Driver Modeling Through Deep Reinforcement Learning and Behavioral Game Theory

Berat Mert Albaba, *Student Member, IEEE*, and Yildiray Yildiz, *Senior Member, IEEE*

***Abstract*— In this work, a synergistic combination of deep reinforcement learning and hierarchical game theory is proposed as a modeling framework for behavioral predictions of drivers in highway driving scenarios. The modeling framework presented in this work can be used in a high-fidelity traffic simulator consisting of multiple human decision-makers. This simulator can reduce the time and effort spent for testing autonomous vehicles by allowing safe and quick assessment of self-driving control algorithms. To demonstrate the fidelity of the proposed modeling framework, game-theoretical driver models are compared with real human driver behavior patterns extracted from two different sets of traffic data.**

***Index Terms*— Autonomous vehicles (AVs), deep learning, driver modeling, game theory (GT), reinforcement learning (RL).**

I. Introduction

SAFETY concerns about autonomous vehicles (AVs) continue to exist, which needs to be addressed for successful integration into daily traffic [1]. In addition to real traffic tests, traffic environments simulated in computers may be used both to accelerate the validation phase and introduce a wide variety of traffic scenarios, which may take several driving hours to encounter [2]–[4]. For reliable simulation results, human driver models should demonstrate human-like driving behavior with reasonable accuracy.

Several approaches are proposed in the literature for modeling human drivers. Markov models in [5]–[7] and support vector machines in [8] and [9] are employed to predict driver actions. Neural networks are also used for this purpose in [10]–[12]. Other tools utilized to model driver actions are dynamic Bayesian networks [13], Gaussian processes [14], [15], and inverse reinforcement learning (RL) [16], [17].

Game-theoretical driver models are also proposed. For example, in [18], a Stackelberg game is used to model highway driving, but dynamic scenarios consisting of several moves are not considered. Stackelberg games are also used in [19], which considers multimove scenarios. However, computations become quite complex once the number of players increases to more than 2. A game-theoretical inverse RL method is proposed in [20] for predicting the interaction between two drivers while assuming a predefined policy for the surrounding vehicles. This approach is also not straightforward for extending to more crowded scenarios, where all the drivers are strategic decision-makers. Deep learning-based driving algorithms are also proposed in [21]–[25]. However, in these studies, the goal is not to obtain human driver models but optimal driving policies to be used in autonomous driving systems. Deep learning and imitation learning-based approaches for modeling human drivers are proposed in [26] and [27], respectively. These methods depend on the training data, which may limit their generalization properties. Control theory-based approaches, such as [28]–[30], are also proposed in the literature for modeling drivers.

This work proposes a deep RL and game theory (GT)-based driver modeling method, which allows simultaneous decision-making for multiagent traffic scenarios. What distinguishes our approach from existing studies is that all the drivers in a multimove scenario make strategic decisions simultaneously, instead of modeling the ego driver as a decision-maker and assuming predetermined actions for the rest of the drivers. This is achieved by combining a hierarchical game-theoretical concept named level-k reasoning [31]–[33] with a deep RL method called deep Q-learning (DQN) [34]. The resulting models have "bounded rationality" since the assumed levels of other players are not always correct. There exist earlier studies that also use RL and GT in modeling driver behavior, such as [2] and [35]–[38]. A tabular RL method is used in these studies, which severely limits the driver observation space. This fact is stated in [2], where it is presented that the main reason behind crashes is the limited observation space. Thus, for the first time, instead of employing a table-based RL method, a deep neural network-based approach, DQN, is used in combination with GT in this work, which not only enabled a dramatically larger observation space but also allowed the introduction of a continuous one, providing infinite resolution to the driver perception. Furthermore, different from similar studies, any possibility of overfitting is eliminated by conducting model-data comparisons using two independent traffic data sets. In this study, a data-based modeling approach is not used. Instead, driver models are derived using the proposed modeling framework, and then, their predictive power is tested using two independent traffic data sets. Finally, the proposed models are compared with the baseline models, IDM [39] and MOBIL [40], and models in the previous work [35]. The contributions of this work over these earlier results, which also use GT and RL, can be listed as follows.

1) It is demonstrated that a dramatically larger class of traffic scenarios, compared to earlier studies, can be successfully modeled.
2) It is shown that the driver crash rates can be reduced to realistic levels, which was not possible earlier.

Manuscript received December 11, 2020; revised April 7, 2021; accepted April 18, 2021. Date of publication May 5, 2021; date of current version February 10, 2022. Manuscript received in final form April 22, 2021. This work was supported by The Scientific and Technological Research Council of Turkey (TUBITAK) under Grant 118E202. Recommended by Associate Editor A. Vahidi. *(Corresponding author: Berat Mert Albaba.)*

The authors are with the Department of Mechanical Engineering, Bilkent University, 06800 Ankara, Turkey (e-mail: mert.albaba@bilkent.edu.tr; yyildiz@bilkent.edu.tr).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCST.2021.3075557.

Digital Object Identifier 10.1109/TCST.2021.3075557

3) A dramatically larger percentage of real driver patterns from two different data are successfully modeled compared to earlier results.
4) It is shown that the proposed models perform significantly better than the baseline models in the literature.

To the best of our knowledge, combining deep RL with GT to model human driver behavior and demonstrating the resulting modeling framework's predictive power through traffic data validation was not reported earlier in the literature.

This work is organized as follows. In Section II, the algorithm combining DQN and the level-k approach is described. In Section III, physical vehicle models, raw traffic data processing, and driver observation and action spaces are explained. In Section IV, the details of the training and simulation of driver policies are given. In Section V, validation studies are presented. A summary is provided in Section VI.

