Optimal Speed Limit Control for Network Mobility and Safety: A Twin-delayed Deep Deterministic Policy Gradient Approach

Fatima Afifah[a], Zhaomiao Guo[b,*]

[a] *Department of Civil, Environmental, and Construction Engineering*
University of Central Florida
[b] *Maseeh Department of Civil, Architectural and Environmental Engineering*
University of Texas at Austin

Abstract

Variable speed limit control (VSLC) has emerged as a promising approach for improving traffic safety and reducing congestion. However, local adjustment of VSLC may have broader impacts on the transportation network performance due to driver rerouting. This study proposes a deep reinforcement learning (DRL) controller based on twin-delayed deep deterministic policy gradient (TD3) algorithm to improve mobility and safety over a small-scale interconnected network considering rerouting behavior. The proposed DRL-based VSLC controller is designed to handle a large number of possible speed limits at each time step by utilizing a deep actor-critic framework. The study also experiments with different reward functions to characterize network mobility, safety, and traffic oscillation. Additionally, we investigate the sensitivity of the control algorithm across different traffic patterns, driving behavior, and VSLC locations, where the proposed TD3 algorithm demonstrated robustness and generalizability. Our findings indicate that implementing network-specific reward functions leads to improvements in traffic safety and mobility. Specifically, it results in a 3.84% enhancement in overall safety, as measured by time-to-collision metrics, and a 33.2% improvement in mobility by reducing total travel time compared to the scenario without VSL control. While comparable in safety performance, TD3 outperforms deep deterministic policy gradient (DDPG) algorithm by 15.1% in terms of mobility. This study contributes to the understanding of the impacts of VSLC on transportation networks and provides insights into effective ways of implementing VSLC to improve network mobility and safety.

Keywords: variable speed limit, twin-delayed deep deterministic policy gradient (TD3), traffic rerouting, network safety, network mobility

[*] (Corresponding Author) Assistant Professor, Maseeh Department of Civil, Architectural and Environmental Engineering, University of Texas at Austin, Email: zguo@utexas.edu

1. Introduction

With increasing levels of vehicle connectivity and automation, variable speed limit control (VSLC) empowers traffic management agencies to dynamically regulate traffic progression based on prevailing system conditions, such as traffic and weather events. VSLC has demonstrated the potential to improve traffic safety by reducing speed variance and increasing average headway (Islam et al., 2013; Li et al., 2016; Abdel-Aty et al., 2006; Lee et al., 2004; Park et al., 2008). Additionally, it offers a cost-effective approach to mitigating re-current (Emmerink et al., 1995) and non-recurring congestion conditions (Chen et al., 2016; Kwon et al., 2006) and reduces the need to upgrade the existing infrastructure (Hoogendoorn et al., 2013; Li et al., 2014; Farrag et al., 2020). As by-products of improving safety and mobility, research also shows the potential of VSLC to reduce vehicle emission (Grumert et al., 2015; Bel and Rosell, 2013) and dampen traffic shock wave (Hegyi et al., 2005b, 2008).

With increasing connectivity and on-board computation capabilities, vehicles can receive and process information on transportation states (e.g., travel time and variable speed limit) in real-time and reroute to their destinations based on their current information. As a result, VSLC not only directly influences the travel speed and travel time of certain links, but also indirectly causes broader traffic redistribution due to traffic rerouting. The traffic redistribution affects the mobility of the whole transportation system despite each vehicle only aiming to improve its mobility. In addition, vehicle rerouting may also lead to a higher frequency of lane changing before intersections/interchanges, which increases the likelihood of vehicle collision and the formation of local bottlenecks. Therefore, it is imperative to consider these direct and indirect effects at a network level when deciding the optimal VSLC strategies.

Different methodologies have been explored for VSLC, such as rule-based (Papageorgiou et al., 2008; Li and Ranjitkar, 2015), model-based including feedback-based control(Carlson et al., 2013; Jin and Jin, 2015; Karafyllis and Papageorgiou, 2019) and model predictive control (MPC) (Hegyi et al., 2005a; Han et al., 2017; Li et al., 2019), and reinforcement learning (RL)-based methods (Kušić et al., 2020a; Li et al., 2017; Walraven et al., 2016; Vrbanić et al., 2021; Kušić et al., 2020a; El-Tantawy et al., 2013; Wu et al., 2020c). Rule-based methods use predefined rules to set speed limits based on traffic conditions, while model-based approaches utilize mathematical traffic flow models to predict the effects of different speed limits. However, these methods may not easily handle complex scenarios or adapt to changing traffic conditions. Model-based methods dynamically adjust speed limits on roadways based on real-time traffic conditions and system feedback or predictions to optimize traffic flow, enhance safety, and mitigate congestion. However, model-based methods can be constrained by their reliance on idealized traffic models, extensive numerical computations, and the need for calibration of numerous parameters. In recent years, RL-based approaches have emerged as a promising alternative solution to overcome the limitations of model-based methods (Kušić et al., 2020b). RL allows the development of smart controllers that automatically adjust speed limits based on traffic conditions. These RL-VSLCs differ in terms of the selection of system states, reward functions, and learning algorithms (Kušić et al., 2020a; Li et al., 2017; Walraven et al., 2016; Vrbanić et al., 2021). Q-learning (QL) is commonly used in RL-VSLCs, but it faces challenges with large state spaces and continuous state-action representations (Kušić et al., 2020a; El-Tantawy et al., 2013; Wu et al., 2020c). More recent approaches, such as actor-critic algorithms, have demonstrated improved efficiency and effectiveness in VSLCs (Wu et al., 2020c; Gregurić et al., 2022).

Although there have been studies using RL methods to achieve optimal VSLC strategies, most studies focus on enhancing traffic performance at intersections or link levels (Kušić et al., 2020a; Wu et al., 2020c; Wang et al., 2019a). For instance, Wu et al. (2020c) used a deep deterministic policy gradient (DDPG) algorithm to alleviate bottlenecks at a freeway section, demonstrating improved safety and mobility through VSLC. Nonetheless, these studies often overlook potential impacts on surrounding links. Zhu and Ukkusuri (2014) presented a dynamic speed limit control model utilizing a link-based dynamic network loading approach based on a cell transmission model (CTM) and reinforcement learning algorithm, demonstrating significant reductions in total travel time and emissions in a stochastic traffic network. However, network-level traffic rerouting and the impacts of VSL placement locations were not considered. VSLCs have also been investigated in other different contexts, such as temporary speed limit control for work zone management (Lin et al., 2004), dynamic lane management (e.g., high-occupancy vehicle (HOV) lanes or reversible lanes) (Radwan et al., 2011), environmental impact reduction (Vrbanić et al., 2022), ramp-metering for traffic flow control (Dadashzadeh and Ergun, 2019). None of these studies focuses on the network-level impacts of VSLC. Optimizing VSLC strategies is challenging due to adaptive decision-making and interconnected network topology. Addressing the broader implications of VSLC implementation within interconnected transportation networks is still limited (Afifah et al., 2023), hindering comprehensive analysis of VSLC's effects on transportation systems.

The main goal of this study is to optimize the VSLC strategies considering potential impacts on safety and mobility at a network level. To achieve this goal, we propose a deep reinforcement learning (DRL) approach enabling dynamic speed limits aiming to reduce network congestion and accident risks. Our approach employs a deep actor-critic framework, efficiently handling a large number of possible speed limits at each time step. The main contributions of this study are summarized as follows.

1) We customized a deep actor-critic RL framework for the optimal control of VSLCs to optimize mobility and safety over a small-scale transportation network considering traffic rerouting.

2) We investigated the transferability of the proposed RL control algorithm with different traffic and driving behavior attributes and compared the efficacy with other state-of-the-art RL algorithms.

The remainder of this paper is structured as follows. Section 2 presents the customized RL algorithm to address our problem, including the appropriate selection of reward functions, state representations, and action spaces. Section 3 presents the environment setup and the result analyses. Section 4 concludes the study and discusses possible future extensions and policy implications.

2. Methodology

In recent years, policy-based algorithms such as DDPG (Lillicrap et al., 2015) have gained popularity in different fields, including robotics (Jesus et al., 2019; Xu et al., 2018), the electricity supply industry (Liang et al., 2020), energy systems (Li et al., 2021; Wu et al., 2020a), wireless communication (Qiu et al., 2019), transportation (Wang et al., 2019b; Casas, 2017; Wu et al., 2020b), and also in VSL related studies (Gregurić et al., 2022; Wu et al., 2020c). As a model-free off-policy RL algorithm, DDPG combines deterministic policy gradient (DPG) and deep Q-network (DQN). Although DDPG has provided promising results for real-world problems in different research fields, it may suffer from the following issues (Fujimoto et al., 2018):

1) Overestimation Bias: The overestimation bias in the DDPG algorithm occurs when the learned value function overestimates the true value of the state-action pairs. This overestimation bias can be problematic as it can lead to unstable and suboptimal behavior of the algorithm. The overestimation bias arises when the policy and value networks are

jointly learned using the same samples. The policy improvement step in DDPG updates the policy based on the current value estimate, which can be inaccurate due to the overestimation bias. This inaccurate value estimate can then lead to the selection of suboptimal actions by the policy, which in turn can lead to poor convergence of the algorithm.

2) Accumulation Error: The accumulation error in DDPG occurs because the value function estimate is based on an estimate of a subsequent state, leading to a buildup of error over time. This error is exacerbated in a function approximation setting resulting in residual TD-error after each update.

For our study, we adapted the twin-delayed deep deterministic policy gradient (TD3) framework (originally proposed by Fujimoto et al. (2018)) to implement VSLC for improving network mobility and safety. TD3 was chosen for its ability to address the overestimation bias and accumulation errors common in reinforcement learning algorithms, providing a stable and effective control. In the remainder of this section, we first discuss the key features of TD3 using general notations in subsection 2.1. Then, we present the selection of state space, reward functions, and action space for our VSLC problem and the simulation environment setup in Section 2.2. Finally, we summarize the TD3 algorithm for our VSLC problem in Section 2.3.

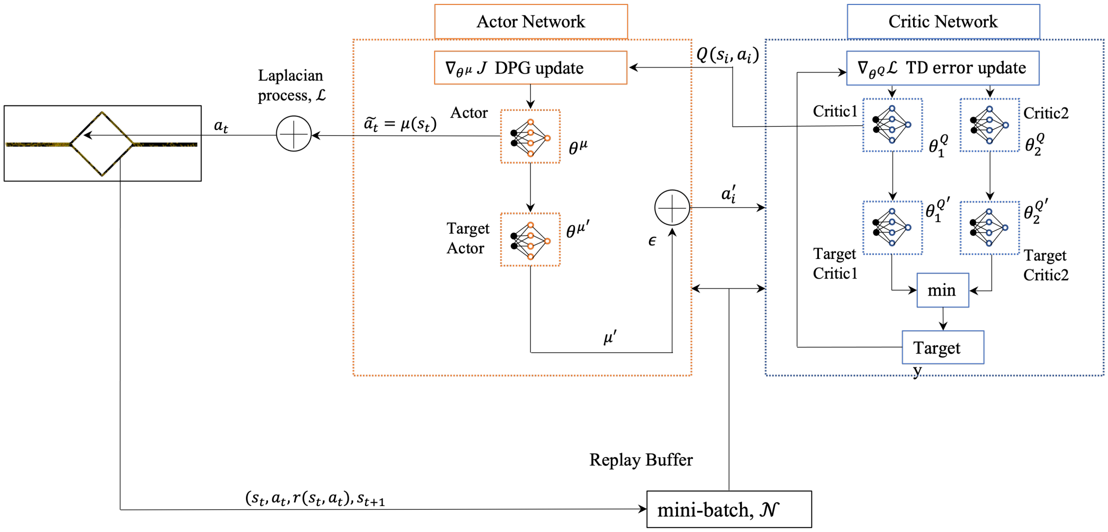


Figure 1: Structure of TD3

2.1. Twin-delayed Deep Deterministic Policy Gradient

Figure 1 shows the algorithm structure of TD3. TD3 consists of two independent critic networks and one actor network, each having an online and a target network. To mitigate overestimation bias in Q-learning, TD3 employs a clipped variant of Double Q-learning that utilizes two independent critic networks to estimate two Q-values for each state-action pair. During the update process, the minimum of the two Q-value estimates is used as the target value in the Bellman equation update, thereby reducing overestimation bias (see Equation (1)).

$$y_i = r_i + \gamma \min_{j=1,2} Q_j^{'}(s_i^{'}, (\mu^{'}(s_i^{'}; \theta^{\mu^{'}}) + \epsilon); \theta_j^{Q^{'}}) \tag{1}$$

where i is an element in a minibatch N (randomly selected for training) which consists of a tuple of s (the current state), a (action), r (reward), $s^{'}$ (the next state). In this Bellman equation, y is the target Q value. $Q^{'}$ and $\theta^{Q^{'}}$ are the Q value and the weights from the target critic network, respectively. $\mu^{'}$ and $\theta^{\mu^{'}}$ are the policy and weights from the target actor network, respectively. γ is a discount factor (e.g., 0.9). However, when the difference between the two Q-value estimates exceeds a predefined clipping range, the target value is clipped to lie within this range.

