Deep Reinforcement Learning Based Lane-level Variable Speed Limit Control

Xiyu Chen
Polytechnic Institute
Zhejiang University
Hangzhou, China
chenxy14@zju.edu.cn

Juntao Jiang
College of Control Science and Engineering
Zhejiang University
Hangzhou, China
juntaojiang@zju.edu.cn

Jiandang Yang*
College of Control Science and Engineering
Zhejiang University
Hangzhou, China
yangjd@zju.edu.cn

Yong Liu*
College of Control Science and Engineering
Zhejiang University
Hangzhou, China
yongliu@iipc.zju.edu.cn

2023 9th International Conference on Control Science and Systems Engineering (ICCSSE) | 979-8-3503-3905-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCSSE59359.2023.10244962

***Abstract*—Variable speed limit (VSL) is an effective traffic control method to alleviate congestion and increase safety. This paper incorporates deep reinforcement learning (DRL) into the VSL control strategy and proposes a twin delayed deep deterministic policy gradient (TD3)-based solution. We set different speed limits between every lane to control the speed of vehicles entering the highway merging area, thereby increasing the traffic flow and improving passing efficiency. The proposed model learns a large number of discrete actions within continuous actions through the actor-critic framework, using the reward signal based on the difference between inflow and outflow to train the agent. We selected real-world road segments and collected corresponding data to test the proposed method. The simulation results show that the VSL control based on TD3 can effectively reduce average travel time and increase the number of passing vehicles.**

***Keywords*—Deep reinforcement learning, TD3, Variable speed limit control, Intelligent transportation system**

I. Introduction

Highways are essential parts of the transportation networks. With the economy's development, the transportation demand continues to increase. Especially during holidays, highways are often plagued with congestion problems. The merging area, where traffic from various sections converges, is a common cause of interruptions in the primary traffic flow, leading to significant congestion. Once congestion occurs, passing capacity drops sharply, further aggravating the situation [1]. To alleviate highway congestion, developing intelligent transportation systems (ITS) is a practical and effective solution.

Variable speed limit (VSL), as a traffic control technology of ITS, effectively mitigates congestion in merging areas, enhancing traffic efficiency. VSL system dynamically detects traffic flow parameters of vehicles on the road and inputs this traffic flow information into the controller. After processing through an algorithm, the system outputs the calculated speed limit value to the variable message signs. VSL has been proven to enhance traffic safety [2, 3] and reduce environmental pollution [4] while simultaneously boosting traffic efficiency.

Traditional VSL strategies include determining the speed limit value based on traffic state thresholds or feedback control [5]. Recently, artificial intelligence has played a more critical role in traffic flow control. Reinforcement learning (RL), a branch of machine learning, involving interacting with the environment and receiving feedback, has been widely applied. The emergence of deep learning significantly improved RL, and deep reinforcement learning (DRL) has achieved impressive success in areas like robotics and gaming. There are many DRL methods, such as deep Q networks (DQN) [6], Deep Deterministic Policy Gradient (DDPG) [7], Proximal policy optimization (PPO) [8], twin delayed deep deterministic policy gradient (TD3) [9], etc. DRL also holds great potential for ITS control tasks. Experimental results demonstrate that DRL methods outperform traditional model-driven traffic control methods in traffic signal control, showing their practical application value [10].

Many scholars have applied RL to VSL [11–14]. In [11], Q-learning (QL) was applied to VSL and compared with the feedback strategy. The study found that VSL based on QL can significantly reduce travel time under different demands. To represent and explore the large state-action space, the study [12] applied DQN to VSL, and the results showed that this method could improve the average travel speed. A VSL algorithm based on DDPG was proposed in [13] to eliminate chronic highway bottlenecks. In [14], multi-agent reinforcement learning was applied to VSL. The proposed distributed QL-VSL method can improve traffic flow by maintaining high traffic density levels close to critical density on highways. This paper applies the TD3 algorithm to VSL and proposes a lane-based VSL algorithm for highway merging areas. Considering that the discrete action space of lane-based control is too large, we use continuous action output and map it to a discrete action space. This algorithm can set corresponding speed limits for both mainline and ramp according to real-time traffic states upstream and downstream.

