A Discrete Soft Actor-Critic Decision-Making Strategy with Sample Filter for Freeway Autonomous Driving

Jiayi Guan[1], Guang Chen[1,2*], Jin Huang[3], Zhijun Li[4], Lu Xiong[1], Jing Hou[1], Alois Knoll[2]

***Abstract*—Autonomous driving is a promising technology to reduce traffic accidents and improve driving efficiency. Although significant progress has been achieved, existing decision-making systems of autonomous vehicle still cannot meet the safety and driving efficiency requirements in highly dynamic environments. In this work, we design a discrete decision-making strategy based on the discrete soft actor-critic with sample filter algorithm (DSAC-SF) to improve driving efficiency and safety on freeways with dynamics traffic. Specifically, we first propose a sample filter method for discrete soft actor-critic, which improves the sample efficiency and stability of the algorithm via enhancing the utilization of effective samples. Subsequently, we construct the discrete decision-making strategy for autonomous driving based on the DSAC-SF algorithm, and further design the area observation method and the multi-objective reward function to improve the driving safety and efficiency. Finally, we carry out comparison and ablation experiments on the the scalable multi-agent reinforcement learning training school (SMARTS) simulation environment. Experimental results indicate that our strategy obtains a high success rate and a fast vehicle speed in the decision-making tasks on freeways. Moreover, our DSAC-SF algorithm also achieves improved performance in training efficiency and stability compared to the commonly used discrete reinforcement learning algorithm.**

***Index Terms*—Discrete decision-making, autonomous driving, deep reinforcement learning, soft actor-critic**

I. Introduction

THE revolution of modern vehicles brings us not only great convenience but also traffic accidents. Statistics show that about 90% of traffic accidents are caused by driver's improper operation [1]. Autonomous driving is a promising solution to reduce traffic accidents and improve traffic efficiency [2–5]. As the brain of autonomous driving, decision-making is critical to the study of autonomous driving technologies [6–8]. Although some significant achievements [9–12] have been obtained in decision-making on various scenarios in recent years, there is still a long way to go for a safe and efficient decision-making strategy for autonomous driving.

The sequential decision properties of reinforcement learning (RL) algorithms are consistent with the decision-making process of autonomous driving. Therefore RL has been widely studied in the decision-making task of autonomous driving [13]. However, there are still many deficiencies in the previous research on autonomous driving decision-making tasks based on RL. In particular, to explore the optimal action sequence, randomness needs to be added to the policy exploration process [14, 15]. This results in the collected transitions containing a mass number of redundant cases, such as vehicles crashing into the same place consecutively. These duplicated redundant samples reduce the sample efficiency and stability of the algorithm. The priority experience replay strategy improves the stability of the algorithm and the utilization of favorable samples via selective replay experience [16, 17], but this setting tends to cause the algorithm to converge to a local optimum. In addition, The value distribution RL [18, 19] replaces the maximization of the Q-value via maximizing the Q-value distribution function, which effectively eliminates the local optimum caused by the Q-value estimation, however it does not eliminate the problem of sample inefficiency caused by repeated redundant samples. The above algorithms improve the performance of the algorithm via various settings, but they do not solve the problems of sample inefficiency and algorithm instability caused by repeated redundant samples.

To solve the problem of sample inefficiency and algorithm instability caused by repeated redundant samples, we propose a discrete soft actor-critic strategy with sample filter algorithm (DSAC-SF). Furthermore, we design a discrete decision-making strategy for autonomous driving based on DSAC-SF to improve driving efficiency and safety. The main contributions of this work are listed as follows:

- We establish two sample pools to store the current episode and training samples respectively. Based on this, we propose a discrete soft actor-critic with sample filter algorithm to eliminate the problem of sample inefficiency caused by repeated redundant samples.
- We propose an area observation method to improve the accuracy of traffic information observation via dividing multiple areas to extract and automatically supplement default information.
- We design a decision-making strategy based on the DSAC-SF, and extensive simulation experiments indicate that our strategy can provide competitive performance under both sparse and dense traffic flows.

II. Preliminaries and Problem Formulation

Before we introduce the decision-making strategy based on DRL, we first briefly describe the reinforcement learning

* Corresponding author: Guang Chen, Email: guangchen@tongji.edu.cn

Author's Affiliation: [1]Tongji University, Shanghai, China; [2]Chair of Robotics, Artificial Intelligence and Real-time Systems, Technical University of Munich, Munich, Germany; [3]School of Vehicle and Mobility, Tsinghua University, Beijing, China; [4] University of Science and Technology of China, Hefei, China.

and decision-making of autonomous driving, and clarify the problem that this study needs to address in this section.

Reinforcement Learning. We consider the environment formulation as a Markov Decision Process (MDP) to be defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ denotes the state transition probability, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ represents the reward function, and $\gamma \in (0, 1]$ is the discount factor, A policy π is a mapping $\mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$. The objective of standard RL is to maximize the expected sum of reward $G_t = \max \sum_{i=0}^{\infty} \gamma^i r(s_{t+i}, a_{t+i})$. This objective can be modified to maximize the entropy $\alpha \mathcal{H}(\pi(\cdot|s_t)) = -\alpha log \pi(\cdot|s_t)$ by augmenting the expected entropy of policy over τ_π:

$$\pi^* = \arg\max_{\pi} \sum_{i=0}^{T} E_{(s_t, a_t) \sim \tau_\pi} \left[\gamma^i \Big(r(s_{t+i}, a_{t+i}) - \alpha log \pi(\cdot|s_t) \Big) \right] \tag{1}$$

where α is the temperature parameter that determines the relative importance of the entropy term versus the reward.