To further enhance the stability of the learning process, TD3 adopts target policy smoothing techniques when forming the target critic. This involves introducing noise to the target action generated by the target actor network during the Bellman equation update. The amount of noise added is determined by a small standard deviation clipped Gaussian distribution (Equation (2)).

$$\epsilon \sim clip(\mathcal{N}(0, \sigma), -c, c) \tag{2}$$

where, $-c$ and c are the minimum and maximum noise allowed. The resulting smoothed action is then used as the target action for computing the Q-values in the critic network update. By regularizing the updates and preventing overfitting to the current policy, target policy smoothing reduces instability in the learning process.

To update the online critic network, the mean-squared error between the predicted Q-value and the target Q-value is calculated and used to compute the gradient with respect to the critic network parameters (Equation (3) and (4)). The weights of the online critic networks (θ_1^Q and θ_2^Q) are then updated using an optimizer, such as stochastic gradient descent (SGD) or Adam, to minimize the mean-squared errors.

$$\nabla_{\theta_1^Q} \mathcal{L}(\theta_1^Q) = \frac{1}{N} \sum_{i=1}^{N} [\nabla_{\theta_1^Q} (y_i - Q_1(s_i, a_i; \theta_1^Q))^2] \tag{3}$$

$$\nabla_{\theta_2^Q}\mathcal{L}(\theta_2^Q) = \frac{1}{N}\sum_{i=1}^{N}[\nabla_{\theta_2^Q}(y_i - Q_2(s_i, a_i; \theta_2^Q))^2] \tag{4}$$

In TD3, the actor network is updated using the DPG algorithm. The gradient of the expected return with respect to the actor parameters is computed using the chain rule (see Equation (5)), and this gradient is used to update the actor weights in the direction that increases the expected return.

$$\nabla_{\theta^\mu} J = \frac{1}{N}\sum_{i=1}^{N}[\nabla_a Q_1(s, a; \theta_1^Q)_{|s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu}\mu(s; \theta^\mu)_{|s=s_i}] \tag{5}$$

Similar to DDPG, the learning process is stabilized using a "soft update". This is accomplished by gradually changing the target network parameters at regular intervals. Equations (6) and (7) are the soft updates for the target actor and the two target critic networks respectively. In this study, τ is selected to be 0.005 to ensure stability.

$$\theta^{\mu^{'}} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu^{'}} \tag{6}$$

$$\theta_j^{Q^{'}} \leftarrow \tau\theta_j^Q + (1-\tau)\theta_j^{Q^{'}}, \forall j \in \{1, 2\} \tag{7}$$

Without loss of generality, we consider a continuous action space for our VSLC problem [1]. The advantage of off-policy algorithms, such as TD3, is that exploration can be separated from the learned networks so that they are more suitable for models with a continuous action space, which are typically challenging to explore all possible combinations in a reasonable amount of time. We add noise to randomize the action using a Laplacian process (Jinnai et al., 2019) which follows the Laplace distribution, also known as a double exponential distribution as shown in Equation (8).

$$a_t \sim \mathcal{L}(\tilde{a_t}, b_t) \tag{8}$$

where $\mathcal{L}(\tilde{a_t}, b_t)$ has a probability density function of $\frac{1}{2b_t}\exp(-\frac{|a_t - \tilde{a_t}|}{b_t})$, $\tilde{a}$ is the action we get from the online actor network, and b_t is the scale parameter with a value of 1.5 in this study.

[1] For conventional VSLC, a speed limit is usually set at a 5 mph increment. However, we envision that when connected and automated vehicles (CAVs) become prevailing, it is potentially beneficial for VSL to be set as a continuous variable. After learning an optimal continuous VSL, we can round up/down to the nearest 5 mph for practical implementation if needed.

By using Equation (8), we get a random action, which can be further clipped into a range $[a_{min}, a_{max}]$ using Equation (9) if needed to ensure practicability (e.g., speed limits cannot be lower than 5 mph.)

$$a_t = clip(\mathcal{L}(\tilde{a_t}|b_t), a_{min}, a_{max}) \tag{9}$$

2.2. State, Reward, Action, and Simulation Environment

The selection of state, reward, and action directly influences the performance of RL algorithms. In this subsection, we discuss the selected state, reward, and action functions for the VSLC problem based on the performance in our numerical experiments.

State:. For the VSLC problem, we have chosen length-based occupancy of all links as the state. Length-based occupancy is the ratio between the sum of the lengths of the vehicles to the length of the road section in which those vehicles are present, usually represented as a percentage.

Reward:. In this study, we explore three different objectives, including safety, mobility, and traffic oscillation. The selection of reward functions for different objectives is presented below. Notice that other reward functions can be flexibly constructed to guide the VSLC towards different objectives, such as driving comfortability, energy efficiency, or a combination of multiple objectives. We will leave other control objectives for future research.

Objective - Improving safety:

Time-to-collision (TTC) is a commonly used metric in the field of traffic safety to assess the likelihood of a collision between two vehicles. It is defined as the estimated time it would take for two vehicles to collide if they maintain their current trajectory and speed (Hydén, 1996). The formula is as follows.

$$TTC_i = \begin{cases} \frac{X_i(t) - X_{i-1}(t) - l_i}{\dot{X}_i(t) - \dot{X}_{i-1}(t)}, & \text{if} \quad \dot{X}_i\left(t\right) > \dot{X}_{i-1}\left(t\right) \\ \infty, & \text{Otherwise} \end{cases} \tag{10}$$

where:

$\dot{X}_i$: the speed of the vehicle i

X_i : the rear position of the vehicle i

l_i : the length of vehicle i

$i-1$: the vehicle in front of vehicle i

TTC is a useful metric for safety rewards in RL. If an agent takes an action that leads to a decrease in TTC, then that action can be considered unsafe and penalized accordingly. However, we note that collision risk should not be linear to TTC as the sensitivity of a larger TTC may not indicate the change of safety performance as much as smaller TTC. In this study, we adopt a log transformation of the TTC, as shown in Equation (11).

$$TTC_{i,new} = \log(TTC_i) \tag{11}$$

Conceptually, Log(TTC) can be an effective safety metric in transportation systems due to its ability to provide a continuous and sensitive measure of safety dynamics. Unlike discrete safety events such as emergency braking, which offer insights into specific instances of safety interventions, log(TTC) offers a nuanced representation of safety risks by continuously monitoring variations in hazard proximity over time. Moreover, log(TTC) offers a more accurate evaluation of safety compared to traditional TTC metrics by emphasizing the smaller TTC at critical moments. This continuous nature allows for proactive interventions to mitigate potential hazards effectively.

Objective - Improving mobility:

In line with previous studies, we adopt total travel time (TTT) as the measurement for mobility, as defined in Equation (12). Intuitively, the reward function is negative of TTT so that a higher TTT value (i.e., vehicles take longer to traverse the network) yields lower rewards.

$$\mathrm{TTT} = \sum_{a \in A} \mathrm{CurrentTT_a} \times N_a \tag{12}$$

where:

TTT : total travel time

$\mathrm{CurrentTT_a}$: the current travel time on a link a

N_a : the number of vehicles on the link a

Objective - Dampening traffic oscillation:

In addition to utilizing VSLC to enhance network mobility and safety, dampening traffic oscillation is a typical traffic control strategy to ensure that vehicles are evenly distributed to each of the alternative routes at all times.

$$\Delta\text{NumVeh} = \frac{1}{2} \sum_{i \neq j \in \mathcal{R}} |\text{NumVeh}_i - \text{NumVeh}_j| \tag{13}$$

When there are only two alternative routes, the reward function for dampening traffic oscillation can be defined in Equation (14).

$$\Delta\text{NumVeh} = |\text{NumVeh}_1 - \text{NumVeh}_2| \tag{14}$$

where:

ΔNumVeh : total relative difference in the number of vehicles in different routes

$\mathcal{R}$: The set of alternative routes

NumVeh_i : number of vehicles in route i

Action:. In this study, the VSLC agent uses TD3-based RL to control traffic flow with variable speed limits. We consider continuous speed limit control actions a_t considering the potential that speed limits can be controlled and enforced in a higher granularity when vehicles are connected and automated. The speed limit is calculated as $V_0 + I * a_t$, where V_0 is the minimum speed (e.g., 30 mph), I is the incremental interval (e.g., 5 mph), and a_t is the action taken at time t, including added noise (Equation (8)). The continuous speed limit control can be converted to discrete speed limits for practical implementation for human drive vehicles using an equation $V_0 + I * \text{round}(a_t)$.

2.3. TD3 algorithm summary

We train our TD3 model with the state, reward, and action defined in the previous subsections. The summary of the TD3 algorithm is presented in Algorithm 1.

While formal hyperparameter optimization techniques, such as grid search and random search, were considered in the literature, they were computationally infeasible for our computational hardware given the involvement of complex microscopic traffic dynamics. Therefore, we employed a practical, iterative tuning approach, adjusting key hyperparameters incrementally based on observed performance in multiple traffic scenarios. This method allowed us to achieve stable convergence and consistent solution quality without excessive computational costs. For

Algorithm 1 Twin-delay deep deterministic policy gradient (TD3)

```
Set observation space, action space, and reward function to implement VSLC.
Set algorithm parameters: $\tau$, $b_t$, $d$, $N$.
Randomly initialize two critic networks $Q_1$, $Q_2$ and actor network $\mu$ with parameters $\theta^{Q_1}$, $\theta^{Q_2}$, and $\theta^{\mu}$, respectively.
Initialize target network weights $\theta^{Q'_1} \leftarrow \theta^{Q_1}$, $\theta^{Q'_2} \leftarrow \theta^{Q_2}$, and $\theta^{\mu'} \leftarrow \theta^{\mu}$.
for each episode do
  Start traffic simulation with SUMO and obtain initial state $s_1$
  for each time step $t$ in episode do
    Select an action from the current policy $\tilde{a_t} = \mu(s_t; \theta^{\mu})$ and perform the exploration method following Laplacian process (Equation (8)) and clipping (Equation (9)) to obtain $a_t$.
    Calculate $V_t = V_0 + Ia_t$ and feed it into SUMO as the speed limit.
    Obtain reward $r(s_t, a_t)$ and the next state $s_{t+1}$
    Store a tuple $(s_t, a_t, r(s_t, a_t), s_{t+1})$ in replay memory
    Sample minibatch of $N$ transitions $(s_{t_i}, a_{t_i}, r_{t_i}, s_{t_i+1}, i \in \{1, 2, ..., N\})$ from replay memory.
    for each $i \in \{1, 2, ..., N\}$ do
      Update target Q value $y_{t_i}$ following (1).
    end for
    Update critics $\theta_j^Q \leftarrow \arg\min_{\theta_j^Q} \frac{1}{N} \sum_{i \in \{N\}} (y_{t_i} - Q_j(s_{t_i}, a_{t_i}; \theta_j^Q))^2, j \in \{1, 2\}$
    if $t \bmod d == 0$ do
      Update $\theta^{\mu}$ by the DPG, with gradient defined in (5)
      Update target actor and critic networks based on (6) and (7)
    end if
  end for
end for
```

instance, the learning rate of 0.002 was selected after testing various values and observing that it provided the best trade-off between stability and convergence speed. Higher values, such as 0.005, led to instability, whereas lower values slowed convergence significantly. Similarly, other parameters like discount factor, batch size, policy noise, and policy update frequency were tuned iteratively.

