2019 American Control Conference (ACC)
Philadelphia, PA, USA, July 10-12, 2019

Stochastic Driver Modeling and Validation with Traffic Data

Mert Albaba[1], Yildiray Yildiz[2], Nan Li[3], Ilya Kolmanovsky[3] and Anouck Girard[3]

Abstract— **This paper describes a stochastic modeling approach for predicting driver responses in highway traffic. Different from existing approaches in the literature, the proposed modeling framework allows *simultaneous* decision making for multiple drivers (>100), in a computationally feasible manner, instead of modeling the decisions of an ego driver and assuming a predetermined driving pattern for other drivers in a given scenario. This is achieved by a unique combination of hierarchical game theory, which is used to model strategic decision making, and stochastic reinforcement learning, which is employed to model multi-move decision making. The proposed approach can be utilized to create high fidelity traffic simulators, which can be used to facilitate the validation of autonomous driving control algorithms by providing a safe and relatively fast environment for initial assessment and tuning. What makes the proposed approach appealing especially for autonomous driving research is that the driver models are strategic, meaning that their responses are based on predicted actions of other intelligent agents in the traffic scenario, where these agents can be human drivers or autonomous vehicles. Therefore, these models can be used to create traffic models with multiple human-machine interactions. To evaluate the fidelity of the framework, created stochastic driver models are compared with real driving patterns, processed from the traffic data collected by US Federal Highway Administration on US101 (Hollywood Freeway) on June 15th, 2005.**

I. Introduction

There are several legal, technical and monetary challenges that the autonomous driving car manufacturers have to overcome before we can see a wide-spread usage of these vehicles on roads. Two of these challenges are developing control algorithms in an environment where hard-to-predict agents, namely humans, also drive; and being able to validate the safety of these vehicles for this uncertain environment. In this paper, a stochastic driver modeling approach is proposed, which can be utilized to address these two issues: If high fidelity human driver models are obtained, then these models can be employed to create traffic simulators to be used for initial evaluation of autonomous driving control algorithms. These simulators can help speed up the validation process, where it is estimated that millions of miles of driving tests are required [1] for autonomous cars to obtain similar levels of safety guarantees as vehicles with human drivers. Furthermore, driver models can be beneficial for developing autonomous driving control algorithms that provide human-like driving experience, which may be useful for operating in traffic with other human drivers.

There are several successful approaches in the literature in human driver modeling. In [2], a framework named "cognitive architecture method", specifying human cognition's behavioral models, is used. Hidden Markov Models (HMMs) are used in [3] and [4], and Markov Dynamic Models (MDMs) are employed in [5], to predict driver responses. In [6] and [7], k-means clustering is used for predicting trajectories of vehicles with drivers. A method named SITRAS (Simulation of Intelligent Transport Systems) is used in [8] for modeling lane changing actions of drivers. In [9], Dynamic Bayesian Networks are used to model the drivers, and in [10] and [11] Gaussian Process Regression is used to detect patterns and predict driver behaviors. For predicting lane change intent, Support Vector Machine (SVM) method is combined with Bayesian filter in [12]. In [13], to predict driver behaviors on intersections, Partially Observable Markov Decision Process (POMDP) is used along with Bayesian Network. COSMODRIVE (combination of artificial intelligence and cognitive psychology methods) is used in [14] to model drivers. Moreover, the same method, COSMODRIVE, is used to predict driver behaviors on intersections in [15]. In [16], Adaptive Predictive Control (APC) is utilized for driver modeling. Inverse reinforcement learning is used in [17] to obtain driving styles of human drivers from their trajectories. In [18], an autonomous intelligent cruise control system (AICC) is proposed to model car following behaviors of human drivers.

A distinguishing feature of the proposed driver modeling framework is the ability to model simultaneous decision making of multiple intelligent agents (>100), in a computationally feasible way. This is achieved thanks to a unique combination of two modeling approaches: Cognitive Hierarchy Theory (CHT) [19], [20], [21], and stochastic reinforcement learning for Partially Observable Markov Decision Process (POMDP) [22], which is inspired from semi net-form games [23]. CHT assumes various levels of reasoning for each driver in a given traffic scenario and helps determine the strategic decisions of drivers. Here, "strategic decision" refers to a decision process where predicted actions of other agents are taken into account. Stochastic reinforcement learning, on the other hand, helps determine driver actions in time-extended scenarios, where drivers consider best action sequences, instead of a single best action.

