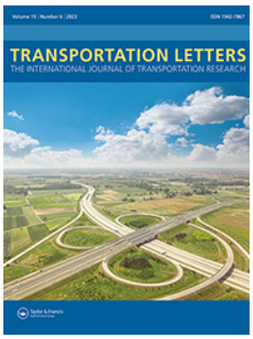

Taylor & Francis
Taylor & Francis Group

Transportation Letters

The International Journal of Transportation Research

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ytrl20

A dynamic self-improving ramp metering algorithm based on multi-agent deep reinforcement learning

Fuwen Deng, Jiandong Jin, Yu Shen & Yuchuan Du

To cite this article: Fuwen Deng, Jiandong Jin, Yu Shen & Yuchuan Du (2023): A dynamic self-improving ramp metering algorithm based on multi-agent deep reinforcement learning, Transportation Letters, DOI: 10.1080/19427867.2023.2231638

To link to this article: https://doi.org/10.1080/19427867.2023.2231638

TRANSPORTATION LETTERS
https://doi.org/10.1080/19427867.2023.2231638

Taylor & Francis
Taylor & Francis Group

A dynamic self-improving ramp metering algorithm based on multi-agent deep reinforcement learning

Fuwen Deng [a], Jiandong Jin [b], Yu Shen [c] and Yuchuan Du [c]

[a]School of Computer Science and Technology, Shandong Technology and Business University, Yantai, China; [b]School of Electronics Engineering and Computer Science, Peking University, Beijing, China; [c]College of Transportation Engineering, Tongji University, Shanghai, China

ABSTRACT

We present a novel ramp metering algorithm that incorporates multi-agent deep reinforcement learning (DRL) techniques, which utilizes monitoring data from loop detectors. Our proposed approach employed a multi-agent DRL framework to generate optimized ramp metering schedules for each ramp meter in real-time, enhancing the operational efficiency of urban freeways with less investment. To simplify the implementation and training of the algorithm, we developed a simulation platform based on SUMO microscopic traffic simulator. We conducted a series of simulation experiments, including local and coordinated ramp metering scenarios with various traffic demands profiles. The simulation results indicate that the proposed DRL-based algorithm outperforms the state-of-the-practice ramp metering methods, considering a comprehensive evaluation index encompassing mainstream speed at the bottleneck and queue length on ramp. Additionally, the method exhibits robustness, scalability, and the potential for further improvement through online learning during implementation.

ARTICLE HISTORY
Received 21 March 2023
Accepted 26 June 2023

KEYWORDS
ramp metering; deep reinforcement learning; adaptive control

Introduction

Urban freeways take a significant role in facilitating personal mobility and freight movement in metropolitan areas, where effective traffic management can yield substantial economic, security, and environmental benefits. Among the available strategies for traffic management on urban freeways, ramp metering, and variable speed limits are commonly employed. In this study, we focus on ramp metering, which regulates the flow of vehicles entering the freeway through on-ramps to alleviate congestion on the mainline.

Review on ramp metering methods

Over the last few decades, efforts have been made toward developing ramp metering algorithms that more effectively alleviate congestion and resultant delays. The existing ramp metering strategies can be classified into three categories: fixed-time strategies, adaptive ramp-metering strategies, and model predictive control strategies.

Fixed-time strategies refer to the offline metering measures that are implemented during specific times of the day. With constant historical demands, the ramp metering problem can be formulated as a linear programming or quadratic programming problem, with the objective of maximizing the number of vehicles served or the total traveled distance. Constraints may include restrictions on mainstream flow and ramp queue length. The early-stage ramp metering strategies mostly belong to this category, such as those described in Chen, Cruz, and Paquet (1974), Papageorgiou (1980), Wang (1972), Wang and May (1973), and Wattleworth (1967).