Table 1 provides all the hyperparameters used in this study through numerical experiments. Future work can incorporate additional learning-based approaches (such as data augmentation (Li et al., 2020), reward shaping (Wu et al., 2021), and exploration strategies (Xu et al., 2021)) to further improve the performance.

Table 1: Hyperparameter and Their Effects

Hyperparameter	**Tested Range**	**Selected Value**	**Effect**
Learning Rate (α)	[0.001, 0.002, 0.005, 0.01]	0.002	Determines the learning step size during the gradient descent update for the actor network. The selected value provided a good balance between convergence speed and stability, avoiding both overshooting and slow learning.
Learning Rate (β)	[0.001, 0.005, 0.01]	0.005	Determines the step size during the gradient descent update for the critic network. A higher learning rate for the critic helped in faster value estimation without causing instability.
Discount Factor (γ)	[0.85, 0.9, 0.95]	0.9	Determines the importance of future rewards. The selected value ensures that the agent takes both immediate and long-term rewards into account, with future rewards moderately discounted.
Batch Size	[32, 64, 128, 256, 512]	256	Affects the stability and efficiency of the learning process. The selected batch size provided stable gradient estimates without overfitting, balancing computational efficiency and learning stability.
Replay Memory Size (M)	[500,000, 1,000,000, 2,000,000]	$1,000,000$	Stores past experiences for training the model. A large buffer size allows the model to learn from a diverse set of experiences, improving generalization and stability.
Tau (τ)	[0.001, 0.005, 0.01]	0.005	The soft update parameter for the target networks. The selected value ensures slow updates, providing stable training and preventing drastic changes that could destabilize the learning process.
Policy Noise	[0.005, 0.01, 0.02]	0.01	Noise added to the target policy during critic update helps in exploration by introducing randomness. The selected value ensured sufficient exploration without causing instability.
Policy Update Frequency (d)	[1, 2, 3, 4, 5]	2	Updating the policy network every 2 iterations helped stabilize the training process by allowing the critic network to provide accurate value estimates.

3. Experiment Results

At the current stage, limited RL algorithms have been directly trained on real transportation systems [2] In order to advance the understanding of the potential of RL algorithms, high-fidelity

[2] One of the main concerns of training RL-based algorithms on real transportation systems is that “bad” actions may be implemented to cause severe congestion or safety issues. There are several methods one can

simulators are used in the existing literature.. We used the open-source Simulation of Urban MObility (SUMO) software (Lopez et al., 2018) for our simulation experiment, which is known for its flexibility and has been used in various RL-based traffic control studies (Zeynivand et al., 2022; Wang et al., 2022; Kušić et al., 2021; Boukerche et al., 2021). We tuned the microscopic car following & lane changing models, and dynamic routing and rerouting behavior to approximate real microscopic driving behaviors and traffic dynamics. The outputs of microscopic driving behavior models are aggregated to reflect realistic macroscopic traffic fundamental diagrams, see Figure 2. Our vehicle types were connected passenger cars, and we assigned each vehicle an individual ”speedFactor” following a normal distribution (mean=1, standard deviation=0.1, range=[0.2,2]) to achieve varied desired driving speeds. For the car-following model, we used the Intelligent Driver Model (IDM) proposed by Treiber and Kesting (2017) with a lane change gap acceptance setting of 1 second. The default lane-changing model in SUMO was used with the default ”lcspeedGain” value of 1, where a higher value of ”lcspeedGain” refers to the eagerness of vehicles to perform lane changing to gain speed. Vehicles were allowed to re-route based on the real-time travel times using the *traci.vehicle.rerouteTraveltime* function [3].

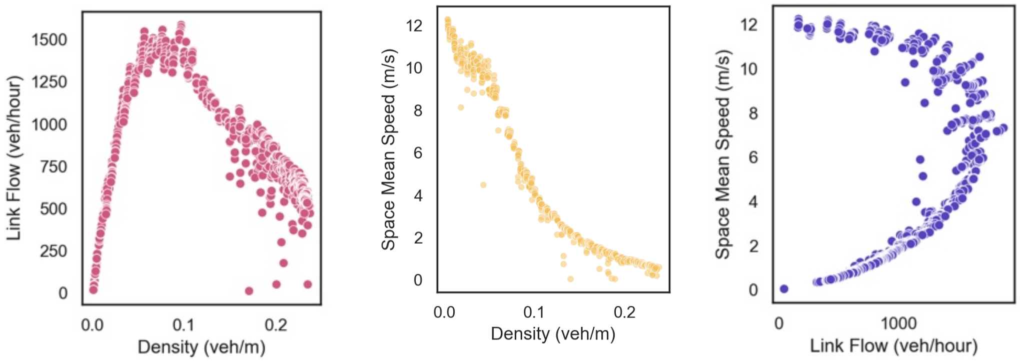


Figure 2: Fundamental Diagram

We implemented the TD3 algorithm on a four-node network (see Figure 3), through which

employ to enhance the robustness of RL-based approaches during online training. For example, from a control algorithm perspective, we can penalize the severe congestion or safety issues more than mild situations. From a data perspective, we can use digital twins to assist the online training of RL-based approaches which can significantly avoid the implementation of adverse actions. From a practical perspective, one can enforce the smoothness and bounds of the speed limit control to avoid adverse control outcomes. However, there is a trade off between the overall control efficiency and reliability. So a quantitative way to decide their balance will also be a promising research topic.

[3] For more information, visit `https://sumo.dlr.de/docs/TraCI/Change_Vehicle_State.html`

vehicles traversed from node 1 to node 6 via two alternative routes (i.e., path 1-2-3-5-6 and path 1-2-4-5-6). All links within the network span 1 mile in length, with Link 1-2 and Link 5-6 accommodating 4 lanes each, while the remaining links have 2 lanes each.

We consider a steady state of OD traffic flow in our network, which is 3000 vehicles/hour. The total simulation time is around 2400 seconds and approximately 2000 vehicles are simulated. These vehicles were assumed to utilize real-time transportation system information, such as travel time, to calculate the shortest route to their destination and reroute accordingly using the traci.vehicle.rerouteTraveltime function. The selection of the four-node network was deliberate, as it effectively captures route choices, lane changes, and traffic diverge/merge behaviors, thereby enabling insights into the impacts of VSLC on network safety and mobility despite its simplicity. There are two main reasons why the simple network serves the purpose for this paper. First, this work is one of the first to investigate the impacts of VSLC considering rerouting. Using a small-scale network allows us to have a more controlled environment to focus on selected rerouting options given local VSLC to clearly observe and analyze the impact of different VSLC objectives. Second, considering a large-scale network is a straightforward process but introduces a significant level of computational and calibration difficulties, which is beyond the scope of this study. For example, the current simulation framework depends on SUMO, which is a microscopic traffic simulator. While SUMO is known to be high fidelity in capturing microscopic driving behavior, it lacks scalability for large networks. The training of the proposed RL algorithm relies on hundreds of episodes to reach a reliable convergence solution, so a microscopic-simulation-based RL algorithm may not be suitable for a large network. To mitigate this issue, there are two potential improvements. First, we can adopt a mesoscopic simulator to balance the accuracy of traffic modeling and computational scalability. Second, to mitigate the computational issues, a multi-agent framework with zero-shot learning can be adopted to leverage the existing single-agent trained model within a multi-agent framework. Zero-shot learning techniques can be employed to enable each agent to apply the learned policies directly to new sections of the network without a system-level retraining. In the base case, VSLC was implemented at Link 2-3 to regulate the traffic. We rounded the optimal speed limits to the nearest 5 mph increments, ranging from 30 mph to 65 mph for VSL-controlled roads, while other roads maintained a constant 40 mph speed limit. In this section, network safety is measured using the $log(TTC)$ and network mobility using TTT as discussed in detail in Section 2.2. A simulation runtime was approximately 54 hours for 800 episodes on a Linux system with a 16 cores Intel(R) Core(TM) i9-9900K CPU @ 3.60GHz processor, NVIDIA GEFORCE RTX 2080 Ti GPU, and 16 GB of

RAM memory.

In the remainder of this section, we first compared different VSLC traffic control strategies for the base case. Then, we compared our proposed model against different algorithms, assessed the proposed model's generalization capabilities, and performed sensitivity analyses on the placement of VSLC in the network to improve traffic safety and mobility.

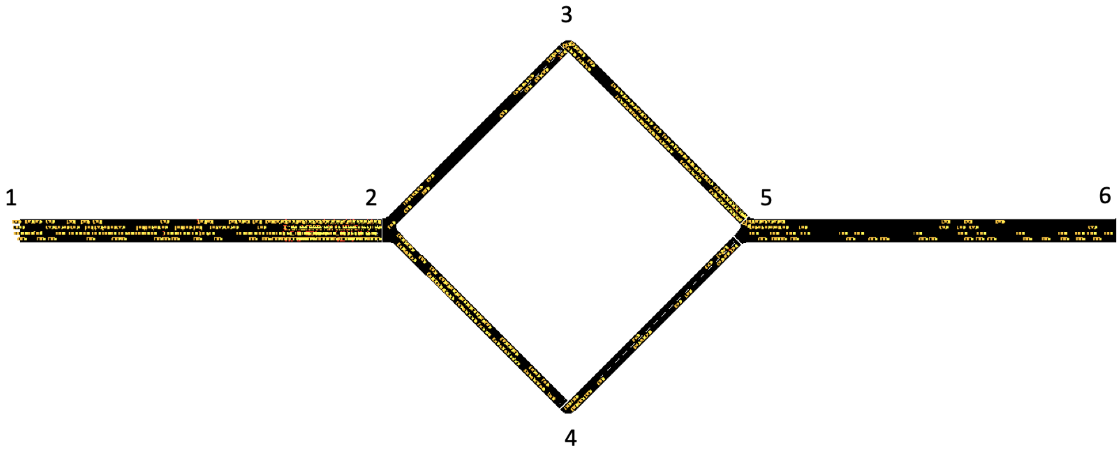


Figure 3: Four-Node Network

3.1. Assessment of Different Control Objectives in the Network

We compared four traffic control objectives: no VSLC, VSLC for mobility, VSLC for safety, and VSLC for dampening traffic oscillation. The comparison of different performance metrics from these four traffic control objectives is summarized in Table 2.

Table 2: Performance of Different Traffic Control Objectives

	Routing		**Lane change frequency (%)**	$\sum \log(TTC)$	**Total travel time (sec)**
	% of vehicles in Route 1	**% of vehicles in Route 2**			
No VSL	49.41	50.59	13.97	9.526×10^5	9.188×10^7
VSLC implementation (mobility)	64.35	35.65	10.13	9.149×10^5	6.576×10^7
VSLC implementation (safety)	26.38	73.62	14.25	1.002×10^6	1.349×10^8
VSLC implementation (traffic oscillation)	50.32	49.68	26.66	1.005×10^6	1.534×10^8

(a) Traffic Distribution

Without VSLC, although the total numbers of vehicles using each route were close, vehicles alternated between routes over time, resulting in pronounced traffic oscillation, as seen in Figure 4a. On the other hand, VSLC for dampening traffic oscillation achieved closer traffic assignment in both routes at all times, as shown in Table 2 and illustrated in Figure 4d. However, we note that dampening traffic oscillation may not be an effective strategy in terms of mobility and safety measurements, as we will discuss later in this section. Implementing VSLC for mobility attracted more vehicles to Route 1 (see Figure 4c) due to the higher speed limit set on Link 2-3, while VSLC for safety discouraged vehicles from using Link 2-3 (see Figure 4b) by providing a lower speed limit (see Figure 5).