[1]Mert Albaba is with Faculty of Electrical and Electronics Engineering, Bilkent University, 06800 Bilkent, Ankara, Turkey mert.albaba@ug.bilkent.edu.tr

[2]Yildiray Yildiz with the Department of Mechanical Engineering, Bilkent University, 06800 Bilkent, Ankara, Turkey yyildiz@bilkent.edu.tr

[3]Nan Li, Ilya Kolmanovsky and Anouck Girard are with the Department of Aerospace Engineering, University of Michigan, Francois-Xavier Bagnoud Aerospace Building, 1320 Beal Avenue, Ann Arbor, MI 48109-2140 nanli@umich.edu, ilya@umich.edu, anouck@umich.edu

The authors have published similar highway driver models in [24] and [25]. The contributions of this study compared to these prior works are that 1) the proposed driver models are compared with real driver responses obtained by processing traffic data provided in [21], 2) unlike earlier results, where a 3-lane traffic was considered, a 5-lane road is modeled, which allows a larger variety of traffic test scenarios.

The organization of the paper is: in Section II, modelling approach, which consists Cognitive Hierarchy Theory (CHT) and Reinforcement Learning, is presented; in Section III, traffic scenario elements consisting drivers' action and observation spaces, vehicle motion model and level-0 policy are defined; in section IV, training results are presented along with training performances; in Section V, model and data comparison results are provided and in Section VI, a summary is given.

II. Modeling Approach

The approach exploited in this paper consists of two main components: One of the components is called Cognitive Hierarchy Theory (CHT) [19], [20], [21], allows for strategic decision making, while allowing incorrect assumptions about other drivers, and hence permitting less-than-optimal human behavior. The other component, stochastic reinforcement learning [23], enables these decisions to be taken in a time extended manner, instead of a single-shot decision, which is crucial for traffic scenarios.

A. Decision Tools: Cognitive Hierarchy Theory (CHT) and Reinforcement Learning

In CHT, it is assumed that humans have various levels of reasoning. The lowest level agent, level-0, acts without any regard to other agents' possible actions, and therefore called as a non-strategic decision maker. All other levels, level-k agents, assume that the rest of the players are level-(k-1) and produce best responses based on this assumption.

Reinforcement learning is learning through trial-and-error actions with a goal of maximizing a cumulative reward over the period of learning [26]. At the end of the learning process, we obtain a policy, a mapping from states to actions, such that when the agent implements this policy, the goal of maximizing the cumulative reward is achieved. In this paper, a reinforcement learning method that produces stochastic policies is used [23]. In this method, the action-value function, which defines the value of choosing a certain action a, given a state s, is determined as given below:

$$\beta_t(m,a) = \left(1 - \frac{X_t(m,a)}{K_t(m,a)}\right)\gamma_t\beta_{t-1}(m,a) + \frac{X_t(m,a)}{K_t(m,a)} \quad (1)$$

$$Q_t(m,a) = \left(1 - \frac{X_t(m,a)}{K_t(m,a)}\right)Q_{t-1}(m,a) + \beta_t(m,a)(R_t - R) \quad (2)$$

where m indicates message (observed state), a indicates action and t indicates the time step. Moreover, X_t is the indicator function (takes 1 if message (m)/message-action pair (m,a) is visited and 0 otherwise), and K_t represents the number of times a particular message (m)/message-action pair (m,a) is visited. γ_t is the discount factor, R_t is the reward obtained in time step t, and R is the average reward, which is recursively estimated. Similarly, the value function $V(m)$, which defines the value of being in a certain state is calculated as:

$$\beta_t(m) = \left(1 - \frac{X_t(m)}{K_t(m)}\right)\gamma_t\beta_{t-1}(m) + \frac{X_t(m)}{K_t(m)} \quad (3)$$

$$V_t(m) = \left(1 - \frac{X_t(m)}{K_t(m)}\right)V_{t-1}(m) + \beta_t(m)(R_t - R). \quad (4)$$