The advantage of the maximum entropy is that it can encourage a wider exploration [20], capture a variety of near-optimal behavior with equally attractive, and meanwhile give up promising avenues. Therefore, the performance of maximum entropy RL is often more robust and better than traditional RL methods.

Decision-making of autonomous driving. The decision-making of autonomous driving is to make reasonable decisions based on the traffic environment information within a certain range around the ego vehicle. The mathematical formula can be written as:

$$d = f(o_1, \cdots, o_i) \tag{2}$$

where d is the decision order, o_i is the observed traffic environment information, including vehicle speed, pose, road curvature, etc.

The sequential decision properties of RL are consistent with the decision-making process of autonomous driving. Therefore, RL is widely adopted for the decision-making task of autonomous driving. However, previous studies over-simplified the scenario in the process of extracting observation information, and they seldom considered the problem of default information supplementation for special scenarios [21, 22]. Furthermore, prior RL algorithms are sample inefficient. Therefore, this work aims to provide a RL algorithm with high sample efficiency, and design a discrete decision-making strategy based on the algorithm to improve the driving efficiency and safety of the decision-making model.

III. DRL-BASED DECISION-MAKING STRATEGY

In this section, the DSAC-SF algorithm is proposed. Furthermore, we present the discrete decision-making strategy based on the proposed algorithm for autonomous driving.

A. Discrete Soft Actor Critic with Sample Filter Algorithm

1) Sample Filter Method.: To explore optimal action sequences, randomness needs to be added to the policy exploration process. This results in the sampled transitions containing a mass of redundant failure cases, such as the vehicle crashing into the same place consecutively. Redundant failure samples can easily reduce the convergence speed and stability of the strategy. Oversampling of high value instances has been shown to accelerate training in a variety of gym and Atari video game environment. This Conclusion can be calculated by the Bellman equation of the Q value:

$$\begin{aligned} Q(s_t, a_t) = \sum_{s_{t+1} \sim p} p(s_{t+1}|s_t, a_t) \Big[r(s_t, a_t, s_{t+1}) \\ + \gamma \sum_{a_{t+1} \sim \pi} \pi(a_{t+1}|s_{t+1}) Q(s_{t+1}, a_{t+1}) \Big] \end{aligned} \tag{3}$$

We assume $\sum_{s_{t+1} \sim p} p(s_{t+1}|s_t, a_t) = \sum_{s_{t+1} \sim p^h} p^h(s_{t+1}|s_t, a_t) + \sum_{s_{t+1} \sim p^l} p^l(s_{t+1}|s_t, a_t)$, then Equation. 3 can be written as:

$$\begin{aligned} Q(s_t, a_t) = \sum_{s_{t+1} \sim p^h} \Big[p^h(s_{t+1}|s_t, a_t) r^h(s_t, a_t, s_{t+1}) \Big] + \\ \sum_{s_{t+1} \sim p^l} \Big[p^l(s_{t+1}|s_t, a_t) r^l(s_t, a_t, s_{t+1}) \Big] + \gamma \sum_{\substack{s_{t+1} \sim p \\ a_{t+1} \sim \pi}} Q(s_{t+1}, a_{t+1}) \end{aligned} \tag{4}$$

where $p^h(s_{t+1}|s_t, a_t)$ is the transition probability corresponding to a high reward of action reward value $r^h(s_t, a_t, s_{t+1})$, and $p^l(s_{t+1}|s_t, a_t)$ is the transition probability corresponding to a low-value reward of action reward value $r^l(s_t, a_t, s_{t+1})$. Equation. 4 shows that increasing the sampling probability $p^h(s_{t+1}|s_t, a_t)$ of high value instances can improve the Q-value estimate for the dominant action, thereby accelerate strategy training. However, unbalanced sampling can easily lead to network over-fitting. Therefore, in the process of improving the sampling of high-value instances, it is necessary to ensure that low-value transitions are sampled.

Algorithm 1 Discrete Soft Actor-Critic with Sample Filter

Initialize $\theta_{m,1}$, $\theta_{m,2}$, ϕ, $\theta_{t,1}$,$\theta_{t,2}$,α
Initialize D_{te}, D_{tr}
for each episode **do**
  for each environment step **do**
    $a_t \leftarrow \pi_\phi(a_t|s_t)$; $s_{t+1} \leftarrow p(\cdot|s_t, a_t)$; $r_t \leftarrow r(a_t, s_t)$
    $D_{tr} \leftarrow D_{tr} \cup \{s_t, a_t, r_t, s_{t+1}\}$
  end for
  if $((m\%N_f == 0) \cup (m < n_f)) \cap (Equation\ 5)$ **then**
    for $j \in$ environment step **do**
      if $i > D_{n,\max}$ **then**
        $i \leftarrow i\%D_{n,\max}$
      end if
      $D_{te}[i] \leftarrow D_{tr}[j]$; $i \leftarrow i + 1$
    end for
  end if
  for each gradiant step **do**
    $J_Q(\theta) \leftarrow E_{D \sim D_{te}} \left[\left(Q_{\theta_{m,i}}(s_t, a_t) - \hat{Q}(s_t, a_t) \right)^2 \right]$
    $J_\pi(\phi) \leftarrow \pi_\phi(\cdot|s_t)^T \left[\alpha \log(\pi_\phi(\cdot|s_t)) - Q_{\theta_{m,i}}(s_t, a_t) \right]$
    $J(\alpha) \leftarrow \pi(\cdot|s_t)^T \left[-\alpha \left(\log(\pi_t(\cdot|s_t)) + \overline{\mathcal{H}} \right) \right]$
    $\theta_{m,i} \leftarrow \theta_{m,i} - \lambda_Q \hat{\nabla}_{\theta_{m,i}} J(\theta_{m,i}); i \in \{1, 2\}$
    $\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi J_\pi(\phi)$
    $\alpha \leftarrow \alpha - \lambda \hat{\nabla}_\phi J_\pi(\phi)$
    $\hat{Q}_{t,i} \leftarrow \tau Q_{m,i} + (1 - \tau) \hat{Q}_{t,i}; i \in \{1, 2\}$
  end for
end for