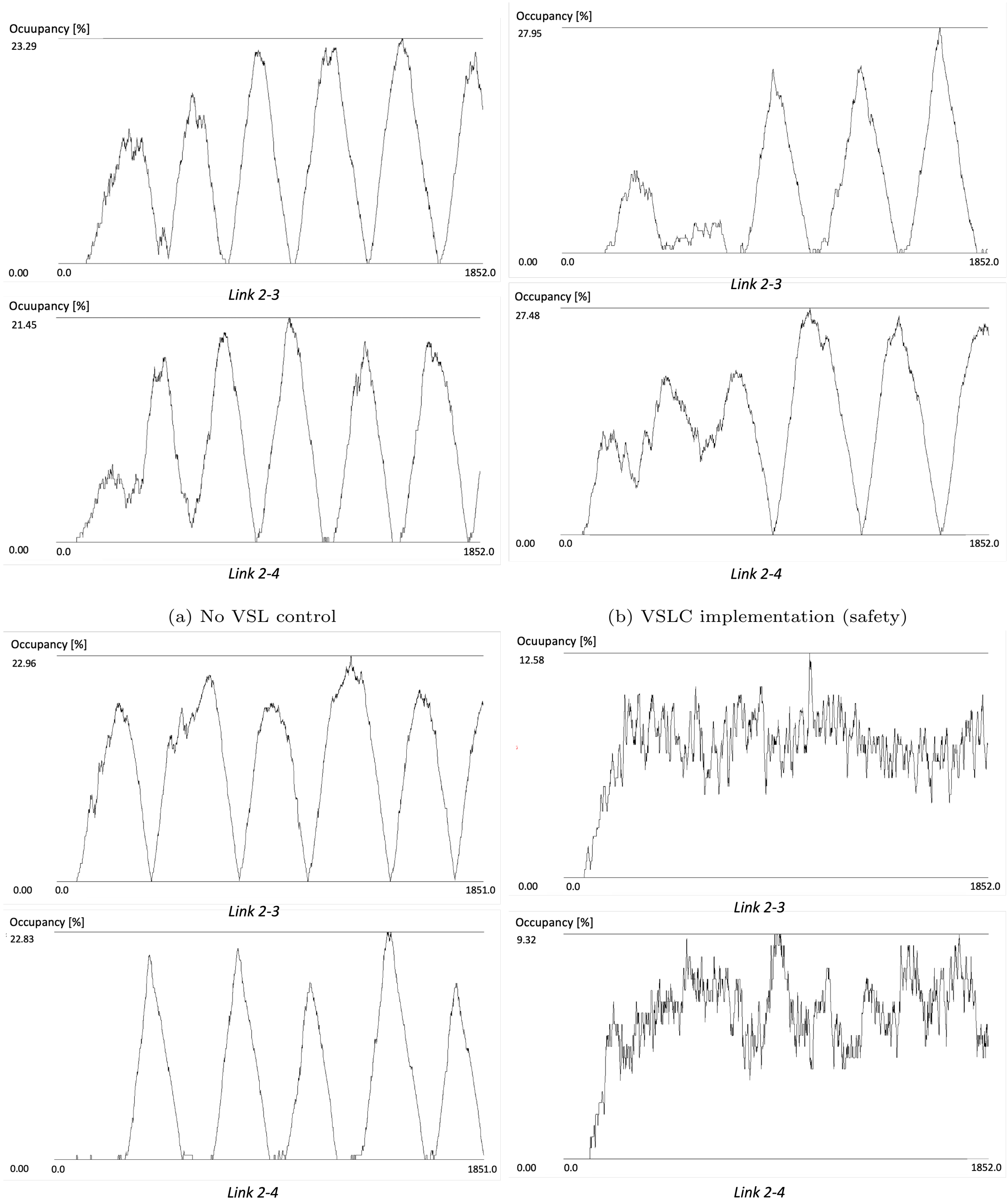
(b) Network Mobility

Table 2 summarizes traffic mobility (measured by TTT) for each control objective. VSLC for mobility achieved the lowest total travel time, which was 28.4% lower compared to no VSLC. This strategy provided a higher speed limit, reducing bottlenecks on Link 1-2 (see Figure 6a). On the other hand, VSLC for safety worsened mobility, which was 46.82% lower than no VSL control, due to the lower speed limits and increased routing complexity, causing a larger queue length as shown in Figure 6a. VSLC for traffic oscillation, aiming to balance occupancy in both routes, yielded the least mobility improvement, resulting in the largest queue on Link 1-2 (see Figure 6a). These results show that naively maintaining equal occupancy of each route might unintentionally deteriorate the network performance.

(c) Network Safety

Table 2 presents a comparison of traffic safety measurements (measured by $log(TTC)$) for various traffic control objectives. The VSLC implemented for traffic oscillation exhibited slightly higher $log(TTC)$ values in comparison to the VSLC aiming at safety improvement. However, the former strategy entailed more frequent lane-changing activities, leading to an increase in emergency brake events, as depicted in Figure 6b, which consequently might result in potentially hazardous situations. In contrast, the VSLC designed for safety demonstrated the fewest emergency occurrences, as well as a high $log(TTC)$. Notably, the VSLC implemented for mobility experienced a marginal 3.95% decline in safety compared to the scenario with no VSL control. In conclusion, the VSLC strategy tailored for safety enhancement stood out as the most effective approach for improving safety in the transportation network and only focusing on mobility performance might inadvertently lead to higher collision risks.

Remark 1: The numerical results suggest that safety and mobility can sometimes be conflicting objectives. This observation aligns with the inherent trade-offs often present in transportation systems, where optimizing for one objective may inadvertently compromise the other. One can engineer a reward function to better balance these two objectives. However, we note that the mobility results we reported in the numerical section are the output of simulation without



(a) No VSL control

(b) VSLC implementation (safety)

(c) VSLC implementation (mobility)

(d) VSLC implementation (traffic oscillation)

Figure 4: Dynamic Traffic Distribution Given Different VSLC Control Objectives (Horizontal Axis is Time in seconds)

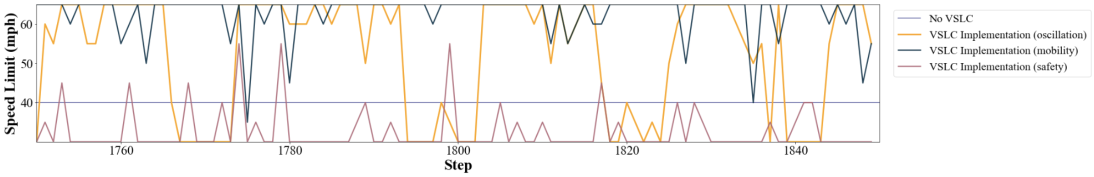
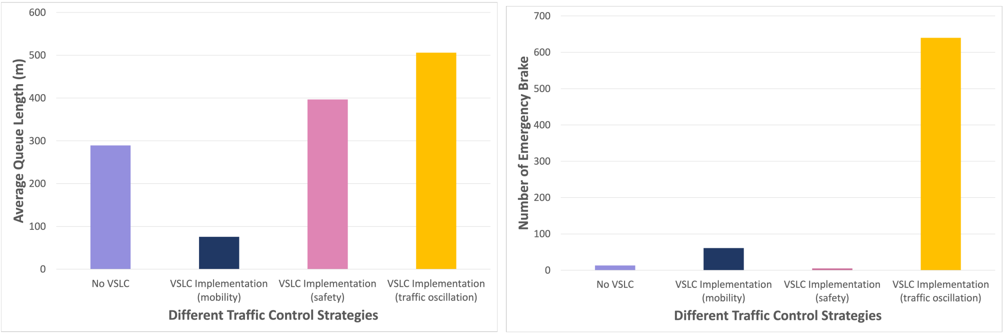


Figure 5: Speed Profiles of Link 2-3 for Different VSLC Control Objectives



(a) Queue Formation

(b) Emergency Brake Made

Figure 6: Microscopic Mobility and Safety Measurements

traffic accidents. If a traffic accident actually happens, it can cause significant delays. In this sense, optimizing safety can also improve the expected mobility and minimize the variance of travel time.

Remark 2: To evaluate the performance of a certain safety measurement as the reward function, we need to evaluate multiple safety measurements in a holistic view. For example, the findings in Table 2 show that the VSLC strategy focused on oscillation leads to slightly higher log(TTC) values compared to the VSLC aimed at safety improvement (i.e., with log(TTC) as the reward function). However, this does not mean minimizing oscillation is the safest since it also results in more frequent lane-changing activities. Increased lane changes can lead to higher instances of emergency braking, which can create potentially hazardous situations despite the higher average log(TTC). On the other hand, although no VSLC has a lower lane change frequency, it produces higher emergency brakes and lower log(TTC) compared with VSLC (safety). In contrast with no VSLC and VSLC for oscillation, the VSLC strategy aims to improve log(TTC) (i.e., reduce collision risks) show fewer emergency braking events, and maintain a relatively high log(TTC). Such results indicate that the VSLC strategy designed for safety successfully balances maintaining safe distances (high log(TTC)) and reducing risky behaviors (fewer lane changes and emergency brakes). Thus, it stands out as the most effective reward function for improving overall safety. While log(TTC) provides good numerical performance in terms of overall system safety and algorithm convergence, valuable further research is to further optimize the selection of safety measures in the context of reinforcement learning or investigate the theoretical properties of alternative safety measures in a more general setting to guide the selection of safety reward functions.

Remark 3: In our analysis, we focused on minimizing the total travel time, which inherently considers various factors contributing to delays, including capacity drops at certain locations caused by VSLC. We do not aim to optimize the mobility at a specific location. Instead, we focus on the mobility of the whole network.

3.2. Comparison with Other Algorithms

We compared the proposed TD3 algorithm with other state-of-the-art RL algorithms, including value-based DQL, policy-based soft actor-critic (SAC), DDPG algorithms, as well as a classic control algorithm and no VSL control to understand the potential of each algorithm on improving network mobility and safety. The control algorithm implemented in this study is a rule-based variable speed limit system. It dynamically adjusts speed limits in real-time based on observed traffic conditions, aiming to enhance network mobility and safety. Specifically, the algorithm lowers speed limits when traffic density surpasses a critical threshold or when average vehicle speeds drop below a predefined level. Conversely, in low-density, high-speed conditions, the algorithm restores speed limits to their maximum allowable values to facilitate smoother traffic flow. The algorithm aligns conceptually with methodologies by Allaby et al. (2007) and Lee et al. (2006), which leverage real-time traffic parameters to adapt speed limits dynamically.

All models were trained for 800 episodes with the same SUMO parameters. The average performance of different methods across the 100 test episodes for the base case is presented in Table 3.

Table 3: Average Performance of Different Algorithms in Improving Network Safety and Mobility in Base Case

Method	**Network safety (unit: seconds)**	**Network mobility (unit: seconds)**
No VSLC	9.5262×10^5	9.1878×10^7
Control algorithm	9.6823×10^5	8.4221×10^7
VSLC-DQL	8.8534×10^5	1.1302×10^8
VSLC-SAC	9.2121×10^5	8.4903×10^7
VSLC-DDPG	**9.9546e+05**	7.2324×10^7
VSLC-TD3	9.8919e+05	**6.1384e+07**

The results indicate that both TD3 and DDPG VSLC strategies significantly enhance network performance in terms of safety and mobility. The TD3 algorithm excels in improving network mobility, showing a 33.2% increase over no VSL control and a 15.1% improvement over the DDPG algorithm. In terms of safety, however, DDPG shows a slight advantage, improving traffic safety by 0.6% in terms of $log(TTC)$ compared to TD3. This can be attributed to DDPG's aggressive exploration mechanism, which effectively navigates sparse but critical safety signals, such as those involving short time to collision. Despite its better performance in $log(TTC)$, DDPG may suffer from overestimation bias, which could affect its reliability. Conversely, TD3 mitigates such bias through its twin Q-networks, enhancing the reliability of its results. In contrast to the policy-based methods, the value-based DQN method failed to improve network safety and mobility performance, which was 7.05% and 23% lower than no VSL control, respectively. This could be because value-based algorithms in continuous action spaces require discretization of the action space, making the learning process slow and computationally expensive. Additionally, discretization could lead to suboptimal actions being chosen, as the algorithm was limited to a finite set of actions. Compared to the no-VSL scenario, the control algorithm led to a 9.1% improvement in mobility and a 1.3% enhancement in safety. However, compared to the RL-based strategies, the control method demonstrates less adaptability to fluctuating traffic conditions, which might explain its limited efficiency in improving network safety and mobility.