II. Method

A. Level-k Reasoning

In order to model the strategic decision-making process of human drivers, a game-theoretical concept named level-k reasoning is used [33]. The level-k approach is a hierarchical decision-making concept and presumes that different reasoning levels exist for different humans. The lowest level of reasoning in this concept is called level-0 reasoning. A level-0 agent is a nonstrategic/naive agent since his/her decisions are not based on other agents' possible actions. All level-k agents, except for level-0, presume that the rest of the agents are level-$(k-1)$ and make their decisions based on this belief. Since this belief may not always hold, the agents have bounded rationality, meaning that they do not act optimally in all situations but provide adequate performance.

B. Deep Q-Learning

In time-extended scenarios, where the agents make a series of decisions before an episode is completed, such as the traffic scenarios focused on in this work, level-k reasoning cannot be used alone. To obtain driver models that provide the best responses to the other agents' likely actions in a multimove setting, we utilize DQN together with level-k reasoning. The main reason for the employment of DQN is the continuous state space that becomes infeasible to handle with other RL methods used in earlier studies [2], [35], [37], [38]. The detailed expositions of DQN can be found in [34].

The neural network architecture for DQN utilized in this work is a four-layer network initialized with the Glorot uniform initialization [41], which has an input layer consisting of 19 nodes that take state representations, an output layer with seven nodes that gives action Q-values, and three hidden layers with rectified linear unit (ReLU) [42] activation that contains 256, 256, and 128 nodes. In this work, experience replay and target network structures are also utilized, explained in [34].

Boltzmann exploration, i.e., softmax, is utilized with an initial temperature of $T = 50$, which decreases exponentially to 1 during the training process. With these temperature values, the training starts with an almost uniform action probability distribution, and then, the probability of taking the actions with high Q values increases gradually. At time step t, the probability of taking action a is given as

$$P_t(a) = \frac{e^{Q_t(a)/T}}{\sum_{i=0}^{n-1} e^{Q_t(a_i)/T}} \tag{1}$$

where n represents the number of actions [43].

Algorithm 1 Training Up to the Level-k Agent by Combining DQN and Level-k Reasoning

1: Load the predetermined level-0 policy, π^0
2: **for** j = 0 to k-1 **do**
3: Assign level-(j) policy to the agents in the environment, p_i, as: $\pi_{p_i} = \pi^j$, $i = 1, 2, ..., n_d$
4: Initialize the ego driver's policy to a uniform action probability distribution over all states: $\pi_{ego} = \pi^{uniform}$
5: Train the ego driver using DQN
6: At the end of the training, save the resulting policy as the level-(j+1) policy, π^{j+1}
7: **end for**

C. Combining Level-k Reasoning With Deep Q-Learning

To generate agents with different reasoning levels for modeling multimove strategic decision-making in traffic scenarios, the learning capability offered by DQN is combined with the level-k reasoning approach. The combination of level-k reasoning and DQN is explained in [35].

In the proposed approach, the predetermined, nonstrategic level-0 policy is the anchoring policy from which all the higher levels are derived using DQN. In order to obtain the level-1 policy, a traffic scenario is created where all drivers are level-0 agents except for the ego driver, and a uniform distribution policy is assigned to the ego driver, i.e., ego driver selects the actions randomly at the beginning. Then, the ego agent is trained in this environment through DQN and learns how to respond best to the level-0 policy. Once the training is over, the ego driver becomes a level-1 agent. For the training of the level-2 agent, a traffic scenario is formed where all drivers are level-1 agents whose policy is obtained previously. A uniform distribution policy is then assigned to the ego driver, and the ego drive is trained in the environment formed by level-1 drivers. Thus, in the end, the ego driver learns the best responses to level-1 agents, i.e., level-1 policy, and the level-2 agent is obtained. Training of higher levels, level-k, $k \geq 1$, is achieved similarly. The procedure for obtaining reasoning levels up to the level-k through the proposed combination of level-k reasoning and DQN is explained in Algorithm 1, where n_d is the number of drivers.

The utilized hierarchical learning process is computationally feasible since, at each stage of learning, the agents other than the ego agent use previously trained policies and become parts of the environment. The computed driver policies can then be used to obtain traffic scenarios containing a mixture of different levels, where all the agents are simultaneously making strategic decisions. This approach contrasts with the conventional driver models used for crowded traffic scenarios, where one or two drivers are strategic decision-makers, and

the rest are assigned predefined policies that satisfy certain kinematic constraints. In this work, the highest level is set to level-3, in accordance with [44].

The main differences of the proposed approach from other RL methods, such as robust adversarial RL [45] and multiagent RL (MARL) [46], are as follows; Robust RL covers zero-sum games, and extending the training for a zero-sum game with a large number of players is nontrivial. MARL also suffers from scalability issues. The round-robin approach addresses the scalability problem of MARL. However, this method assumes the knowledge of all players in the game, and thus, it is not applicable for partially observable Markov decision Processes (POMDPs). Besides, it is developed only for cooperative games. Finally, in MARL, it is assumed that agents are rational and do not deviate from the optimal converged policy, and therefore, resulting policies are not bounded rational.

Remark 1: Since level-k reasoning reduces all the agents, except for the ego agent, to becoming a part of the environment, the training process and its convergence properties are equivalent to those of the conventional DQN.

Remark 2: Different driver levels may represent different driving characteristics, as well as the depth of reasoning. For example, a level-1 driver trained in traffic consisting of level-0 agents that never change lanes can be more aggressive compared to a level-2 agent that is trained in an environment consisting of these aggressive level-1 types.

III. Traffic Scenario

The traffic scenario comprises a five-lane highway and multiple vehicles. The lane width is 3.7 m, and each vehicle's size is 5 m $\times$ 2 m. The vehicles have continuous dynamics. Specific numerical values needed to create the traffic scenario, such as observation and action space parameters, are determined based on one of the two traffic data sets. Therefore, we first explain how data are processed before providing the scenario details. It is noted that data are only used for determining parameters of action and observation spaces and for comparison with the proposed policies.