Finally, the policy update rule is given as:

$$\pi(a|m) \rightarrow (1-\varepsilon)\pi(a|m) + \varepsilon\pi^1(a|m) \quad (5)$$

where ε is the learning rate and takes values between 0 and 1, and $\pi^1(a|m)$ is defined as the policy which makes $J^{\pi^1} > 0$, where J^{π^1} is defined as:

$$J^{\pi^1} = \max_a [Q^\pi(m,a) - V^\pi(m)]. \quad (6)$$

where Q^π is the action-value function of the policy π and V^π is the value function of the policy π. This policy update creates an incremental change in the average reward as:

$$\Delta R^\pi = \varepsilon \sum_m P^\pi(m) J^{\pi^1}(m) + O(\varepsilon^2) \quad (7)$$

where $P^\pi(m)$ is the occupancy probability of message m in policy π.

B. Combining CHT and Stochastic Reinforcement Learning

To obtain the driver models for a traffic scenario where each driver makes simultaneous decisions in a time-extended manner, CHT and stochastic reinforcement learning explained earlier are employed in parallel: First, a level-0 policy is defined, which is discussed later in the following sections. Then, the level-0 policy is assigned to all of the drivers in the traffic scenario but the ego driver. The policy of the ego driver is obtained using the reinforcement learning method. Since all of the other drivers are level-0 thinkers, the ego driver learns to respond best to the level-0 policy and therefore becomes a level-1 driver. The process continues by assigning all the drivers the level-1 policy and training a policy in this environment, which is then called a level-2 policy. The same procedure is implemented for higher levels. It is important to note that during the training of policies, all the drivers except the trainee (the ego driver) are assigned a policy that is obtained in the previous step, and thus these drivers become part of the environment, which permits using only one instance of the reinforcement learning process to obtain a certain level-k driver. This is the key that makes this approach computationally feasible.

III. Elements of the Traffic Scenario

Traffic scenarios modeled in this paper consist of several vehicles (> 100) in a 5-lane highway (see Fig. 1). Each lane is 3.7 meters wide, and the vehicles are assumed to drive constantly without any specified end point. Vehicle dynamics are continuous. Vehicles sizes are selected as 5m$\times$2m. Below, the elements necessary to create the traffic scenario are explained.

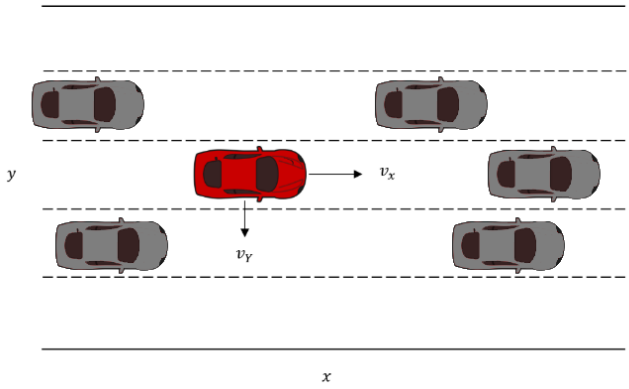


Fig. 1: The ego vehicle (red) and the vehicles observed by the ego driver (grey).