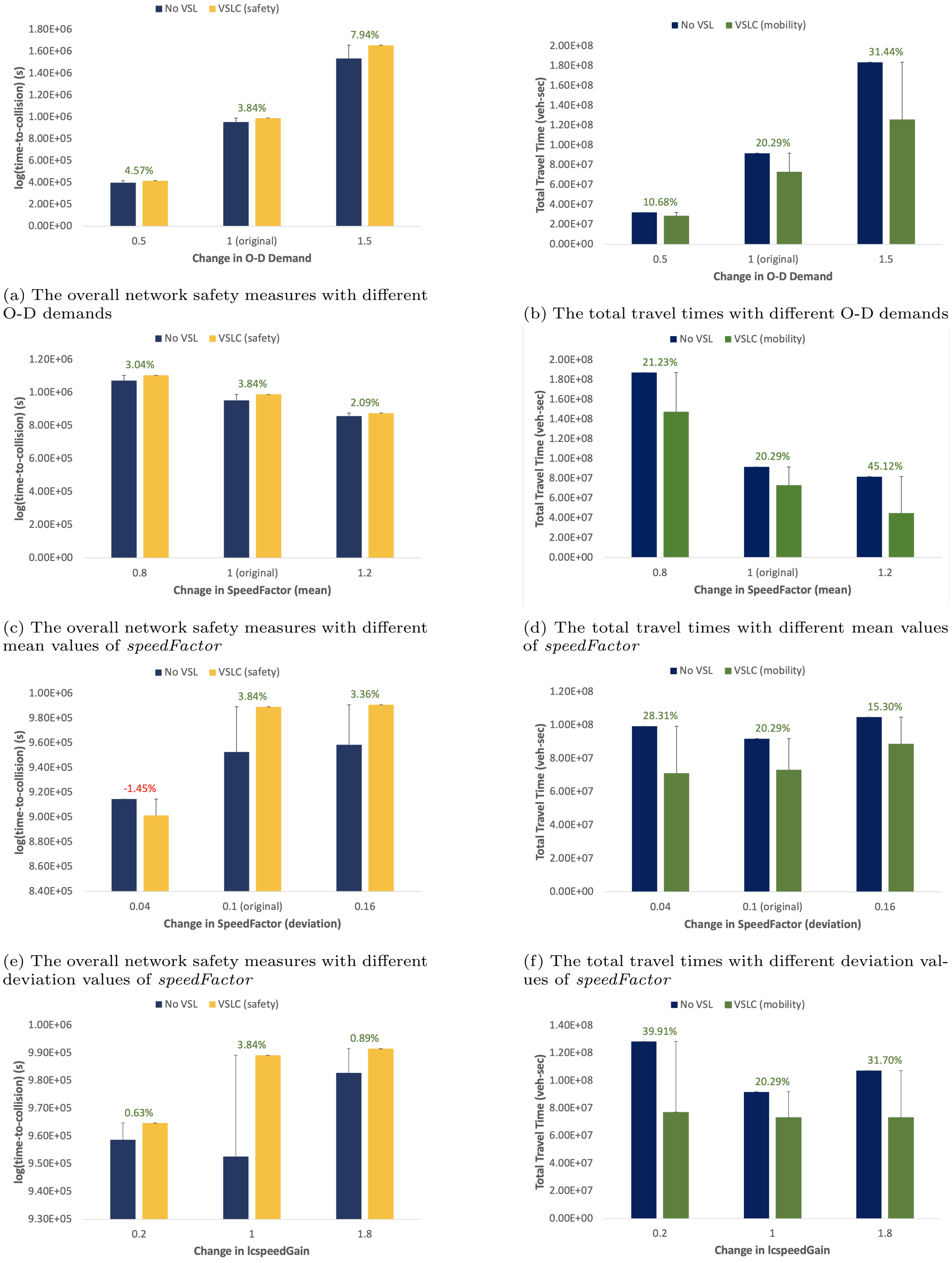
3.3. Assessment of RL Generalizability

A generalizable agent can adapt to new and unseen scenarios and demonstrate effective performance even when environmental conditions have changed. In the context of road traffic, evaluating the generalizability of an RL agent is critical due to the complex, dynamic, and ever-changing nature of traffic environments that traffic simulators can not capture completely. Studies, e.g., (Wang et al., 2012), have shown that the variation in traffic conditions and unforeseen events in real-world transportation systems can cause the failure of VSL control. To test the generalizability of the algorithm, we evaluated the learned algorithm in environments with different demands, “speedFactor”, and “lcspeedGain” scenarios. While assessing the model for one of these attributes, the other attributes were kept the same as the base case. We evaluated the performance of all models on 100 test episodes. The results are presented in Figure 7. We found that our learned VSLC strategies based on the base case settings could be applied to an environment with different parameters with an improved safety or mobility performance compared with the no VSLC case. Since different factors led to similar observations, we will only explain the generalization capability of the trained model in detail for O-D demand as an example in the remainder of this subsection.

Figure 7a and Figure 7b show that although the VSLC RL algorithm is learned given the base case O-D demand, the algorithm can improve both safety and mobility when O-D demand deviates from the base case. In Figure 7a, the VSLC algorithm can increase the safety performance, measured by $log(TTC)$, by 3.84% in the base case compared with the case without VSLC. When we applied the learned VSLC algorithm to a lower O-D demand case (0.5 times the original O-D demand) and a higher O-D demand case (1.5 times the original O-D demand), the proposed VSLC was capable of increasing the network safety performance by 4.57% and 7.94%, respectively. In Figure 7b, a similar observation holds for mobility, where the learned VSLC algorithm based on the base-case O-D demand is capable of improving mobility, measured by TTT, by 10.68% - 31.44%.

3.4. Assessment of Different VSLC Control Scopes

To demonstrate the value of considering network-wide effects of VSLC, we conducted experiments comparing the effectiveness of different control scopes, including link-specific, neighborhood-specific (only consider the control links and the adjacent link(s)), and network-specific mobility and safety rewards. We implemented VSLC on Link 2-3. Figure 8a and 8b illustrate the resulting Log(TTC) and total travel time, respectively. We can see that a network-specific reward function can enhance these safety and mobility metrics around 2 to 9 folds. More specifically, when training based on the link-specific reward, the Log(TTC) value is the lowest around 10^5 seconds, indicating minimal safety improvements confined to the controlled link with potential risks elsewhere in the network. The neighborhood-specific reward shows moderate improvement to $5 * 10^5$ seconds, while the network-specific reward yields the highest Log(TTC) values close to 10^6 seconds. Similarly, from Figure 8b, the total travel times reduce from $6.2 * 10^7$ to 10^7 seconds by using a network-specific reward instead of link-specific reward. Considering neighboring links slightly reduces the total travel time compared to the link-specific approach, indicating improved traffic flow and reduced congestion in the vicinity. These result highlights the benefits of considering the network impacts of VSLC beyond the implemented link or neighboring links on enhancing overall traffic safety and mobility.



(a) The overall network safety measures with different O-D demands

(b) The total travel times with different O-D demands

(c) The overall network safety measures with different mean values of *speedFactor*

(d) The total travel times with different mean values of *speedFactor*

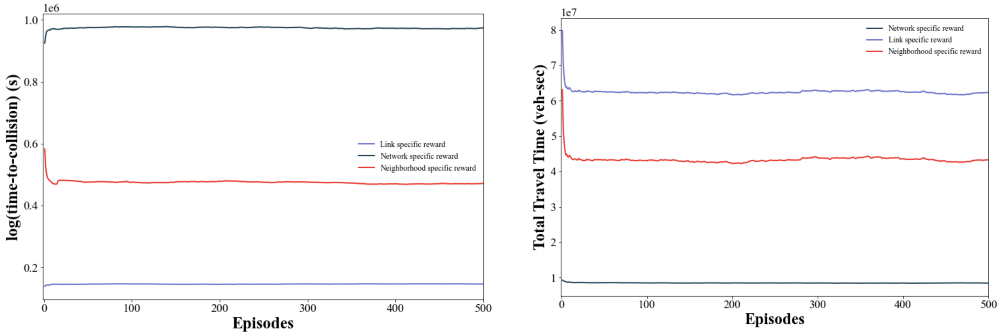
(e) The overall network safety measures with different deviation values of *speedFactor*

(f) The total travel times with different deviation values of *speedFactor*

(g) The overall network safety measures with different values of *lcspeedGain*

(h) The total travel times with different values of *lcspeedGain*

Figure 7: Effects of Different Traffic Attributes on Network Safety and Mobility



(a) Log(Time-to-Collision)

(b) Total Travel Time

Figure 8: Comparison of Safety and Mobility Measures for Different Control Scopes with VSLC Implementation at Link 2-3

3.5. Sensitivity of VSLC Placement in the Network

Although VSLC has the potential to enhance transportation safety and mobility, implementing VSL control throughout the entire network can be costly and may cause unnecessary burdens for information processing. In this section, we explored the potential impacts of different VSLC placement locations in the test example shown in Figure 9 to identify the optimal VSLC placement. Given the network's symmetry, our analysis strategically focused on selected segments to demonstrate the impact of VSLC placement links on network performance. Figure 9(a) shows VSLC implementation on the Link 1-2, which is utilized by vehicles prior to node 2 where they have two alternative routes to choose from. Figure 9(b) controls the speed of Link 2-3, which is the base case of the study. Figure 9(c) and (d) assess whether dividing the links (Link 1-2 and 2-3 respectively) into upstream and downstream and controlling their speed by separate VSLC devices can further improve the network mobility and safety. The reasons why we choose to implement VSLs on Link 2-3 instead of Link 2-4 or Link 3-5 are as follows. First, Link 2-4 serves as a parallel route to Link 2-3 in the symmetrical four-node network used in this study (see Figure 3). Given their functional similarity, implementing VSLs on both links would result in redundant findings. Therefore, we selected Link 2-3 as a representative upstream segment for VSL placement. Second, VSLs are known to be effective in managing upstream traffic flow to mitigate congestion before it propagates downstream (Carlson et al., 2011). Implementing VSLCs on a downstream link, such as Link 3-5, will have a limited impact on controlling the congestion formation and the overall network performance compared to regulating traffic flow at upstream segments. The motivation behind these scenarios is that if there is congestion downstream of a link, increasing the speed limit throughout the entire link could exacerbate the queue. In such scenarios, setting a higher speed limit downstream and a lower speed limit upstream of a link may mitigate queue formation. In the remainder of this subsection, we first examine the placement of VSLC for enhancing network safety, followed by a discussion on the placement of VSLC for improving network mobility.

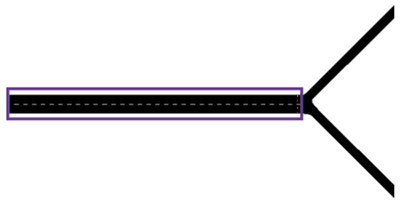
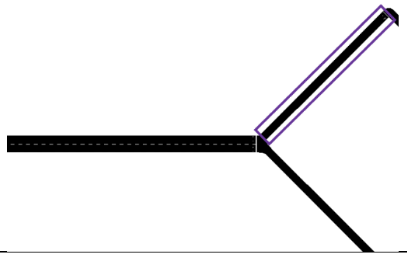
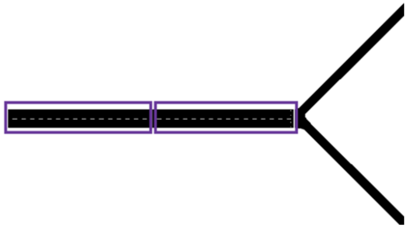
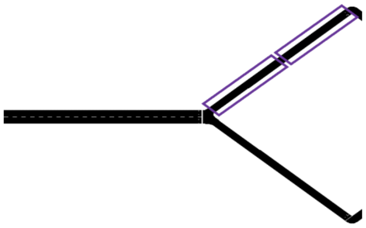
Position of VSL	Diagram
(a) One VSL in the Link 1-2	
(b) One VSL in the Link 2-3 (Base case)	
(c) Two VSLs in the upstream and the downstream of the Link 1-2	
(d) Two VSLs in the upstream and the downstream of the Link 2-3	

Figure 9: Locations of VSLC Placements

(a) Network Safety

Figure 10 presents the average safety reward over 800 training episodes. Notably, the base case was not the most effective in terms of improving the average safety reward. Figure 10 shows that the two VSLCs deployed upstream and downstream of Link 1-2 yield the most desirable