A. Traffic Data Processing

In this work, two sets of traffic data, collected on US101 and I80 highways [47], [48], are used for model validation. Among these two, the US101 set is employed to determine the observation and action space parameter values. Raw data are first processed to eliminate unrealistic velocity changes, such as 12-m/s increase/decrease in 0.1 s. The problem of large velocity jumps is solved by applying a linear curve fitting. To exemplify, if among the velocity values $v_{i-5}, v_{i-4}, \ldots, v_i, v_{i+1}, v_{i+2}, \ldots, v_{i+5}$, where the subscripts denote the time steps, the values v_{i+1} and v_{i+2} show jumps, and these values are replaced with the appropriate values $v_{i+1} = v_i + (v_{i+3} - v_i)/3$ and $v_{i+2} = v_i + 2(v_{i+3} - v_i)/3$. Then, acceleration values are obtained by using the five-point stencil method [49] given as $a_i = (-v_{i+2} + 8v_{i+1} - 8v_{i-1} + v_{i-2})/12$.

B. Driver Observation Space

In this work, it is assumed that a driver on lane l observes the closest front and rear cars on lanes $l-2, l-1, l+1$, and $l+2$, along with the front car on lane l. Therefore, up to nine surrounding cars are observable by the driver (see Fig. 1). Observations are coded as relative positions and velocities. Specifically, a driver on lane l can detect: 1) relative positions and velocities of the vehicles that are in front of the driver, on lanes $l-2, l-1, l, l+1$, and $l+2$; 2) relative positions and velocities of the vehicles that are at the back of the driver, on lanes $l-2, l-1, l+1$, and $l+2$; and 3) own lane number (l).

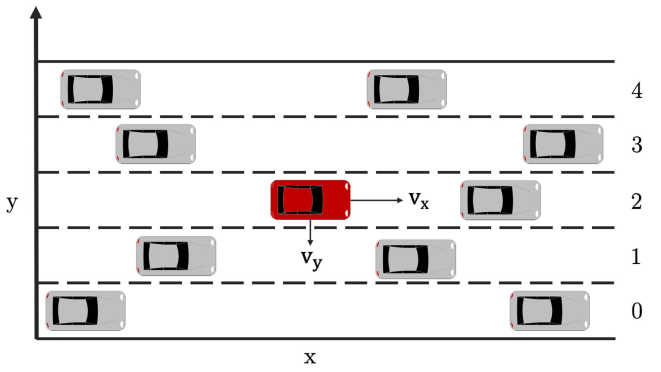


Fig. 1. Ego vehicle (red, center) and the vehicles the ego driver can observe. Lane numbers are shown on the right.

C. Driver Action Space

Drivers have two action types: changing lane and changing acceleration. For lane change, two actions are defined: moving to the left lane and moving to the right lane, which are assumed to be completed in 1 s. To determine acceleration changing actions, the distribution of vehicle accelerations, obtained by processing the US101 data, is used. Fig. 2 presents the acceleration distribution. In the figure, five regions are identified and approximated by known continuous distributions that are shown in red color and superimposed on the original figure. Based on this acceleration data analysis, the driver actions in terms of accelerations are defined as follows.

1) *Maintain:* Acceleration is sampled from normal distribution with $\mu = 0$ and $\sigma = 0.075$ m/s^2.
2) *Accelerate:* Acceleration is sampled from a uniform distribution between 0.5 m/s^2 and 2.5 m/s^2.
3) *Decelerate:* Acceleration is sampled from a uniform distribution between -0.5 m/s^2 and -2.5m/s^2.
4) *Hard Accelerate:* Acceleration is sampled from an inverse half normal distribution with $\mu = 3.5$ m/s^2 and $\sigma = 0.3$ m/s^2.
5) *Hard Decelerate:* Acceleration is sampled from a half normal distribution with $\mu = -3.5$ m/s^2 and $\sigma = 0.3$ m/s^2.

Remark 3: Distributions superimposed on the histogram in Fig. 2 are continuous. Therefore, although the DQN samples actions from five separate distributions, the actions that were taken by the drivers are represented by continuous variables. In other words, there are infinitely many actions that are available for the drivers to take.

D. Equations of Motion

In Fig. 1, the variable x is used to represent the longitudinal position and y represents the lateral position. Similarly, v_x

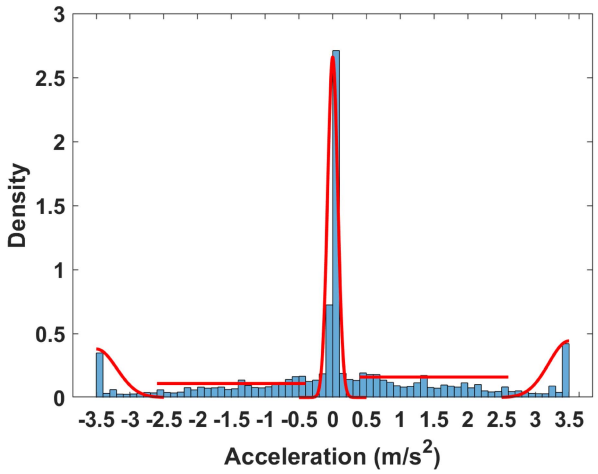


Fig. 2. Acceleration distribution (blue line), together with superimposed standard distributions (red line).

and v_y represent the longitudinal and lateral velocities, respectively. The equations of motion for the vehicles in the traffic are given by

$$x(t_0+t) = x(t_0) + v_x(t_0) + \frac{1}{2}a(t_0)t^2 \tag{2}$$

$$y(t_0+t) = y(t_0) + v_y(t_0) \tag{3}$$

$$v_x(t_0+t) = v_x(t_0) + a(t_0)t \tag{4}$$

where t_0 is the initial time step and a is the acceleration. Furthermore, it is assumed that lane changing takes 1 s.

