Improved Driver Modeling for Human-in-the-Loop Vehicular Control

Katherine Driggs-Campbell, Victor Shia, and Ruzena Bajcsy

***Abstract*— In order to develop provably safe human-in-the-loop systems, accurate and precise models of human behavior must be developed. Driving is a good example of such a system because the driver has full control of the vehicle, and her likely actions are highly dependent on her mental state and the context of the current situation. This paper presents a testbed for collecting driver data that allows us to collect realistic data, while maintaining safety and control of the environmental surroundings. We extend previous work that focuses on set predictions consisting of trajectories observed from the nonlinear dynamics and behaviors of the human driven car, accounting for the driver mental state, the context or situation that the vehicle is in, and the surrounding environment in both highway and intersection scenarios. This allows us to predict driving behavior over long time horizons with extremely high accuracy. By using this realistic data and flexible algorithm, a precise and accurate driver model can be developed that is tailored to an individual and usable in semi-autonomous frameworks.**

I. Introduction

There are number of ways to approach safety when considering safety in human-machine interaction. Some robotic systems approach safety from a mechanical point of view by creating systems that physically cannot harm the human [6]. Another approach is to develop controllers and sensor systems that can guarantee safety for a given system [15]. However, when considering systems that involve or interact with humans (called human-in-the-loop systems), deriving safety boundaries is not a simple task. Human actions and behaviors are often unpredictable and cannot easily be described by normal dynamical methods. Therefore, in order to develop provably safe human-in-the-loop systems, first a useful model of the human must be developed that can be incorporated into a safety algorithm for semiautonomous control. In this paper, we consider driver behavior. This application is a key example as the driver has full control of the system, and her likely actions are highly dependent on her mental state and the context of the current situation.

It is well-known that while humans have many innate skills that make them adept at driving (e.g. flexible and adaptable to new situations, good at decision making), they are fallible and prone to error. In fact, according to a 2008 study, 93% of car crashes are due to human error. On top of that, studies show that at any given moment in America, approximately 660,000 drivers are using cell phones or another electronic device while driving, even though doing so increases the risk of getting into an accident by three times [1], [28]. This has brought rise to a great deal of research in driver modeling and autonomous vehicles to mitigate or hopefully eliminate these collisions [2]. Additionally, many semiautonomous and human-in-the-loop systems are being developed as many believe that autonomous vehicles should be introduced incrementally [16].

One example of a successful semiautonomous system that uses a reachable set methodology is the Volvo City Safety system. When driving in the city (below 35 miles per hour), the system calculates the reachable set of the vehicle for the future 500ms and predicts collisions by checking to see if a detected object is within that set [11]. As noted, this method does not work at high speeds as the reachable set of the vehicle itself becomes too large, leading to an overly invasive system. When considering high speeds, the human can no longer be considered as a disturbance in the system, as the driver has significant influence over the future trajectories of the vehicle. Ideally, the system would function at high speeds and consider the likely actions of the human by modeling the driver to create a more informative reachable set.

Modeling the driver has been considered by a variety of communities, ranging from the human factors to computer vision. The human factors and psychology community have focused on quantifying the amount of attention required for driving and level of distraction for different tasks during driving [7], [24]. Other communities have used a variety of sensors: eye and facial trackers [8], [20], [30], body sensors [26] or other sensors to predict driver intent [12], [14]. In [9], intersection data was collected from traffic to design driver assistance systems, but no individualized models were created. However, these modeling methods fall short of easily being employed in a control framework.

We propose the use of a driver model that incorporates knowledge of an individual driver's likely set of actions to create a set of likely states the driver will visit, to minimize the amount of intervention by an active safety system, as first described in [29]. In this paper, we introduce a new experimental setup that allows us to collect realistic driver data that can be used to create a model of the human mental state, in a safe manner. We hypothesize that by incorporating a mental model of the human and by developing a context aware system, we can better predict driver behavior and thus create a better active safety system. This system can be tailored to a specific driver, and if highly precise, it can be less invasive, while maintaining accuracy.

This material is based upon work supported by the National Science Foundation Award ECCS-1239323 and by the Office of Naval Research MURI Award ONR-N000141310341 and DURIP Award N000141310679.

The study protocol was approved by the University of California Institutional Review Board for human protection and privacy, under Protocol ID 2013-07-5459. Each subject was first informed on the experimental procedure and written informed consent was obtained.

Katherine Driggs-Campbell, Victor Shia, and Ruzena Bajcsy are with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720 USA (e-mail: {krdc,vshia,bajcsy}@eecs.berkeley.edu).

The work here expands upon the data-driven methodology presented in [27] by creating a more precise and accurate model of human driving behavior. Our contributions include (1) using a human-in-the-loop testbed to collect realistic driving and mental state data; (2) enhancing the existing driver modeling algorithm with improved results; and (3) extending the model to consider intersection scenarios in addition to highway driving. We relax some of the assumptions from previous work on the underlying model of the vehicle, by including all inputs (steering, throttle, and braking) into the model. This creates more variability in the predictions, as we consider nonlinear behaviors of the human inputs.

The paper is structured as follows. In Section II, we will present our methods and its advantages over other methods. The setup and execution of the experiments will be described in Section III. Section IV will present our metrics and results for analyzing our system when considering highway driving. Section V will present an extensions of this algorithm to intersection behaviors. While this work focuses on the development of the human model, the methods for applying the model are discussed in Section VI. Finally, we will conclude with a discussion of the results and of the future works.