Adaptive or traffic responsive ramp metering strategies use real-time measurements to capture the dynamic traffic demand, then variable metering rates are allocated to ramps in response to actual traffic conditions. Generally, the objective of the strategies is to keep the freeway traffic conditions close to prespecified values. The most widely deployed adaptive ramp metering strategies are ALINEA (Markos Papageorgiou and Hadj-Salem 1990) and its variants (Smaragdis and Papageorgiou 2003; Smaragdis, Papageorgiou, and Kosmatopoulos 2004). ALINEA-like algorithms were designed based on classical automatic control theory and can be deemed as proportional-integral (PI) feedback controllers that keep the downstream occupancy near a desired value. METALINE (Papageorgiou, Blosseville, and Hadj-Salem 1990) is an extension of ALINEA for coordinated control of ramp meters. It was designed based on the linear quadratic (LQ) optimization theory. Other controllers, e.g. linear-quadratic-integral (LQI), were adapted for coordinated ramp metering in literature (Stylianopoulou et al. 2020). The queue-control policy for selected ramps is an alternative strategy for achieving coordinated ramp metering. Within this category, the HERO linked-control algorithm serves as a typical example (Papamichail and Papageorgiou 2008). More classical examples include Zone (Stephanedes 1994), Bottleneck (Jacobson, Henry, and Mehyar 1989), HELPER (Lipp, Corcoran, and Hickman 1991) and fuzzy logic algorithm (Taylor, Meldrum, and Jacobson 1998). Related problems such as on-ramp queue length estimation (Yang et al. 2019) was also discussed in existing studies.

Model predictive control (MPC) strategies (Camacho and Bordons 1995) adopt a freeway traffic model to predict the future traffic behavior in a rolling horizon framework, then optimize the ramp metering rates over the prediction horizon. The upside of MPC is the proactive feature, i.e. the pre-consideration of traffic conditions in the near future. A study by van de Weg et al. (van de Weg et al. 2019) discussed how MPC can optimize ramp metering and variable speed limits for improving freeway throughput. Tabadkani Aval et al. (Tabadkani Aval and Eghbal 2020) proposed a ramp metering strategy combining sliding mode control (SMC) and MPC to provide robustness and optimality. Han et al. (Han et al. 2020) developed a two-level controller, using an MPC approach to optimize total network travel time and distribute the optimal total inflows to each on-ramp based on local feedback.

CONTACT Fuwen Deng dengfw@sdtbu.edu.cn School of Computer Science and Technology, Shandong Technology and Business University, Yantai 264005, China

Several recent studies, e.g. Hegyi, De Schutter, and Hellendoorn (2005), evaluated model predictive controllers in simulation frameworks.

Despite systematic investigation and many successful applications, the existing ramp metering options are not without drawbacks. Most existing strategies predefine traffic models to predict traffic patterns and calculate control actions, while the models probably fail in unusual conditions, e.g. extreme weather and traffic incidents, then lead to misbehaving of controllers due to model mismatch. Moreover, administrators have to make an effort to fine-tune the parameters of control algorithms for different segments, so that large-scale models and parallel implementations are excessively time-consuming and costly. In addition, the existing strategies have no capability to improve the control efficiency in long-term implementation. The existing ramp metering approaches require further enhancement to achieve robustness, scalability, and self-improvement.

Reinforcement learning and its application

Recent years have witnessed the rapid development of reinforcement learning (RL). RL refers to an artificial intelligence technique in which agents interact with the environment, learn an optimal policy by trial-and-error, then make sequential decision to solve a specific problem. It is a powerful tool for control and has demonstrated success in many complex problem settings such as video games (Mnih et al. 2015), robot control (Van De Panne 2017), self-driving (Shalev-Shwartz, Shammah, and Shashua 2016), and chess and go (Silver et al. 2016). In most cases, RL (specifically, model-free RL) requires no explicit model of the environment beforehand and learns only from the past experiences. Moreover, RL has the ability to adapt to different environments without changing the algorithm and can improve performance through continuous self-learning. For these rea-sons, RL shows promise in overcoming the downsides of traditional ramp metering strategies. The application of RL to adaptive traffic signal control at intersections has been shown to be efficient in the literature (Jin and Ma 2015; Khamis and Gomaa 2014; Li, Lv, and Wang 2016; Mosharafian, Afzali, and Mohammadpour Velni 2022). It is convincing that this paradigm can be effective for solving the ramp metering problem.