E. Vehicle Placements

At the beginning of the training and simulations, vehicles are randomly placed on a 600-m endless circular road segment. It is observed from the US101 data that 50% of the time, intervehicle distances remain between 11 and 27 m. Therefore, initial distances between vehicles are constrained to be larger than or equal to 11 m during initial vehicle placement. Initial velocities are selected to prevent impossible-to-handle cases at the beginning of the training or simulation. A driver who is in close proximity to the vehicle in front should be able to prevent a crash using the hard decelerate action.

F. Reward Function

In the reward function, a variable is defined for each of these goals, and the weights are assigned to these variables to emphasize their relative importance as

$$R = w_1 * c + w_2 * s + w_3 * d + w_4 * e \tag{5}$$

where w_i are the weights. The first term of the reward function, c, is included in order to prevent crashes. c equals -1 if a crash occurs and 0, otherwise. The second term, s, used to have a high enough speed and calculated as $s = (v(t) - (v_{\max} + v_{\min})/2)/v_{\max}$, where $v_{\max} = 24.59$ m/s and $v_{\min} = 2.78$ m/s. The purpose of the third term, d, is to keep a safe distance from the front car and equal to -1 if the distance to the car in front is smaller than 11 m, 0 if the distance to the car in front is between 11 and 27 m, and 1, otherwise. The last term, e, is introduced to reduce the amount of unnecessary driver actions and is equal to 0 if the action of the driver is maintain, -0.25 if the action is accelerate or decelerate, -0.5 if the action is hard accelerate or hard decelerate, and -1 if the action is move left or move right.

Remark 4: $v_{\max}$ is selected as 24.59 m/s (55 mi/h), which is the speed limit at US101 for the selected road section. Driver models are not allowed to pass this speed limit. The average velocity, $(v_{\max} + v_{\min})/2$, is not the desired velocity. As shown in (17), drivers take positive rewards as they approach the maximum velocity and can be penalized for velocities smaller than the nominal velocity.

IV. Training and Simulation

During the training of a level-k driver, 125 level-$(k-1)$ vehicles are placed on the road, together with the ego vehicle. This placement makes the traffic density approximately equal to that of US101 data [47]. The number of cars is decreased to 100 at the end of the 1300th episode and increased to 125 again at the end of the 3800th episode, to increase the number of states that the drivers are exposed to during training.

A. Level-0 Policy

The nonstrategic level-0 policy must be determined first before obtaining other levels. In earlier studies, where approaches similar to the one proposed in this work are used, level-0 policies are set as a single persisting action regardless of the state being observed [50]–[52] or as a conditional logic based on experience [53]. The level-0 policy used in this study is defined as follows:

1) hard decelerate if the car in front is closer than 11 m and approaching;
2) decelerate if the car in front is closer than 11 m and stable or its relative position is between 11 and 27 m and approaching;
3) accelerate if the car in front has a relative position that is between 11 and 27 m and it is moving away or the relative position is larger than 27 m;
4) maintain otherwise.

In the definition of level-0 policy given above, the terms *approaching*, *stable*, and *moving away* are defined precisely. A vehicle having a relative velocity smaller than -0.1 m/s is considered as *approaching*, a vehicle having a relative velocity between -0.1 and 0.1 m/s is considered as *stable*, and a vehicle having a relative velocity larger than 0.1 m/s is considered as *moving away*. Relative velocity is defined as $v_{\text{front}} - v_{\text{back}}$.

B. Simulation Performance

The following scenarios are simulated.

1) Level-1 driver is placed on a traffic environment consisting $n_d - 1$ level-0 drivers on a 600-m road segment.
2) Level-2 driver is placed on a traffic environment consisting $n_d - 1$ level-1 drivers on a 600-m road segment.
3) Level-3 driver is placed on a traffic environment consisting $n_d - 1$ level-2 drivers on a 600 - road segment.

Here, n_d corresponds to the total number of drivers on the road. Simulations are performed for $n_d = 75$, 100, and 125, for each scenario. In all of the above scenarios, simulations are run for 100 episodes, each covering a 100-s simulation. In these simulations, drivers do not experience any crashes.

When compared with previous studies [2], [36], policies proposed in this work show more realistic driving behavior since the average crash rate is 2 per million miles driven nationally [54].

Remark 5: In the level-k reasoning approach, a driver assumes that the other drivers are one level below him/her. For scenarios where this assumption is violated, the drivers respond to an incorrect assumption, and therefore, crash rates naturally increase. For example, in our simulations, we observed 2%, 1%, and 0.67% crash rates for level-1 versus level-1, level-2 versus level-2, and level-3 versus level-3 encounters, respectively. Developing models for mixed level-k encounters is an active area of research, where the models use a "dynamic level-k method." A recent related work can be found in [55].

V. Validation With Traffic Data

In order to compare the proposed policies with the policies obtained by processing the real traffic data, Kolmogorov–Smirnov test (K-S test) for discontinuous distributions [56] is used. The test is explained briefly in Section V-A, and a more detailed description can be found in [56].

A. K-S Test for Discontinuous Distributions

For an unknown discrete probability distribution function (pdf) $F(x)$ and a hypothesized pdf $H(x)$, the null hypothesis of the K-S test is

$$H_0 : F(x) = H(x) \text{ for all x.} \tag{6}$$

To test the null hypothesis, first, empirical cumulative pdf of the observed data, $S_n(x)$, and hypothesized cumulative pdf, $H_c(x)$, are calculated. Second, the test statistics, which are measures of the difference between $S_n(x)$ and $H_c(x)$, are calculated as $D = \sup_x |H_c(x) - S_n(x)|$, $D^- = \sup_x (H_c(x) - S_n(x))$ and $D^+ = \sup_x (S_n(x) - H_c(x))$, where D is the two sided and D^- and D^+ are one sided test statistics. Third, critical levels of D^- and D^+, $P(D^- \leq d^-)$ and $P(D^+ \leq d^+)$ are calculated, where d^- and d^+ denote the observed values of D^- and D^+, respectively. Finally, the critical value for the two-sided test statistic is determined as

$$P(D \geq d) \doteq P(D^+ \geq d) + P(D^- \geq d) \tag{7}$$

where d is the observed value of D. It is noted that this critical value describes the percentage of data samples whose test statistics are larger than or equal to d, given that the null hypothesis is true. Thus, the probability of observation (data point) being sampled from the hypothesized model, $H(x)$, or, equivalently, the probability of the null hypothesis being true, increases with the increase in the critical value. The null hypothesis is rejected if the critical value is smaller than a certain threshold called the significance value α. Two significance values are used: 0.05 and 0.10. The first one is selected since it is a commonly used value [56] and the second is selected to show its effect on the results.