II. Methodology

The following section will introduce the formulation and implementation of the proposed model. In this work, we are interested in the driver modeling that will influence the control. Refer to [27] for the larger control framework which this driver model can be applied.

A. Modeling Human Behavior

In this subsection, we provide a mathematical formulation of the driver model. We denote the past observed information with $\mathcal{O}$ and the information at the current time with $\mathcal{I}$.

Existence of a Driver State Function: Suppose in the current information set, we are able to obtain driver monitoring data such that:

$$\theta : \mathcal{O} \times \mathcal{I} \to Q \tag{1}$$

where Q is our set of discrete mental states that the driver could be in. In this work, we assume this set of mental states to be attentive, partially attentive, and distracted. This is similar to work in psychology and discrete event systems, which identifies tasks to have no, low, or high mental workload or cognitive distraction, and adjusts based on discrete mental modes [19], [22].

Existence of Distinct Driver Modes: Given that driver behavior heavily depends on context and mental state, we assume that there exist distinct driver modes that depend on this information. As was previously mentioned, we are interested in long time horizon trajectory predictions that will encapsulate the uncertainties and the bounds of the potential future states of the vehicle. This is described as follows:

$$\begin{aligned} \underset{\Delta \subset \mathbb{R}^n}{\text{argmin}} \quad & |\Delta| \\ \text{subject to} \quad & P_{\theta(\mathcal{O},\mathcal{I})}\left[(X(k) - x_0) \subset \Delta | \mathcal{O}, \mathcal{I}\right] \geq \alpha \\ & \forall k \in \{0, \dots, N\} \end{aligned} \tag{2}$$

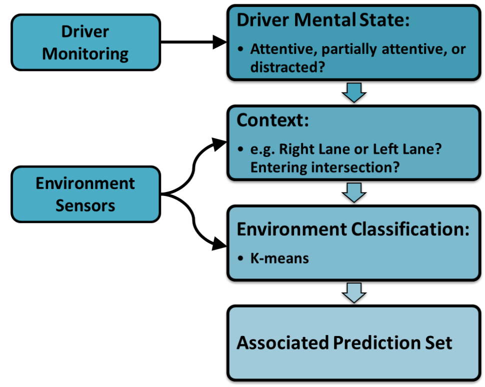


Fig. 1: A flow diagram of the parsing used to identify the driver modes.

where X is the random variable representing the future vehicle trajectory, $x_0 \in \mathbb{R}^n$ is the initial position, N is the time horizon, P_θ is the probability distribution on the trajectories given the driver state and the information sets, and Δ the minimum area set that contains the future trajectory of the vehicle with at least probability $\alpha \in [0, 1]$. This can be interpreted as the α-likely reachable set of the vehicle given the past and current information we have observed from the driver. The output of this optimization program at time k return a set that has a probability distribution that will be referred to as $\Delta_k(\mathcal{O}, \mathcal{I}, \alpha)$.

B. Model Implementation

In this subsection, we describe how we implement the driver model described in Section II-A. Rather than making assumptions on human behavior and fit P_θ to a known probability distribution, we estimate an empirical distribution from the observed data. Here, suppose that for a given context and environment observations, there are distinct modes of behavior. While we assume that the level of distraction and context are known (i.e. we have full knowledge of θ and know when the vehicle is on a highway or approaching an intersection), identifying a mode associated with the environment constraints is more difficult. These constraints are related to the road boundaries and curvature and the surrounding obstacles. We uncover the modes of the environment from sensor data using the k-means algorithm on observed scenario data [17]. This environment representation also includes the level of traffic and relative states via sensing the surrounding vehicles.

The sensors chosen to collect environment data mimic current commercially available sensors such as front and side radar and lane detection modules [4]. Using the front and side radar, we detect the number of obstacles in the vicinity of the ego vehicle and observable states (i.e. relative position and velocity), relative to the ego vehicle. The lane sensors allow us to extract future road bounds.

To create a context aware system, we implement a hierarchical algorithm with the levels as shown in Figure 1. The first two levels determine driver mode using the driver

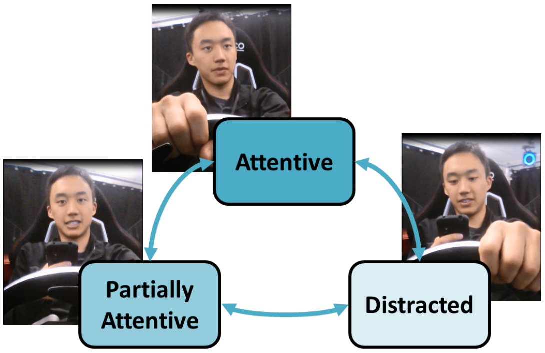


Fig. 2: Diagram showing the transitions between mental states when distractions or actions occur.

mental state (described in Eq. 1 and shown in Fig. 2) and context (e.g. approaching an intersection, position on the road, etc.), which provide insight to the future trajectories of the vehicle. For instance, drivers behave differently during city driving than highway driving and drivers demonstrate different behaviors when in the right or left lane of a two lane road. By parsing the data based off of these detectable contexts, we hypothesize that the system will be able to more accurately identify driver modes. This is a significant improvement to the model proposed in [27] which uses k-means on a time-series vector of driver state, context and environment.