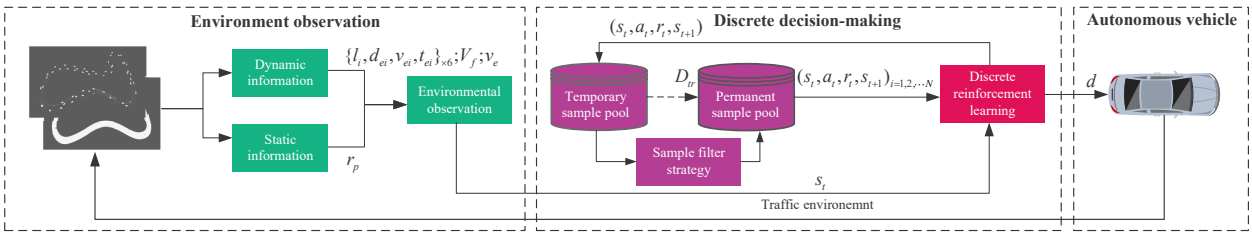


Fig. 1. The system framework of discrete decision-making strategy for autonomous driving. The system includes three modules: environment observation module, discrete decision-making model module and intelligent vehicle module.

According to the above analysis, we propose a sample filter method to improve the utilization of effective samples for DRL-based decision-making strategy, thereby improving the learning efficiency and stability of the model. Two sample pools are constructed, and then a sample filter strategy is established in this method. Specifically, a temporary sample pool and a permanent sample pool are built, with the temporary sample pool is used to collect samples of the current episode, and the permanent sample pool is used to store training samples for the DRL model. Furthermore, the sample filter strategy is used to determine whether the current episode sample can be added to the permanent sample pool.

As shown in Algorithm 1, we store the collected samples $\{s_t, a_t, r_t, s_{t+1}\}$ of the current episode into the temporary sample pool D_{tr}. After the end of the current episode, the following conditions are used to determine whether the samples of the temporary sample pool D_{tr} need to be added to the permanent sample pool D_{te}. The judgment condition can be written as:

$$E\left(r_{ep}\right)_{D_{tr}} > k_{ep} E(r_{ep})_{D_{te}} \cup E\left(r_{st}\right)_{D_{tr}} > k_{st} E\left(r_{st}\right)_{D_{te}} \quad (5)$$

where k_{ep} and k_{st} are the relaxation factors of the judgment condition, and $k_{ep}, k_{st} \in (0, 1]$. $E\left(r_{ep}\right)_{D_{te}}$, $E\left(r_{st}\right)_{D_{te}}$ denote the average expected reward of each episode and step of permanent sample pool D_{te}, respectively. $E\left(r_{st}\right)_{D_{tr}}$, $E\left(r_{st}\right)_{D_{tr}}$ represent the average expected reward of each episode and step of temporary sample pool D_{tr}, respectively. In addition $E\left(r_{ep}\right)_{D_{tr}} = \sum_{i=1}^{n} \gamma^{i-1} r_i$, $E\left(r_{st}\right)_{D_{tr}} = \frac{1}{n}\sum_{i=1}^{n} \gamma^{i-1} r_i$, $E\left(r_{ep}\right)_{D_{te}} = \frac{1}{m}\sum_{j=1}^{m}\left[\sum_{i=1}^{n} \gamma^{i-1} r_i\right]_j$, $E\left(r_{st}\right)_{D_{te}} = \frac{1}{m \cdot n}\sum_{j=1}^{m}\left[\sum_{i=1}^{n} \gamma^{i-1} r_i\right]_j$. where the n is the number of steps in the current episode, m is the number of episodes.

To prevent overfitting, we add samples from the temporary pool to the permanent pool every m_f episodes without filtering. In addition, samples need to be collected quickly in the early stage of model training, so the samples of the samples of the first n_f episodes when the model start sampling does not need to be filtered.