The organization of this paper is as follows. Section II describes the mechanism of VSL. Section III introduces the TD3 algorithm and its application in VSL. Section IV presents the selected traffic scenarios and traffic demands. Section V analyzes the simulation results. Finally, the main conclusions are drawn in the last section.

*Corresponding author

II. Variable Speed Limit Control

When traffic demand exceeds the capacity of the merging area, congestion occurs, accompanied by a sharp decrease in capacity and stop-and-go traffic. Previous studies have investigated traffic flow at bottleneck locations and found that a sharp decrease in capacity is a common phenomenon [15]. As vehicle density increases and road capacity gradually reaches its maximum, further increases in vehicle density will cause road capacity to drop sharply within a short period. As shown in Fig. 1, when vehicle density exceeds K_m, the capacity drops from Q_m to Q_d.

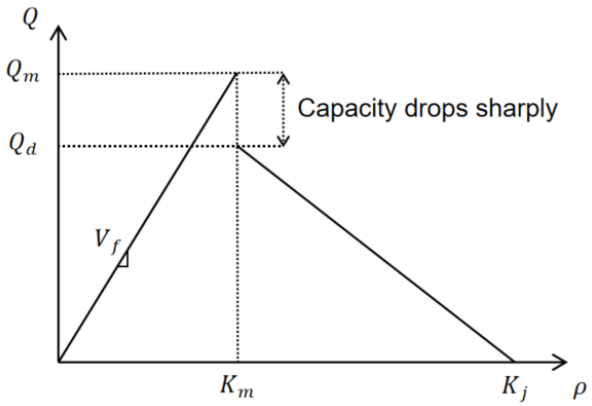

Fig. 1. Traffic flow relationships with a sharp drop in capacity

Related research shows that VSL can effectively alleviate congestion in bottleneck areas [16]. VSL on bottleneck sections cannot directly improve the theoretical capacity of the section, but it can influence traffic flow density and speed on the bottleneck section by controlling the speed of upstream vehicles in a timely manner, preventing a sharp drop in road capacity, achieving a traffic load level equal to or close to the maximum traffic capacity of the bottleneck area, and improving road efficiency and actual traffic flow volume.

Fig. 2 simplifies the impact process of VSL on traffic flow in the bottleneck area. The black line represents the actual traffic flow basic map of the bottleneck area, while the green line represents the basic map after applying VSL. Initially, vehicles travel at free-flow speeds. As the number of vehicles gradually increases and traffic density increases, the capacity drops sharply to Q_d after reaching the threshold K_m of capacity reduction. At this time, the maximum passing flow of the bottleneck section is Q_d. By applying VSL upstream, with a speed limit value of V_{VSL}, the maximum passing flow of the upstream section can be increased to Q_{VSL}. Through VSL, a high-density artificial traffic flow area can be formed on the upstream side of the bottleneck section, increasing the traffic capacity from Q_d to Q_{VSL} and avoiding queues of the bottleneck area that may result in congestion on the upstream section. At the same time, it can quickly relieve queuing vehicles to restore the original traffic capacity, thereby avoiding a decrease in traffic capacity at bottleneck areas.

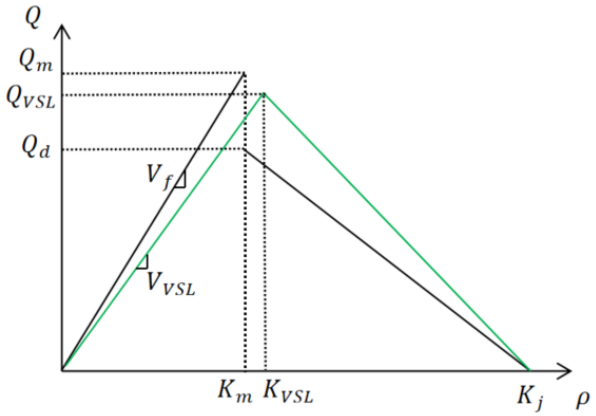

Fig. 2. Impact of VSL on bottleneck areas

III. TD3 for VSL

A. Reinforcement Learning

The agent selects actions based on the state of the environment, and the actions affect the environment, which provides feedback in the form of rewards. By iterating the policy function in this way, the agent is guided to choose more appropriate actions in the future, and the basic model framework for reinforcement learning is the Markov decision process (MDP).