To summarize the algorithm, driver data is parsed using a decision tree-like method. The first level utilizes the driver state and the second utilizes the position of the car on the road. The final level is dictated by which cluster the observed surroundings belong to. By separating the training data in this manner, we then build predictive sets using the future observed trajectories at each instance of this mode, effectively linking the dataset to the current point in time. From these sets, we can calculate an empirical probability distribution for future trajectories and inputs. In essence, this model answers the question: given the context and the driver's mental state, what has the driver done in the past?

III. Experiments

In this section, we describe the simulator and experimental setup. A challenging component of studying human-in-the-loop systems is collecting data, especially when safety is of concern. To address this, we have developed an experimental setup for studying human-in-the-loop systems in vehicles, tailored toward driving applications. The testbed was designed to recreate the feeling of moving in a vehicle and is equipped with monitoring devices to observe the human.

Driver data was collected using a Force Dynamics CR401, a 4-axis motion platform simulator, which recreates the forces experienced while driving [3]. This system has been integrated with PreScan software, which provides vehicle dynamics and customizable driving environments [5]. In addition, this testbed is equipped with driver monitoring devices to sense and observe the driver state [13]. Using this human-in-the-loop testbed, we are able to reliably and

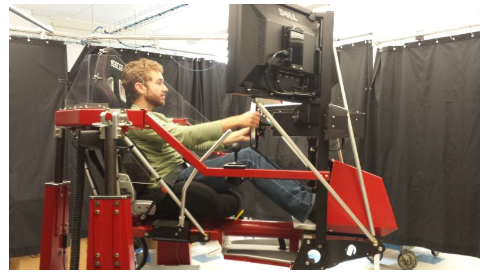


Fig. 3: Picture of test subject driving the simulator.

Fig. 4: PreScan simulation environment from the driver's perspective.

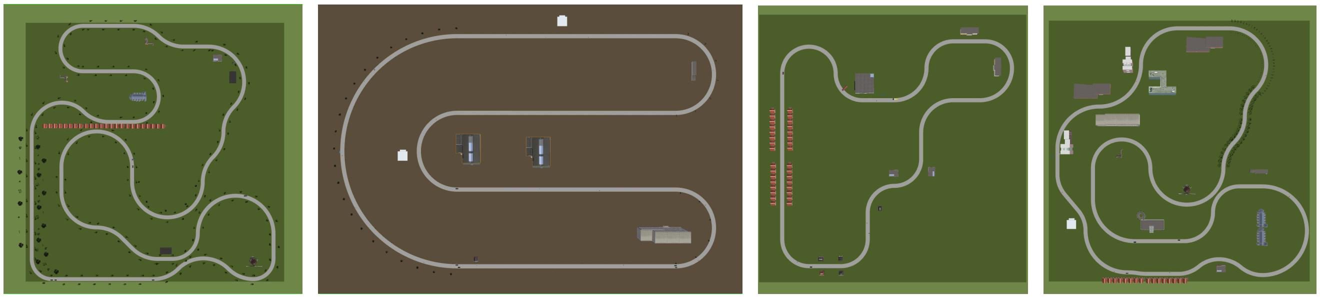
realistically obtain driver data that can illustrate the utility of our models and provide useful motion feedback to the drivers. Evidence of the utility of motion simulators are can be found in [18], [21]. The simulator and visualization seen by the driver is shown in Fig. 3 and Fig. 4, respectively.

Thirteen subjects between the ages of 21 and 36 were recruited to test this prediction algorithm for highway and intersection scenarios. The intersection model is presented in Section V. For the highway scenario, subjects were asked to drive on four courses each lasting twenty minutes to generate three training sets and one test set. These courses consisted of two lane roads with turns of various curvatures, with different levels of traffic that moved independently of the ego vehicle with no opposing traffic. On these courses, the drivers faced a number of obstacles, some that were static (e.g. trailers in the road, cardboard boxes, etc.) and others that were moving (e.g. balls rolling in the road, other vehicles, etc.). The driver was asked to drive as they would normally at about 50 mph. The final test course consisted of obstacles and road patterns that had not been experienced in the training set, to verify the flexibility of the model. The top views of the courses are illustrated in Fig. 5.

To simulate distraction, the driver was given an android phone with a custom application to randomly ping the driver to respond to a text message 30-60s after the driver responded to the previous text. The application also recorded phone acceleration and touch to determine when the driver state in real time [13]. A few example questions are provided below:

1. What did you have for lunch today?
2. What is your major?
3. Where are the Olympics this year?

To determine the driver state, we assume that the driver is attentive when there is no distraction, partially distracted after the phone rings and she considers answering, and fully distracted when she is physically typing on the phone.



(a) Training Course 1 (b) Training Course 2 (c) Training Course 3 (d) Test Course

Fig. 5: Training and test courses used to build the data sets.

IV. Evaluation

In this section, we describe the metrics used to evaluate the model and results on the above experiments.

