2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

AD4RL: Autonomous Driving Benchmarks for Offline Reinforcement Learning with Value-based Dataset

Dongsu Lee[†], Chanin Eom[†], and Minhae Kwon[†,*]

Abstract— **Offline reinforcement learning has emerged as a promising technology by enhancing its practicality through the use of pre-collected large datasets. Despite its practical benefits, most algorithm development research in offline reinforcement learning still relies on game tasks with synthetic datasets. To address such limitations, this paper provides autonomous driving datasets and benchmarks for offline reinforcement learning research. We provide 19 datasets, including real-world human driver's datasets, and seven popular offline reinforcement learning algorithms in three realistic driving scenarios. We also provide a unified decision-making process model that can operate effectively across different scenarios, serving as a reference framework in algorithm design. Our research lays the groundwork for further collaborations in the community to explore practical aspects of existing reinforcement learning methods. Dataset and codes can be found in https://sites.google.com/view/ad4rl.**

I. INTRODUCTION

Considerable progress in intelligent machines has made remarkable strides, fueled by deep neural networks trained on large datasets [1]–[3]. In contrast, an intelligent automated system such as autonomous driving has seen limited improvement. Deep reinforcement learning was a promising solution for modern control systems [4], [5], but it faces challenges due to its technical characteristics. The practical challenges of online reinforcement learning are as follows [6], [7]. Firstly, trial-and-error based learning may lead to financial loss and social damage in mission-critical systems, e.g., car accidents. Secondly, simulator-based training has an inherent gap between the simulator and real-world dynamics, resulting in limited performance in real-world deployment. Lastly, the active data collection during the training, i.e., online interaction between the agent and the environment, is expensive and hampers the ability to exploit vast previously collected datasets. Addressing these challenges is critical to realizing the full potential of reinforcement learning.

To overcome the challenges, offline reinforcement learning, also known as batch reinforcement learning [8], has recently gained attention as a promising approach for autonomous systems. This paradigm utilizes large-scale pre-collected datasets to train policies for agents. Since online data collection is no longer necessary for training, it allows us to avoid having agents perform immature and risky actions with an unstable policy in the early training phase. Offline reinforcement learning offers a secure and efficient learning method by leveraging insights from successful data-driven deep learning, providing the potential to shift the paradigm of mission-critical applications [9]–[11].

Despite recent attention to offline reinforcement learning, research on autonomous driving still heavily relies on online reinforcement learning [12]. Some recent efforts have attempted to shift the research paradigm toward offline reinforcement learning, but they are still in the early stages [9], [13], [14]. For example, [13] provides synthetic datasets and benchmarks using the FLOW framework [15]. While it serves as a valuable dataset and benchmark in offline reinforcement learning studies, it focuses exclusively on acceleration maneuvers, neglecting lane-changing and presenting unrealistic, simplified driving scenarios. Additionally, it assumes that safety modules can prevent all accidents, which is an unrealistic assumption [16]. Another challenge with existing benchmarks is the absence of real-world datasets. Most studies rely solely on synthetic datasets collected by online reinforcement learning agents without incorporating any real-world human-driving datasets [13], [14], [17].

Contributions: Our primary contribution is the incorporation of real-world human-driving datasets as well as synthetic datasets in offline reinforcement learning for autonomous driving tasks. We employ the US Highway 101 dataset [18] collected by the next generation simulator (NGSIM) project of the Federal Highway Administration (FHWA) [19]. To provide a useful dataset and benchmark, we pre-process the NGSIM dataset by labeling the reward, correcting errors, and normalizing values. We also propose a unified partially observable Markov decision process (POMDP) that can be applied across various driving scenarios. Finally, we benchmark offline reinforcement learning algorithms within the FLOW framework and extend its functionality.[1]

This work was supported in part by the National Research Foundation of Korea (NRF) grant (RS-2023-00278812) and in part by the ITRC (Information Technology Research Center) support program (IITP-2022-2020-0-01602) funded by the Korea government (MSIT). Dongsu Lee is grateful for financial support from Hyundai Motor Chung Mong-Koo Foundation. (Corresponding author: M. Kwon)

[†]Authors are with the Department of Intelligent Semiconductors, Soongsil University, Seoul 06978, Republic of Korea (e-mail:{movementwater, eci0623}@soongsil.ac.kr, minhae@ssu.ac.kr)

[*]M. Kwon is with the School of Electronic Engineering, Soongsil University, Seoul 06978, Republic of Korea

[1]This study introduces a benchmark specifically tailored for autonomous driving, aiming to ensure widespread accessibility and reproducibility. Consistent with previous literature, the study employs a simulated environment. The decision to rely on a simulated environment is motivated by the inherent challenges associated with evaluating the performance of autonomous driving policies. Existing research suggests that off-policy evaluation approaches lack the necessary reliability [7], [13], [20]. Consequently, policy candidates are evaluated exclusively within the simulated environment as a practical approach to mitigate the risks inherent in policy evaluation. It is worth noting that while policy evaluation takes place in the simulator, real-world datasets are incorporated for policy training.

979-8-3503-8457-4/24/$31.00 ©2024 IEEE

II. Related Works

Reinforcement Learning Based Autonomous Driving The advancement of deep reinforcement learning has played a pivotal role in propelling the progress of autonomous driving research. Researchers have diligently tackled a spectrum of control tasks, encompassing acceleration, lane-changing, and intersection negotiation [21], [22]. Previous studies have predominantly focused on specific roadway configurations, encompassing single-lane roads, multi-lane highways, on/off-ramps, and scenarios involving lane reduction [15], [23]–[25]. Nonetheless, these decision-making frameworks remain tailored to distinct scenarios, lacking universality across diverse driving contexts. The paramount objective of an autonomous driving decision-making model lies in its capacity to seamlessly and safely operate across a spectrum of driving scenarios. Consequently, this paper introduces a comprehensive decision-making process model designed to accommodate diverse driving scenarios.