The simplest MDP consists of four elements $\{S, A, P, r\}$, where S represents the state of the environment, A represents the actions of the agent, r represents the reward function and p represents the probability of state transitions. At each time step t, the agent selects an action based on the current state s and policy π. The environment then transitions to state S_{t-1} with probability P and provides a reward $r : S \times A \times S \to R$. The policy π is a mapping from states to actions, and its performance can be evaluated by the state-value function $V(s)$ or Q-value function $Q(s, a)$. The ultimate goal of reinforcement learning is to find a policy $\pi(a \mid s) = P_{\mathrm{r}}(A_t = a \mid S_{t-1} = s)$ that maximizes the expected cumulative reward G_t for the agent.

$$G_t = \sum_{t=i}^{\infty} \gamma^{t-i} r(s_t, a_t) \tag{1}$$

Where γ is the discount factor, which measures the magnitude of future rewards in the cumulative rewards at the current state. a is the action, s is the state, t is the time, and $r(s_t, a_t)$ is the reward at each time step.

To find the optimal policy function π^*, many reinforcement learning algorithms use the Q-value function $Q_\pi(s_t, a_t)$ to evaluate the policy. The Bellman equation is given by

$$Q_\pi(s_t, a_t) = r(s_t, a_t) + \gamma \mathrm{E}_\pi [Q_\pi(s_{t+1}, a_{t+1})] \tag{2}$$

where E_π is the expectation.

When the optimal Q-value function $Q^*\pi(s_t, a_t)$ is known, the policy function π^* can be obtained by $a = \operatorname{argmax} Q^*_\pi(s_t, a')$. The value of $Q^*\pi(s_t, a_t)$ can be learned

using a temporal difference in the QL algorithm. However, when the dimensions of states and actions are large, the QL algorithm faces a "dimensional disaster." This problem can be solved by using neural networks.

B. TD3 Algorithm

The twin delayed deep deterministic policy gradient (TD3) algorithm is a DRL algorithm based on the actor-critic (AC) framework, an improved version of the DDPG algorithm. The algorithm uses two neural networks called the actor and critic networks to approximate the policy function and Q value function. The actor-network is responsible for interacting with the environment to obtain the most suitable action a for the current state s. The critic network evaluates the policy network's action a and the system's current state s based on the reward r. The TD3 algorithm can handle problems with continuous action spaces. Due to the potential overestimation of the Q-value function by the critic network in the DDPG algorithm, TD3 makes three improvements to address the shortcomings of DDPG.

- The TD3 algorithm draws on the experience of the DDQN [17] algorithm by using two sets of critic networks and considering smaller target Q values in the computation, which helps to suppress the overestimation problem in neural networks.
  Updating the Critic network with minimized loss function.
  $$L(\theta) = \mathrm{E}\left[\sum_{i=1}^{2}\left(y_t - Q_{\theta_i}\left(s_t, a_t\right)\right)^2\right] \tag{3}$$
  Where $Q_{\theta_i}\left(s_t, a_t\right)$ is the output of the critical network, y_t is the target Q value, which is defined as
  $$y_t = r\left(s_t, a_t\right) + \gamma \min Q_{\theta_i'}\left(s_{t+1}, a_{t+1}\right) \tag{4}$$
  $$a_{t+1} \sim \pi_{\phi'}\left(s_{t+1}\right) \tag{5}$$
  Where $Q_{\theta'}$ is the output of the two critical target networks, the smaller Q value is selected to calculate the target Q value, and $\pi_{\phi'}$ is the target actor policy.
  Updating actor networks by deterministic policy gradient algorithm.
  $$\nabla_\phi J(\phi) = \mathrm{E}_{s\sim p\pi}\left[\nabla_a Q_\theta(s,a)|_{a=\pi(s)} \nabla_\phi \pi_\phi(s)\right] \tag{6}$$
- The TD3 algorithm, like DDPG, uses a soft update strategy to update the target network. However, TD3 updates the policy network at a lower frequency than the critic network. This delayed policy update can reduce accumulated errors, thus reducing variance. Additionally, reducing errors in the target network prior to updating the actor-network enhances the stability of the TD3 algorithm.
- In order to further reduce the impact of Q-value function errors on updating target values, the TD3 algorithm introduces noise into the target actor-network, which makes the Q-value function smoother and enhances the robustness of the algorithm.
  $$a' = \pi_{\phi'}\left(s'\right) + \varepsilon, \varepsilon \sim \mathrm{clip}(\mathcal{N}(0,\sigma), -c, c) \tag{7}$$
  Where a' is the next action, s' is the next state, ε is noise, $\mathcal{N}$ is a normal distribution, σ is the standard deviation, and c is the range of noise.