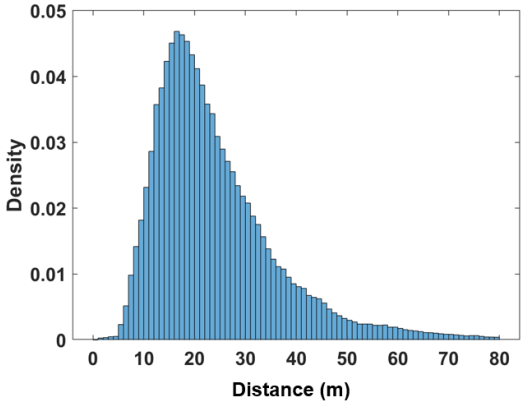


Fig. 2: Distribution of Distances to Car in Front

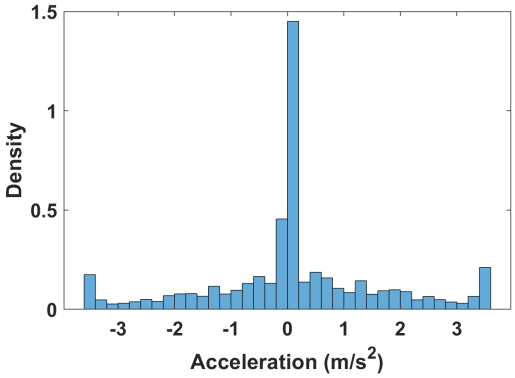


Fig. 3: Acceleration Distribution

A. Driver Observation Space

It is assumed that a driver can observe his/her distance to the cars that are 1) in front of him/her driving in the same lane, 2) the car in the front left lane, 3) the car in the front right lane, 4) the car in the rear left lane, and 5) the car in the rear right lane. However, these distances are not measured exactly but instead perceived as discretized amounts of *close*, *nominal* and *far*. The ranges of these distance amounts are determined by processing the traffic data of US101, Hollywood Freeway provided in [21] and obtaining the distance distribution in the traffic, as shown in Fig. 2. It is estimated that around 50% of the time the distances are in the range 11m - 27m and therefore we defined the distance as *nominal* if it's between 11m and 27m, *close* if it's smaller than 11m and *far* if it's larger than 27m. Furthermore, the driver can observe the changes in distances in three groups: *approaching*, *stable* and *moving away*. As a result, observation space, which defines what a driver can perceive during driving, consists of the following items:

- Distances to the cars in front, front left, front right, rear left and rear right lanes: *close*, *nominal* or *far*,
- Distance changes of the cars in front, front left, front right, rear left and rear right lanes: *approaching*, *stable* or *moving away*

B. Driver Action Space

There are two types of actions: "changing lane" or "changing acceleration". In this paper, it is assumed that drivers could not change their accelerations while changing lanes. There are two actions for lane change: moving to the left lane and moving to the right lane.

To determine the driver actions, the acceleration distribution data is obtained from the traffic data given in [21], which is presented in Fig. 3. This distribution is investigated in five separate parts (see Fig. 4), where continuous distributions are used (red curves) to approximate each of these five sub-distributions. Based on these approximated distributions, we determine the following action space for drivers:

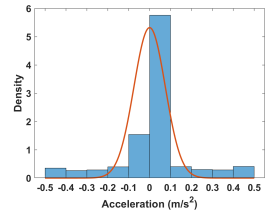
1) *Maintain*, where the acceleration is sampled from normal distribution with zero mean and a standard deviation of 0.075.
2) *Accelerate*, where the acceleration is sampled from a uniform distribution between 0.5 m/s^2 and 2.5 m/s^2.
3) *Decelerate*, where the acceleration is sampled from a uniform distribution between -0.5 m/s^2 and -2.5 m/s^2.
4) *Hard Accelerate*, where the acceleration is sampled from a half-normal distribution with a mean of 3.5 m/s^2 and a standard deviation of 0.3 m/s^2.
5) *Hard Decelerate*, where the acceleration is sampled from a half-normal distribution with a mean of -3.5 m/s^2 and a standard deviation of 0.3 m/s^2.
6) *Move Left*.
7) *Move Right*.

C. Driver Objective Function

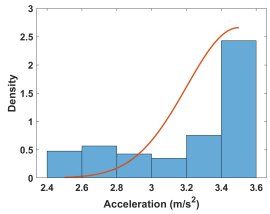
The preferences of drivers, such as spending a minimum amount of effort while keeping a large headway and avoiding crashes are expressed in mathematical form using a reward function, which is given below:

$$R = w_1 * c + w_2 * v + w_3 * d + w_4 * e \tag{8}$$

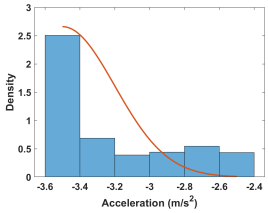
where, w_1, w_2, w_3 and w_4 are weights determining the relative emphasis on different terms, c equals to -1 if a crash occurs and 0 otherwise, v equals to the difference between the speed of the vehicle and the mean speed divided by the maximum