Reinforcement Learning Using Pre-collected Data There is a growing interest in leveraging pre-collected dataset-based training approaches for reinforcement learning, primarily driven by its practical constraints. The new paradigm encompasses offline reinforcement learning [8], imitative learning [26], and imitation learning [27], which leverage pre-collected datasets to build an initial policy. However, these methods suffer from two significant limitations. Firstly, in both training and deployment phases, offline reinforcement learning often suffers from extrapolation errors where the policy samples out-of-distribution actions that are not included in the training dataset [7], [20]. This issue is commonly referred to as the distribution shift problem between the trajectory distribution of the trained policy and the dataset. Recent research attempts to mitigate this issue by introducing conservative or penalizing terms to align the policy's actions more closely with the training dataset, but such constraints may impose performance limitations [28]–[31]. Secondly, the majority of existing studies are confined to synthetic datasets generated by pre-trained policies within online reinforcement learning settings rather than utilizing real-world datasets [13]. In response to these limitations and with the aim of providing a comprehensive benchmark for offline reinforcement learning, this paper assesses the performance of cutting-edge offline reinforcement learning algorithms using both human driver-generated datasets and synthetic datasets.

Value-based Datasets for Autonomous Driving System In the realm of autonomous driving research using reinforcement learning, the predominant approach relies on image-based data for observation [32]–[34]. This approach offers the advantage of an end-to-end pipeline, where neural networks handle the entire process from perception to decision-making. However, this end-to-end methodology poses challenges in terms of interpretability, making it challenging to pinpoint the source of failures. In contrast, some studies opt for a value-based approach, employing sensor data as inputs for policy training rather than relying on images [13], [15], [24], [35]. This approach allows researchers to focus more directly on enhancing the decision-making capabilities of the policy without consideration of image processing ability. This study contributes value-based datasets designed for training driving policies across diverse driving scenarios.

III. Problem Formulations for Autonomous Driving Tasks

This section introduces driving scenarios and datasets for offline reinforcement learning to train an autonomous driving policy. We consider the human driver dataset to capture the fundamental essence of offline reinforcement learning and synthetic datasets generated by online reinforcement learning agents. Subsequently, we look deeper into a unified POMDP that can work across driving scenarios.

A. Driving Scenarios

This subsection aims to extend the FLOW framework's driving scenario to a more realistic level by introducing three complex driving scenarios: 1) highway, 2) lane reduction, and 3) cut-in traffic. These scenarios are carefully designed to reflect real-world road environments more accurately.

Highway Traffic (H): Fig. 1 illustrates the highway traffic scenario. We aim to simulate a more realistic driving environment by incorporating the patterns of the US-101 highway dataset. This includes capturing the diversified characters of vehicles on the road (e.g., desired velocity, safety distance). The goal of the autonomous vehicle is to navigate the complex environment by making adaptive decisions, with the aim of selecting a reliable and optimal path.

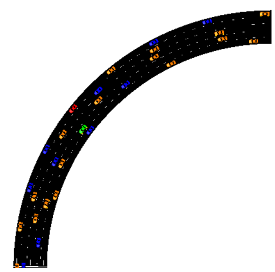

Fig. 1: Highway scenario

Lane Reduction (L): Fig. 2 represents the lane reduction/expansion scenario. Lane reduction road structures involve the conversion of a multi-lane road into a road with fewer lanes. Reducing the number of lanes limits the vehicle capacity of the road, resulting in a bottleneck that concentrates all vehicles in a narrower area and causes traffic congestion. Therefore, drivers are forced to slow down and be more cautious. Additionally, the drivers negotiate with other vehicles to transit through the lane reduction area. The autonomous vehicle aims to find a low-density lane and move while keeping a safe distance.

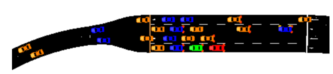

Fig. 2: Lane reduction scenario

Cut-in Scenario (C): In Fig. 3, the cut-in scenario is illustrated, which highlights the autonomous vehicle's capability to overtake other vehicles obstructing the agent's path. If the autonomous vehicle successfully overtakes other slow vehicles and maintains its desired

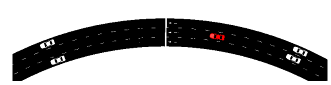

Fig. 3: Cut-in scenario

velocity, it can maximize its rewards. To construct this scenario, we consider two setups: 1) adjusting the autonomous vehicle's desired velocity to be higher than that of the non-autonomous vehicle, and 2) regularly placing non-autonomous vehicles on the road. We expect that the autonomous vehicle will overtake other vehicles to maintain its desired velocity.

To configure realistic traffic in simulation, we have analyzed the US Highway 101 dataset provided by the NGSIM project [18], [19]. We introduce several attributes and properties of this dataset (Fig. 4). **1) Length of road and vehicle:** Fig. 4**A** represents the maximum longitudinal position as approximately 2195.4 feet, and Fig. 4**B** depicts the average vehicle length as approximately 14.6 feet. **2) Target velocities:** We have first confirmed quartile 3 of velocity distribution per vehicle. Fig. 4**D** shows the distribution comprising quartile 3 velocities of all observed vehicles in the dataset. We have selected five values ($\min$, $Q1$, $Q2$, $Q3$, and $\max$) as the target velocities. **3) The number of vehicles:** Fig. 4**C** shows that approximately 117 vehicles have been driving on average per time unit. Subsequently, we distribute the 117 vehicles into five vehicle types, which mean vehicles with five different target velocities (i.e., $\min$, $Q1$, $Q2$, $Q3$, and $\max$ in Fig. 4**D**).