The TD3 algorithm employs experience replay to train the actor and critic networks. An experience replay buffer is used to store the $< s, a, r, s' >$ samples that are made up of state s, action a, reward value r, next state s'. These samples are obtained by interacting with the environment. During training, the neural networks are updated using small, random batches of samples taken from the replay buffer. Using experience replay improves the sample efficiency, and the correlation between training samples is reduced, leading to more stable convergence.

In summary, the TD3 algorithm comes with the benefit of reducing the overestimation of the Q-value, and improving training efficiency and stability.

C. Architectures of Neural Networks

This paper uses general Deep Neural Networks (DNNs) for the policy and value networks. The established policy network and the value network of TD3 both include 5 MLP layers as the input layer, hidden layers and the output layer. The specific architectures of the neural networks are shown in Fig. 3. The numbers in the figures represent the sizes of each input or output.

D. TD3-Based VSL Strategy

The key elements in the TD3 agent include:

- **Agent**: The agent in this article is a VSL controller, which reduces the average travel time and improves traffic efficiency by setting corresponding speed limits for each lane.
- **State**: The state describes the real-time traffic environment, and the state variables can be any traffic parameters obtained by sensors. In this study, we set up corresponding detectors on the upstream mainline, upstream ramp, and bottleneck area. We defined three variables to define the road segment traffic state, corresponding to the vehicle occupancy rate data collected by detectors at these three locations. These three variables can be used to monitor changes in traffic conditions.
- **Action**: The action taken by the agent at a time step t, which is the speed limit value set by the agent for each lane in the control area, is discretized into multiples of 5 in the published speed limit value in practical applications. Considering that when there are many lanes, the discrete action space will become very large, and the algorithm for handling discrete action spaces such as QL and DQN will be very difficult. Therefore, we adopt a method based on continuous space actions and map the continuous action space into a discrete action space by evenly dividing it.

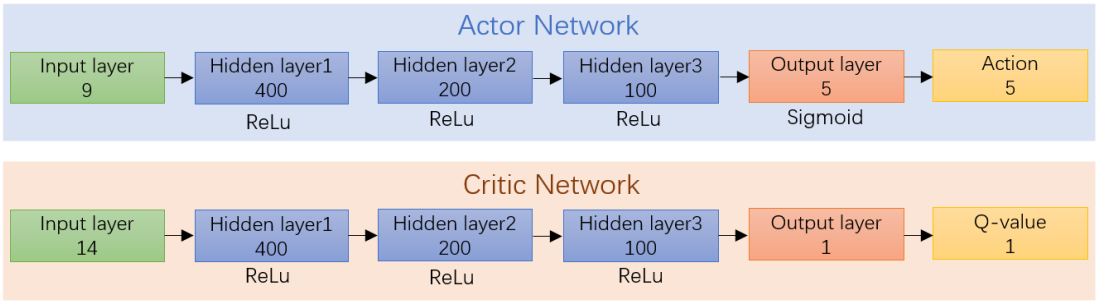

Fig. 3. Actor and critic network structure

- **State transition probability**: The training of the agent is conducted through simulation on the SUMO platform [18], where the state transition probability is represented by the changes in micro traffic flow presented by the SUMO platform at each time step. These state changes can be manifested explicitly as variations in the values of different detectors.
- **Reward**: The main goal is to improve traffic efficiency through VSL. An essential indicator of traffic efficiency is the average travel time. However, since the travel time cannot be calculated until the vehicle completes its route, using average travel time as a reward function can lead to a delay in reward. To solve this problem, refer to [19], where it is shown that there is a direct relationship between the average travel time and the number of arriving vehicles f^{in} at the entrance of a road segment and the number of departing vehicles f^{out} at the exit of a road segment, that is:

$$TTS = T\sum_{k=1}^{K}\left[N(0) + T\sum_{t=0}^{k-1} f^{\text{in}}(t) - T\sum_{t=0}^{k-1} f^{\text{out}}(t)\right]$$
$$= T\sum_{k=1}^{K} N(0) + T^2\sum_{k=1}^{K}\sum_{t=0}^{k-1}(f^{\text{in}}(t) - f^{\text{out}}(t)) \quad (8)$$