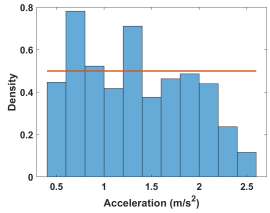
(a) Normal Distribution Fit near Center



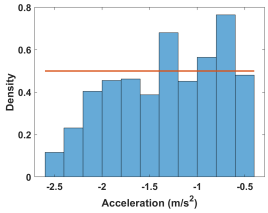
(b) Half-Normal Distribution Fit near Right End



(c) Half-Normal Distribution Fit near Left End



(d) Uniform Distribution Fit near 0.5-2.5



(e) Uniform Distribution Fit near -0.5–2.5

Fig. 4: Continuous approximations, given in red, of five sub-distributions.

(division is made to normalize the variable and fit its value between -0.5 and 0.5) $v = \frac{v(t) - \frac{v_{max}+v_{min}}{2}}{v_{max}}$, d equals to -1 if the distance to the car in front is *close*, 0 if it is *nominal* and 1, otherwise. e takes the values of 0, -0.25 and -0.5, if the selected action is *maintain*, *-accelerate* or *decelerate-*, and *-hard accelerate* or *hard decelerate-*, respectively. e becomes equal to -1 if the action is move left or move right.

The weight values should be selected such that w_1, determining the importance of not having an accident, gets a high enough value to prevent accidents at all costs. Similarly, w_3, penalizing unsafe distances between cars, should have a high enough value for safety. The values of w_2 and w_4, emphasizing the importance of reaching one's destination as quickly as possible and keeping a minimum effort in the traffic, respectively, can be decided based on the driver type to be modeled. In this study, we use weight values such that $w_1 > w_4 > w_3 > w_2$. It is noted the selection of these values can be done using numerical analysis and trade-off studies.

D. Vehicle Placements and Physical Models

During both the training and simulation studies, vehicles are placed randomly within a 600m length of road segment with initial velocities ranging between 5 m/s and 7.5 m/s. The initial distances between the vehicles are always greater than 11m, which is a little larger than the minimum distance required to prevent an accident if the relative velocity between two cars are at its highest allowable value. Vehicle accelerations and directions are modified according to the actions of drivers. As explained in earlier sections, drivers can make acceleration decisions of: *maintain*, *accelerate*, *decelerate*, *hard accelerate* and *hard decelerate*, together with lane changing actions. In this work, it is assumed that cars change lanes with constant velocity. In other words, it is assumed that the acceleration is zero during a lane change.

Cars change their velocities and positions based on the equations given below.

$$x(t+1) = x(t) + v_x(t) * \Delta t + \frac{1}{2} a(t) \Delta t^2 \quad (9)$$

$$y(t+1) = y(t) + v_y(t) * \Delta t \quad (10)$$

$$v_x(t+1) = v_x(t) + a(t) * \Delta t \quad (11)$$

As seen in Fig. 1, in (9)-(11), x and y are the longitudinal and lateral positions, respectively, v_x is the longitudinal velocity and v_y is the lateral velocity. Moreover, a is the acceleration and Δt is the step time.

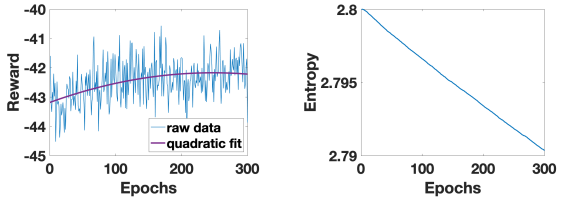
E. Level-0 Model

As explained in previous sections, in Cognitive Hierarchy Theory, policies are developed in a hierarchical manner, and the process begins with defining a non-strategic level-0 policy. In this study, we define level-0 policy as: *hard decelerate* if the car in front is *close* and *approaching*; *decelerate* if the car in front is *close* and *stable* or *nominal* and *approaching*; *accelerate* if the car in front is *nominal* and *moving away* or *far*; *maintain* otherwise.