Several up-to-date studies (Fares and Gomaa 2014; Han et al. 2022; Liu et al. 2021; Wang et al. 2022; Xu, Liu, and Xu 2022) have demonstrated the effectiveness of employing RL for ramp metering and expressway traffic management. For instance, Fares et al. (Fares and Gomaa 2014) designed a density control agent based on Markovian modeling and Q-learning algorithm to alleviate freeway mainstream congestion. Liu et al. (2021) proposed an RL-based method that uses preprocessed traffic video as input to learn optimal metering strategies. Xu, Liu, and Xu (2022) utilized traffic flow parameters to design state space and applied Double-Deep Q Network to ramp metering. The results of these studies have demonstrated significant improvements over ALINEA and other classical ramp metering methods mentioned earlier.

The contribution of the paper

In this paper, a novel ramp metering algorithm based on multi-agent deep reinforcement learning (DRL) framework is proposed. Multi-Agent Proximal Policy Optimization (MAPPO) architecture is introduced to solve ramp metering problem. A ramp metering testbed is developed based on SUMO microscopic traffic simulation software, providing a simulation environment encompassing several ramp metering scenarios, which is similar to OpenAI Gym (Brockman et al. 2016), to process the interaction between agent and freeway environment, then evaluate the performance of RL-based ramp metering algo-rithms. Simulation experiments are undertaken to illustrate the power of the proposed approach. Superior performance com-pared to the baseline is presented.

Our study contributes to the field by presenting two main advances beyond existing research. First, we designed an RL-based ramp metering approach that operates in a continuous action space, addressing limitations of prior research (Liu et al. 2021; Xu, Liu, and Xu 2022) that relied mostly on Deep Q-Learning. While Deep Q-Learning has yielded impressive results in games (e.g. Atari and go) where decisions are made within discrete action spaces, it is not well-suited for ramp metering scenarios. Specifically, it tends to produce inaccuracies in metering rate adjustments and may cause system instability due to abrupt action shifts. To address this issue, our proposed approach employs policy gradient-based method and allows for effective handling of continuous action space. Second, we have developed a test platform of ramp metering algorithms based on an open-source simulation software, and preset various traffic demand profiles. Therefore, the testbed can generate more realistic traffic flow dynamics and facilitate obtaining detailed and interpretable results during testing experiments.

Preliminary

RL is a learning paradigm concerned with learning to control a system: An RL agent receives the controlled system's state and a reward value associated with the last transition, then determines an action which is sent back to the system to maximize the total reward that expresses a long-term objective. In this section, basic concepts of RL are introduced as necessary background of the following contents.

Markov decision process

RL sequential decision-making problem is usually formulated as a *Markov Decision Process* (MDP). An MDP is a discrete time stochastic control process. It can be defined as a triplet $M = (S, A, \mathcal{P}_0)$, where S is the set of states, A is the set of actions and $\mathcal{P}_0$ is the transition probability kernel. Let $t \in \mathbb{N}$ denote the current time step; then, once an action is selected the system makes a transition:

$$(S_{t+1}, R_{t+1}) \sim \mathcal{P}_0(\cdot|S_t, A_t), \tag{1}$$

where R denote immediate reward. The decision-maker can select its actions at any time step based on a specified polity π:

$$A_t = \pi(S_t) \tag{2}$$

The return of the transition history $S_0, A_0, R_1, \ldots, S_{t-1}, A_{t-1}, R_t, S_t$ is defined as the total discounted sum of the rewards incurred:

$$R = \sum_{t=0}^{\infty} \gamma^t R_{t+1}, \tag{3}$$

where γ is discount rate.

The value function of a state s underlying π, denoted $V^{\pi}(s)$, is the expected return when starting in s and following π thereafter:

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+1} | S_t = s\right], s \in \mathcal{S} \tag{4}$$

For any policy π and any state s, the following consistency condition holds between the value of s and the value of its possible successor states s':

$$V^{\pi}(s) = \pi(s) \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma V^{\pi}(s')], s \in \mathcal{S} \quad (5)$$

which is called *Bellman equation* for V^{π}.