where K represents the total number of time steps, T represents the time interval, and k represents the time index. $f^{\text{in}}(t)$ and $f^{\text{out}}(t)$ respectively represent the number of vehicles arriving at the entrance of a road segment and leaving at the exit at time t. Therefore, the reward function within a time step t can be represented by $f^{\text{in}}(t) - f^{\text{out}}(t)$, the overall reward function can be represented by $F^{\text{in}} - F^{\text{out}}$. The number of vehicles departing and arriving at a certain moment can be collected respectively from upstream and downstream detectors.

In Fig. 4, we present the overall control framework of VSL. The actor network outputs the speed limit values for each lane based on the traffic state. The detector provides feedback on the difference between inflow and outflow as a reward signal, samples are stored in the replay buffer and mini-batch is sampled to train the neural networks.

IV. Simulation Network

We selected a road segment of about 2.5 kilometers long on the Hangzhou-Ningbo Expressway in Zhejiang Province, China. The geometric shape of this section in the map and SUMO is shown in Fig. 5. This section has four lanes, and there is a bottleneck on upstream of the highway due to the merging of traffic flows from the mainline and the ramp. The maximum and minimum speed limits for the right two lanes on the mainline are $100km/h$ and $60km/h$, respectively, while those for the left two lanes are $120km/h$ and $60km/h$, respectively. The speed limit for the ramp is $40km/h$. According to the setting principles of discrete actions described in Section III, the action set for the right two lanes on the mainline is $\{60, 65, 70, 75, 80, 85, 90, 95, 100\}(km/h)$, while that for the left two lanes is $\{60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 115, 120\}(km/h)$. The action set for ramps is $\{5, 10, 15, 20, 25, 30, 35, 40\}(km/h)$.

We set up four detectors at the upstream mainline entrance, upstream ramp entrance, bottleneck area and downstream exit to detect traffic occupancy rates. There are two vehicle travel routes on the section: Mainline to Mainline (M2M) and On-ramp to Mainline (On2M). In order to test the control effect under high traffic volume, Shown in Table I, we collected the passing vehicle volume data on point A of the M2M from 10 am to 4 pm on January 27th (the last day of the Spring Festival holiday). And the passing vehicle volume data of the On2M were simulated. The simulation lasts for 6 hours. The VSL control zone length for the mainline is set to 500m, and that for ramps is set to 300m. The control period is set to 5 minutes. The number of training episodes is 200, and the traffic volume in each period followed a Poisson distribution. Trucks are set to drive on the right side by default. Considering that frequent acceleration/deceleration and large speed differences between adjacent lanes can increase accident rates [20], we set a limit that the speed difference between adjacent time steps in the same lane should not exceed $20km/h$, and that between adjacent lanes in the same time-step should not exceed $20km/h$.

V. Simulation Result

The reward value obtained by the agent in each round can reflect the quality of the training results. The higher the reward value, the better the training effect. We trained a total of 200

