \sigma(t) \in Inv$, where τ_i for $i \in \{0, ..., N\}$ are the times at which a discrete transition takes place and are such that $\tau_i \leq \tau_i' = \tau_{i+1}$, $q(\tau_{i+1})$ denotes the value of q after the i^{th} transition, $q(t) := q(\sup_{\tau_i \leq t} \tau_i)$ for $t \in \mathcal{T}$ and $\sigma(t) \in Inv$, $x(0) = x_0 \in X$, and $q(\tau_0) = q_0 \in Q$. We assume without loss of generality that $\tau_0 = 0$. Since discrete transitions change only the discrete state, we have that $x(\tau_{i+1}) = x(\tau_i')$ for all i. For input signal $\boldsymbol{\sigma} : \mathcal{T} \to \Sigma$, we denote the discrete state trajectory by $\phi_q(t, q_0, \boldsymbol{\sigma}) = q(t)$ with $q(0) = \phi_q(0, q_0, \boldsymbol{\sigma})$. We define the set of reachable modes from any initial set of modes $\bar{q} \subset Q$ by $\mathcal{R}(\bar{q}) := \bigcup_{q_0 \in \bar{q}} \bigcup_{t \geq 0} \bigcup_{\boldsymbol{\sigma}} \phi_q(t, q_0, \boldsymbol{\sigma})$.

Definition 2. A *Hidden Mode Hybrid System* (HMHS) is a hybrid automaton with uncontrolled mode transitions in which the discrete state $q(t)$ is not measured and q_0 is only known to belong to a set $\bar{q}_0 \subseteq Q$.

We denote a HMHS by H in the remainder of the paper. The only information about the mode is its initial uncertainty, denoted $\bar{q}_0 \subseteq Q$, the measured signals $x(t)$ and the control signal $u(t)$. 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 all the available information. In order to keep track of the current mode uncertainty, we introduce a discrete state estimate and formulate the control problem as one with perfect state information [13–15].

Definition 3. A *discrete state estimate* is a time-dependent set, denoted $\hat{q}(t) \in \hat{Q}$, with the properties that (i) $q(t) \in \hat{q}(t)$ for all $t \geq 0$; (ii) For $t_2 \geq t_1$, we have that $\hat{q}(t_2) \subseteq \mathcal{R}(\hat{q}(t_1))$.

Define the new hybrid automaton $\hat{H} = (\hat{Q}, X, U, D, Y, \hat{Inv}, \hat{R}, f)$, in which $\hat{Q}$ is a new set of discrete states, Y is a set of discrete events, $\hat{Inv} = \{\epsilon\}$ is a set of silent events with $Y \cap \hat{Inv} = \emptyset$, $\hat{R} : \hat{Q} \times Y \to \hat{Q}$ is a discrete state transition map. Let $\hat{\mathcal{T}} = \bigcup_{i=0}^{\hat{N}}[\hat{\tau}_i, \hat{\tau}_i')]$ be a hybrid time trajectory such that $\hat{\tau}_0 = \tau_0$, $y(\hat{\tau}_i') \in Y$ and $y(t) \in \hat{Inv}$ for $t \in [\hat{\tau}_i, \hat{\tau}_i')$ for all i such that $\hat{\tau}_i < \hat{\tau}_i'$. We represent $\hat{H}$ by $\hat{q}(\hat{\tau}_{i+1}) = \hat{R}(\hat{q}(\hat{\tau}_i'), y(\hat{\tau}_i'))$, $y(\hat{\tau}_i') \in Y$ and $\dot{\hat{x}}(t) \in f(\hat{x}(t), \hat{q}(t), u(t), d(t)), d(t) \in D, y(t) \in \hat{Inv}$, where we have defined $\hat{q}(t) := \hat{q}(\sup_{\hat{\tau}_i \leq t} \hat{\tau}_i)$ for all $t \in \hat{\mathcal{T}}$. The map $\hat{R}$ is such that $\hat{q}(t)$ is a discrete state estimate, $\hat{x}(0) = x_0$ and $\hat{q}(\hat{\tau}_0) = \bar{q}_0$. This in turn implies that (a) $\hat{R}(\hat{q}, y) \subseteq \mathcal{R}(\hat{q})$ for all $y \in Y$ and $\hat{q} \in \hat{Q}$ and that (b) $\hat{\tau}_0' = \hat{\tau}_0 = 0$ and $y(\hat{\tau}_0')$ is such that $\hat{R}(\hat{q}(\hat{\tau}_0'), y(\hat{\tau}_0')) := \mathcal{R}(\hat{q}(\hat{\tau}_0)) = \mathcal{R}(\bar{q}_0)$. The discrete input $y(t)$ derives information from the measured continuous state signal about the values of $\dot{x}(\tau)$ for $\tau < t$ and uses this information to determine the current values of q compatible with such a derivative (see [3, 5, 6] for more information on mode estimators).