Remark 6: Our null hypothesis is that the investigated model is representative of data. We retain this hypothesis when the analysis shows that it is not rejected. Therefore, in the rest of this work, we call a data-model comparison "successful" when the null hypothesis is not rejected.

B. Comparing Game-Theoretical Models With Traffic Data

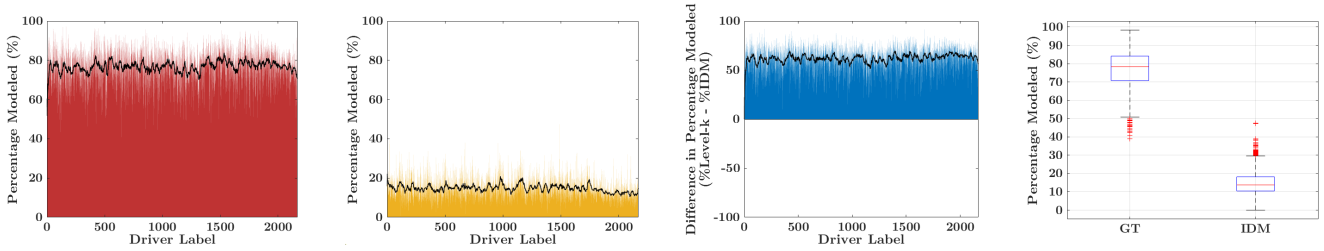
The proposed continuous GT models are pdfs over the action space defined in Section III. Since observing the same state is nearly impossible in the continuous case, a probability distribution cannot be formed. To solve this problem, states are binned, and probability distributions are obtained based on the frequency of actions taken by the real drivers (or models) for each bin. For both the GT policies and the ones obtained from the data, action probabilities that are lower than 0.01 are replaced with 0.01 with renormalizations to eliminate close-to-zero probabilities.

GT and the data-based policies are compared for each driver. Details are explained, and the comparison process for each driver is given in [35]. However, in this work, since DQN is used instead of traditional RL methods, $n_{V_{\text{model}}} \geq n_{\text{limit}}$ constraint in [35, Algorithm 4] is removed. Thus, a significantly larger portion of the real data is used in comparisons.

To compare the performance of these models with an alternative model, the alternative model should have the same stochastic map structure of proposed GT policies. In this work, policies in the previous work [35] previous GT (pGT) policies, discrete GT (dGT) models, IDM [39], and MOBIL [40] are used as baseline models. For MOBIL, since the rear vehicle on the same lane is not included in the observation space of the ego driver in this work, this car is omitted, and policies are generated for two different politeness values, p: 0 and 1, referred to as $M - 0$ and $M - 1$. IDM and MOBIL do not have stochastic map structures. Thus, first, policies of IDM and MOBIL are generated. In other words, for each state, actions taken by IDM and MOBIL are generated for 100 random distance and velocity samples. Then, action probability distributions are obtained for IDM and MOBIL.

C. Results

The following definitions are employed when reporting the validation results. Given two discrete pdf p and q, the mean absolute error (MAE) between p and q measures the average error between pdfs and is defined as $\text{MAE} = 1/n \sum_{i=1}^{n} |p(x_i) - q(x_i)|$, where x_is are random variables. aMAE is the average of the $\text{MAE}_j s$ between the GT policies and the data-based policies, for which the null hypothesis is not rejected. Therefore, aMAE is calculated as $\text{aMAE} = 1/M \sum_{j=1}^{M} \text{MAE}_j$, where M is the number of comparisons for which the null hypothesis is not rejected. Finally, rMAE is the average of the $\text{MAE}_k s$ between the GT policies and the data-based policies, for which the null hypothesis is rejected. Therefore, rMAE is calculated as $\text{rMAE} = 1/K \sum_{k=1}^{K} \text{MAE}_k$, where K is the number of comparisons for which the null hypothesis is rejected.



(a) (b) (c) (d)

Fig. 3. Comparison results for $n_{\text{limit}} = 3$ (US101). (a) Percentages of modeled states by the GT policies. (b) Percentages of modeled states by the IDM policy. (c) Differences in the percentages of the modeled states. (d) Box map showing median, maximum, and minimum. Standard error in the mean: 0.22% for GT and 0.13% for IDM.

TABLE I

HUMAN DRIVER MODELING PERFORMANCES OF GT POLICIES AND THE BASELINE METHODS FOR US101 DATA

		n_{state} = 3	n_{state}= 5
α = 0.05	Mean % modeled by GT policies	**76.73%**	**72.69%**
	Mean % modeled by pGT policies	60.92%	54.34%
	Mean % modeled by dGT policies	40.26%	34.76%
	Mean % modeled by IDM	17.74%	8.76%
	Mean % modeled by M-0	6.09%	2.80%
	Mean % modeled by M-1	1.34%	0.66%
	Mean % difference: %GT - %pGT	15.81%	18.35%
	Mean % difference: %GT - %dGT	36.47%	37.93%
	Mean % difference: %GT - %IDM	58.99%	63.93%
	Mean % difference: %GT - %M-0	70.64%	69.89%
	Mean % difference: %GT - %M-1	75.39%	72.03%
	aMAE	0.10	0.08
	rMAE	0.22	0.20
α = 0.10	Mean % modeled by GT policies	**68.22%**	**64.28%**
	Mean % modeled by pGT policies	52.48%	45.73%
	Mean % modeled by dGT policies	31.88%	30.10%
	Mean % modeled by IDM	11.73%	5.31%
	Mean % modeled by M-0	3.77%	1.23%
	Mean % modeled by M-1	0.89%	0.41%
	Mean % difference: %GT - %pGT	15.74%	18.55%
	Mean % difference: %GT - %dGT	36.34%	34.18%
	Mean % difference: %GT - %IDM	56.49%	58.97%
	Mean % difference: %GT - %M-0	64.45%	63.05%
	Mean % difference: %GT - %M-1	67.33%	63.87%
	aMAE	0.09	0.08
	rMAE	0.21	0.19