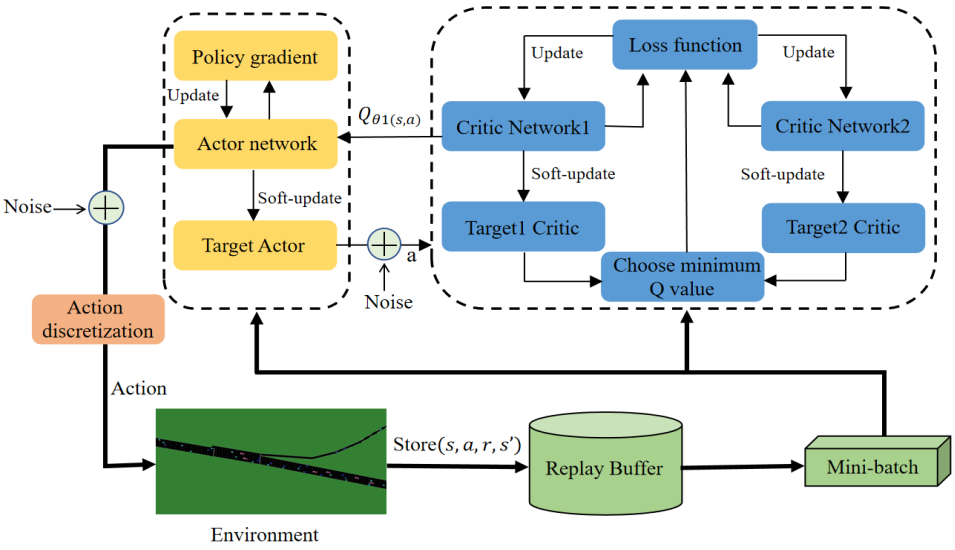

Fig. 4. TD3-based algorithm framework for VSL

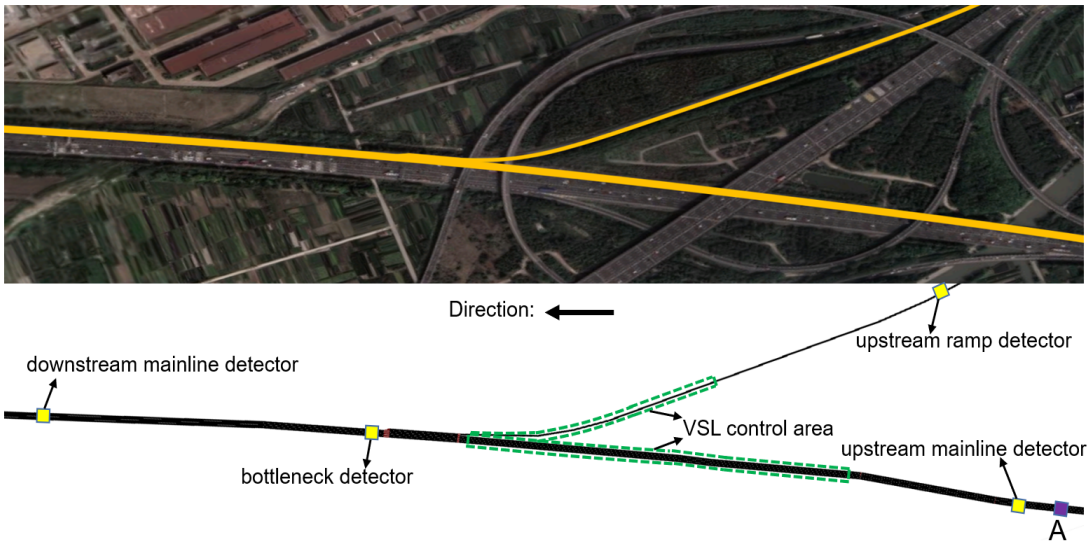

Fig. 5. Road section directional diagram in map and SUMO

TABLE I. TRAFFIC DEMAND FOR TWO ROUTES

Time	M2M	On2M
10:00-11:00	4933	300
11:00-12:00	6108	800
12:00-13:00	6356	1000
13:00-14:00	6692	1100
14:00-15:00	6054	1000
15:00-16:00	5246	800

episodes, and the change in reward value during the learning process is shown in Fig. 6. It can be seen that there is a clear convergence trend in the reward value, which roughly shows an upward trend in the first 100 episodes, and then gradually stabilizes.

The speed limit values for each lane vary over time, as shown in Fig. 7. In most cases, the speed limit values are lower than those without control ($\{100, 100, 120, 120, 40\}(km/h)$). Due to the larger number of vehicles during the middle time period compared to the two sides, the speed limit value in the middle position is slightly lower than that on both sides. Among the four lanes on the mainline, Lane 1 has a lower speed limit value than the other three lanes. One important reason is that when vehicles on the ramp merge onto the mainline, the lane-changing behavior initially affects the rightmost lane of the mainline, leading to a lower speed limit on the rightmost lane compared to the speed limits of other lanes on the mainline.