We now define the safety control problem with perfect state information for system $\hat{H}$ in which, the state $\hat{q}(t)$ and $\hat{x}(t) = x(t)$, is measured. Let $\hat{\pi} : \hat{Q} \times X \to U$ be a feedback map. We denote the $\hat{x}$-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 $\hat{C} := \bigcup_{\hat{q} \in \hat{Q}} \left(\hat{q} \times \hat{C}_{\hat{q}}\right)$, in which $\hat{C}_{\hat{q}} := \{x_0 \in X \,|\, \forall\, \hat{\pi}, \exists\, \mathbf{d}, \mathbf{y}, t \geq 0 \; s.t. \; \text{some} \; \phi_{\hat{x}}^{\hat{\pi}}(t, (\hat{q}, x_0), \mathbf{d}, \mathbf{y}) \in Bad\}$ is called mode-dependent capture set. It represents 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. (Control Problem with Perfect State Information) Determine the set $\hat{C}$ and a feedback map $\hat{\pi}$ that keeps any initial condition $(\bar{q}_0, x_0) \notin \hat{C}$ outside $\hat{C}$.

The solution to Problem 1 can be obtained by leveraging results available for control of hybrid automata with perfect state information [14, 15]. For this purpose, for any $\hat{q} \in \hat{Q}$ and $S \subseteq X$ define the operator Pre as $\text{Pre}(\hat{q}, S) := \{x \in X \mid \forall \hat{\pi},\ \exists\ \mathbf{d},\ t \geq 0 \text{ s.t. some } \phi_{\hat{x}}^{\hat{\pi}}(t, (\hat{q}, x), \mathbf{d}, \epsilon) \in S\}$. The set $\text{Pre}(\hat{q}, S)$ is the set of all continuous states that are taken to S for all feedback maps when the mode estimate is kept constant to $\hat{q}$. An algorithmic procedure is defined in [14, 15] for obtaining set $\hat{C}_{\hat{q}}$ on the basis of the Pre operator.

III. Application scenario

Referring to Figure 1, vehicle 1 is autonomous and communicates with the infrastructure, while vehicle 2 is

human-driven and does not communicate its intents. We assume that the infrastructure measures the position and speed of vehicle 2 through road-side sensors such as cameras and magnetic-induction loops and that it transmits this information to the on-board controller of vehicle 1. Vehicle 1 has to use this information to avoid a collision.

Vehicle 1 longitudinal dynamics along its path are 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), $u \in [u_L, u_H]$ represents the input command, $b < 0$ represents the static friction term, and $c > 0$ with the cv_1^2 term modeling air drag (see [16] for more details on the model).

Vehicle 2 is controlled by the driver decisions. There has been a wealth of work on the modeling of human driving behavior for vehicle safety applications [11]. In this work, we model human driving behavior in the proximity of an intersection through a simple hybrid system with two modes: braking and acceleration, that is,

$$\dot{p}_2 = v_2, \dot{v}_2 = \beta_q + \gamma_q d, \tag{1}$$

with $q \in \{A, B\}, d \in [-\bar{d}, \bar{d}]$, 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 and thus we have that $\beta_B < 0$ and that $\beta_A > 0$. 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 several ways in which mode A or mode B can be realized. It also captures (as we shall see in the experimental section) variability among drivers. We finally assume that there is confusion between the modes, that is, $\{\beta_B + \gamma_B[-\bar{d}, \bar{d}]\} \cap \{\beta_A + \gamma_A[-\bar{d}, \bar{d}]\} \neq \emptyset$, which leads to having $\beta_B + \gamma_B\bar{d} \geq \beta_A - \gamma_A\bar{d}$.