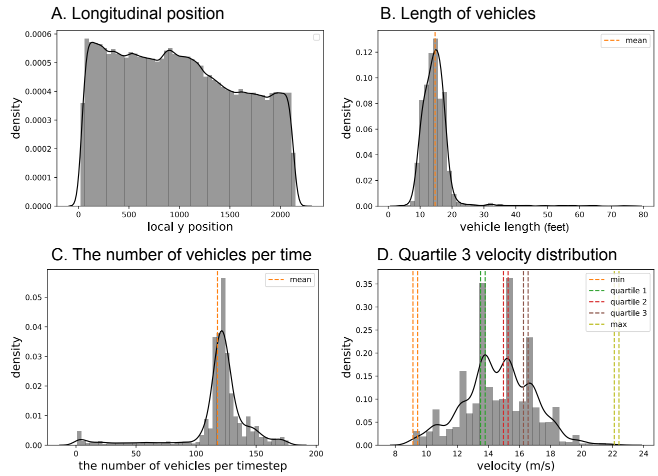

Fig. 4: A: Distribution of all vehicles' longitudinal position. B: Distribution of vehicle length over all driving vehicles in dataset. C: Distribution regarding the number of vehicles per time frame. D: Distribution over quartile 3 velocity of all vehicles.

B. Datasets

This subsection describes the actual and synthetic driving datasets for applying offline reinforcement learning. The number of transitions included in each dataset is approximately one million.

Human Driver Dataset. This research objective is to assess the feasibility and practicability of implementing offline reinforcement learning using an actual driving dataset. We utilize the Highway US-101 dataset, which is selected from the FHWA traffic analysis tool of the NGSIM Project [18], [19]. We use the value-based dataset of NGSIM and pre-process the raw data fitted to the proposed POMDP (e.g., error correction, value normalization, and alignment with POMDP in Section IV-A). Note that this dataset only applies to highway traffic scenarios due to the unique properties of the roads where the data was gathered.

Synthetic Driving Dataset. These datasets were generated by an online reinforcement learning agent using the deep deterministic policy gradient (DDPG) [36] algorithm, which works on continuous action and state spaces. We adopt diverse datasets to comprehend how their quality influences the performance of offline algorithms.

- **Human-like**: It is a synthetic dataset generated by the Intelligent Driver Model (IDM) controller [37] designed to reflect the human driving pattern. This control-theoretic model focuses solely on acceleration control; to incorporate a lane-changing maneuver, we also employ the LC2013 model [38], a manually designed lane-changing controller available within the SUMO simulator.
- **Final**, **Medium**, and **Random**: These datasets are collected by exploiting different policies, which could be obtained at different points in the training phase of online reinforcement learning. When synthesizing the "Final" dataset, we consider the fully pre-trained policy through online reinforcement learning. Subsequently, the "Medium" dataset is gathered using an intermediate policy obtained by an early-stopping method. The "Random" dataset is based on the randomly initialized and unrolled policy in all scenarios.
- **Final-Medium**, and **Final-Random**: These datasets are a blend of two other datasets in equal proportions. The "Final-Medium/Final-Random" dataset is literally combined the "Final" and "Medium/Random" datasets.

IV. Reinforcement Learning for Autonomous Driving System

The realistic reinforcement learning problem is generally formalized as a POMDP $M = \langle \mathcal{S}, \mathcal{O}, \mathcal{A}, r, \mathcal{T}, \Omega, \rho_0, \gamma \rangle$ that includes a state $\mathbf{s} \in \mathcal{S}$, an observation $\mathbf{o} \in \mathcal{O}$, an action $\mathbf{a} \in \mathcal{A}$, a reward $r(\mathbf{s}, \mathbf{a}, \mathbf{s}') \in \mathbb{R}$, a state transition probability $\mathcal{T}(\mathbf{s}'|\mathbf{s}, \mathbf{a})$, an observation probability $\Omega(\mathbf{o}|\mathbf{s})$, an initial state distribution ρ_0, and temporal discount factor $\gamma \in [0, 1)$. The agent aims to maximize the expected discounted cumulative reward $\mathbb{E}_{s_0 \sim \rho_0, \mathbf{o} \sim \Omega(\cdot|\mathbf{s}), \mathbf{a} \sim \pi(\cdot|\mathbf{o}), \mathbf{s}' \sim \mathcal{T}(\cdot|\mathbf{s}, \mathbf{a})} \Big[\sum_t \gamma^t r(\mathbf{s}, \mathbf{a}, \mathbf{s}') \Big]$.

The offline reinforcement learning samples the transitions $(\mathbf{o}, \mathbf{a}, \mathbf{o}', r)$ from the fixed dataset $\mathcal{D}$, thereafter minimizing an estimate of the Bellman error as follows.[2]

$$\mathcal{L}(\theta) = \mathbb{E}_{(\mathbf{o}, \mathbf{a}, \mathbf{o}', r) \sim \mathcal{D}} \Big[Q_\theta(\mathbf{o}, \mathbf{a}) - r + \gamma Q_{\theta'}(\mathbf{o}', \pi(\cdot|\mathbf{o}')) \Big]$$

A. Unified POMDP Model

This subsection presents a unified POMDP structure, which can address the three different scenarios as a decision-making process for autonomous driving. We employ an actor-critic algorithm to train an autonomous vehicle as an agent on continuous observation and action spaces. The agent learns a

[2]In this paper, the superscript '$'$' implies the information on next timestep. For instance, v_n' represents the velocity of the vehicle e_n at the next timestep.

driving policy π that determines optimal behavior based on observable information. To elaborate, the agent enhances its policy by taking into account the reward signal that arises from the observation-action pair $(\mathbf{o}, \mathbf{a})$. The primary goal of the agent is to maximize the accumulated reward.

We define road conditions as two sets: the number of vehicles and the number of lanes. The set $E = \{e_1, e_2, \cdots, e_N\}$ represents the deployed vehicles on the road. It composes the set of autonomous vehicles E_{av} and the set of non-autonomous vehicles E_{non}, i.e., $E = E_{av} \cup E_{non}$. Subsequently, the set $K = \{1, 2, \cdots, L\}$ means the number of lanes configured on the road. Note that the number of lanes in specific segments can be less than the total number of lanes (e.g., lane reduction scenario).