A. Model Metrics

Before presenting the results, the evaluation techniques will be briefly described. Since we are considering a set prediction for the driver model, there is a trade-off between the precision and the accuracy. To clarify, if we were only interested in a model with high accuracy, then the reachable set of the car would suffice. However, this does not provide us any useful information into what the driver is likely to do. Therefore, we define accuracy as the number of samples that fall within the boundaries of the set (Eq. 3) and precision as the area of the set relative to the size of the reachable set (Eq. 4). More information about these metrics can be found in [27]. These metrics are formally defined as follows:

$$A = \frac{1}{M}\sum_{i=1}^{M}\prod_{k=0}^{N}\mathbb{1}\left\{(x_i(k) - x_i(0)) \in \Delta_k(\mathcal{O}, \mathcal{I}, \alpha)\right\} \quad (3)$$

where M is the number of observed trajectories, $x_{i,N}(k)$ is the state of vehicle of the i^{th} trajectory at time k, and $\mathbb{1}$ is the indicator function.

$$P = \max\left\{1 - \frac{1}{M}\sum_{i=1}^{M}\frac{\left|\cup_{k=0}^{N}\Delta_k(\mathcal{O}, \mathcal{I}, \alpha)\right|}{\left|\cup_{k=0}^{N}\mathcal{R}(k, \mathcal{I})\right|}, 0\right\} \quad (4)$$

where $\mathcal{R}(k, \mathcal{I})$ is the reachable set beginning at $x_i(0)$, for a constant velocity. For a given mode, the median velocity of the associated set of observed trajectories is used. Since we consider the reachable set for a constant velocity (meaning there is no throttle input), it is possible that the predicted set is larger than the reachable set used for comparison. The precision is set to zero if this occurs, the prediction is deemed less useful than the standard reachable set.

B. Highway Experiments

We evaluate this metric at various time horizons $T = \{0.5, 1.0, 1.2, 1.5, 2.0\}$ seconds. For comparison, we compare these results to the reachable set (denoted RS) of the vehicle, which is always accurate, but lacks precision. The accuracy and precision metrics versus the number of clusters are shown in Tables I and II, respectively. These results are the averaged metrics for each of the individualized models.

From these results, a few key observations can be made. As expected, the accuracy decreases over time. This can partially be explained by the uncertainty of the environment, which can drastically change over the course of two seconds. Therefore, this model is best used at a time horizon of 1 to 1.5s. Also note that the accuracy decreases as a function of the number of clusters. Here, we note the trade off between the precision, which improves with an increase in clusters. This intuitively makes sense, as we finely separate the data, we expect smaller set predictions, sacrificing the accuracy of the model. When the environment is ignored, meaning only a single cluster is used, precision is completely lost. This implies that the environment must be taken into consideration for useful prediction.

It can be seen that for a time horizon of 0.5s, the driver model has zero precision. This implies that when considering a short time horizon, the directly using the reachable set is justified. However, when predicting over long time horizons,

TABLE I: Driver Model Accuracy Results, where k is the number of clusters used and RS denotes the reachable set.

Method	**0.5 s**	**1.0 s**	**1.2 s**	**1.5 s**	**2.0 s**
RS	1.000	1.000	1.000	1.000	1.000
$k = 1$	0.998	0.998	0.998	0.998	0.967
$k = 5$	0.993	0.992	0.992	0.991	0.894
$k = 10$	0.985	0.983	0.982	0.981	0.818
$k = 15$	0.961	0.959	0.958	0.956	0.757
$k = 20$	0.955	0.952	0.950	0.948	0.721
$k = 25$	0.925	0.922	0.920	0.916	0.676
$k = 30$	0.911	0.908	0.906	0.902	0.642

TABLE II: Driver Model Precision Results, where k is the number of clusters used and RS denotes the reachable set.

Method	**0.5 s**	**1.0 s**	**1.2 s**	**1.5 s**	**2 s**
RS	0.000	0.000	0.000	0.000	0.000
$k = 1$	0.000	0.000	0.000	0.000	0.000
$k = 5$	0.000	0.000	0.087	0.244	0.337
$k = 10$	0.000	0.182	0.336	0.457	0.528
$k = 15$	0.000	0.313	0.447	0.551	0.613
$k = 20$	0.000	0.400	0.521	0.614	0.671
$k = 25$	0.000	0.458	0.570	0.657	0.709
$k = 30$	0.000	0.488	0.596	0.679	0.729

the model is becomes more useful than the alternative. The trade-off between accuracy and precision is visualized in Fig. 6, which shows both metrics versus the number of environment clusters used.

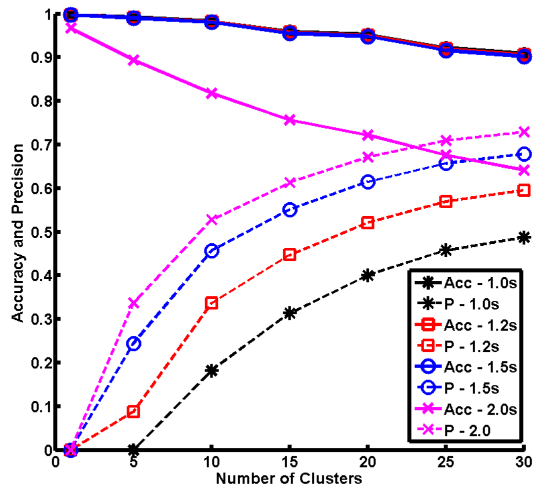
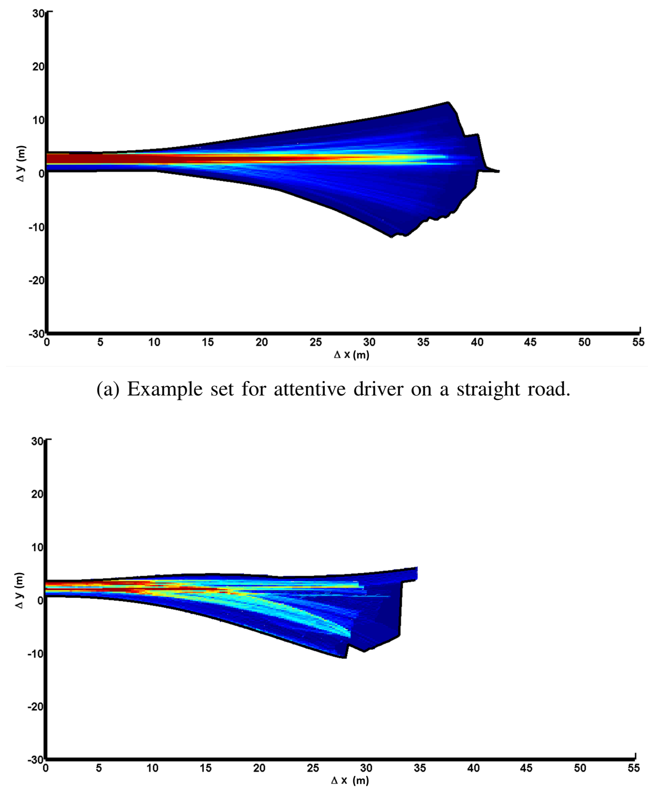