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 $x = (p_1, v_1, p_2, v_2)$; $U = [u_L, u_H] \subset \mathbb{R}$; $D = [-\bar{d}, \bar{d}] \subset \mathbb{R}$; $\Sigma = \emptyset$; $R : Q \times \Sigma \to Q$ is the mode update map, which is trivial as $\Sigma = \emptyset$, that is, the mode can start in either A or B and no transitions occur between these two modes and $f : X \times Q \times U \times D \to X$ is the vector field, which is piecewise continuous and 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

$$f_1(p_1, v_1, u) = \begin{pmatrix} v_1 \\ \begin{cases} 0 & \text{if } (v_1 = v_{min} \text{ and } \alpha_1 < 0) \\ & \text{or } (v_1 = v_{max} \text{ and } \alpha_1 > 0) \\ \alpha_1 & \text{otherwise} \end{cases} \end{pmatrix}, \tag{2}$$

$$f_2(p_2, v_2, q, d) = \begin{pmatrix} v_2 \\ \begin{cases} 0 & \text{if } (v_2 = v_{min} \text{ and } \alpha_2 < 0) \\ & \text{or } (v_1 = v_{max} \text{ and } \alpha_2 > 0) \\ \alpha_2 & \text{otherwise} \end{cases} \end{pmatrix}, \tag{3}$$

with $\alpha_1 = au + b - cv_1^2$ and $\alpha_2 = \beta_q + \gamma_q d$. There is a lower non-negative speed limit, v_{min}, and upper speed limit, v_{max}, implying that vehicles cannot go in reverse and guaranteeing liveness of the system. Referring to Figure 1, the set of bad states for system H models collision configurations and it is given by $Bad := \{(p_1, v_1, p_2, v_2) \in \mathbb{R}^4 \mid (p_1, p_2) \in [L_1, U_1] \times [L_2, U_2]\}$.

In this scenario, system $\hat{H} = (\hat{Q}, X, U, D, Y, \hat{Inv}, \hat{R}, f)$, 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 $\hat{q}(0) = \hat{q}_1$, is uniquely defined once the set Y and map $\hat{R}$ are defined. 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 $\hat{R}(\hat{q}_2, y_B) := \hat{q}_3$. In order to establish when A or B are ruled out given the measurement of x, we consider the following estimate $\hat{\beta}(t) = \frac{1}{t}\int_0^t \dot{v}_2(\tau)d\tau,\ t \geq T$, where $T > 0$ is a time window. If the mode is q, then necessarily we 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. For the mode estimator, property (i) is satisfied as if q is currently possible (i.e., $|\hat{\beta} - \beta_q| \leq \gamma_q\bar{d}$), it cannot be discarded starting from $\hat{q}_1$. Similarly, once mode q is discarded, since R does not allow transitions, q cannot be possible even when $|\hat{\beta} - \beta_q| \leq \gamma_q\bar{d}$. Condition (ii) is satisfied as $\hat{q}_2 \subseteq \mathcal{R}(\hat{q}_1)$ and $\hat{q}_3 \subseteq \mathcal{R}(\hat{q}_1)$. For system $\hat{H}$, we have $\hat{C}_{\hat{q}_1} = \text{Pre}(\hat{q}_1, Bad)$, $\hat{C}_{\hat{q}_2} = \text{Pre}(\hat{q}_2, Bad)$ and $\hat{C}_{\hat{q}_3} = \text{Pre}(\hat{q}_3, Bad)$ (refer to [14, 15]).