IV. Policy Training Results

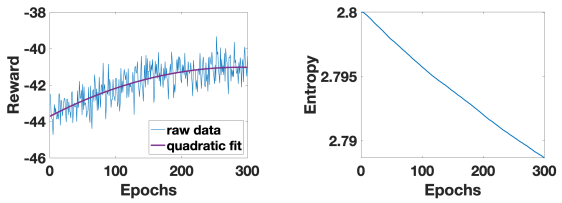
To train each level-k driver policy, first, 75 vehicles are placed on the road. After each 25 epochs, where each epoch corresponds to 100 episodes, 25 more vehicles are added to increase the number of different states visited by the drivers if the number of cars on the road is less than 125. At its maximum, there are 125 drivers interacting with each-other, for a given training instance.

Figures 5, 6 and 7 show the time evolution of the average epoch rewards, together with average epoch entropy of the overall action distributions, during the training of level-1, level-2 and level-3 driver policies, respectively. For an N element discrete probability distribution, entropy is calculated as $-\sum_1^N p_i(log_2 p_i)$ which, in the context of driver modeling, can be interpreted as the degree of randomness in action selection. As seen from the figures, while the average rewards converge relatively faster, the entropies continue to drop, at a much slower rate. The main reason for this result is that there are states in the observation space that are either not visited or visited only a few times. This fact can also be observed from the entropy evolution graphs of the two frequently visited states, given in Fig. 8, where it is seen that the entropies are converging at a much faster rate. For simulation studies, one way to overcome this issue is to assign the level-0 policy for states that are not visited frequently during training.



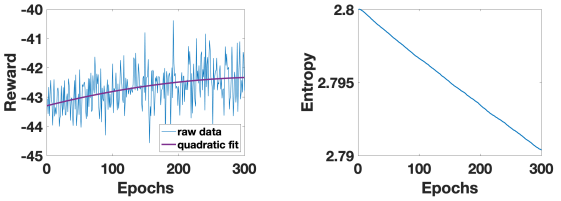
(a) Average Reward per Epoch in Level-1 Training (b) Average Entropy per Epoch in Level-1 Training

Fig. 5: Level-1 Training



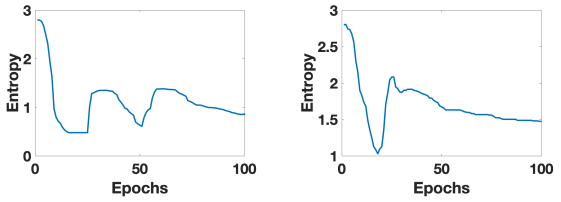
(a) Average Reward per Epoch in Level-2 Training (b) Average Entropy per Epoch in Level-2 Training

Fig. 6: Level-2 Training



(a) Average Reward per Epoch in Level-3 Training (b) Average Entropy per Epoch in Level-3 Training

Fig. 7: Level-3 Training



(a) Entropy per Epoch of a State (b) Entropy per Epoch of a State

Fig. 8: Entropy per Epoch plots of two randomly selected frequently visited states.

V. Model and Data Comparison

In this section, the method of comparison, which is Kolmogorov-Smirnov Test for Discontinious Distributions [27], and the comparison procedure are explained.

A. Kolmogorov-Smirnov Test for Discontinious Distributions

In this paper, the null hypothesis H_0 is that the probability distribution of actions obtained with the game theoretical method, is equal to that of the real data. The distributions $F(x)$ and $H(x)$ used in the Kolmogorov-Smirnov Test are defined as the unknown cumulative action probability distribution functions of a real human driver and the game theoretical driver model, respectively. Then, the null hypothesis can be defined as

$$H_0 : F(x) = H(x) \ for \ all \ x \tag{12}$$

For this test, test statistics presented below need to be calculated.

$$D = sup_x |H(x) - S_n(x)| \tag{13}$$

$$D^- = sup_x (H(x) - S_n(x)) \tag{14}$$

$$D^+ = sup_x (S_n(x) - H(x)) \tag{15}$$

where $S_n(x)$ is the cumulative probability distribution function which is obtained through data samples.