TABLE II

HUMAN DRIVER MODELING PERFORMANCES OF GT POLICIES AND THE BASELINE METHODS FOR I80 DATA

		n_{state} = 3	n_{state}= 5
α = 0.05	Mean % modeled by GT policies	**71.42%**	**64.37%**
	Mean % modeled by pGT policies	35.97%	30.66%
	Mean % modeled by dGT policies	46.43%	41.83%
	Mean % modeled by IDM	4.56%	2.46%
	Mean % modeled by M-0	1.86%	0.96%
	Mean % modeled by M-1	0.15%	0.05%
	Mean % difference: %GT - %pGT	35.45%	33.71%
	Mean % difference: %GT - %dGT	24.99%	22.54%
	Mean % difference: %GT - %IDM	66.86%	61.91%
	Mean % difference: %GT - %M-0	69.56%	63.41%
	Mean % difference: %GT - %M-1	71.27%	64.32%
	aMAE	0.09	0.07
	rMAE	0.22	0.21
α = 0.10	Mean % modeled by GT policies	**63.04%**	**56.37%**
	Mean % modeled by pGT policies	28.73%	24.24%
	Mean % modeled by dGT policies	41.15%	38.70%
	Mean % modeled by IDM	3.31%	1.43%
	Mean % modeled by M-0	1.15%	0.44%
	Mean % modeled by M-1	0.08%	0.03%
	Mean % difference: %GT - %pGT	34.31%	32.13%
	Mean % difference: %GT - %dGT	21.89%	17.67%
	Mean % difference: %GT - %IDM	59.73%	54.94%
	Mean % difference: %GT - %M-0	61.89%	55.93%
	Mean % difference: %GT - %M-1	62.96%	56.34%
	aMAE	0.08	0.06
	rMAE	0.21	0.20

Model versus data comparisons are made for two different n_{limit} values, specifically for $n_{\text{limit}} = 3$ and $n_{\text{limit}} = 5$. As explained earlier, n_{limit} is the minimum number of state visits in the traffic data for the corresponding policy to be considered in the K-S test. It is observed that the minimum number of state visits is approximately equal to 3 for the K-S test to acknowledge that the policy is sampled from a nonuniform distribution, with a significance value of 0.05. Therefore, we report the results for $n_{\text{limit}} = 3$. Moreover, we also report the results for $n_{\text{limit}} = 5$ to demonstrate this variable's effect on the test outcomes.

1) Model Validation Using US-101 Data: In this section, we give comparison results between the policies obtained by processed US-101 Data and the GT policies. The data are collected between 7.50 and 8.05 A.M. and consists of 2168 different drivers [47].

Table I presents the performances of the proposed GT policies, pGT policies, dGT policies, IDM, and MOBIL along with aMAE and rMAE values. In the table, "mean % modeled" refers to the average of the successfully modeled state percentages over all 2168 drivers. As shown in this table, the proposed policies can model human driver behavior better than the baseline models. As explained in [35], since a tabular/traditional RL method is utilized, data are filtered in the previous work, and policies are compared with a limited part of the data. However, in this work, data are not filtered, and all of the data are used for comparisons. Thus, although much larger data are used for comparisons, there is a significant performance increase when the proposed policies are compared with the policies in the previous work. In addition, as a visual demonstration, Fig. 3 shows the performances of GT policies and IDM for $n_{\text{state}} = 3$ in US101 data. In Fig. 3(a)–(c), each vertical line belongs to an individual driver, and the black line plots the moving average for every 20 drivers. Fig. 3(d) shows the median, 25th–75th percentiles, maximum, minimum, and outliers (red pluses).

2) Model Validation Using I80 Data: I-80 data collected between 5.00 and 5.15 P.M. is used as the second validation test, which contains 1835 drivers.

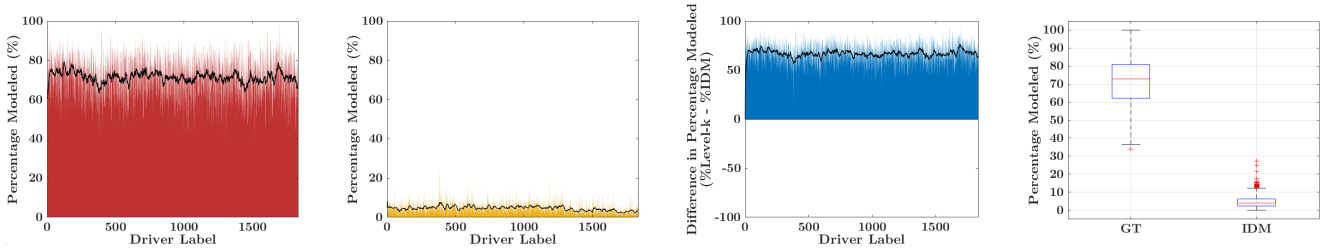


Fig. 4. Comparison results for $n_{\text{limit}} = 3$ (I80). (a) Percentages of modeled states by the GT policies. (b) Percentages of modeled states by the IDM policy. (c) Differences in the percentages of modeled states. (d) Box map showing median, maximum, and minimum. Standard error in the mean: 0.27% for GT and 0.07% for IDM.