Fig. 6: Plot illustrating the trade-off between accuracy and precision as the number of clusters increase, when T = 1.0, 1.2, 1.5, and 2.0s.

Once these sets are created, the empirical probability distributions can be derived to show the likelihood of the trajectory sets over time. Example sets and their distributions are visualized in Figure 7.



(a) Example set for attentive driver on a straight road.

(b) Example set for distracted driver on a slightly curved road.

Fig. 7: Visualization of prediction sets for an individual driver ($k = 20$), where the probability distribution is plotted over Δx and Δy, representing the longitudinal and latitudinal change in position in meters. The dark red regions represent areas of high probability and darker blue regions represent low probability.

V. Extension to Intersections

In this section, we describe an extension to the model on city driving with intersections. At intersections, we assume knowledge of the driver intention to turn or drive straight, similar to if the driver where to obey navigation commands from a Global Position System navigation device.

A. Model Extension Formulation

The model proposed in Section II-B easily extends to intersection with some slight modifications to the context detection level. In the context detection level of the decision tree, we detect if the driver is driving on a straightaway or approaching an intersection, which is defined as being within a distance r of the intersection. In the model presented here, we set $r = 60m$. This is then conditioned on the status of the intersection (e.g. stop sign, traffic light), which would influence the behavior. We include the status of the traffic light for the previous two seconds in the feature set. The model considers the following four scenarios: (1) approaching a stop sign with the intent of continuing straight; (2) approaching a stop sign and turning right; (3) approaching a traffic light; and (4) city driving.

Since the time scale at which events occur in city driving, this model is formulated to test for a 5 second prediction, tested in 1 second increments. Generally, accurate and precise predictions over such long time horizons are extremely difficult if not impossible due to the potential changes in the environment. However, this long time horizon allows us to analyze the entire execution of a maneuver that might occur at an intersection.

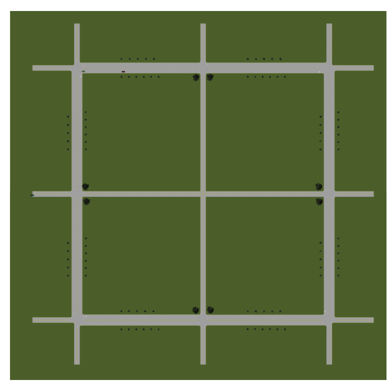


Fig. 8: Course used to build the intersection model.

B. Experimental Setup

The course for this extension is shown in Fig. 8, where the training and testing data was completed in the same road configuration, with different traffic flows, obstacles, and environmental distractions. The driver was asked to navigate

TABLE III: Intersection Model Accuracy Results, where k is the number of clusters used and RS stands for the reachable set.

Method	1 s	2 s	3 s	4 s	5 s
RS	1.000	1.000	1.000	1.000	1.000
$k = 1$	0.999	0.998	0.997	0.997	0.967
$k = 5$	0.993	0.990	0.986	0.984	0.912
$k = 10$	0.986	0.981	0.976	0.973	0.874
$k = 15$	0.979	0.972	0.966	0.961	0.835
$k = 20$	0.974	0.965	0.959	0.952	0.797
$k = 25$	0.969	0.958	0.951	0.944	0.772

TABLE IV: Intersection Model Precision Results, where k is the number of clusters used and RS stands for the reachable set.

Method	1 s	2 s	3 s	4 s	5 s
RS	0.000	0.000	0.000	0.000	0.000
$k = 1$	0.868	0.894	0.844	0.704	0.00
$k = 5$	0.865	0.858	0.855	0.833	0.00
$k = 10$	0.898	0.899	0.898	0.883	0.529
$k = 15$	0.933	0.930	0.927	0.912	0.382
$k = 20$	0.948	0.948	0.944	0.927	0.470
$k = 25$	0.966	0.967	0.9607	0.941	0.531

the loop of intersections by driving as they would normally at about 25 mph while stopping at stop signs and at traffic lights, as one would in city driving for half an hour. It was noted that the drivers generally respond to the texts while stopped, so the driver state was excluded in this section.