2) Discrete Soft Actor-Critic Algorithm.: The DSAC algorithm is based on the Soft Actor-Critic (SAC) algorithm to change the mean μ and variance σ of the output of the policy network to the probability of discrete actions. In the process of strategy training, we apply the categorical distribution sampling method instead of the epsilon-greedy strategy. Since the uncertainty of the categorical distribution, the action exploration ability of the strategy is improved and the hyperparameters of the strategy are reduced. If the variable X obeys the categorical distribution and the probability vector of the variable X in the action spaces is $P = [\alpha_1, \alpha_2, \cdots, \alpha_k]^T$, then the probability of the variable X in the sampling can be written as:

$$p(X = x_i) = \alpha_i; \qquad x_i \in \{1, 2, \cdots, k\} \quad (6)$$

where $x_i \in \{1, 2, \cdots, k\}$ is the action spaces of the Variable X, and the probability vector of the variable X satisfies $\sum_{i=1}^{k} \alpha_i = 1$.

Since DSAC is based on the SAC algorithm, it has an actor-critic network framework and includes three processes: actor update process, critic update process and temperature coefficient update process.

Actor update process. This process explores the policy by maximinzing the reward value. For this reason, the soft Q-function value is employed to guide changes to the policy. So the policy loss function $J_\pi(\phi)$ can be written as:

$$J_\pi\left(\phi\right) = \pi_\phi(\cdot|s_t)^T \left[\alpha \log\left(\pi_\phi\left(\cdot|s_t\right)\right) - \min_{i=1,2} Q_{\theta_{m,i}}(s_t, a_t)\right] \quad (7)$$

where $\pi_\phi(\cdot|s_t)^T$ is the vector transpose of the $\pi_\phi(\cdot|s_t)$.

Critic update process. This process sets up the mean-squared Bellman error (MSBE) loss for each Q-functions using this kind of sample approximation for the target. As a result, the loss functions $J_Q\left(\theta\right)$ for the Q-networks and the parameter method can be written as:

$$J_Q\left(\theta\right) = E_{(s_t, a_t, r_t, s_{t+1}) \sim D}\left[\left(Q_{\theta_i}\left(s_t, a_t\right) - \hat{Q}\left(s_t, a_t\right)\right)^2\right] \quad (8)$$

$$\hat{Q}\left(s_t, a_t\right) = r\left(s_t, a_t\right) + \gamma(1-d) E_{s_{t+1} \sim p(\cdot|s_t, a_t)}\left[\hat{V}(s_{t+1})\right] \quad (9)$$

$$\hat{V}\left(s_{t+1}\right) = \pi\left(\cdot|s_{t+1}\right)^T \left[\min_{i=1,2} Q_{\theta_{t,i}}\left(s_{t+1}, a_{t+1}\right) - \alpha \log\left(\pi\left(\cdot|s_{t+1}\right)\right)\right] \quad (10)$$

where D is the sample buffer. $Q_{\theta_{m,i}}(s_t, a_t)$ is the action value of a_t action in state s_t. $\theta_{m,i}$,$\theta_{t,i}$ are the main and target Q network parameters, respectively.

The temperate parameter update process. The temperate parameter α as a hyper-parameter is hard to set. Thus, an optional way of learning the parameter which refer to the research provides by the study [20]. The loss function of $J(\alpha)$ can be written as:

$$J\left(\alpha\right) = \pi(\cdot|s_t)^T \left[-\alpha\left(\log\left(\pi_t\left(\cdot|s_t\right)\right) + \overline{\mathcal{H}}\right)\right] \quad (11)$$

The above DSAC with sample filter algorithm is given by algorithms 1.

B. Decision-making Strategy for Autonomous Driving

In this subsection, we construct an autonomous driving decision model based on our DSAC-SF algorithm.

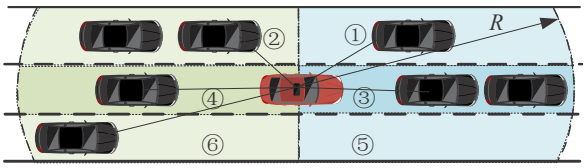


Fig. 2. Schematic diagram of dynamic observation environment with area observation method.

1) Autonomous Driving Decision-making Framework: As shown in Fig. 1, it is the framework of our autonomous driving decision-making model based on the DSAC-SF algorithm. This framework includes three modules: environment observation module, discrete decision-making module and intelligent vehicle module. The environment observation module includes an environmental information extraction method, which extracts the structured information around the ego vehicle depending on the decision-making requirements of the intelligent vehicle. The discrete decision-making module is an autonomous driving decision-making model designed based on the above DSAC-SF algorithm. The intelligent vehicle module is to receive the decision-making commands and perform related actions. We will clarify in detail the environment method and the decision-making model.

2) Environment Observation: The environmental observation method extracts the structured information of the traffic environment via sub-region as the observation state of the decision-making model. We first divide the traffic environment into six areas for observation via the lane lines and the virtual line that passes through the center and is perpendicular to the longitudinal axis of the ego vehicle. The six observation areas are contained in a circle with the ego vehicle as the center and R as the radius, as shown in Fig. 2. Subsequently, we extract dynamic observation information from each area separately, and improve the accuracy of observation status via supplementing default values. In addition, we combine dynamic and static observation information to form the observation state, enhancing the integrity of the observation state. The dynamic traffic environmental information includes the relative coordinates d_{ei}, relative speed v_{ei} and relative headway time t_{ei} to the vehicle closet to the ego vehicle in each observation area. Furthermore, we add an attribute flag l_i for each area to distinguish the observation area in special cases. In addition, we set the average radius r_p of a segment of trajectories in front of the ego vehicle as the road curvature radius and as static traffic information. The parameters l_i, d_{ei}, v_{ei}, t_{ei} for each region can be written as:

$$l_i = \begin{cases} -1, & \text{area } i \text{ is not exist} \\ 0, & \text{area } i \text{ is exists but without vehicles} \\ 1, & \text{area } i \text{ is exists and has some vehicles} \end{cases} \tag{12}$$

$$d_{ei} = \big[\mathrm{sgn}((l_i-1))+1\big]\|P_e - P_i\|_2 - \big[\mathrm{sgn}(|l_i|-1)\big]R \tag{13}$$

$$v_{ei} = \mathrm{sgn}(l_i+1)v_e - \big[\mathrm{sgn}(l_i-1)+1\big]v_i + (i\%2)\,\mathrm{sgn}\big(|l_i|-1\big)v_{max} \tag{14}$$

where $P_e(x_e, y_e)$ is the coordinates of the ego vehicle. $P_i(x_i, y_i)$ is the coordinates of the surrounding vehicle in the i area. The time headway $t_{ei} = \frac{d_{ei}}{v_e}$. R is the maximum distance of observation. v_i is the speed of the vehicle closet to the ego vehicle in the area i. $v_{\max}$ is the maximum speed of the scenario. v_e is the speed of the ego vehicle.

3) Decision-making model: We design a discrete decision-making model for autonomous driving based on our DSAC-SF algorithm. The model takes the observation matrix extracted via the above-mentioned environmental observation method as the observation method as the observation state of the model. Discrete actions serve as control instructions for the model. Furthermore, we define a multi-objective reward function to improve the safety and efficiency of the decision-making model for autonomous driving. The rewards r_t can be written as:

$$r_t = k_e \int_t^{t+T} v_x dt + k_c(1-d)C_c + k_d r_d \tag{15}$$

where k_e, k_c and k_d are the weight coefficients of each reward. T is the simulation period. v_x is the component of vehicle speed in the forward direction of the road. d is the flag used to record whether the vehicle collide or rush out the road $d \in \{0, 1\}$. C_c is the constant penalty for collide or rush out the road. r_d is the reward for hazardous working conditions, which can be written as:

$$r_d = \begin{cases} 0, & (t_h > t_s) \\ -\dfrac{1}{t_h}, & (t_c < t_h \le t_s) \\ C_t, & (t_h \le t_c) \end{cases} \tag{16}$$

where t_h is the relative TH between the leading vehicle and the ego vehicle. t_s is the threshold of safety relative TH. t_c is the threshold of hazard relative TH. C_t is the constant penalty of the hazardous working conditions, and $C_t < 0$.

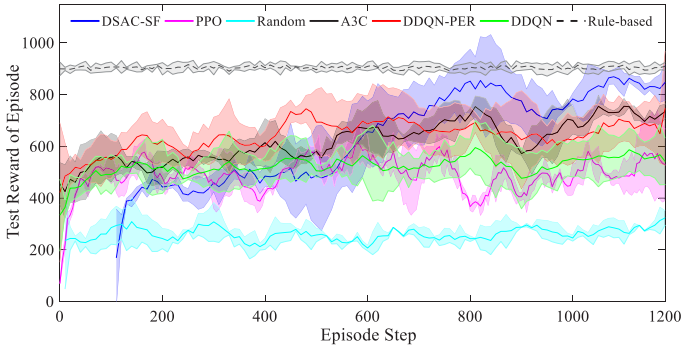


Fig. 3. Test reward of episode of DSAC-SF, PPO,DDQN, DDQN-PER, A3C, rule-based and random-action decision-making strategy. The curves describe in the figure represent the average values of multiple tests, and the shaded boundaries represent the maximum and minimum values. To ensure sample balance, DSAC-SF strategies is only randomly adopted without training for the first 100 episodes in the training process, thus the reward curve of the DSAC-SF strategy starts from 100 episodes.

IV. Experiment Evaluation

In this section, we compare the DSAC-SF to rule-based and prior RL methods in the SMARTS simulation platform. In addition, we performed ablation experiments on sample filter strategy and area observation method.

TABLE I
PERFORMANCE COMPARISON OF DISCRETE DECISION-MAKING STRATEGIES UNDER DIFFERENT TRAFFIC FLOW DENSITIES

Traffic	Methods	Success rate $P_s(\%)\uparrow$	Speed v $\mu(m/s)\uparrow$	$\sigma\downarrow$	TH $\mu\downarrow$	$\sigma\downarrow$	$2s(\%)\downarrow$
Dense	DSAC-SF	**92.75**	**12.23**	1.45	**4.27**	0.79	0.2
	PPO	72.7	11.21	1.66	5.05	0.82	**0.01**
	DDQN	57.78	11.32	2.64	**3.49**	0.98	6.42
	DDQN-PER	88.96	**11.59**	2.33	**3.42**	0.91	5.93
	A3C	90.86	11.34	2.05	4.38	0.88	0.34
	Rule-based	**93.47**	10.64	1.19	6.67	0.63	0
	Random	30.85	4.91	3.05	10.9	5.45	5.12
Sparse	DSAC-SF	**95.76**	**13.72**	1.81	4.47	0.83	0.16
	PPO	85.26	12.12	1.92	5.12	0.92	**0.03**
	DDQN	88.75	12.41	2.91	**3.76**	1.05	4.69
	DDQN-PER	91.29	12.69	2.51	**3.65**	0.95	4.12
	A3C	92.38	**12.73**	2.35	4.44	0.93	0.43
	Rule-based	**96.68**	10.94	1.38	6.75	0.7	**0**
	Random	35.77	4.46	3.09	11.83	5.31	3.21