State: The state $\mathbf{s} \in \mathcal{S}$ contains information about all vehicles on the road, as follows:

$$\mathbf{s} = [v_1, p_1, k_1, v_2, p_2, k_2, \cdots, v_N, p_N, k_N]^\top,$$

where v_n, p_n, and k_n represent the velocity, position, and lane number of vehicle e_n, respectively. When the N vehicles exist on the road, the dimension of the state $\mathbf{s} \in \mathbb{R}^{3\times N}$.

Observation: The agent cannot access the complete state information, thereby relying on partial information about the state to make decisions. This constraint arises from the observability limitations of an autonomous vehicle. Specifically, the agent can observe vehicles within a restricted perceivable space $\mathcal{V}$ surrounding them. The perceivable space can be defined as the area that covers both the longitudinal space $\mathcal{V}^{long} \in [V_{\min}^{long}, V_{\max}^{long}]$ (front and behind areas) and the lateral space $\mathcal{V}^{lat} \in [V_{\min}^{lat}, V_{\max}^{lat}]$ (left and right sides). The vehicles in this space are defined as perceivable vehicles $e_i \in E$ of the agent e_n. Let us define a set of perceivable vehicles as E_{sv}. The set E_{sv} can be formulated as $\{e_i | e_i \in E - \{e_n\}, p_i - p_n \leq |\frac{\mathcal{V}^{long}}{2}|, k_i - k_n \leq |\frac{\mathcal{V}^{lat}}{2}|\}$.

Observable vehicles are defined as the closest leading and following vehicles per visible lane among perceivable vehicles. Namely, the maximum number of observable vehicles is $2(2\mathcal{V}_{\max}^{lat} + 1)$, comprising $(2\mathcal{V}_{\max}^{lat} + 1)$ leading and $(2\mathcal{V}_{\max}^{lat} + 1)$ following vehicles: the set of observable leading vehicles $E_L = \{e_{LL\mathcal{V}_{max}^{lat}}, \cdots, e_{LL2}, e_{LL1}, e_{LS}, e_{LR1}, e_{LR2}, \cdots, e_{LR\mathcal{V}_{max}^{lat}}\}$, and the set of observable following vehicles $E_F = \{e_{FL\mathcal{V}_{max}^{lat}}, \cdots, e_{FL2}, e_{FL1}, e_{FS}, e_{FR1}, e_{FR2}, \cdots, e_{FR\mathcal{V}_{max}^{lat}}\}$

Herein, the first subscripts L, F mean the leading and following vehicle; the second subscripts L, S, R represent the left, same, and right lanes. To elaborate, additional subscripts of L, R mean the lane gap based on the same lane S, i.e., $L1$ and $L\mathcal{V}_{max}^{lat}$ are the nearest and farthest left lane, respectively.

The observation $\mathbf{o} \in \mathcal{O}$ of the agent e_n is defined as follows.

$$\mathbf{o} = [v_n, \Delta\mathbf{v}^\top, \Delta\mathbf{p}^\top, \boldsymbol{\rho}^\top, \boldsymbol{\zeta}^\top]^\top \tag{1}$$

In (1), $\Delta\mathbf{v}$ denotes a vector of relative velocities between the agent and observable vehicles, $\Delta\mathbf{p}$ means a vector of relative distances between the agent and observable vehicles, $\boldsymbol{\rho} = [\rho_{L\mathcal{V}_{max}^{lat}}, \cdots, \rho_S, \cdots, \rho_{R\mathcal{V}_{max}^{lat}}]^\top$ represents a vector of lane traffic density, and $\boldsymbol{\zeta} = [\zeta_{L\mathcal{V}_{max}^{lat}}, \cdots, \zeta_S, \cdots, \zeta_{R\mathcal{V}_{max}^{lat}}]^\top$ depicts a vector of existence of lanes beyond the longitudinal perceivable space.

Action: The vector of action $\mathbf{a} \in \mathcal{A}$ of the agent comprises acceleration and lane-changing, i.e., $\mathbf{a} = [a^{acc}, a^{lc}]^\top$. The acceleration behavior a^{acc} is decided in continuous action space $[A_{\min}, A_{\max}]$, range between maximum deceleration and acceleration. Next, the lane-changing maneuver a^{lc} is controlled in discrete action space $\{-1, 0, 1\}$. Herein, $a^{lc} = -1$ and $a^{lc} = 1$ indicate that the agent decides to move to the right and left lane, respectively; $a^{lc} = 0$ represents keeping a lane.

Reward: The agent executes the action $\mathbf{a}$ in a given state $\mathbf{s}$, thereby receiving a reward r. The reward is determined by a reward function, which is expressed as a linear combination of five reward components, i.e.,

$$r = R(\mathbf{s}, \mathbf{a}, \mathbf{s}') = \eta_1\mathcal{R}_1 + \eta_2\mathcal{R}_2 + \eta_3\mathcal{R}_3 + \eta_4\mathcal{R}_4 + \eta_5\mathcal{R}_5 + C.$$

The non-negative coefficient $\eta_n \geq 0$ refers to the weight assigned to the n-th reward component $\mathcal{R}_n$, and C represents a scaling constant.

The first reward component $\mathcal{R}_1$ is designed to keep the speed near the desired speed v^* without overspeeding the speed limit v_{limit}, which always satisfies $v_{\text{limit}} - v^* > 0$.

$$\mathcal{R}_1 = \begin{cases} \frac{v_n'}{v^*} & v_n' \leq v^* \\ \frac{v_{\text{limit}} - v_n'}{v_{\text{limit}} - v^*} & v_n' > v^* \end{cases}$$

When $0 \leq v_n' \leq v_{\text{limit}}$, $\mathcal{R}_1$ becomes positive value, and if $v_n' = v^*$, it is maximized (i.e., $\mathcal{R}_1 = 1$); when $v_n' > v_{\text{limit}}$, $\mathcal{R}_1$ becomes negative value as penalty.