Solving an RL task means finding an optimal policy by $\pi^{\star}$. This can be achieved by finding an optimal value function defined as:

$$V^{*}(s) = \sup_{\pi \in \prod} V^{\pi}(s), s \in \mathcal{S} \quad (6)$$

The commonly used solution of MDP is dynamic programming (DP) algorithm. However, it is common that we have no prior knowledge of the environment, i.e. transition probability is usually unknown in advance for a realistic problem, hence we must estimate the value function based on experience. We can solve the problem through Monte Carlo (MC) sampling and temporal difference (TD) methods such as SARSA and Q-learning.

Function approximation

In many RL tasks, the state and action space can be highly dimensional and destructively large in number. This gives rise to the problem named *curse of dimensionality*. In such cases, generalization is necessary to make sensible decisions when some state encountered is not seen before. One feasible generalization measure is function approximation.

We often seek an estimate of the values in the form:

$$V_{\theta}(s) \doteq V^{\pi}(s), s \in \mathcal{S} \quad (7)$$

where $\theta \in \mathbb{R}$ is a d-dimensional vector of parameters. The formulation of $V_{\theta}(s)$ can be linear function, coarse coding, tile coding, radial basis functions, etc.

It is also possible to learn a parameterized policy function without consulting a value function such as:

$$\pi_{\theta}(a|s) \doteq \frac{\exp[h(s, a, q)]}{\sum_b \exp[h(s, b, q)]}, s \in \mathcal{S} \quad (8)$$

where $\theta \in R^{d'}$ is a d'-dimensional vector of parameters and $h(s, a, \theta)$ is a feature function. This strategy is known as policy-gradient method.

Recent literatures demonstrated the powerful generalization capability of deep neural network (DNN). The DRL method refers to the RL methods combined with DNN. This approach extends RL to the entire process from observation to action (end to end) without explicitly designing the state space or action space. Leading contenders of DRL include DQN, DDPG, TRPO, etc.

Function approximation keeps track of a significantly smaller number of parameters, instead of a large number of state-action pairs. This helps to reduce requirements on computing resource and memory capacity.

Multi-agent reinforcement learning

Consider a set of agents in the operating environment. With the increasing number of agents, the number of state-action pairs increases exponentially. Albeit the adoption of function approximation method, the learning process will be time-consuming. For computing tractability, *multi-agent reinforcement learning* (MARL) is introduced to allow the agents to exchange information (e.g. the observation of environment or the delayed reward) and coordinate their respective actions in order to achieve global optimization. This approach decomposes a complex problem into smaller problems solved by the agents via collaboration and parallelism, making it more scalable and robust.

Proposed algorithm

A dynamic self-improving ramp metering algorithm is developed for agents to learn how to execute on-ramp metering actions in a freeway environment. The following section gives a description of the RL-based ramp metering algorithm.

State, action, and reward representation

State

According to the classical traffic flow theory, characteristics of traffic state on the freeway can be depicted by three fundamental quantities, i.e. volume, density, and speed. In the practice of traffic engineering, occupancy which is proportional to density is directly collected by the inductive loop detectors. Hence, the occupancy should be a natural and interpretable state representation of freeway traffic environment. Compared to other traffic state detectors such as millimeter wave radar and video camera, loop detector is more adaptable (e.g. less sensitive to weather conditions) and low-budget (pre-facilitated in most freeways).

To configure a ramp metering system, the freeway can be divided into several segments. Each segment S_i encompasses an entrance ramp R_i facilitated with a ramp meter (Figure 1). The instantaneous traffic state of a segment will be detected by a set of mainstream loop detectors. Define the occupancy of detection profile k in segment i as $\bar{o}^{(i)}_{l,m,k,t}$, then the average profile occupancy in control period t is:

$$\bar{o}^{(i)}_{k,t} = \frac{1}{LT} \sum_{l=1}^{L} \sum_{m=1}^{M} o^{(i)}_{l,m,k,t} \quad (9)$$

where l is the index of lane, m is the index of aggregation period of detectors, L and M are the number of lanes and aggregation periods in a control period, respectively. The traffic state of a segment in control period t is defined as a vector $o^{(i)}_t = \left(\bar{o}^{(i)}_1, \bar{o}^{(i)}_2, \ldots, \bar{o}^{(i)}_K\right)$, where K is the number of detection profiles which include a group of loop detectors each. We recommend that K is at least 4 to depict the traffic state comprehensively. In addition, both upstream and downstream (especially bottleneck area) should be covered by detector groups.