A. Experiment Setup

We evaluate our proposed strategy on the SMARTS simulation platform. Currently, SMARTS is recognized as the most prevalent simulation platform in decision-making and motion-control for the autonomous driving experiments [23], which supports the setting of vehicle agents, simulation scenarios and traffic flow. We choose a multi-lane loop scenario without intersection for this experiment to evaluate the performance of our decision-making strategy on freeways. In addition, we set up simulation scenarios of sparse and dense traffic flow to experiment the performance of the decision-making strategy under different traffic flow densities. Five surrounding vehicles per second are generated under spare traffic flow, and ten surrounding vehicles per second are generated under dense traffic flow. Since the surrounding vehicles do not automatically disappear until they collide, the traffic density increases as the simulation time of each episode increases.

B. Performance comparison experiment

To evaluate the performance of the proposed DSAC-SF and decision-making strategy, we not only compare the episode rewards in the training process of DSAC-SF, PPO [24], DDQN [25], DDQN-PER [26], A3C [27], rule-based [28] and random-action, but also experiment the decision-making performance of these strategies in the same simulation environment. The episode reward tested during training is shown in Fig. 3. It can be seen that the reward curve of the DSAC-SF strategy grows slower than strategies such as DDQN, DDQN-PER, A3C and PPO, but the reward value of DSAC-SF is higher than these strategies after the training is stable. The higher the reward value after the training is stabilized under the same experimental conditions, the better the efficiency of the algorithm. Therefore, the DSAC-SF strategy has better training efficiency than algorithms such as DDQN, PPO, DDQN-PER and A3C.

The result in Table I shows that the success rate of DSAC-SF strategy with area observation method is higher than that of DDQN, DDQN-PER, A3C, PPO and random-action strategies in both sparse and dense traffic flow. In addition, it can be seen from the table that the probability of the DSAC-SF strategy's TH is smaller than DDQN, DDQN-PER, A3C and random-action strategies. The higher the strategy success rate, the greater the probability of reaching the end point safely, and the better the safety of the vehicle. Therefore, we can concluded that the vehicle safety based on DSAC-SF strategy is higher than the vehicle based on DDQN, DDQN-PER, A3C, PPO and Random-action strategies.

1) Ablation Study: The decision-making performances of DSAC-SF with area observation method, DSAC-SF with distance observation method, and DSAC with area observation method strategies are shown in Table II. As shown in the table, the success rate of DSAC-SF with area observation strategy in both sparse and dense traffic flow is improved than DSAC-SF with distance observation and DSAC with area observation strategies. In addition, the probability that the TH of the DSAC-SF with area observation method is less than 2 seconds in both parse and dense traffic flow, which is lower than the DSAC-SF with distance observation method and DSAC with area observation method. The higher the success rate and smaller the probability that TH is less than 2s, the higher the security of the decision-making strategy. Therefore, we can conclude that the area observation method and sample filter approach in this work can improve the safety of decision-making. In addition, it can be seen from Table II that the mean μ of the average speed of DSAC-SF strategy with area observation method in both parse and dense traffic flow is higher than that of DSAC-SF strategy with distance observation method and DSAC with area observation method. The higher the speed, the higher the efficiency of the agent vehicle. Thus, the area observation method and sample filter approach proposed in this paper can improve the driving efficiency of the decision-making. In summary, the performance and stability of the DSAC-SF decision-making strategy can be improved through the area observation method and sample filter strategy. The DSAC-SF decision-making strategy proposed in this paper has better safety, efficiency, and stability than PPO and random strategy.

TABLE II
ABLATION STUDY FOR THE SAMPLE FILTER STRATEGY AND AREA OBSERVATION METHODS.

Traffic	Methods	Success Rate $P_s(\%)\uparrow$	Speed v_e $\mu(m/s)\uparrow$	$\sigma\downarrow$	TH $\mu\downarrow$	$\sigma\downarrow$	$<2s(\%)\downarrow$
Dense	DSAC-SF (Area-obs)	**92.75**	**12.23**	**1.45**	4.27	0.79	**0.2**
	DSAC-SF (Distance-obs)	75.49	12.07	1.51	4.08	0.78	0.38
	DSAC (Area-obs)	90.26	12.11	1.53	**3.78**	**0.66**	0.35
Spare	DSAC-SF (Area-obs)	**95.75**	**13.72**	**1.81**	4.45	0.83	**0.16**
	DSAC-SF (Distance-obs)	80.98	12.88	1.85	4.29	0.81	0.23
	DSAC (Area-obs)	93.08	12.91	1.92	**4.08**	**0.73**	0.21

V. CONCLUSION

In this work, a DSAC-SF algorithm is proposed to improve the sample efficiency and stability of the algorithm. Furthermore, an decision-making strategy of autonomous driving

is designed based on the algorithm to improve the driving efficiency and safety of the vehicle. Specifically, we propose a sample screening strategy that includes two sample pools to improve the DSAC algorithm. Subsequently, we design a discrete decision-making strategy for autonomous driving based on the DSAC-SF algorithm. The strategy constructs a multi-objective reward function to improve driving efficiency and safety. In addition, we propose a area observation method to enhance the accuracy of extracting structured information. Finally, we experiment our decision-making strategy on the SMARTS simulation platform, and the results indicate that our strategy can provide competitive performance in both sparse and dense traffic flows.