The second reward component $\mathcal{R}_2$ induces the agent to perform lane-changing that frees up driving space.

$$\mathcal{R}_2 = |a^{lc}|(\Delta p_{SL}' - \Delta p_{SL})$$

This component is activated when the agent performs lane-changing (e.g., $|a^{lc}| = 1$). In other words, if $|a^{lc}| = 0$, then $\mathcal{R}_2 = 0$. Subsequently, if driving space is secured after changing lanes (i.e., $\Delta p_{SL}' \geq \Delta p_{SL}$), and it means desirable action, resulting in $\mathcal{R}_2 > 0$. On the other hand, if the agent fails to secure the driving space after changing lanes (i.e., $\Delta p_{SL}' < \Delta p_{SL}$), thereby receiving a penalty $\mathcal{R}_2 < 0$.

The third and fourth reward components are related to the safe distance between the leading and following vehicles, respectively. These components prevent the violation of the safe distance s^*.

The third reward component is defined as follows:

$$\mathcal{R}_3 = \min\left[0, 1 - \left(\frac{s_{LS}^*}{\Delta p_{LS}'}\right)^2\right], \tag{2}$$

where s_{LS}^* means that the safe distance between the agent e_n and e_{LS} and is defined as follows [37].

$$s_{LS}^* = s_0 + \max\left[0, v_n'\left(t^* + \frac{\Delta v_{LS}'}{2\sqrt{|A_{min} \times A_{max}|}}\right)\right]$$

Herein, s_0 indicates the minimum safe distance between vehicles, and t^* means the minimum time headway, which is

TABLE I: Average performance (± confidence interval with two standard deviations) on driving scenarios and datasets. The scores are based on 10 evaluations with five random seeds. Cyan and red highlight boxes depict the best score in each dataset and each scenario, respectively.

Task Name	**BC**	**Imitative** [26]	**BCQ** [28]	**CQL** [30]	**IQL** [31]	**EDAC** [39]	**PLAS** [40]
`highway-US101-NGSIM`	1202.07 ± 216.79	1446.87 ± 242.17	1501.99 ± 101.02	1260.59 ± 134.1	1261.58 ± 155.81	1253.35 ± 16.46	1468.11 ± 25.89
`highway-final`	1565.64 ± 28.03	821.77 ± 438.46	1563.26 ± 33.76	1473.31 ± 162.78	1537.80 ± 28.13	1219.17 ± 59.66	1551.31 ± 24.56
`highway-medium`	1393.42 ± 52.39	419.57 ± 361.79	1402.91 ± 30.69	1377.20 ± 73.71	1423.18 ± 27.39	1244.65 ± 60.38	1407.75 ± 30.69
`highway-random`	622.15 ± 234.29	-2.12 ± 13.05	884.28 ± 93.03	-10.42 ± 14.06	433.69 ± 49.06	1266.66 ± 65.83	1543.02± 104.65
`highway-final-medium`	1543.33 ± 49.49	20.87 ± 291.16	1450.23 ± 65.49	1434.55 ± 144.34	1615.46 ± 25.18	1181.51 ± 53.19	1431.30 ± 20.94
`highway-final-random`	963.19 ± 218.79	25.90 ± 83.13	659.38 ± 76.95	240.73 ± 30.09	835.75 ± 335	1261.58 ± 38.62	1569.594 ± 65.28
`highway-human-like`	980.80 ± 477.69	123.02 ± 339.94	797.98 ± 226.53	766.19 ± 250.68	1146.09 ± 270.23	1245.43 ± 4.25	771.11 ± 180.78
`lanereduction-final`	968.46 ± 36.73	967.75 ± 67.39	1266.84 ± 240.33	984.36 ± 151.21	1248.22 ± 235.20	268.79 ± 18.89	965.67 ± 30.83
`lanereduction-medium`	311.11 ± 82.90	303.06 ± 33.04	1260.16 ± 280.21	344.17 ± 45.61	1319.43 ± 190.68	271.62 ± 18.74	387.01 ± 31.95
`lanereduction-random`	106.99± 54.20	139.36 ± 14.19	31.05 ± 3.73	3.18 ± 1.82	312.86 ± 101.68	269.61 ± 23.16	1088.64 ± 45.48
`lanereduction-final-medium`	502.11 ± 44.96	790.79 ± 120.77	1162.62 ± 276.56	554.56 ± 83.02	1257.35± 175.99	278.61 ± 21.85	366.21 ± 35.64
`lanereduction-final-random`	157.98 ± 54.20	791.42 ± 133.52	9.82 ± 18.74	66.40 ± 12.58	157.27 ± 119.49	312.99 ± 24.59	604.6 ± 86.55
`lanereduction-human-like`	1160.63 ± 91.30	960.40 ± 154.79	1449.62 ± 176.54	470.34 ± 71.13	1443.84 ± 175.25	166.88 ± 142.23	1228.03 ± 8.66
`cutin-final`	1711.55 ± 159.27	953.72 ± 74.21	1155.46 ± 215.64	1012.24 ± 180.62	1737.22 ± 244.82	946.1732 ± 19.3026	1110.21 ± 5.94
`cutin-medium`	1634.74 ± 79.01	971.67 ± 93.78	1607.19 ± 80.29	1201.13 ± 190.26	1513.23 ± 88.41	946.1717 ± 19.3038	1596.30± 24.12
`cutin-Random`	972.58 ± 147.16	1243.71 ± 257.91	1101.06 ± 194.16	638.37 ± 106.86	756.27 ± 128.54	946.1699 ± 19.3031	1477.45 ± 333.78
`cutin-final-medium`	1084.62 ± 135.39	866.66 ± 66.61	1144.52 ± 196.31	1085.40 ± 128.06	1446.98 ± 143.01	946.1731 ± 19.3008	531.35± 41.69
`cutin-final-random`	599.56 ± 136.57	1135.27 ± 283.30	1425.52 ± 310.94	1101.67 ± 188.81	798.89 ± 119.34	946.1719 ± 19.3014	1655.24 ± 95.89
`cutin-human-like`	863.22 ± 166.86	724.22 ± 113.87	1391.40 ± 250.34	1082.06 ± 178.75	1105.40 ± 290.50	946.1704 ± 19.3011	1354.08 ± 19.02

the shortest time that a following vehicle can achieve without reducing velocity. This component induces that the agent maintains the safe distance s^*_{LS} from the leading vehicle in the same lane. Specifically, if $\Delta p'_{LS} < s^*_{LS}$, then $\mathcal{R}_3$ in (2) becomes the negative value; otherwise, $\mathcal{R}_3$ is not activated (i.e., $\mathcal{R}_3 = 0$).