A. Computational tools

The sets $\text{Pre}(\hat{q}, Bad)$ can be efficiently computed for the application under study. This is because for every mode estimate $\hat{q}$ the continuous dynamics are the parallel composition of two order preserving systems [7]. Specifically, for the application example, define the restricted Pre operators for $i \in \{1, 2, 3\}$ $\text{Pre}(\hat{q}_i, Bad)_{u_L} := \{x \in X \mid \exists\ \mathbf{d},\ t \geq 0$ s.t. some $\phi_{\hat{x}}(t, (\hat{q}_i, x), u_L, \mathbf{d}, \epsilon) \in Bad\}$ and $\text{Pre}(\hat{q}_i, Bad)_{u_H} := \{x \in X \mid \exists\ \mathbf{d},\ t \geq 0$ s.t. some $\phi_{\hat{x}}(t, (\hat{q}_i, x), u_H, \mathbf{d}, \epsilon) \in Bad\}$. Then, we have that (refer to [7])

$$\text{Pre}(\hat{q}_i, Bad) = \text{Pre}(\hat{q}_i, Bad)_{u_L} \cap \text{Pre}(\hat{q}_i, Bad)_{u_H}, \tag{4}$$

for $i \in \{1, 2, 3\}$. Each of the sets $\text{Pre}(\hat{q}_i, Bad)_{u_L}$ and $\text{Pre}(\hat{q}_i, Bad)_{u_H}$ can be computed by linear complexity discrete time algorithms.

For each mode $\hat{q}_i$ for $i \in \{1, 2, 3\}$, a safe control map $\hat{\pi}(\hat{q}_i, x)$ makes the vector field point outside set $\hat{C}_{\hat{q}_i}$ when

x is on the boundary of $\hat{C}_{\hat{q}_i}$. This keeps the state outside $\hat{C}_{\hat{q}}$. For $\hat{H}$, we have $\hat{C}_{\hat{q}_i} = \text{Pre}(\hat{q}_i, Bad)$ for all $i \in \{1, 2, 3\}$, in which the sets $\text{Pre}(\hat{q}_i, Bad)$ for every $i \in \{1, 2, 3\}$ satisfy relation (4). Because of this relation, one can show (refer to [7]) that a control map $\hat{\pi}(\hat{q}_i, x)$ that maintains the state x outside $\text{Pre}(\hat{q}_i, Bad)$ is given by

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

Since we have that $\text{Pre}(\hat{q}_i, Bad) \subseteq \text{Pre}(\hat{q}_1, Bad)$ 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 $\text{Pre}(\hat{q}_1, Bad)$ implies that it is also outside $\text{Pre}(\hat{q}_2, Bad)$ or $\text{Pre}(\hat{q}_3, Bad)$. Therefore, the feedback map above guarantees that the state never enters the capture set.

IV. Experimental setup

The two-vehicle conflict scenario of Figure 1 is realized experimentally in a multi-vehicle test-bed, which we describe here.

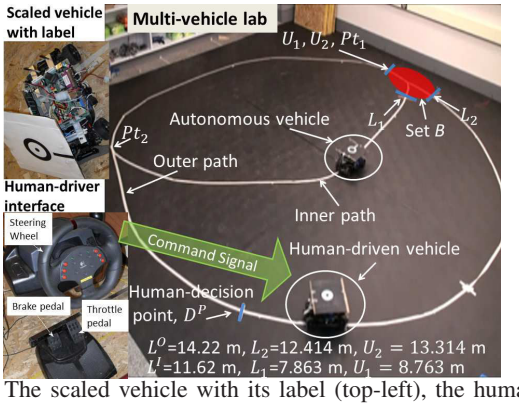


Fig. 2. The scaled vehicle with its label (top-left), the human-driver interface (bottom-left) and the roundabout system (right), L^O is the length of the outer path while L^I is the length of the inner path.

A. Scaled vehicle and human-driver interface

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 vehicle, as shown in Figure 2 (top left), is equipped with an on-board computer (Mini ITX) and a motion controller. The longitudinal response of this vehicle is dynamically similar to that of a high mobility multipurpose wheeled vehicle (HMMWV) [16]. 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 using a human-driver interface. The human-driver interface comprises of a steering wheel and two pedals for throttle and brake commands (see Figure 2). The hardware used is a Logitech MOMO force feedback racing wheel and pedal set. The control algorithms are programmed on the on-board computer.