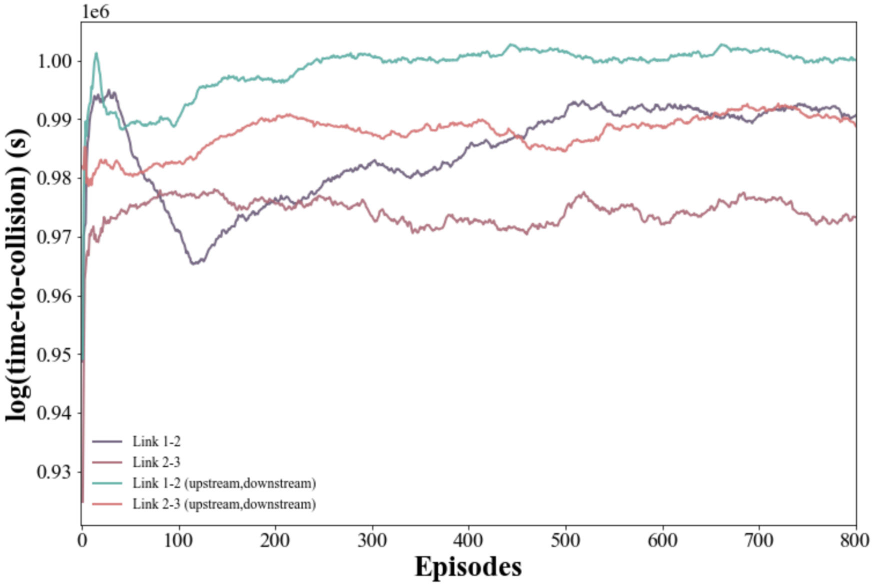
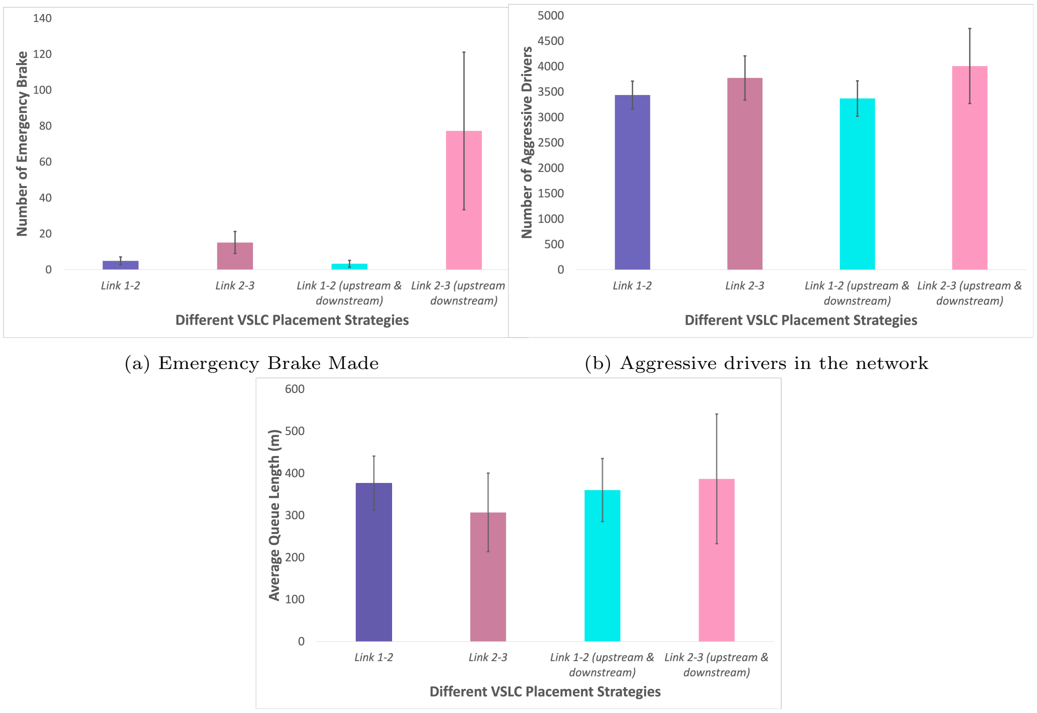


Figure 10: Convergence Patterns of TD3 Algorithms Aiming to Improve Network Safety

outcome for improving the average safety reward ($log(TTC)$). VSLC placement in the upstream and downstream links of Link 2-3 and one VSLC placement on Link 1-2 also provided a higher average safety reward than the base case, with the former reaching a steady state faster than the latter. To better understand the reasons leading to these different safety impacts of the various VSLC placements, a further investigation into the microscopic driving behavior is needed, as discussed below.

To analyze the microscopic driving behavior, 15 episodes of SUMO simulation were randomly selected and investigated. The average performance, driving behaviors, and speed limit of each strategy are presented in Table 4, Figure 11, and Figure 12, respectively. Figure 12b shows that placing VSLC on Link 2-3 results in an average speed limit ranging from 30 to 35 mph during peak periods. However, due to the lower speed limit on this route compared to Route 2, the majority of vehicles (on average 67.73%, see Table 4) preferred to use Route 2, which had a constant speed limit of 40 mph, to reach their destination. The worse safety performance of placing VSLC on Link 2-3 could be because the potential lane changing incentives to Route 2 results in emergencies that require the use of emergency brakes (on average 15 times) (see Figure 11a).

The placement of VSLCs in the upstream and downstream of Link 1-2 yielded the highest improvement of average safety reward (1.56%) compared to the base case (see Table 4). This



(a) Emergency Brake Made

(b) Aggressive drivers in the network

(c) Queue Formation

Figure 11: Traffic Behaviors Given Different VSLC Placements Aiming to Improve Network Safety

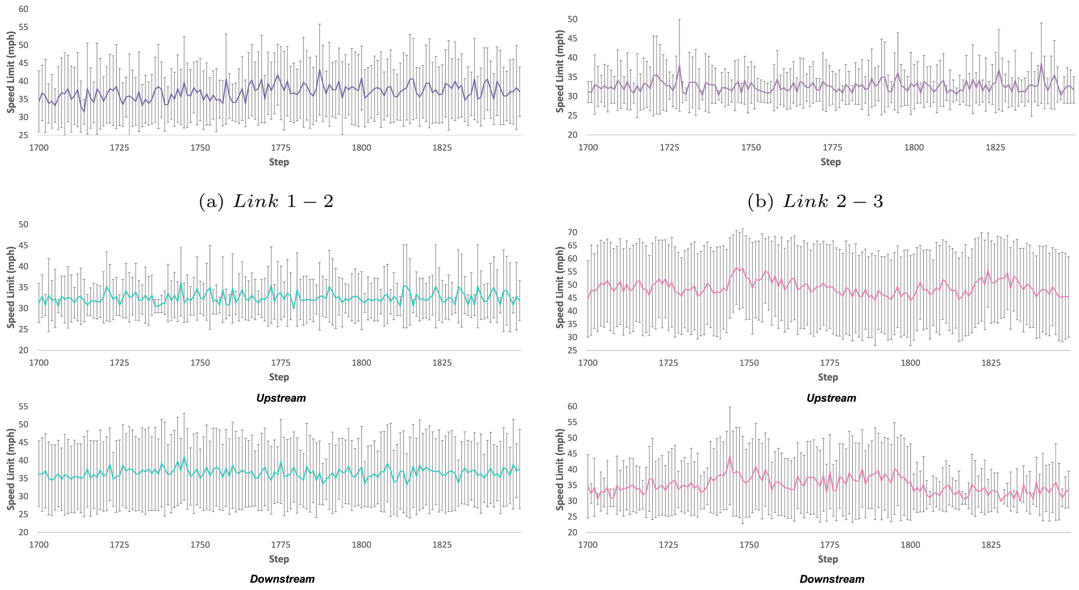
Table 4: Average Performance of Different VSLC Locations Aiming to Improve Network Safety

	Routing		**Lane change frequency (%)**	**TTT of all vehicles (sec)**	$\sum \log(TTC)$
	Route 1 (%)	**Route 2 (%)**			
Link 1-2	50.34 (2.97)	49.66 (2.97)	16.22 (2.43)	1.220e+08 (1.651e+07)	9.936e+05 (2.1186e+04)
Link 2-3	32.27 (7.17)	67.73 (7.17)	13.20 (2.51)	8.778e+07 (1.7735e+07)	9.799e+05 (1.5826e+04)
Link 1-2 (upstream, downstream)	49.77 (2.77)	50.23 (2.77)	16.92 (3.31)	9.774e+07 (1.6262e+07)	9.952e+05 (2.5877e+04)
Link 2-3 (upstream, downstream)	43.99 (5.84)	56.01 (5.84)	19.26 (6.71)	1.228e+08 (3.2419e+07)	9.891e+05 (3.3130e+04)

is attributed to the lower speed limit in the upstream (ranging on average from 30 to 35 mph) compared with downstream (ranging on average from 35 to 40 mph) which allows the traffic to smoothly change to the desired lane before reaching the intersection node 2. The lane change frequency for this strategy was higher than the base case at 16.9% on average. But due to the reduced variance in the speed limit provided by VSLCs, the number of aggressive drivers who changed lanes to gain speed was 10.71% lower than the base case (see Figure 11b), resulting in the lowest number of emergency brakes (with an average of 3.26 times) being pressed compared to all other strategies (see Figure 11a). As both routes had a similar speed limit, vehicles were distributed to both routes almost equally.

The placement of one VSLC on Link 1-2 resulted in slightly worse performance compared with two VSLCs on Link 1-2 but better performance on average safety reward in comparison to the base case. On one hand, one VSLC on Link 1-2 implied a homogeneous speed limit for Link 1-2, which averages between 35 to 40 mph. This led to a higher average queue formation (376 meters), which resulted in a higher number of aggressive drivers and emergency breaks, compared with the case of two VSLCs on Link 1-2. On the other hand, similar to two VSLCs on Link 1-2, vehicles choose to use both routes almost equally. The less intention to choose a particular route leads to a decrease of 8.88% in the number of aggressive drivers changing lanes compared with the base case, resulting in an average of 4.88 emergency brake activations (see Figure 11a).

VSLC placement in both the upstream and downstream of Link 2-3 also provided a slightly



(a) *Link* $1-2$

(b) *Link* $2-3$

(c) Upstream and downstream of *Link* $1-2$

(d) Upstream and downstream of *Link* $2-3$

Figure 12: Speed Profiles for Different VSLC Placements Aiming to Improve Network Safety

higher average safety reward improvement compared with the base case. However, the abrupt changes in speed limits in the upstream and downstream areas (see Figure 12d) may not be as effective as the base case, as the number of aggressive drivers was higher on average (Figure 11b), leading to a higher number of emergency brakes being used compared to other cases (Figure 11a). These results suggest that while more degrees of control freedom may improve the reward function, it may cause unintended risks if those risks are not effectively measured by the reward. The optimal coordination of multiple distributed speed limit controllers is certainly beyond the scope of this paper and will be left for future studies.

(b) Network Mobility

The average mobility reward over 800 training episodes is presented in Figure 13[4]. The base case performed better in terms of improving network mobility compared with other strategies. One VSLC placement on Link 1-2 and two VSLC placements in the upstream and downstream of Link 1-2 exhibited similar levels of improvement, with the former achieving stability faster than the latter, which is as expected because a higher degree of control freedom is harder to learn the optimal strategy. Conversely, VSLC placement in the upstream and downstream of Link 2-3 resulted in the worst network mobility improvement. Similar to the safety assessment,

[4] In Figure 13, we show the negative total travel time. Therefore, the higher the curve, the better the mobility.

15 episodes were randomly selected for each strategy to compare their network mobility/safety performance, driving behavior, and optimal speed limits, as summarized in Table 5, Figure 14, and Figure 15, respectively.