The fourth reward component is as follows:

$$\mathcal{R}_4 = |a^{lc}| \min \left[0, 1 - \left(\frac{s^*_{FS}}{\Delta p'_{FS}}\right)^2\right]. \quad (3)$$

In (3), s^*_{FS} denotes that the safe distance between the agent e_n and e_{FS} and is defined as follows [37].

$$s^*_{FS} = s_0 + \max\left[0, v'_{FS}\left(t^* + \frac{\Delta v'_{FS}}{2\sqrt{|A_{min} \times A_{max}|}}\right)\right]$$

The same as (2), $\mathcal{R}_4 \leq 0$ is always satisfied. In (3), $\mathcal{R}_4$ can be non-zero only if $|a^{lc}| \neq 0$, which indicates that the agent changes the lanes. In contrast to $\mathcal{R}_3$, $\mathcal{R}_4$ switches on only when the agent changes the lane, as maintaining a safe distance while staying in the lane is contingent on the following vehicle. Specifically, if $\Delta p'_{FS} < s^*_{FS}$ and $|a^{lc}| \neq 0$, then $\mathcal{R}_4 < 0$ is satisfied.

Finally, the fifth reward component $\mathcal{R}_5$ is related to the accident (e.g., inter-vehicle crash, changing the nonexistent lanes) and is defined as follows.

$$\mathcal{R}_5 = \begin{cases} 0 & \text{accident not happened} \\ -1 & \text{accident happened} \end{cases} \quad (4)$$

The agent receives the penalty when the agent's action contributes to the accident (i.e., if the agent cannot continue driving). Typically, the fifth balancing weight η_5 is assigned the highest value among coefficients $\eta_1, \cdots, \eta_5$.

Regardless of the driving scenario, the agent takes the decision-making process using the proposed POMDP. Note that the proposed POMDP is designed for safe and efficient driving without considering the routing of the destination.

V. Benchmarking Baseline Performances

This section provides the simulation results across all driving scenarios and datasets. To provide the performance baseline for autonomous driving scenarios, we performed extensive experiments and evaluated the performance of state-of-the-art offline reinforcement learning algorithms, including Behavioral Cloning (BC), imitative learning (DDPG + BC) [26], batch constrained Q (BCQ) [28], conservative Q learning (CQL) [30], implicit Q learning (IQL) [31], ensemble-diversified actor-critic (EDAC) [39], and policy in the latent action space (PLAS) [40].[3] As discussed in section III-B, we utilize the DDPG as the online reinforcement learning algorithm.[4]

A. Metrics

We use three evaluation metrics to verify the algorithm's and dataset's performance per driving scenario.

1) **Normalized Score**: It measures the effectiveness of offline reinforcement learning algorithms when the expert online agent's performance $\text{score}_{\text{final}}$ is set as a baseline. The normalized score is calculated as

$$\text{normalized score} = \frac{(\text{score} - \text{score}_{\text{random}})}{(\text{score}_{\text{final}} - \text{score}_{\text{random}})},$$

where $\text{score}_{\text{random}}$ represents the score of a random policy.

[3]Note that all algorithms cannot aim to work on hybrid action space but well work on discrete action space through simple quantization.

[4]**Why not use the TD3?** We utilize the negative reward trick to prevent adverse effects on the approximation process when considering negative and positive reward signals. In such a setup, TD3 [41], a higher version of DDPG, can lead to empirically overestimating the absolute Q-value. Therefore, we employ the DDPG.

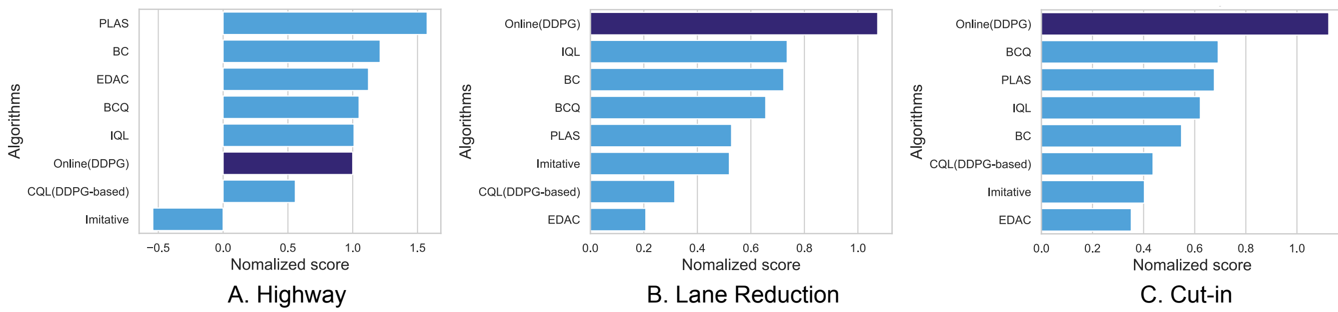

Fig. 5: Normalized score for each algorithm per driving scenario. The average performance across synthetic datasets (except for `Human-like`) is provided to mitigate the impact of the dataset and to highlight the impact of the algorithm.