ACKNOWLEDGMENT

The authors would like to express their deep appreciation to the financial support from the National Natural Science Foundation of China (No. 61906138), the Shanghai Municipal Science and Technology Major Project (No.2018SHZDZX01), ZJ Lab, and Shanghai Center for Brain Science and Brain-Inspired Technology, and the Shanghai Rising Star Program (No.21QC1400900).

REFERENCES

[1] Y. Fu, C. Li, F. R. Yu, T. H. Luan, and Y. Zhang, "A decision-making strategy for vehicle autonomous braking in emergency via deep reinforcement learning," *IEEE transactions on vehicular technology*, vol. 69, no. 6, pp. 5876–5888, 2020.

[2] L. Li, K. Ota, and M. Dong, "Humanlike driving: Empirical decision-making system for autonomous vehicles," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 8, pp. 6814–6823, 2018.

[3] D. Isele, R. Rahimi, A. Cosgun, K. Subramanian, and K. Fujimura, "Navigating occluded intersections with autonomous vehicles using deep reinforcement learning," in *2018 IEEE International Conference on Robotics and Automation*. IEEE, 2018, pp. 2034–2039.

[4] G. Chen, K. Chen, L. Zhang, L. Zhang, and A. Knoll, "Vcanet: Vanishing-point-guided context-aware network for small road object detection," *Automotive Innovation*, vol. 4, no. 4, pp. 400–412, 2021.

[5] W. Liu, X. Xia, L. Xiong, Y. Lu, L. Gao, and Z. Yu, "Automated vehicle sideslip angle estimation considering signal measurement characteristic," *IEEE Sensors Journal*, vol. 21, no. 19, pp. 21 675–21 687, 2021.

[6] S. Aradi, "Survey of deep reinforcement learning for motion planning of autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 740–759, 2022.

[7] W. Wang, X. Na, D. Cao, J. Gong, J. Xi, Y. Xing, and F.-Y. Wang, "Decision-making in driver-automation shared control: A review and perspectives," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 5, pp. 1289–1307, 2020.

[8] B. Peng, Q. Sun, S. E. Li, D. Kum, Y. Yin, J. Wei, and T. Gu, "End-to-end autonomous driving through dueling double deep q-network," *Automotive Innovation*, vol. 4, no. 3, pp. 328–337, 2021.

[9] D. Chen, L. Jiang, Y. Wang, and Z. Li, "Autonomous driving using safe reinforcement learning by incorporating a regret-based human lane-changing decision model," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 4355–4361.

[10] H. Wang, W. Wang, S. Yuan, X. Li, and L. Sun, "On social interactions of merging behaviors at highway on-ramps in congested traffic," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 11 237–11 248, 2021.

[11] Y. Liu, G. Chen, and A. Knoll, "Globally optimal vertical direction estimation in atlanta world," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 4, pp. 1949–1962, 2020.

[12] W. Liu, L. Xiong, X. Xia, Y. Lu, L. Gao, and S. Song, "Vision-aided intelligent vehicle sideslip angle estimation based on a dynamic model," *IET Intelligent Transport Systems*, vol. 14, no. 10, pp. 1183–1189, 2020.

[13] X. Ma, J. Li, M. J. Kochenderfer, D. Isele, and K. Fujimura, "Reinforcement learning for autonomous driving with latent state inference and spatial-temporal relationships," in *2021 IEEE International Conference on Robotics and Automation*. IEEE, 2021, pp. 6064–6071.

[14] J. Yang, J. Zhang, M. Xi, Y. Lei, and Y. Sun, "A deep reinforcement learning algorithm suitable for autonomous vehicles: Double bootstrapped soft-actor-critic-discrete," *IEEE Transactions on Cognitive and Developmental Systems*, 2021.

[15] P. Christodoulou, "Soft actor-critic for discrete action settings," *arXiv preprint arXiv:1910.07207*, 2019.

[16] W. Ko and D. E. Chang, "A dual memory structure for efficient use of replay memory in deep reinforcement learning," in *2019 19th International Conference on Control, Automation and Systems*. IEEE, 2019, pp. 1483–1486.

[17] C. Hu, M. Kuklani, and P. Panek, "Accelerating reinforcement learning with prioritized experience replay for maze game," *SMU Data Science Review*, vol. 3, no. 1, p. 8, 2020.

[18] M. G. Bellemare, W. Dabney, and R. Munos, "A distributional perspective on reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2017, pp. 449–458.

[19] J. Duan, Y. Guan, S. E. Li, Y. Ren, Q. Sun, and B. Cheng, "Distributional soft actor-critic: Off-policy reinforcement learning for addressing value estimation errors," *IEEE transactions on neural networks and learning systems*, pp. 1–15, 2021.

[20] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning*. PMLR, 2018, pp. 1861–1870.

[21] N. Li, H. Chen, I. Kolmanovsky, and A. Girard, "An explicit decision tree approach for automated driving," in *Dynamic Systems and Control Conference*, vol. 58271. American Society of Mechanical Engineers, 2017, p. V001T45A003.