B. Roundabout system

The roundabout system (Figure 2) is designed to replicate a collision situation at a road intersection where two vehicles merge. There are two circular paths that share a common section on a 6 m by 6 m arena. The human-driven vehicle follows the outer path while the autonomous vehicle follows the inner path in an anti-clockwise direction. A collision is possible at the intersection when both vehicles are in the area shaded red, in Figure 2, at the same time. This area corresponds to the set, $B = \{(p_1, p_2) | (p_1, p_2) \in [L_1, U_1] \times [L_2, U_2]\}$, with $L_1 = 7.863$ m, $L_2 = 12.414$ m, $U_1 = 8.763$ m and $U_2 = 13.314$ m. The length of the outer path is 14.22 m and the inner path is 11.62 m. The human-driver controls the vehicle from the human-driver interface and has the full view of the roundabout system. Point D^P is referred to as the human-decision point, this is the point where the human-driver has to decide if he/she wants to break or accelerate in order to force both vehicles to enter the bad set at the same time. Point D^P is located 6 m before the point L_2 on the outer path. The maximum allowable speed that can be achieved by a vehicle in the roundabout system is 1100 mm/sec and the minimum speed is 350 mm/sec. A PID controller maintains the speed at minimum or maximum if the speed violates these limits. When the two vehicles are simultaneously present in the shared path (between points Pt_1 and Pt_2), another PID controller prevents rear end collision.

An overhead camera based positioning system can simultaneously monitor 6 vehicles with an accuracy in position of 50 mm. Each vehicle is mounted with a track-able and distinguishable, black and white label. The cameras are connected via FireWire to three dedicated desktop computers. Each computer receives input from two cameras and runs image processing and tracking algorithms developed in the lab [1]. The positioning information is transmitted to the vehicles over the wireless network.

C. Learning of human driving model

We model the human-driven vehicle using a hybrid automaton whose discrete state models the intention of the human-driver. We assume that the human either decides to brake or accelerate near the intersection. A set of experiments are performed in which human subjects drive a vehicle on the outer path in the roundabout system (Figure 2). Since we intend to characterize the human driving model, the subjects are directed to either brake or accelerate at the human-decision point, while also avoiding a moving target on the inner path. The data collected is then analyzed to estimate the parameters β_q and γ_q as described in Section II.

[1] https://wikis.mit.edu/confluence/display/DelVecchioLab/Home

In the trials, the vehicle is started 2 m after the collision point (see Figure 2) at a random velocity and approximately follows the outer path. About 4 m before the collision point, the driver is allowed to take control of the vehicle ans is asked randomly to either pass the moving target on the inner path (acceleration trial) or allow the moving target to pass the human-driven vehicle (braking trial). We used 5 different subjects to run 10 acceleration and 10 braking trials each. The sample time and the position of the vehicle are recorded. The data is analyzed starting 3 m before the collision point. We denote the position measurement at time step k by $p(k)$ and the time lapsed between two consecutive steps is dT =0.1 sec. The acceleration/deceleration at time step k is denoted $a(k)$ and can be calculated as $a(k) = \frac{p(k)-2p(k-1)+p(k-2)}{dT^2}$. The average acceleration/deceleration is calculated for the trial as $\bar{a} = \frac{1}{N-1}\sum_{k=2}^{N} a(k)$. A total of 99 trial runs are obtained from 5 subjects.

These trials are divided into the training set, comprising 79 trials with 40 braking and 39 acceleration trials, and the test set comprising of the remaining trials. The model depicting the driver behavior is created by fitting two Gaussian distributions to the training data for braking and acceleration trials. The test data is used to verify the model. In order to obtain the best model, more than 1000 randomly chosen training and testing sets are considered. The average training and testing errors are .56% and .96% respectively. For use as the final model, we chose a model with no training and testing error. The associated Gaussian distribution is shown in Figure 3. From these results, we have that the mean of the acceleration mode is 350.5 mm/sec^2 and that of braking mode is -282.7 mm/sec^2. We thus take the value of parameters in equation (1) as β_B = -282.7 mm/sec^2 and β_A=350.5 mm/sec^2. The values of γ_B and γ_A are given by γ_A =139.6 mm/sec^2 and γ_B =106.6 mm/sec^2. The value of $\bar{d}$ is set to $\bar{d}$ = 3 and corresponds to three standard deviations. This also results in an overlap of human input range in braking and acceleration modes.