Table II presents the performances of the proposed GT policies and baseline models for I80 data. This table also shows that the proposed GT policies overperformed baseline methods, IDM and MOBIL, in terms of modeling human drivers. Besides, again, the performances of the proposed policies are significantly better than the policies in the previous work. Also, a comparison between Tables II and III reveals that I80 data are harder to model human driver behaviors for most of the models. For visual demonstration, Fig. 4 shows the performances of GT policies and IDM for $n_{\text{state}} = 3$ in I80 data for each individual driver along with statistics.

Remark 7: US101 data are used only to determine the observation and action set boundaries. It is not used to train the GT driver models. Therefore, the GT policies are not obtained by fitting the model parameters to the data. However, since these data are used to set the observation-action space boundaries, it still affected, albeit indirectly, the obtained models. To test the resulting GT policies with data that are not used in any way to obtain these policies, additional model validation tests are conducted with the I80 data. To summarize, although the US101 data are not used to train the models and, therefore, overfitting is not a concern, additional validation tests are conducted with the I80 data for further assurance of the validity of the GT models.

VI. Summary

In this work, a modeling framework combining a GT concept named level-k reasoning and a deep RL method called DQN is proposed. It is demonstrated that, compared with earlier similar studies, the crash rates of the proposed driver models are more realistic. For evaluating the predictive power of the GT models, two independent traffic data sets, obtained from highways US101 and I80, are used. GT models and the driver policies derived from processing the data are compared using the K-S test. The results show that the proposed GT policies can model up to 77% and 72% of the driver policies obtained from the US101 and the I-80 data sets, respectively. Furthermore, GT policies perform better than the baseline models, IDM [39] and [40], and the policies in [35].

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*. Santa Monica, CA, USA: Rand Corporation, 2014.

[2] 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 Trans. Control Syst. Technol.*, vol. 26, no. 5, pp. 1782–1797, Sep. 2018.

[3] J. Lygeros, D. N. Godbole, and S. Sastry, "Verified hybrid controllers for automated vehicles," *IEEE Trans. Autom. Control*, vol. 43, no. 4, pp. 522–539, Apr. 1998.

[4] T. Wongpiromsarn and R. M. Murray, "Formal verification of an autonomous vehicle system," in *Proc. Conf. Decis. Control*, 2008, pp. 1–7.

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

[6] N. Wan, C. Zhang, and A. Vahidi, "Probabilistic anticipation and control in autonomous car following," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 1, pp. 30–38, Jan. 2019.

[7] W. Wang, D. Zhao, W. Han, and J. Xi, "A learning-based approach for lane departure warning systems with a personalized driver model," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9145–9157, Oct. 2018.

[8] A. Burton *et al.*, "Driver identification and authentication with active behavior modeling," in *Proc. 12th Int. Conf. Netw. Service Manage. (CNSM)*, Oct. 2016, pp. 388–393.

[9] M. Zhao, D. Kathner, M. Jipp, D. Soffker, and K. Lemmer, "Modeling driver behavior at roundabouts: Results from a field study," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 908–913.

[10] J. Morton, T. A. Wheeler, and M. J. Kochenderfer, "Analysis of recurrent neural networks for probabilistic modeling of driver behavior," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1289–1298, May 2017.

[11] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, "Imitating driver behavior with generative adversarial networks," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 204–211.

[12] J. Hu and S. Luo, "A car-following driver model capable of retaining naturalistic driving styles," *J. Adv. Transp.*, vol. 2020, pp. 1–16, Jan. 2020.

[13] I. Dagli, M. Brost, and G. Breuel, "Action recognition and prediction for driver assistance systems using dynamic belief networks," in *Proc. Net. ObjectDays, Int. Conf. Object-Oriented Internet-Based Technol., Concepts, Appl. Netw. World*. Berlin, Germany: Springer, 2002, pp. 179–194.

[14] 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," *Auton. Robots*, vol. 35, no. 1, pp. 51–76, Jul. 2013.

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

[16] M. Kuderer, S. Gulati, and W. Burgard, "Learning driving styles for autonomous vehicles from demonstration," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2015, pp. 2641–2646.

[17] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for autonomous cars that leverage effects on human actions," in *Proc. Robot., Sci. Syst.*, Ann Arbor, MI, USA, vol. 2, 2016.

[18] J. H. Yoo and R. Langari, "Stackelberg game based model of highway driving," in *Proc. 5th Annual Dyn. Syst. Control Conf. Joint JSME 11th Motion Vib. Conf.*, Oct. 2012, pp. 499–508.

[19] C. Dextreit and I. V. Kolmanovsky, "Game theory controller for hybrid electric vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 2, pp. 652–663, Mar. 2014.

[20] L. Sun, W. Zhan, and M. Tomizuka, "Probabilistic prediction of interactive driving behavior via hierarchical inverse reinforcement learning," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2111–2117.

[21] P. Wang and C.-Y. Chan, "Formulation of deep reinforcement learning architecture toward autonomous driving for on-ramp merge," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–6.

[22] J. Liu, P. Hou, L. Mu, Y. Yu, and C. Huang, "Elements of effective deep reinforcement learning towards tactical driving decision making," 2018, *arXiv:1802.00332*. [Online]. Available: https://arxiv.org/abs/1802.00332

[23] K. Min, H. Kim, and K. Huh, "Deep Q learning based high level driving policy determination," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 226–231.

[24] K. Min, H. Kim, and K. Huh, "Deep distributional reinforcement learning based high-level driving policy determination," *IEEE Trans. Intell. Vehicles*, vol. 4, no. 3, pp. 416–424, Sep. 2019.

[25] M. Bouton, A. Nakhaei, K. Fujimura, and M. J. Kochenderfer, "Safe reinforcement learning with scene decomposition for navigating complex urban environments," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 1469–1476.