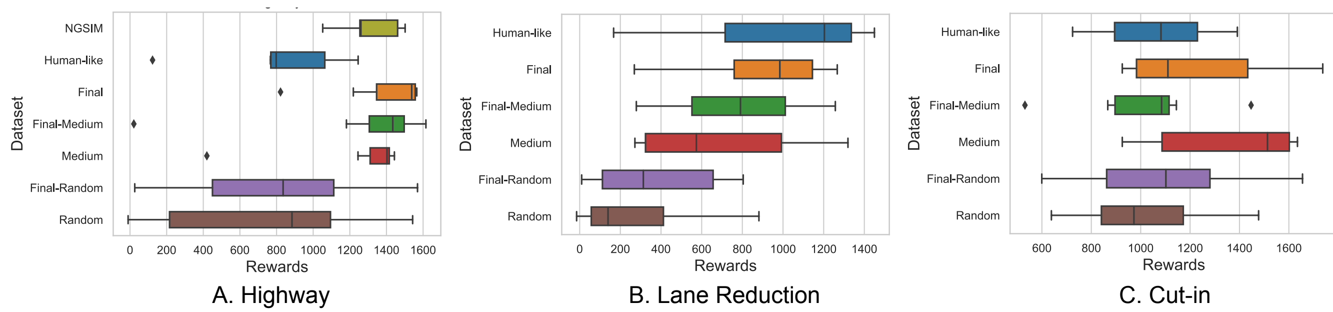

Fig. 6: The IQR range of performance for each dataset per driving scenario. To mitigate the impact of the algorithms and to highlight the impact of the dataset, the IQR range includes the results of all algorithms. The diamonds indicate outliers, and the boxes represent the interquartile range. Herein, the start, middle, and end of the box indicate the quartile 1, 2, and 3, respectively.

2) **Inter-quartile Range (IQR)**: We use the IQR to display the performance range for each dataset [42]. The IQR mitigates the impact of outliers, so it can be a more robust and statistically efficient measure than the median or mean.

B. Simulation Results

The results offer the following advantages: 1) facilitating autonomous driving research by exploring offline reinforcement learning possibilities; 2) discussing the usability of the actual dataset in offline reinforcement learning. All simulation results are presented in Table I. The presented score is the average normalized score over five random seeds. Each policy is evaluated by averaging the performance over ten executions.

Performance over algorithms: Fig. 5 shows the average normalized score per driving scenario and algorithm. This evaluation result presents the average performance across datasets. We comprehend that online reinforcement learning performance generally outperforms offline reinforcement learning performance. It is intuitive because online reinforcement learning can guarantee higher performance if there is enough exploration period.

Performance over dataset: Fig. 6 presents the IQR of the evaluation score per driving scenario and dataset. This evaluation results include the performance of all algorithms. In Fig. 6**B-C**, the results empirically imply two interesting insights: 1) the performance of NGSIM and Human-like datasets are comparable to synthetic datasets (close to Final and Medium for the most part), and 2) The performance ranks of the synthetic datasets is fair-minded (Final $>$ Final-Medium $>$ Medium $>$ Final-Random $>$ Random). On the other hand, In Fig. 6**C**, the average performance of the dataset is far from expected. The Final dataset contains samples with the highest performance, but the overall performance is highest in the Medium.

VI. Conclusion

This study presents an autonomous driving framework based on offline reinforcement learning, accompanied by benchmark performances and datasets that are readily accessible and reproducible. The driving scenarios within the FLOW framework have been expanded to include three realistic road structures: Cut-in, Lane Reduction, and Highway. A unified POMDP has been developed, applicable to all driving scenarios. The contribution of this work extends beyond providing synthetic datasets obtained from online reinforcement learning for each driving scenario, as it also includes pre-processed real-world driving datasets, such as the NGSIM dataset, aligned with the proposed POMDP. As a result, interesting insights have been obtained through the analysis of the results.

In summary, the primary aim of this study is to utilize pre-collected datasets to facilitate research in the field of autonomous driving with offline reinforcement learning. We expect to accelerate progress in this domain and open up new avenues for further exploration.

References

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[2] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M. P. Reyes, M. Shyu, S. Chen, and S. Iyengar, "A survey on deep learning: Algorithms, techniques, and applications," *ACM Computing Surveys*, vol. 51, no. 5, pp. 1–36, 2018.

[3] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[5] R. Sutton and A. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[6] H. Niu, Y. Qiu, M. Li, G. Zhou, J. HU, X. Zhan, *et al.*, "When to trust your simulator: Dynamics-aware hybrid offline-and-online reinforcement learning," in *Neural inf. process. syst.*, vol. 35, 2022, pp. 36 599–36 612.

[7] S. Levine, A. Kumar, G. Tucker, and J. Fu, "Offline reinforcement learning: Tutorial, review, and perspectives on open problems," *arXiv preprint arXiv:2005.01643*, 2020.

[8] S. Lange, T. Gabel, and M. Riedmiller, "Batch reinforcement learning," *Reinforcement learning*, pp. 45–73, 2012.

[9] X. Fang, Q. Zhang, Y. Gao, and D. Zhao, "Offline reinforcement learning for autonomous driving with real world driving data," in *IEEE Intell. Transp. Syst. Conf.*, 2022, pp. 3417–3422.

[10] X. Liang, T. Wang, L. Yang, and E. Xing, "CIRL: Controllable imitative reinforcement learning for vision-based self-driving," in *Eur. Conf. on Comput. Vision*, 2018, pp. 584–599.

[11] A. Kumar, A. Singh, S. Tian, C. Finn, and S. Levine, "A workflow for offline model-free robotic reinforcement learning," in *Conf. on Robot Learn.*, 2022, pp. 417–428.

[12] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. Al Sallab, S. Yogamani, and P. Pérez, "Deep reinforcement learning for autonomous driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4909–4926, 2021.

[13] J. Fu, A. Kumar, O. Nachum, G. Tucker, and S. Levine, "D4RL: Datasets for deep data-driven reinforcement learning," *arXiv preprint arXiv:2004.07219*, 2020.

[14] T. Shi, D. Chen, K. Chen, and Z. Li, "Offline reinforcement learning for autonomous driving with safety and exploration enhancement," *arXiv preprint arXiv:2110.07067*, 2021.