In this study, we conducted a baseline simulation under the condition without control (the speed limit values for all

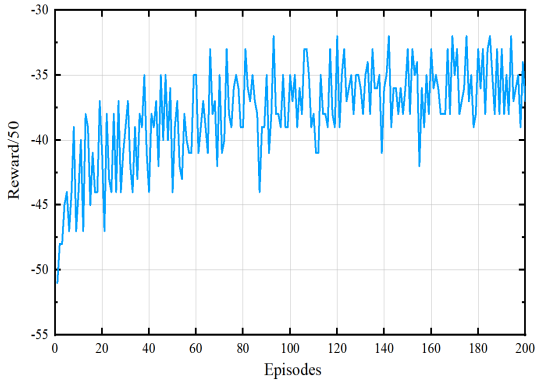

Fig. 6. Reward variation with learning process

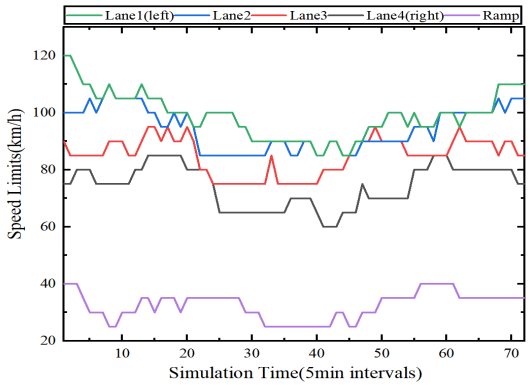

Fig. 7. Speed limit variation on five Lanes with simulation time

five lanes were constant at $\{100, 100, 120, 120, 40\}(km/h)$). We collected the average travel speeds (ATS) at each time step under two scenarios: with and without VSL. As shown in Fig. 8, it can be seen that when the traffic volume is low at the beginning and has not reached the critical density K_m, the ATS under both scenarios is roughly similar. As the number of vehicles increases to the critical density K_m, traffic congestions occur without control, leading to a significant decrease in ATS. However, when adopting VSL control, the ATS can still be maintained at a certain level, avoiding the sharp decrease in traffic capacity described in Section II.

Table II shows the ATS per hour on the road section with and without VSL. It can be seen that during the period of 13:00-14:00, when there is a relatively high traffic flow, the effect of ATS is the best, reaching 34.7%. In the first hour, due to low traffic volume, the ATS with and without VSL are similar, and VSL does not help much in improving traffic efficiency. The ATS during the entire simulation period is increased by 12.3%. Combining Table II and Fig. 8, it can be concluded that when the traffic volume continues to increase to reach the critical density K_m, using VSL can improve traffic efficiency to a certain extent. However, when traffic volume is low, VSL does not help much in improving traffic efficiency.

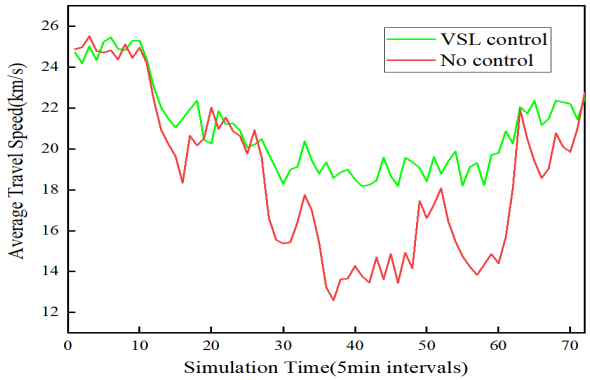

Fig. 8. Variation of ATS with simulation time

TABLE II. PERFORMANCE OF ATS PER HOUR WITH AND WITHOUT VSL

Time	ATS with VSL(m/s)	ATS without VSL(m/s)	Improvement
10:00-11:00	24.7	24.6	0.5%
11:00-12:00	21.3	20.5	3.9%
12:00-13:00	19.5	16.9	15.2%
13:00-14:00	18.8	13.9	34.7%
14:00-15:00	19.1	15.6	22.2%
15:00-16:00	21.7	20.0	8.4%
10:00-16:00	20.9	18.6	12.3%

VI. CONCLUSION

This paper proposes a VSL method based on TD3 to reduce traffic congestion in highway merging areas. The designed controller can adjust the speed limits of the mainline and ramp to keep the flow in the bottleneck area close to its capacity. We selected a road segment of the Hangzhou-Ningbo Expressway in Zhejiang Province, China, collected traffic flow data during the Spring Festival holiday, conducted simulations on the SUMO platform, and set a series of conditions to make the simulation as close as possible to real traffic conditions.