After calculating these test statistics, the "critical level" or $P(D \geq d)$ is calculated with the equation given below.

$$P(D \geq d) = P(D^+ \geq d) + P(D^- \geq d) \tag{16}$$

In this equation, d is the observed value of D and calculation procedures of $P(D^+ \geq d)$ and $P(D^- \geq d)$ can be found in [27], which are omitted here due to space limitations. The critical level can be defined as the percentage of data samples with test statistics larger than or equal to the d when H_0 is true. As this value increases, the probability of the null hypothesis being true increases. The null hypothesis is rejected if this critical level is smaller than a predetermined threshold. This threshold is selected as 0.05 in this work.

B. Comparison Procedure

In this work, for each state, both the model and the data provide a discrete probability distribution of actions. In order to understand whether or not our policies successfully model a human driver for each state visited by the driver, action distribution obtained from data and from the policies are compared.

To obtain meaningful comparison results, any action probability that is less than 0.01 is set to 0.01 and the distribution is renormalized. Furthermore, any state that is visited less than 3 times, either during the training or in the human driving data, is ignored in the comparisons.

C. Results

Fig. 9 shows the percentage of states visited whose action distributions can be modeled by level-k policies for each human driver. We have also created a "dumb policy" which simply provides a uniform probability distribution over all actions, meaning that the driver has no preference for any actions in any state. We then compared the percentages of successfully modeled states for each driver by the game theoretical (GT) policies and the dumb model and the results are provided in Fig. 10. It is seen that although the GT models perform much better than the dumb model, the latter can still model a reasonable percentage of the states. The main reason for this is that although the states that are visited less than 3 times are omitted in the analysis, average number of visits for states is still relatively low, which makes it hard for the Kolmogorov-Smirnov test to fail the dumb policy.

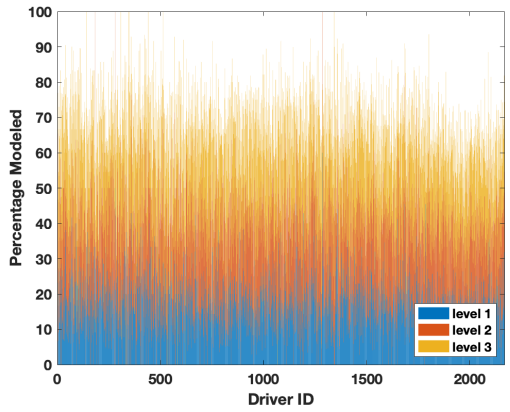


Fig. 9: Percentages of states that are modeled by level-k policies, for each driver.

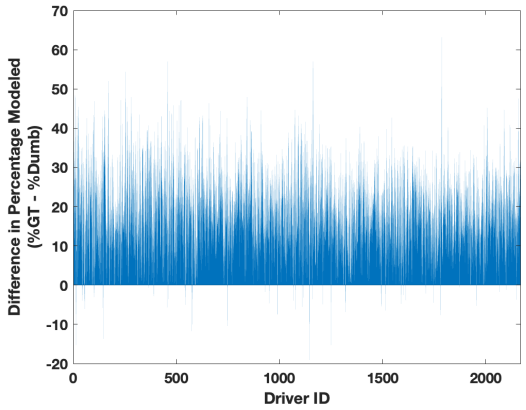


Fig. 10: Differences in modelled percentages between level-k models and the dumb model.

VI. Summary

In this study, a stochastic driver model is presented together with real traffic data comparisons. Hierarchical decision making, and a stochastic reinforcement learning algorithm are utilized in order to predict driver interactions in multiple scenarios. Via Kolmogorov-Smirnov Test for Discontinious Distributions, developed game theoretical models are compared with real human driving data.

References

[1] J. M. Anderson, K. Nidhi, K. D. Stanley, P. Sorensen, C. Samaras, and O. A. Oluwatola, *Autonomous vehicle technology: A guide for policymakers*. Rand Corporation, 2014.

[2] D. D. Salvucci, "Modeling driver behavior in a cognitive architecture," *Human factors*, vol. 48, no. 2, pp. 362–380, 2006.

[3] S. Lefevre, A. Carvalho, and F. Borrelli, "Autonomous car following: A learning-based approach," in *Intelligent Vehicles Symposium (IV), 2015 IEEE*. IEEE, 2015, pp. 920–926.

[4] S. Lefevre, Y. Gao, D. Vasquez, H. E. Tseng, R. Bajcsy, and F. Borrelli, "Lane keeping assistance with learning-based driver model and model predictive control," in *12th International Symposium on Advanced Vehicle Control*, 2013.