C. Results

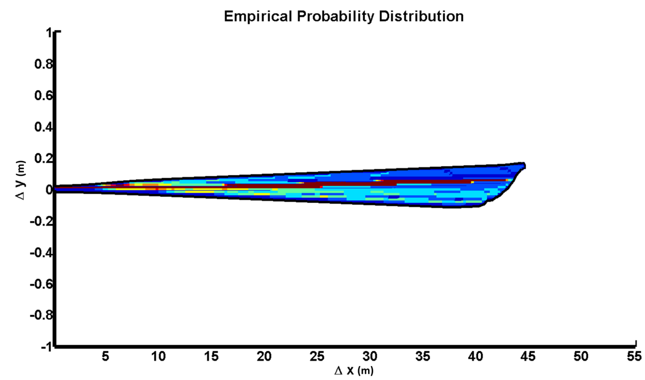
We present the evaluation of this model using the metrics in Section IV-A. The accuracy and precision results are also shown in Table III and IV, respectively. A visualization of the predictions sets for intersections are shown in Figure 9.

From these metrics, a similar relationship between precision and accuracy can be observed as before. The accuracy for this model is quite high. This was expected from the data, as it was observed driver behaviors are more consistent and spontaneous behavior is less likely to occur (e.g. a driver will change lanes with very low probability when approaching an intersection, in our dataset).

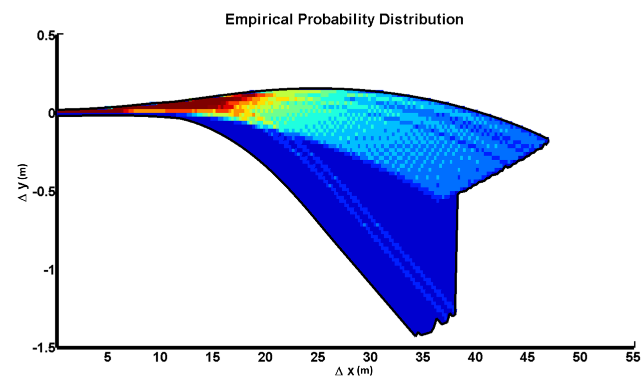
This formulation also allows us to closely examine the control aspect of the human when approaching the intersection. Using the driver's inputs to the vehicle, we can build an empirical probability distribution similar the trajectory sets previously presented to consider driver behaviors in terms of control. An example of this is illustrated in Fig. 10. Because we are directly observing the human behavior and creating the boundaries based of these observations, the predicted sets are not necessary smooth, however realistic and accurate.

VI. Applying the Model

As was discussed in Section I, accurate and precise models of human behavior are crucial for human-in-the-loop systems for developing provably safe control mechanisms or giving feedback to the driver. This model is able to identify the likely set of actions, which can be thought of a highly probably reachable set. This set formulation also allows us to examine the varying behaviors of people depending on the



(a) Example set when approaching a stop sign.



(b) Example set when turning.

Fig. 9: Prediction set for intersection model, where $r = 50m$ and $k = 5$. The probability distribution is plotted over Δx and Δy, representing the longitudinal and latitudinal change in position in meters. Dark red areas represent high probability and dark blue represents low probability.

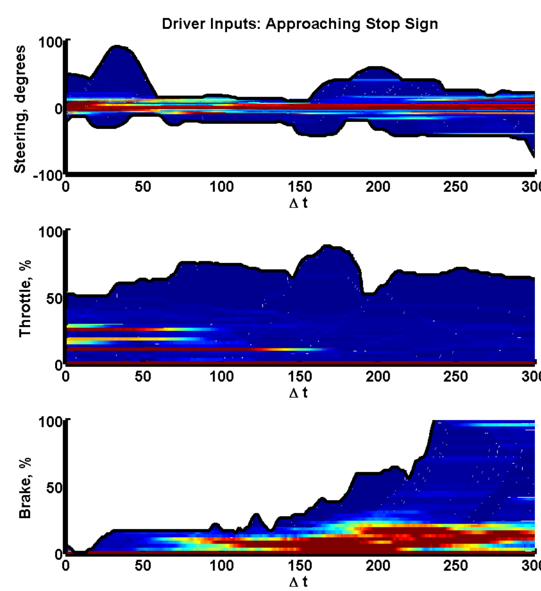


Fig. 10: An example set of driver's inputs when approaching a stop sign, associated with the trajectory set in Fig. 9a.

context in a quantitative manner. Using this empirical model, we can quantify the likelihood of "good" driving behavior, as was shown in [25]. This is valuable as the driver would be able to receive useful feedback on their regular driving behaviors and can be used to develop a provably correct controller.

In addition, this model can be incorporated in a semi-autonomous framework. There are two main control frameworks in which this model will easily integrate: switched and augmented control. Consider the following vehicle dynamics:

$$x(k+1) = f(x(k), u(k)), \quad \forall k \in \mathbb{N} \tag{5}$$

where $x(k) \in \mathbb{R}^n$ is the state at time k, $u(k) \in U$ is the input to the vehicle at time k where $U \subset \mathbb{R}^m$ is a compact, connected set containing the origin, and the initial state of the car, $x(0)$, is given.

Suppose that given $\mathcal{I}$, the unsafe regions of the environment can be estimated, denoted as $\mathcal{C}$. It is assumed that for a given fixed time horizon, $N \in \mathbb{N}$, and a given cost function, there exists an optimal control algorithm that is able to keep the vehicle outside the unsafe set. This assumption can be satisfied by model predictive control (MPC) [10].