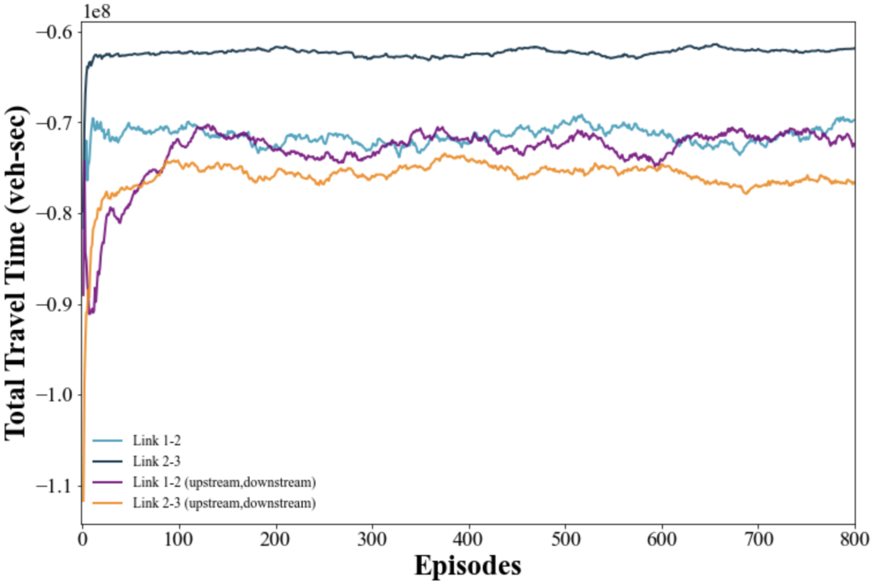


Figure 13: Convergence Patterns of TD3 Algorithms Aiming to Improve Network Mobility

The average speed limit achieved through the placement of VSLC on Link 2-3 is presented in Figure 15b, where the speed limit ranges from 60 to 65 mph over the subset of episodes. The higher speed limit in Route 1 attracted about 65% of vehicles opting for Route 1. Consequently, the average queue formation was the lowest, at an average of 140 meters. The total delay resulting from this strategy was the lowest among all the strategies, with a value of $7.166X10^7$ seconds for all vehicles. As a byproduct, this strategy resulted in the highest $log(TTC)$ among all the considered strategies aimed to maximize mobility.Furthermore, due to the reduced variance in speed limit provided by VSLC, the base case recorded the least number of aggressive drivers (on average 2640 drivers) and emergencies (on average 83 emergency brakes) (see Figure 14). Therefore, while aiming to improve network mobility, one VSLC on Link 2-3 balances the safety performance as well.

VSLC placement on Link 1-2 did not perform as well as the base case mainly because the two alternative routes were equally attractive to drivers approaching node 2 and significant traffic weaving and delay were observed when drivers changed lanes and chose their desired routes. This resulted in 58.67% higher average queue formation than the base case (see Figure 14c).

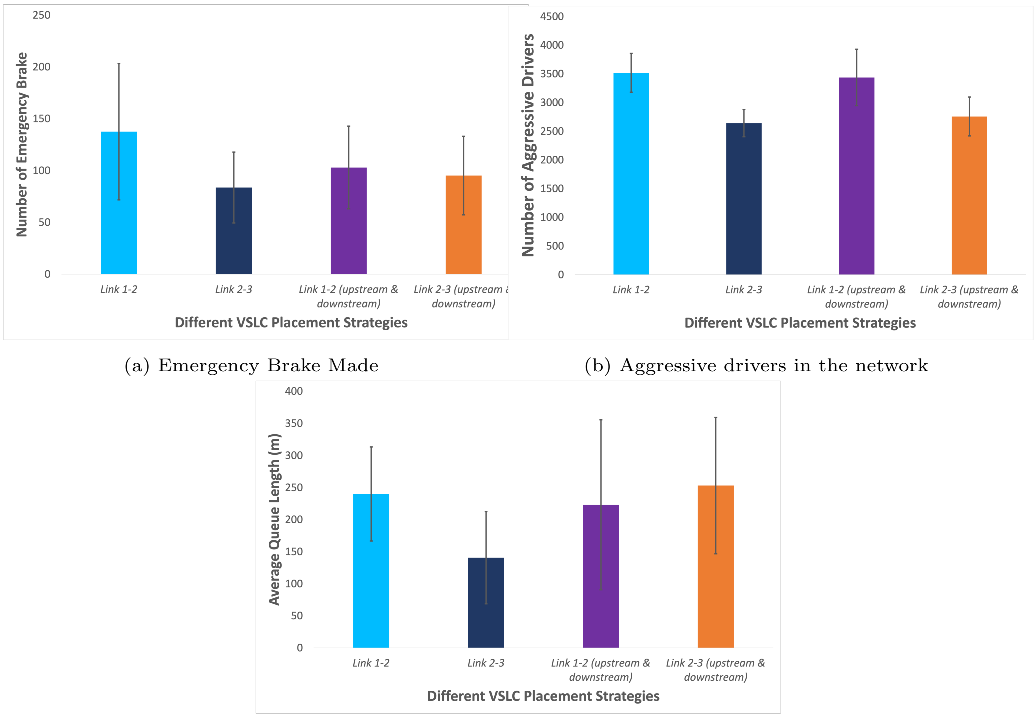
Table 5: Average Performance of Different VSLC Locations Aiming to Improve Network Mobility

	Routing		**Lane change frequency (%)**	**TTT of all vehicles (sec)**	$\sum \log(TTC)$
	Route 1 (%)	**Route 2 (%)**			
Link 1-2	52.99 (2.19)	47.01 (2.19)	11.30 (2.09)	8.056e+07 (9.338e+06)	8.845e+05 (1.776e+04)
Link 2-3	65.04 (2.77)	35.96 (2.77)	11.56 (1.89)	7.166e+07 (9.951e+06)	9.274e+05 (1.348e+04)
Link 1-2 (upstream, downstream)	52.50 (2.33)	47.50 (2.33)	11.98 (5.02)	7.283e+07 (1.956e+07)	8.785e+05 (4.801e+04)
Link 2-3 (upstream, downstream)	69.91 (9.10)	30.09 (9.10)	14.13 (3.36)	8.366e+07 (1.770e+07)	9.204e+05 (1.869e+04)

These can be also seen from Table 5 that the total delay on average is 16.75% higher than the base case.

Different from VSLC for safety, when VSLC was aiming to improve mobility, separate VSLCs in the upstream and downstream of Link 2-3 led to the worst mobility performance. This could be because separately controlling two VSLCs significantly increases the learning complexity. Without good coordination between upstream and downstream VSLCs, a queue formed downstream of Link 2-3 when the majority of traffic chose Route 1. Although increasing the number of VSL can increase the optimal reward theoretically, it introduces complexity for RL algorithms to learn and converge to global optimal. Thus, the performance of multiple VSLs can be worse than a single VSL case in practice.

To summarize, implementing one VSLC on Link 2-3 provided the best network mobility improvement as well as a good balance of network safety with fewer emergencies. Implementing VSLC on Link 1-2 with either single or separate control strategies resulted in suboptimal network mobility improvement due to the delay caused by lane changing and longer queue formation before intersection node 2. In addition, implementing VSLCs in the upstream and downstream of Link 2-3 could be challenging to learn and resulted in additional queue formation on Link 2-3 due to a lack of speed coordination between the upstream and the downstream.

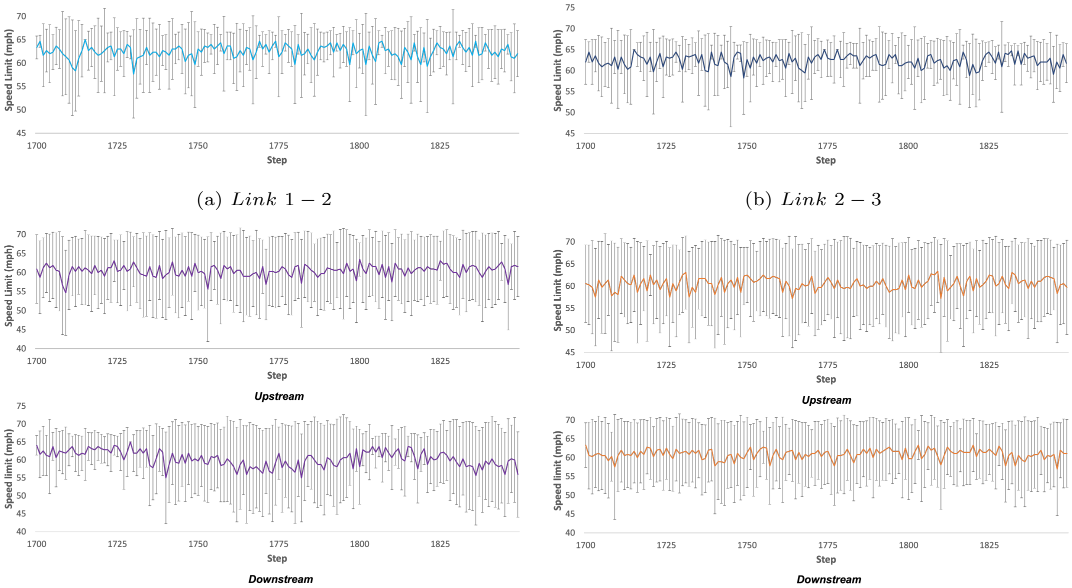


(a) Emergency Brake Made

(b) Aggressive drivers in the network

(c) Queue Formation

Figure 14: Traffic Behaviors Given Different VSLC Placements Aiming to Improve Network Mobility



(a) *Link* $1-2$

(b) *Link* $2-3$

(c) Upstream and downstream of *Link* $1-2$

(d) Upstream and downstream of *Link* $2-3$

Figure 15: Speed Profiles for Different VSLC Placements Aiming to Improve Network Mobility

4. Discussion

This study proposes a twin-delayed deep deterministic policy gradient (TD3) model to learn and control VSL for a subset of links for network mobility and safety. The simulation experiments show that the proposed TD3 algorithm can lead to better performance in terms of network mobility and safety compared with other DRL-based models. The study also examines various traffic limit control objectives and quantifies the performance of the learned VSLC model on other non-optimized objectives. The study finds that when the objective is to enhance network mobility, network safety may be compromised, and vice versa. Additionally, sensitivity analysis reveals the impact of VSLC placement locations on network mobility and safety, offering valuable insights for consideration when implementing VSLC in transportation networks.

Despite the promising results, there are important future extensions to consider. First, the study uses a simplified four-node network to ensure clear and controlled analysis. While this approach effectively highlights the impacts of VSLC on network safety and mobility, future research should extend the findings to more complex urban or highway networks to further validate the model's scalability and robustness. Second, the reward functions in this study focus on optimizing either safety or mobility independently. Exploring combined reward functions that simultaneously consider both objectives may offer a more balanced traffic management strategy. Third, the study relies on high-fidelity simulations for evaluation. Incorporating real-world traffic data or suitable resolution of digital twins in future research would enhance the realism and applicability of the findings, ensuring the model's effectiveness in practical implementations. Fifth, from a computational perspective, future research can also explore multi-agent models to handle more complex information availability scenarios and distributed control of VSL in larger-scale transportation networks. Addressing these limitations and extensions can further optimize VSLC implementation and enhance traffic management in transportation networks, considering the increasing availability of detailed data and vehicle connectivity.

Acknowledgement

The paper is partially supported by the National Science Foundation under Grant No. 2041446 and Safety Research using Simulation University Transportation Center (SAFER-SIM). SAFER-SIM is led by NADS at the University of Iowa, and is funded by a grant from the U.S. Department of Transportation's University Transportation Centers Program (69A3551747131). However, the U.S. Government assumes no liability for the contents or use thereof.

Author Contribution Statement

The authors confirm their contribution to the paper as follows: study conception and design: Zhaomiao Guo and Fatima Afifah; data collection: Fatima Afifah; analysis and interpretation of results: Fatima Afifah and Zhaomiao Guo; draft manuscript preparation: Fatima Afifah. All authors reviewed the results and approved the final version of the manuscript.

References

Abdel-Aty, M., Dilmore, J., Dhindsa, A., 2006. Evaluation of variable speed limits for real-time freeway safety improvement. Accident analysis & prevention 38, 335–345.

Afifah, F., Guo, Z., Abdel-Aty, M., 2023. System-level impacts of en-route information sharing considering adaptive routing. Transportation Research Part C: Emerging Technologies 149, 104075.

Allaby, P., Hellinga, B., Bullock, M., 2007. Variable speed limits: Safety and operational impacts of a candidate control strategy for freeway applications. IEEE Transactions on Intelligent Transportation Systems 8, 671–680.

Bel, G., Rosell, J., 2013. Effects of the 80 km/h and variable speed limits on air pollution in the metropolitan area of barcelona. Transportation Research Part D: Transport and Environment 23, 90–97.