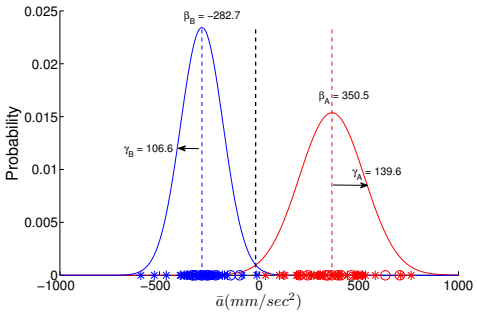


Fig. 3. Gaussian distribution for braking and acceleration trials.

D. Trials experimental conditions

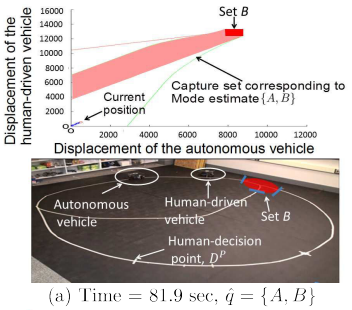
To make sure that the human driving model can generalize and is able to identify the intent of human subjects not present during training, a set of eight subjects that are different from the set used to generate the human-driving model is used. The experiment is started with an introduction to the setup. This is followed by a practice session in which the subject drives the human controlled vehicle on the outer path. Next, the autonomous vehicle is run on the inner path at a constant speed of 500 mm/sec. The speed limits are v_{min} = 350 mm/sec and v_{max} = 1100 mm/sec. The subjects are free to drive the human-driven vehicle at any speed between the points Pt_1 and Pt_2. Since we are interested in scenarios where the mode can be distinguished, the subjects are instructed not to apply any control between point Pt_2 and D^P, while the vehicle speed is maintained at 600 mm/sec. This is done to avoid situation in which the vehicle speed is v_{min} (or v_{max}) at the intersection and the subject decides to apply brake (or throttle) at the decision point, which will in the mode being identified as both braking and acceleration. Thus, we instruct the human subjects to either accelerate or decelerate as soon as they cross the decision point so as to hit the autonomous vehicle or to force the two vehicles in the bad set at the same time.

V. Experimental results

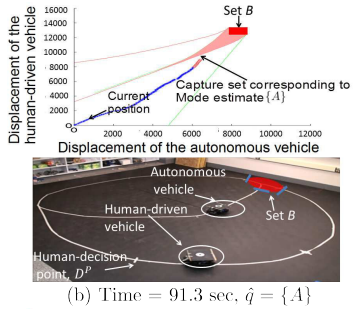
A total of eight subjects took part in the experiments. The duration of each trial depends on the time each vehicle can operate on a single battery charge. A fully charged battery yields an operating time of around 10 to 15 minutes. This operation time is divided into the driver training time and the actual experimentation time. Some subjects learn to drive the vehicle and follow the outer path closely in less than 5 minutes while others take a longer time. This variation in subjects results in the variation of trial length. The shortest trial length that we obtained is 230 seconds while the longest is 600 seconds. The cumulative time for which the trials are conducted is 3479 seconds resulting in a total of 97 instances of collision avoidance in which the human-driver tried to force a collision and the autonomous vehicle applied control in order to avoid the collision. In doing so, the autonomous vehicle entered the capture set in 3 such instances and resulted in a collision in 1 such instance resulting in an overall success rate of 96.9 %. Figure 4 show collision avoidance instances when the human-driven vehicle mode is identified as A.

VI. Conclusion

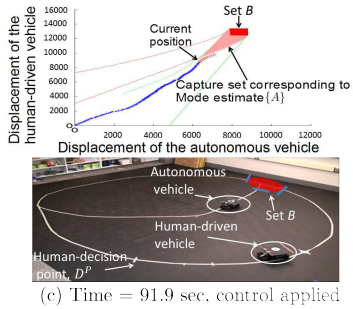
In this paper, we have applied formal techniques for safety control to develop a semi-autonomous cooperative active safety system for collision avoidance between an autonomous and a human-driven vehicle at an intersection. We experimentally validated the safety system in the multi-vehicle lab. The experimental 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