Ideally, the optimal semiautonomous system would be minimally invasive. Using this model, determining when the system should intervene can be calculated using the following probabilistic intervention function, denoted G:

$$G(\alpha, \tau, \mathcal{O}, \mathcal{I}) = \begin{cases} 1 & \text{if } \exists k \text{ s.t. } P\left[\Delta_k(\mathcal{O}, \mathcal{I}, \alpha) \cap \mathcal{C}_k(\mathcal{I})\right] \geq \tau \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $k \in \{0, \ldots, N\}$ represents the time step in the time horizon N, $\Delta(k, \alpha, \mathcal{O}, \mathcal{I})$ refers to the probability distribution on the set of α probable trajectories at time k as defined in Eq. 2, $\mathcal{C}(k, \mathcal{I})$ is the unsafe set given the current information $\mathcal{I}$, and $\tau \in [0, 1]$ is a predefined threat threshold. This means that the vehicle intervention function is 1 if the probability of the α probable trajectory set intersecting with an obstacle at any time k is greater than the threshold τ, indicating that the driver is unsafe.

This framework allows for the uncertainty of modeling and prediction to be incorporated in the threat assessment of the driver in a particular situation. By using the intervention functions, a decision can be made by the semiautonomous system as to whether or not control should be applied. Assuming the prediction is accurate and has good precision, the system will intervene only when necessary leading to fewer interventions than a simpler method using the reachable set of the vehicle.

Switched Control: The most obvious method of semi-autonomous intervention is switched control. By this, it is meant that if the system detects danger for the driver, complete control will be taken from the driver, operating under the assumption that the system can outperform the driver. In the framework presented here, the controller would take over whenever the intervention function was set to 1.

There are a number of issues that arise from this method. The most prominent issue is dealing with handing control back to the human. This is an interesting engineering question, but also has some implications for human factors and psychology, concerning how a driver will react to the intervention and determining when it safe to hand control back to the human.

Augmented Control: Instead of completely relieving the human of his duties as driver, augmented control adds the minimum amount of control to the driver's input to keep the driver safe. The augmented control is always on, removing the need to switch between autonomous and human control. The simulation described in the experimental setup also tested an augmenting control strategy using MPC. The controller algorithm runs in real-time to minimize a quadratic cost function as well as the following minimization problem:

$$\begin{array}{ll} \text{minimize} & \delta u^2 \\ \text{subject to} & G(\alpha, \tau, \mathcal{O}, \mathcal{I}) \leq 0 \\ & x(k+1) = f(x(k), u_{dm}(k) + \delta u(k)), \\ & \qquad \forall k = \{0, \ldots, N\} \end{array} \tag{7}$$

where all variables are as before and u_{dm} is given by the driver model. This minimization problem adds the minimal input needed to keep the driver safe. This method has been implemented in a real-time framework and has shown, promising and successful results [27].

VII. Conclusion

The contributions of this paper extend previous work by developing a realistic testbed for data collection, and increasing the utility and accuracy of this driver modeling method. In addition, we relax some of the assumptions on the underlying model of the vehicle, by including all inputs (steering, throttle, and braking) into the model. This creates more variability in the generated prediction sets, as we consider nonlinear behaviors of the human. Regardless, this implementation of the model exhibits comparable precision and significantly improved accuracy. The accuracy of the previous model ranged from 79.2% to 92.0% at 1.2 second time horizon.

By developing this testbed and this extended algorithm, we are able to collect realistic driving data and accurately predict driver behavior. This experimental setup is unique in that it allows us to collect data for and test human-in-the-loop systems, while maintaining safety measures and control of the environmental surroundings. This aids in creating a robust system as we can push the data collection to the search out corner cases or infrequent events that often arise in driving scenarios. By creating a flexible, context aware system, the identification is limited to regions that it has seen before yet is flexible enough to handle variances in scenarios. As was shown in [27], this formulation can be used in a semi-autonomous framework that is able to robustly respond to uncertain human behaviors.

By using these realistic data and flexible algorithm, a precise and accurate driver model can be developed that is tailored to an individual and usable in semi-autonomous frameworks and in driver behavior analysis. Future works include adding more contexts, like night-time driving, poor weather conditions, icy roads, levels of traffic, etc.; examining different distractions and the resulting variation in behaviors; and testing various control methods while the human is driving to verify that the system is minimally invasive and maintains appropriate safety margins. In particular, implementing and identifying parameters for the probabilistic control framework will be explored to verify feasibility and reliability. We will also consider use in a real vehicle, through new, more realistic experiments and by examining the relationship between driving behaviors in a simulator and in an actual vehicle with respect to this model, as has been studied here [23].

References

[1] Traffic safety facts, research note. driver electronic device use in 2011. `http://www-nrd.nhtsa.dot.gov/Pubs/811719.pdf`, 2011. Accessed: 2014-09-30.

[2] GM studying operator behavior in self-driving vehicles: Staying aware considered key to autonomous vehicle operation. `http://media.gm.com/media/us/en/gm/news.detail.html/content/Pages/news/us/en/2012/Jun/0620_humanfactors.html`, 2012. Accessed: 2014-09-30.

[3] Force Dynamics. `http://www.force-dynamics.com`, 2014. Accessed: 2014-09-30.

[4] Mobileye. `http://www.mobileye.com`, 2014. Accessed: 2014-09-30.

[5] PreScan Simulation Software. `https://www.tassinternational.com/prescan`, 2014. Accessed: 2014-09-30.

[6] Rethink robotics: Safety and compliance. `http://www.rethinkrobotics.com/safety-compliance/`, 2014. Accessed: 2014-09-27.