[26] M. Jain, K. Brown, and A. K. Sadek, "Multi-fidelity recursive behavior prediction," 2018, *arXiv:1901.01831*. [Online]. Available: https://arxiv.org/abs/1901.01831

[27] R. P. Bhattacharyya, D. J. Phillips, C. Liu, J. K. Gupta, K. Driggs-Campbell, and M. J. Kochenderfer, "Simulating emergent properties of human driving behavior using multi-agent reward augmented imitation learning," in *Proc. Int. Conf. Robot. Automat. (ICRA)*, May 2019, pp. 789–795.

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

[29] M. D. Lio, A. Mazzalai, K. Gurney, and A. Saroldi, "Biologically guided driver modeling: The stop behavior of human car drivers," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2454–2469, Aug. 2018.

[30] J. Han, D. Karbowski, N. Kim, and A. Rousseau, "Human driver modeling based on analytical optimal solutions: Stopping behaviors at the intersections," in *Proc. Dyn. Syst. Control Conf.*, vol. 59162, Jan. 2019, Art. no. V003T18A010.

[31] R. Nagel, "Unraveling in guessing games: An experimental study," *Amer. Econ. Rev.*, vol. 85, no. 5, pp. 1313–1326, 1995.

[32] D. O. Stahl and P. W. Wilson, "On players' models of other players: Theory and experimental evidence," *Games Econ. Behav.*, vol. 10, no. 1, pp. 218–254, Jul. 1995.

[33] C. F. Camerer, T.-H. Ho, and J.-K. Chong, "A cognitive hierarchy model of games," *Quart. J. Econ.*, vol. 119, no. 3, pp. 861–898, Aug. 2004.

[34] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015.

[35] B. M. Albaba and Y. Yildiz, "Modeling cyber-physical human systems via an interplay between reinforcement learning and game theory," *Annu. Rev. Control*, vol. 48, pp. 1–21, Jan. 2019.

[36] M. Albaba, Y. Yildiz, N. Li, I. Kolmanovsky, and A. Girard, "Stochastic driver modeling and validation with traffic data," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2019, pp. 4198–4203.

[37] 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 *Proc. IEEE 55th Conf. Decis. Control (CDC)*, Dec. 2016, pp. 727–733.

[38] D. W. Oyler, Y. Yildiz, A. R. Girard, N. I. Li, and I. V. Kolmanovsky, "A game theoretical model of traffic with multiple interacting drivers for use in autonomous vehicle development," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2016, pp. 1705–1710.

[39] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 62, no. 2, p. 1805, 2000.

[40] A. Kesting, M. Treiber, and D. Helbing, "General lane-changing model MOBIL for car-following models," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1999, no. 1, pp. 86–94, Jan. 2007.

[41] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.

[42] A. F. Agarap, "Deep learning using rectified linear units (ReLU)," 2018, *arXiv:1803.08375*. [Online]. Available: https://arxiv.org/abs/1803.08375

[43] S. Ravichandiran, *Hands-on Reinforcement Learning with Python: Master Reinforcement and Deep Reinforcement Learning Using OpenAI gym and TensorFlow*. Birmingham, U.K.: Packt Publishing Ltd, 2018.

[44] M. A. Costa-Gomes, V. P. Crawford, and N. Iriberri, "Comparing models of strategic thinking in van huyck, battalio, and Beil's coordination games," *J. Eur. Econ. Assoc.*, vol. 7, nos. 2–3, pp. 365–376, Apr. 2009.

[45] L. Pinto, J. Davidson, R. Sukthankar, and A. Gupta, "Robust adversarial reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2817–2826.

[46] B. Fernandez-Gauna, I. Etxeberria-Agiriano, and M. Graña, "Learning multirobot hose transportation and deployment by distributed round-robin Q-Learning," *PLoS ONE*, vol. 10, no. 7, Jul. 2015, Art. no. e0127129.

[47] U. F. H. Administration. *Us101 Dataset*. Accessed: Apr. 29, 2021. [Online]. Available: https://www.fhwa.dot.gov/publications/research/operations/07030

[48] U. F. H. Administration. *I80 Dataset*. Accessed: Apr. 29, 2021. [Online]. Available: https://www.fhwa.dot.gov/publications/research/operations/07030

[49] T. Sauer, *Numerical Analysis*, 3rd ed. London, U.K.: Pearson, 2017.

[50] N. Musavi, D. Onural, K. Gunes, and Y. Yildiz, "Unmanned aircraft systems airspace integration: A game theoretical framework for concept evaluations," *J. Guid., Control, Dyn.*, vol. 40, no. 1, pp. 96–109, Jan. 2017.

[51] N. Musavi, K. B. Tekelioğlu, Y. Yildiz, K. Gunes, and D. Onural, "A game theoretical modeling and simulation framework for the integration of unmanned aircraft systems in to the national airspace," in *Proc. AIAA Infotech Aerosp.*, 2016, p. 1001.

[52] Y. Yildiz, A. Agogino, and G. Brat, "Predicting pilot behavior in medium scale scenarios using game theory and reinforcement learning," in *Proc. AIAA Model. Simul. Technol. (MST) Conf.*, Aug. 2013, p. 4908.

[53] S. Backhaus *et al.*, "Cyber-physical security: A game theory model of humans interacting over control systems," *IEEE Trans. Smart Grid*, vol. 4, no. 4, pp. 2320–2327, Dec. 2013.

[54] *Traffic Safety Facts 2017: A Compilation of Motor Vehicle Crash Data*, N. H. T. S. Administration, Washington, DC, USA, 2019.

[55] C. Köprülü and Y. Yıldız, "Act to reason: A dynamic game theoretical model of driving," 2021, *arXiv:2101.05399*. [Online]. Available: https://arxiv.org/abs/2101.05399

[56] W. J. Conover, "A kolmogorov goodness-of-fit test for discontinuous distributions," *J. Amer. Stat. Assoc.*, vol. 67, no. 339, pp. 591–596, Sep. 1972.