[15] E. Vinitsky, A. Kreidieh, L. Le Flem, N. Kheterpal, K. Jang, C. Wu, F. Wu, R. Liaw, E. Liang, and A. M. Bayen, "Benchmarks for reinforcement learning in mixed-autonomy traffic," in *Conf. on Robot Learn.*, 2018, pp. 399–409.

[16] A. R. Kreidieh, C. Wu, and A. M. Bayen, "Dissipating stop-and-go waves in closed and open networks via deep reinforcement learning," in *IEEE Intell. Transp. Syst. Conf.*, 2018, pp. 1475–1480.

[17] C. Gong, Z. Yang, Y. Bai, J. He, J. Shi, A. Sinha, B. Xu, X. Hou, G. Fan, and D. Lo, "Mind your data! Hiding backdoors in offline reinforcement learning datasets," *arXiv preprint arXiv:2210.04688*, 2022.

[18] U. S. Department of Transportation Federal Highway Administration, "Next generation simulation (NGSIM) program US-101 videos. [Dataset]. Provided by ITS DataHub through data.transportation.gov," 2016, Accessed 2022-06-05 from http://doi.org/10.21949/1504477.

[19] U. D. of Transportation, "NGSIM—next generation simulation," 2008.

[20] R. Prudencio, M. Maximo, and E. Colombini, "A survey on offline reinforcement learning: Taxonomy, review, and open problems," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[21] D. Troullinos, G. Chalkiadakis, I. Papamichail, and M. Papageorgiou, "Collaborative multiagent decision making for lane-free autonomous driving," in *Int. Conf. on Auton. Agents and Multi-agent Syst.*, 2021, pp. 1335–1343.

[22] M. Stryszowski, S. Longo, E. Velenis, and G. Forostovsky, "A framework for self-enforced interaction between connected vehicles: Intersection negotiation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 6716–6725, 2020.

[23] K. Jang, E. Vinitsky, B. Chalaki, B. Remer, L. Beaver, A. A. Malikopoulos, and A. Bayen, "Simulation to scaled city: zero-shot policy transfer for traffic control via autonomous vehicles," in *Int. Conf. on Cyber-Physical Syst.*, 2019, pp. 291–300.

[24] C. Wu, A. Kreidieh, K. Parvate, E. Vinitsky, and A. Bayen, "FLOW: A modular learning framework for mixed autonomy traffic," *IEEE Transactions on Robotics*, vol. 38, no. 2, pp. 1270–1286, 2021.

[25] Q. Li, Z. Peng, L. Feng, Q. Zhang, Z. Xue, and B. Zhou, "Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[26] S. Fujimoto and S. Gu, "A minimalist approach to offline reinforcement learning," in *Neural inf. process. syst.*, vol. 34, 2021, pp. 20 132–20 145.

[27] T. Osa, J. Pajarinen, G. Neumann, A. Bagnell, P. Abbeel, J. Peters, *et al.*, "An algorithmic perspective on imitation learning," *Foundations and Trends in Robotics*, vol. 7, no. 1-2, pp. 1–179, 2018.

[28] S. Fujimoto, D. Meger, and D. Precup, "Off-policy deep reinforcement learning without exploration," in *Int. Conf. on Mach. Learn.*, 2019, pp. 2052–2062.

[29] A. Kumar, J. Fu, M. Soh, G. Tucker, and S. Levine, "Stabilizing off-policy Q-learning via bootstrapping error reduction," in *Neural inf. process. syst.*, vol. 32, 2019.

[30] A. Kumar, A. Zhou, G. Tucker, and S. Levine, "Conservative Q-learning for offline reinforcement learning," in *Neural inf. process. syst.*, vol. 33, 2020, pp. 1179–1191.

[31] I. Kostrikov, A. Nair, and S. Levine, "Offline reinforcement learning with implicit Q-learning," in *Int. Conf. on Learn. Representations*, 2022.

[32] R. McAllister, B. Wulfe, J. Mercat, L. Ellis, S. Levine, and A. Gaidon, "Control-aware prediction objectives for autonomous driving," in *IEEE Int. Conf. on Robot. and Automat.*, 2022, pp. 01–08.

[33] S. Pini, C. Perone, A. Ahuja, A. Ferreira, M. Niendorf, and S. Zagoruyko, "Safe real-world autonomous driving by learning to predict and plan with a mixture of experts," in *IEEE Int. Conf. on Robot. and Automat.*, 2023, pp. 10 069–10 075.

[34] H. Chiu, J. Li, R. Ambruş, and J. Bohg, "Probabilistic 3D multi-modal, multi-object tracking for autonomous driving," in *IEEE Int. Conf. on Robot. and Automat.*, 2021, pp. 14 227–14 233.

[35] N. Kheterpal, E. Vinitsky, C. Wu, A. Kreidieh, K. Jang, K. Parvate, and A. Bayen, "FLOW: Open source reinforcement learning for traffic control," in *Neural inf. process. syst.*, 2018.

[36] T. Lillicrap, J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *Int. Conf. on Learn. Representations*, 2016.

[37] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, p. 1805, 2000.

[38] J. Erdmann, "SUMO's lane-changing model," *Modeling Mobility with Open Data*, pp. 105–123, 2015.

[39] G. An, S. Moon, J.-H. Kim, and H. O. Song, "Uncertainty-based offline reinforcement learning with diversified Q-ensemble," in *Neural inf. process. syst.*, vol. 34, 2021, pp. 7436–7447.

[40] W. Zhou, S. Bajracharya, and D. Held, "PLAS: Latent action space for offline reinforcement learning," in *Conf. on Robot Learn.*, 2021, pp. 1719–1735.

[41] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *Int. Conf. on Mach. Learn.*, 2018, pp. 1587–1596.

[42] R. Agarwal, M. Schwarzer, P. S. Castro, A. Courville, and M. Bellemare, "Deep reinforcement learning at the edge of the statistical precipice," in *Neural inf. process. syst.*, vol. 34, 2021, pp. 29 304–29 320.