Boukerche, A., Zhong, D., Sun, P., 2021. A novel reinforcement learning-based cooperative traffic signal system through max-pressure control. IEEE Transactions on Vehicular Technology 71, 1187–1198.

Carlson, R.C., Papamichail, I., Papageorgiou, M., 2011. Local feedback-based mainstream traffic flow control on motorways using variable speed limits. IEEE Transactions on intelligent transportation systems 12, 1261–1276.

Carlson, R.C., Papamichail, I., Papageorgiou, M., 2013. Comparison of local feedback controllers for the mainstream traffic flow on freeways using variable speed limits. Journal of Intelligent Transportation Systems 17, 268–281.

Casas, N., 2017. Deep deterministic policy gradient for urban traffic light control. arXiv preprint arXiv:1703.09035 .

Chen, Z., Liu, X.C., Zhang, G., 2016. Non-recurrent congestion analysis using data-driven spatiotemporal approach for information construction. Transportation Research Part C: Emerging Technologies 71, 19–31.

Dadashzadeh, N., Ergun, M., 2019. An integrated variable speed limit and alinea ramp metering model in the presence of high bus volume. Sustainability 11, 6326.

El-Tantawy, S., Abdulhai, B., Abdelgawad, H., 2013. Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (marlin-atsc): methodology and large-scale application on downtown toronto. IEEE Transactions on Intelligent Transportation Systems 14, 1140–1150.

Emmerink, R.H., Axhausen, K.W., Nijkamp, P., Rietveld, P., 1995. Effects of information in road transport networks with recurrent congestion. Transportation 22, 21–53.

Farrag, S., El-Hansali, M.Y., Yasar, A., Shakshuki, E.M., 2020. Simulation-based evaluation of using variable speed limit in traffic incidents. Procedia Computer Science 175, 340–348.

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

Gregurić, M., Kušić, K., Ivanjko, E., 2022. Impact of deep reinforcement learning on variable speed limit strategies in connected vehicles environments. Engineering Applications of Artificial Intelligence 112, 104850.

Grumert, E., Ma, X., Tapani, A., 2015. Analysis of a cooperative variable speed limit system using microscopic traffic simulation. Transportation research part C: emerging technologies 52, 173–186.

Han, Y., Hegyi, A., Yuan, Y., Hoogendoorn, S., Papageorgiou, M., Roncoli, C., 2017. Resolving freeway jam waves by discrete first-order model-based predictive control of variable speed limits. Transportation Research Part C: Emerging Technologies 77, 405–420.

Hegyi, A., De Schutter, B., Hellendoorn, H., 2005a. Model predictive control for optimal coordination of ramp metering and variable speed limits. Transportation Research Part C: Emerging Technologies 13, 185–209.

Hegyi, A., De Schutter, B., Hellendoorn, J., 2005b. Optimal coordination of variable speed limits to suppress shock waves. IEEE Transactions on intelligent transportation systems 6, 102–112.

Hegyi, A., Hoogendoorn, S.P., Schreuder, M., Stoelhorst, H., Viti, F., 2008. Specialist: A dynamic speed limit control algorithm based on shock wave theory, in: 2008 11th international ieee conference on intelligent transportation systems, IEEE. pp. 827–832.

Hoogendoorn, S., Daamen, W., Hoogendoorn, R., Goemans, J., 2013. Assessment of dynamic speed limits on freeway a20 near rotterdam, netherlands. Transportation research record 2380, 61–71.

Hydén, C., 1996. Traffic conflicts technique: state-of-the-art. Traffic safety work with video processing 37, 3–14.

Islam, M.T., Hadiuzzaman, M., Fang, J., Qiu, T.Z., El-Basyouny, K., 2013. Assessing mobility and safety impacts of a variable speed limit control strategy. Transportation research record 2364, 1–11.

Jesus, J.C., Bottega, J.A., Cuadros, M.A., Gamarra, D.F., 2019. Deep deterministic policy gradient for navigation of mobile robots in simulated environments, in: 2019 19th International Conference on Advanced Robotics (ICAR), IEEE. pp. 362–367.

Jin, H.Y., Jin, W.L., 2015. Control of a lane-drop bottleneck through variable speed limits. Transportation Research Part C: Emerging Technologies 58, 568–584.

Jinnai, Y., Park, J.W., Machado, M.C., Konidaris, G., 2019. Exploration in reinforcement learning with deep covering options, in: International Conference on Learning Representations.

Karafyllis, I., Papageorgiou, M., 2019. Feedback control of scalar conservation laws with application to density control in freeways by means of variable speed limits. Automatica 105, 228–236.

Kušić, K., Dusparic, I., Guériau, M., Gregurić, M., Ivanjko, E., 2020a. Extended variable speed limit control using multi-agent reinforcement learning, in: 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), IEEE. pp. 1–8.

Kušić, K., Ivanjko, E., Gregurić, M., Miletić, M., 2020b. An overview of reinforcement learning methods for variable speed limit control. Applied Sciences 10, 4917.

Kušić, K., Ivanjko, E., Vrbanić, F., Gregurić, M., Dusparic, I., 2021. Spatial-temporal traffic flow control on motorways using distributed multi-agent reinforcement learning. Mathematics 9, 3081.

Kwon, J., Mauch, M., Varaiya, P., 2006. Components of congestion: Delay from incidents, special events, lane closures, weather, potential ramp metering gain, and excess demand. Transportation Research Record 1959, 84–91.

Lee, C., Hellinga, B., Saccomanno, F., 2004. Assessing safety benefits of variable speed limits. Transportation Research Record 1897, 183–190.

Lee, C., Hellinga, B., Saccomanno, F., 2006. Evaluation of variable speed limits to improve traffic safety. Transportation research part C: emerging technologies 14, 213–228.

Li, D., Ranjitkar, P., 2015. A fuzzy logic-based variable speed limit controller. Journal of advanced transportation 49, 913–927.

Li, D., Ranjitkar, P., Ceder, A., 2014. A logic tree based algorithm for variable speed limit controllers to manage recurrently congested bottlenecks. Technical Report.

Li, G., Wei, Y., Chi, Y., Gu, Y., Chen, Y., 2020. Breaking the sample size barrier in model-based reinforcement learning with a generative model. Advances in neural information processing systems 33, 12861–12872.

Li, J., Yu, T., Zhang, X., Li, F., Lin, D., Zhu, H., 2021. Efficient experience replay based deep deterministic policy gradient for agc dispatch in integrated energy system. Applied Energy 285, 116386.

Li, Z., Liu, P., Xu, C., Duan, H., Wang, W., 2017. Reinforcement learning-based variable speed limit control strategy to reduce traffic congestion at freeway recurrent bottlenecks. IEEE transactions on intelligent transportation systems 18, 3204–3217.

Li, Z., Liu, P., Xu, C., Wang, W., 2016. Optimal mainline variable speed limit control to improve safety on large-scale freeway segments. Computer-Aided Civil and Infrastructure Engineering 31, 366–380.

Li, Z., Zhu, X., Liu, X., Qu, X., 2019. Model-based predictive variable speed limit control on multi-lane freeways with a line of connected automated vehicles, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), IEEE. pp. 1989–1994.

Liang, Y., Guo, C., Ding, Z., Hua, H., 2020. Agent-based modeling in electricity market using deep deterministic policy gradient algorithm. IEEE transactions on power systems 35, 4180–4192.

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

Lin, P.W., Kang, K.P., Chang, G.L., 2004. Exploring the effectiveness of variable speed limit controls on highway work-zone operations, in: Intelligent transportation systems, Taylor & Francis. pp. 155–168.

Lopez, P.A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., Wießner, E., 2018. Microscopic traffic simulation using sumo, in: The 21st IEEE International Conference on Intelligent Transportation Systems, IEEE. URL: `https://elib.dlr.de/124092/`.

Papageorgiou, M., Kosmatopoulos, E., Papamichail, I., 2008. Effects of variable speed limits on motorway traffic flow. Transportation Research Record 2047, 37–48.

Park, J.J., Lee, Y.M., Park, J.B., Kang, J.G., 2008. The effect of point to point speed enforcement systems on traffic flow characteristics. Journal of Korean Society of Transportation 26, 85–95.

Qiu, C., Hu, Y., Chen, Y., Zeng, B., 2019. Deep deterministic policy gradient (ddpg)-based energy harvesting wireless communications. IEEE Internet of Things Journal 6, 8577–8588.

Radwan, E., Zaidi, Z., Harb, R., 2011. Operational evaluation of dynamic lane merging in work zones with variable speed limits. Procedia-Social and Behavioral Sciences 16, 460–469.

Treiber, M., Kesting, A., 2017. The intelligent driver model with stochasticity-new insights into traffic flow oscillations. Transportation research procedia 23, 174–187.

Vrbanić, F., Ivanjko, E., Mandžuka, S., Miletić, M., 2021. Reinforcement learning based variable speed limit control for mixed traffic flows, in: 2021 29th Mediterranean Conference on Control and Automation (MED), IEEE. pp. 560–565.

Vrbanić, F., Miletić, M., Tišljarić, L., Ivanjko, E., 2022. Influence of variable speed limit control on fuel and electric energy consumption, and exhaust gas emissions in mixed traffic flows. Sustainability 14, 932.

Walraven, E., Spaan, M.T., Bakker, B., 2016. Traffic flow optimization: A reinforcement learning approach. Engineering Applications of Artificial Intelligence 52, 203–212.

Wang, C., Xu, Y., Zhang, J., Ran, B., 2022. Integrated traffic control for freeway recurrent bottleneck based on deep reinforcement learning. IEEE Transactions on Intelligent Transportation Systems .

Wang, C., Zhang, J., Xu, L., Li, L., Ran, B., 2019a. A new solution for freeway congestion: Cooperative speed limit control using distributed reinforcement learning. IEEE Access 7, 41947–41957.

Wang, P., Li, H., Chan, C.Y., 2019b. Continuous control for automated lane change behavior based on deep deterministic policy gradient algorithm, in: 2019 IEEE Intelligent Vehicles Symposium (IV), IEEE. pp. 1454–1460.

Wang, Y., Zhang, Y., Hu, J., Li, L., 2012. Using variable speed limits to eliminate wide moving jams: a study based on three-phase traffic theory. International Journal of Modern Physics C 23, 1250060.

Wu, J., Wei, Z., Liu, K., Quan, Z., Li, Y., 2020a. Battery-involved energy management for hybrid electric bus based on expert-assistance deep deterministic policy gradient algorithm. IEEE Transactions on Vehicular Technology 69, 12786–12796.

Wu, T., Zhou, P., Liu, K., Yuan, Y., Wang, X., Huang, H., Wu, D.O., 2020b. Multi-agent deep reinforcement learning for urban traffic light control in vehicular networks. IEEE Transactions on Vehicular Technology 69, 8243–8256.

Wu, Y., Mozifian, M., Shkurti, F., 2021. Shaping rewards for reinforcement learning with imperfect demonstrations using generative models, in: 2021 IEEE International Conference on Robotics and Automation (ICRA), IEEE. pp. 6628–6634.

Wu, Y., Tan, H., Qin, L., Ran, B., 2020c. Differential variable speed limits control for freeway recurrent bottlenecks via deep actor-critic algorithm. Transportation research part C: emerging technologies 117, 102649.

Xu, D., Zhu, F., Liu, Q., Zhao, P., 2021. Improving exploration efficiency of deep reinforcement learning through samples produced by generative model. Expert Systems with Applications 185, 115680.

Xu, J., Hou, Z., Wang, W., Xu, B., Zhang, K., Chen, K., 2018. Feedback deep deterministic policy gradient with fuzzy reward for robotic multiple peg-in-hole assembly tasks. IEEE Transactions on Industrial Informatics 15, 1658–1667.

Zeynivand, A., Javadpour, A., Bolouki, S., Sangaiah, A., Ja'fari, F., Pinto, P., Zhang, W., 2022. Traffic flow control using multi-agent reinforcement learning. Journal of Network and Computer Applications 207, 103497.

Zhu, F., Ukkusuri, S.V., 2014. Accounting for dynamic speed limit control in a stochastic traffic environment: A reinforcement learning approach. Transportation research part C: emerging technologies 41, 30–47.