[7] D. Basacik and A. Stevens. Scoping Study of Driver Distraction. *Transport Research Laboratory. Road Safety Research Report*, 95, Feb 2008.

[8] L. M. Bergasa, J. Nuevo, M. A. Sotelo, R. Barea, and M. E. Lopez. Real-Time System for Monitoring Driver Vigilance. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):63–77, Mar. 2006.

[9] H. Berndt, S. Wender, and K. Dietmayer. Driver braking behavior during intersection approaches and implications for warning strategies for driver assistant systems. In *IEEE Intelligent Vehicles Symposium*, pages 245–251, June 2007.

[10] F. Borrelli, M. Baotic, A. Bemporad, and M. Morari. Efficient online computation of constrained optimal control. In *Proceedings of the 40th IEEE Conference on Decision and Control*, volume 2, pages 1187–1192 vol.2, 2001.

[11] E. Coelingh, L. Jakobsson, H. Lind, and M. Lindman. Collision Warning With Auto Brake - A Real-Life Safety Perspective, Apr. 2007.

[12] A. Doshi, B. T. Morris, and M. M. Trivedi. On-Road Prediction of Driver's Intent with Multimodal Sensory Cues. *IEEE Pervasive Computing*, 10(3):22 – 34, Sept. 2011.

[13] K. Driggs-Campbell, G. Bellegarda, V. Shia, S. S. Sastry, and R. Bajcsy. Experimental design for human-in-the-loop driving simulations. *CoRR*, abs/1401.5039, 2014.

[14] L. Fletcher, N. Apostoloff, L. Petersson, and A. Zelinsky. Vision In and Out of Vehicles. *IEEE Intelligent Systems*, pages 12 – 17, June 2003.

[15] J. H. Gillula and C. Tomlin. Guaranteed safe online learning via reachability: tracking a ground target using a quadrotor. In *2012 IEEE International Conference on Robotics and Automation*, pages 2723–2730, May 2012.

[16] N. Gordon-Bloomfield. Nissan changes expectations, timeline for autonomous drive technology. `https://transportevolved.com/2014/07/17/nissan-changes-expectations-timeline-autonomous-drive-technology/`, 2014. Accessed: 2014-09-30.

[17] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, 18(1):100–108, 1979.

[18] K. Hulme, E. Kasprzak, K. English, D. Moore-Russo, and K. Lewis. Experiential learning in vehicle dynamics education via motion simulation and interactive gaming. *International Journal of Computer Games Technology*, 2009.

[19] B. Hyun, C. Park, W. Wang, and A. Girard. Discrete event modeling of heterogeneous human operator team in classification task. In *American Control Conference*, pages 2384–2389, June 2010.

[20] M. Jabon, J. Bailenson, E. Pontikakis, L. Takayama, and C. Nass. Facial Expression Analysis for Predicting Unsafe Driving Behavior. *IEEE Pervasive Computing*, 10(4):84 – 95, Apr. 2011.

[21] R. S. Jacobs and S. N. Roscoe. Simulator cockpit motion and the transfer of initial flight training. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 19(2):218–226, 1975.

[22] L. Jin, Q. Niu, H. Hou, H. Xian, Y. Wang, and D. Shi. Driver cognitive distraction detection using driving performance measures. *Discrete Dynamics in Nature and Society*, 2012.

[23] N. A. Kaptein, J. Theeuwes, and R. V. D. Horst. Driving simulator validity: Some considerations. *Transportation Research Record: Journal of the Transportation Research Board*, 1550:30–36, January 1996.

[24] M. Regan, J. D. Lee, and K. L. Young. *Driver Distraction: Theory, Effects, and Mitigation*. CRC, 2008.

[25] D. Sadigh, K. Driggs Campbell, A. A. A. Puggelli, W. Li, V. Shia, R. Bajcsy, A. L. Sangiovanni-Vincentelli, S. S. Sastry, and S. A. Seshia. Data-driven probabilistic modeling and verification of human driver behavior. Technical Report UCB/EECS-2013-197, EECS Department, University of California, Berkeley, Dec 2013.

[26] A. Sathyanarayana, S. Nageswaren, H. Ghasemzadeh, R. Jafari, and J. H. L. Hansen. Body Sensor Networks for Driver Distraction Identification. In *IEEE International Conference on Vehicular Electronics and Safety*, pages 120 – 125. IEEE, Sept. 2008.

[27] V. Shia, Y. Gao, R. Vasudevan, K. Campbell, T. Lin, F. Borrelli, and R. Bajcsy. Semiautonomous vehicular control using driver modeling. *IEEE Transactions on Intelligent Transportation Systems*, PP(99):1–14, 2014.

[28] B. W. Smith. Human error as a cause of vehicle crashes. `http://cyberlaw.stanford.edu/blog/2013/12/human-error-cause-vehicle-crashes`, 2013. Accessed: 2014-09-30.

[29] R. Vasudevan, V. Shia, Y. Gao, R. Cervera-Navarro, R. Bajcsy, and F. Borrelli. Safe semi-autonomous control with enhanced driver modeling. In *American Control Conference*, pages 2896–2903, 2012.

[30] E. Wahlstrom, O. Masoud, and N. Papanikolopoulos. Vision-Based Methods for Driver Monitoring. In *Intelligent Transportation Systems*, volume 2, pages 903 – 908. IEEE, Oct. 2003.