[5] A. Liu and D. Salvucci, "Modeling and prediction of human driver behavior," in *Intl. Conference on HCI*, 2001.

[6] 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 (ACC), 2012*. IEEE, 2012, pp. 2896–2903.

[7] V. A. Shia, Y. Gao, R. Vasudevan, K. D. Campbell, T. Lin, F. Borrelli, and R. Bajcsy, "Semiautonomous vehicular control using driver modeling," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 6, pp. 2696–2709, 2014.

[8] P. Hidas, "Modelling lane changing and merging in microscopic traffic simulation," *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 5-6, pp. 351–371, 2002.

[9] R. Kowalczyk, *Agent Technologies, Infrastructures, Tools, and Applications for E-Services: NODe 2002 Agent-Related Workshop, Erfurt, Germany, October 7-10, 2002, Revised Papers*. Springer Science & Business Media, 2003, vol. 2592.

[10] G. S. Aoude, B. D. Luders, J. M. Joseph, N. Roy, and J. P. How, "Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns," *Autonomous Robots*, vol. 35, no. 1, pp. 51–76, 2013.

[11] Q. Tran and J. Firl, "Modelling of traffic situations at urban intersections with probabilistic non-parametric regression," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 334–339.

[12] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 797–802.

[13] T. Gindele, S. Brechtel, and R. Dillmann, "Learning driver behavior models from traffic observations for decision making and planning," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 69–79, 2015.

[14] B. Song and D. Delorme, "Human driver model for smartahs based on cognitive and control approaches," in *ITS America 10th Annual Meeting and Exposition: Revolutionary Thinking, Real ResultsIntelligent Transportation Society of America (ITS America)*, 2000.

[15] Y. Liu and U. Ozguner, "Human driver model and driver decision making for intersection driving," in *Intelligent Vehicles Symposium, 2007 IEEE*. IEEE, 2007, pp. 642–647.

[16] A. Y. Ungoren and H. Peng, "An adaptive lateral preview driver model," *Vehicle system dynamics*, vol. 43, no. 4, pp. 245–259, 2005.

[17] M. Kuderer, S. Gulati, and W. Burgard, "Learning driving styles for autonomous vehicles from demonstration," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2641–2646.

[18] P. A. Ioannou and C.-C. Chien, "Autonomous intelligent cruise control," *IEEE Transactions on Vehicular Technology*, 1993.

[19] C. F. Camerer, T.-H. Ho, and J.-K. Chong, "A cognitive hierarchy model of games," *The Quarterly Journal of Economics*, vol. 119, no. 3, pp. 861–898, 2004.

[20] R. Nagel, "Unraveling in guessing games: An experimental study," *The American Economic Review*, vol. 85, no. 5, pp. 1313–1326, 1995.

[21] U. F. H. Administration, "Us101 dataset." [Online]. Available: https://www.fhwa.dot.gov/publications/research/operations/07030

[22] T. Jaakkola, S. P. Singh, and M. I. Jordan, "Reinforcement learning algorithm for partially observable markov decision problems," in *Advances in neural information processing systems*, 1995, pp. 345–352.

[23] R. Lee and D. Wolpert, "Game theoretic modeling of pilot behavior during mid-air encounters," in *Decision Making with Imperfect Decision Makers*. Springer, 2012, pp. 75–111.

[24] N. Li, D. W. Oyler, M. Zhang, Y. Yildiz, I. Kolmanovsky, and A. R. Girard, "Game theoretic modeling of driver and vehicle interactions for verification and validation of autonomous vehicle control systems," *IEEE Transactions on control systems technology*, 2017.

[25] N. Li, D. Oyler, M. Zhang, Y. Yildiz, A. Girard, and I. Kolmanovsky, "Hierarchical reasoning game theory based approach for evaluation and testing of autonomous vehicle control systems," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 727–733.

[26] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[27] W. J. Conover, "A kolmogorov goodness-of-fit test for discontinuous distributions," *Journal of the American Statistical Association*, vol. 67, no. 339, pp. 591–596, 1972.