[22] A. Baheri, S. Nageshrao, H. E. Tseng, I. Kolmanovsky, A. Girard, and D. Filev, "Deep reinforcement learning with enhanced safety for autonomous highway driving," in *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2020, pp. 1550–1555.

[23] M. Zhou, J. Luo, J. Villela, Y. Yang, D. Rusu, J. Miao, W. Zhang, M. Alban, I. Fadakar, Z. Chen *et al.*, "Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving," in *Proceedings of the 4th Conference on Robot Learning*, 11 2020.

[24] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[25] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[26] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," in *International Conference on Learning Representations*, 2016.

[27] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*. PMLR, 2016, pp. 1928–1937.

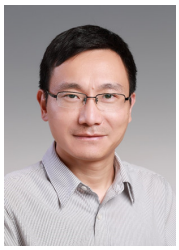
[28] M. Zhang, N. Li, A. Girard, and I. Kolmanovsky, "A finite state machine based automated driving controller and its stochastic optimization," in *Dynamic Systems and Control Conference*, vol. 58288. American Society of Mechanical Engineers, 2017, p. V002T07A002.

Jiayi Guan received the M.E. degree in vehicle operation engineering from Wuhan University of Technology, Wuhan, China in 2017. He was an algorithmic engineer of autonomous driving in GAC R&D Center from 2017 to 2020. He is currently pursuing the Ph.D. degree in Transportation Engineering at Tongji University, Shanghai, China. His research interests include the reinforcement learning, decision-making and motion planning.

Guang Chen is a research professor at Tongji University and a senior research associate (guest) at Technical University of Munich. His research interests include bio-inspired vision, machine learning, and computer vision with applications in autonomous systems. He received his Ph.D degree from Technical University of Munich in 2016. He received the M.Eng degree from Hunan University in 2011. He was awarded the program of Shanghai Rising Star 2021, and Shanghai S&T 35U35 2021. He serves as Associate Editor of several international journals. He is the program chair of IEEE MFI 2022.

Jin Huang received the B.E. and Ph.D. degrees from the College of Mechanical and Vehicle Engineering, Hunan University, Changsha, China, in 2006 and 2012, respectively, during which he joined a joint Ph.D. program in the George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, Georgia, USA, during 2009–2011. He has been a Post-Doctoral Fellow and an Assistant Research Professor with the School of Software, Tsinghua University, Beijing, China, since 2013 and 2016, respectively. His current research interests include artificial intelligence in intelligent transportation systems, cyber-physical systems, automation and mechatronics, and fuzzy engineering.

Zhijun Li (M'07-SM'09) received the Ph.D. degree in mechatronics, Shanghai Jiao Tong University, P. R. China, in 2002. From 2003 to 2005, he was a postdoctoral fellow in Department of Mechanical Engineering and Intelligent systems, The University of Electro-Communications, Tokyo, Japan. From 2005 to 2006, he was a research fellow in the Department of Electrical and Computer Engineering, National University of Singapore, and Nanyang Technological University, Singapore. From 2017, he is a Professor in Department of Automation, University of Science and Technology, Hefei, China. From 2019, he is the Vice Dean of School of Information Science and Technology, University of Science and Technology of China, China. From 2016, he has been the Co-Chairs of IEEE SMC Technical Committee on Bio-mechatronics and Bio-robotics Systems (B^2S), and IEEE RAS Technical Committee on Neuro-Robotics Systems. He is serving as an Editor-at-large of Journal of Intelligent & Robotic Systems, and Associate Editors of several IEEE Transactions. Dr. Li's current research interests include wearable robotics, tele-operation systems, nonlinear control, neural network optimization, etc.



Lu Xiong received the B.E., M.E., and the Ph.D. degrees in vehicle engineering from the School of Automotive Studies, Tongji University, Shanghai, China, in 1999, 2002, and 2005, respectively. From November 2008 to 2009, he was a Postdoctoral Fellow with the Institute of Automobile Engineering and Vehicle Engines, University of Stuttgart, Germany, with Dr. Jochen Wiedemann. He is currently a Professor with Tongji University. He is also an Executive Director of the Institute of Intelligent Vehicles and an Associate Director of the Clean Energy Automotive Engineering Center, Tongji University. His research interests include perception, decision-making, motion planning, and evaluation of autonomous driving.

Jing Hou received the B.E. degree in vehicle engineering from Hunan University, Changsha, China in 2019. He is currently working toward the Ph.D. degree in vehicle engineering at the Tongji University, Shanghai, China. His research interests include deep reinforcement learning and autonomous driving.

Alois Knoll (Senior Member, IEEE) received the diploma (M.Sc.) degree in Electrical/Communications Engineering from the University of Stuttgart, Germany, in 1985 and his Ph.D. (summa cum laude) in Computer Science from the Technical University of Berlin, Germany, in 1988. He served on the faculty of the Computer Science department of TU Berlin until 1993. He joined the University of Bielefeld, as a full professor and the director of the research group Technical Informatics until 2001. Since 2001 he has been a professor at the Department of Informatics, TU München. He was also on the board of directors of the Central Institute of Medical Technology at TUM (IMETUM). From 2004 to 2006, he was Executive Director of the Institute of Computer Science at TUM. His research interests include cognitive, medical and sensor-based robotics, multi-agent systems, data fusion, adaptive systems, multimedia information retrieval, model-driven development of embedded systems with applications to automotive software and electric transportation, as well as simulation systems for robotics and traffic.