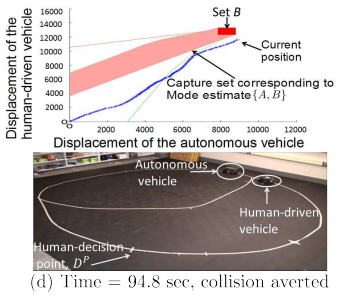
(a) Time = 81.9 sec, $\hat{q} = \{A, B\}$



(b) Time = 91.3 sec, $\hat{q} = \{A\}$



(c) Time = 91.9 sec, control applied



(d) Time = 94.8 sec, collision averted

Fig. 4. Sub-figures (a), (b), (c) and (d) top show the displacement of autonomous and human-driven vehicles along their paths on the x-axis and y-axis respectively, while bottom shows the corresponding snapshot 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 light red. When the hidden mode is not known, both braking and acceleration are taken as possible modes resulting in a larger capture set (sub-figure (a)). With more data, the estimator is able to identify the mode as braking and thus the capture set shrinks (sub-figure (b)). The control input is applied in sub-figure (c) since the predicted state (denoted by red circles) enters the capture set. The applied control keeps the two vehicles from entering the bad set at the same time and thus prevents collision, as shown in sub-figure (d).

specification. They also highlight how the incorporation of these models in a safety control system makes the control actions required for safety less conservative. The experimental data shows that a collision was averted in 97% of the possible conflict situations. The failures can be attributed to the delays in the wireless communication network that can cause the current measurement to be different from the actual value resulting in an erroneous control input. In our future work, these delays will be formally accounted for in the theory.

References

[1] Crash Avoidance Metrics Partnership (CAMP). http://www.camp-ivi.com.

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

[3] A. Balluchi, L. Benvenuti, M. D. Di Benedetto S, and A. L. Sangiovanni-vincentelli. Design of observers for hybrid systems. In *In Hybrid Systems: Computation and Control, volume 2289 of LNCS*, pages 76–89. Springer-Verlag, 2002.

[4] D. Del Vecchio. Observer-based control of block triangular discrete time hybrid automata on a partial order. *International Journal of Robust and Nonlinear Control*, 19(14):1581–1602, 2009.

[5] D. Del Vecchio, R. M. Murray, and E. Klavins. Discrete state estimators for systems on a lattice. *Automatica*, 42(2):271–285, 2006.

[6] D. Del Vecchio, R. M. Murray, and P. Perona. Decomposition of human motion into dynamics-based primitives with application to drawing tasks. *Automatica*, 39(12):2085–2098, 2003.

[7] M. Hafner and D. Del Vecchio. Computation of safety control for uncertain piecewise continuous systems on a partial order. In *Conference on Decision and Control*, pages 1671 –1677, 2009.

[8] 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.), pages 193–205, 2005.

[9] J. Lygeros, C. J. Tomlin, and S. Sastry. Controllers for reachability specifications for hybrid systems. *Automatica*, 35(3):349–370, 1999.

[10] O. Shakernia, G. J. Pappas, and Shankar 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.), Springer Verlag, 2001.

[11] T. Suzuki. Advanced motion as a hybrid system. In *Electronics and Communications in Japan*, 2010.

[12] C. J. Tomlin, I. Mitchell, A. M. Bayen, and M. Oishi. Computational techniques for the verification of hybrid systems. *Proceedings of the IEEE*, 91(7):986–1001, 2003.

[13] B. Tovar and S. M. LaValle. Visibility-based pursuit-evasion with bounded speed. In *Workshop on Algorithmic Foundations of Robotics*, 2006.

[14] R. Verma and D. Del Vecchio. Continuous control of hybrid automata with imperfect mode information assuming separation between state estimation and control. In *Conference on Decision and Control*, pages 3175 –3181, 2009.

[15] R. Verma and D. Del Vecchio. Control of hybrid automata with hidden modes: translation to a perfect state information problem. In *Conference on Decision and Control*, pages 5768 –5774, 2010.

[16] 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 Transactions on Mechatronics*, 13:46–57, 2008.

[17] 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.), Springer-Verlag, pages 153–168, 2006.