The simulation results show that our proposed method is effective in high-traffic volume situations. Using ATS as the indicator, using VSL improves traffic efficiency by 12.3% compared to not using it. At the same time, VSL shows different effects under different traffic volumes and can significantly improve the traffic parameters of highways when the traffic volume is high. Therefore, VSL can be used reasonably to improve traffic efficiency during peak travel periods such as holidays.

The limitation of this study is that we assumed all vehicles fully comply with the posted speed limits, which may be challenging to achieve in real-world scenarios. Moreover, further investigation of VSL's performance under different traffic volumes would be beneficial to determine the conditions for activating VSL. In future work, the proposed controller will be deployed and tested in real-world scenarios.

In summary, our proposed method has broad application prospects and practical significance in the field of intelligent

transportation systems. This approach can provide technical support for intelligent traffic management and improve people's travel experiences.

REFERENCES

[1] J. Sun, Z. Li, and J. Sun, "Study on traffic characteristics for a typical expressway on-ramp bottleneck considering various merging behaviors," *Physica A: Statistical Mechanics and its Applications*, vol. 440, pp. 57–67, 2015.

[2] M. T. Islam, M. Hadiuzzaman, J. Fang, T. Z. Qiu, and K. El-Basyouny, "Assessing mobility and safety impacts of a variable speed limit control strategy," *Transportation research record*, vol. 2364, no. 1, pp. 1–11, 2013.

[3] R. Yu and M. Abdel-Aty, "An optimal variable speed limits system to ameliorate traffic safety risk," *Transportation research part C: emerging technologies*, vol. 46, pp. 235–246, 2014.

[4] B. Othman, G. De Nunzio, D. Di Domenico, and C. Canudas-de Wit, "Variable speed limits control in an urban road network to reduce environmental impact of traffic," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 1179–1184.

[5] R. C. Carlson, I. Papamichail, and M. Papageorgiou, "Local feedback-based mainstream traffic flow control on motorways using variable speed limits," *IEEE Transactions on intelligent transportation systems*, vol. 12, no. 4, pp. 1261–1276, 2011.

[6] 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.

[7] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

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

[9] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International conference on machine learning*. PMLR, 2018, pp. 1587–1596.

[10] H. Wei, G. Zheng, H. Yao, and Z. Li, "Intellilight: A reinforcement learning approach for intelligent traffic light control," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2496–2505.

[11] Z. Li, P. Liu, C. Xu, H. Duan, and W. Wang, "Reinforcement learning-based variable speed limit control strategy to reduce traffic congestion at freeway recurrent bottlenecks," *IEEE transactions on intelligent transportation systems*, vol. 18, no. 11, pp. 3204–3217, 2017.

[12] M. Gregurić, K. Kušić, F. Vrbanić, and E. Ivanjko, "Variable speed limit control based on deep reinforcement learning: A possible implementation," in *2020 International Symposium ELMAR*. IEEE, 2020, pp. 67–72.

[13] Y. Wu, H. Tan, L. Qin, and B. Ran, "Differential variable speed limits control for freeway recurrent bottlenecks via deep actor-critic algorithm," *Transportation research part C: emerging technologies*, vol. 117, p. 102649, 2020.

[14] C. Wang, J. Zhang, L. Xu, L. Li, and B. Ran, "A new solution for freeway congestion: Cooperative speed limit control using distributed reinforcement learning," *IEEE Access*, vol. 7, pp. 41 947–41 957, 2019.

[15] F. L. Hall and K. Agyemang-Duah, "Freeway capacity drop and the definition of capacity," *Transportation research record*, no. 1320, 1991.

[16] E. Ivanjko, G. O. Melo, K. Kušić, and M. Gregurić, "Comparison of controllers for variable speed limit using realistic traffic scenarios," in *2018 International Symposium ELMAR*. IEEE, 2018, pp. 39–42.

[17] 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.

[18] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *2018 21st international conference on intelligent transportation systems (ITSC)*. IEEE, 2018, pp. 2575–2582.

[19] M. Papageorgiou and A. Kotsialos, "Freeway ramp metering: An overview," *IEEE transactions on intelligent transportation systems*, vol. 3, no. 4, pp. 271–281, 2002.

[20] M. Quddus, "Exploring the relationship between average speed, speed variation, and accident rates using spatial statistical models and gis," *Journal of Transportation Safety & Security*, vol. 5, no. 1, pp. 27–45, 2013.