In multi-agent ramp metering tasks, the reward values ($r^{(i-1)}_t$, $r^{(i+1)}_t$ or both) calculated by the neighboring agent(s) can be appended into the state vector, which will be introduced below.

Action

Consistent with most practical ramp metering operations, the proposed system adopts a one-car-per-green realization. In such a setting, a constant-duration green phase permits exactly one vehicle to pass, and thus, the ramp volume is controlled by varying the red-phase duration between a minimum and a maximum value. So, the action determined by the agent is a metering rate $MR^{(i)}_t$ calculated based on the observation of state, then the red-phase duration is set as:

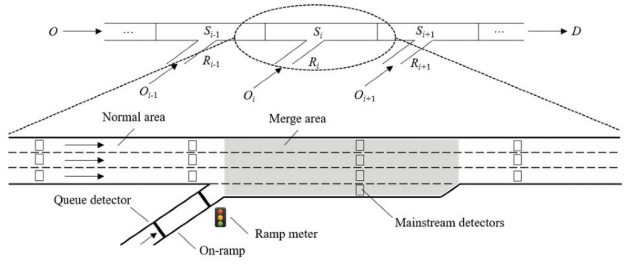

Figure 1. Layout of a freeway section.

$$a_t^{(i)} = \begin{cases} a_{\max} & MR^{(i)} < MR_{\min} \\ a_{\min} & MR^{(i)} > MR_{\max} \\ \left(1 \Big/ MR_t^{(i)} - 1\right) g_t^{(i)} & otherwise \end{cases} \quad (10)$$

where $a_t^{(i)}$ denotes the red-phase duration and $g_t^{(i)}$ denotes green-phase duration. The metering rate will be reevaluated at the end of each control period. Each agent only needs to determine one metering rate of the corresponding ramp in multi-ramp scenarios under the decentralized framework, which can significantly reduce the computational burden.

Reward function

Selection of the reward function has significant impact on the eventual policy arrived at. The objective of ramp metering is to prevent the downstream traffic flow from potential breakdown, i.e. to maintain the travel speed in the merge area. It is also necessary to keep the on-ramp queue below the storage capacity limit, thus preventing long queues to interfere with surface street traffic. Hence, the delayed reward function is designed considering the trade-off between (i) downstream speed and (ii) queue size on the ramp, denoted as:

$$r_{t+1}^{(i)} = \bar{v}_{k_\zeta,t}^{(i)} - \eta \bar{q}_t^{(i)} \quad (11)$$

where $\bar{v}_{k_\zeta,t}^{(i)}$ represents the average speed of profile k_ζ in the merge area and $\bar{q}_t^{(i)}$ represents the average queue size on the ramp i in the last control period. The coefficient η reflects the stress on queue size: a larger η encourages eliminating queue on the ramp. The adopted value of η is 0.1 due to the good performance based on the parameter searching process. In multi-ramp scenarios, the reward value can be shared with other agents to achieve coordinated control.

Policy parameterization for continuous action space

Ramp metering problem differs from Atari video games (20) in that the action, i.e. metering rate, can be continuous real numbers rather than discrete keystrokes. Furthermore, the value of metering rate is interval variable, thus a straightforward discretization leads to the loss of the scale information and makes the learning progress slow. Therefore, the agents should learn a continuous probability distribution function (or a parameterized policy) instead of many discrete actions.

To produce a policy parameterization, the policy is defined as the normal probability density over a real-valued scalar action, with mean and standard deviation given by parametric functions that depend on states, denoted as:

$$\pi_\theta\left(a^{(i)}|o_t^{(i)}\right) = \frac{1}{\sigma_{\theta_\sigma}\left(o_t^{(i)}\right)\sqrt{2\pi}} \exp\left(-\frac{\left(a^{(i)} - \mu_{\theta_\mu}\left(o_t^{(i)}\right)\right)^2}{2\sigma_{\theta_\sigma}\left(o_t^{(i)}\right)^2}\right) \quad (12)$$

where μ and σ are parameterized function approximators for mean and standard deviation, respectively, and $\theta = [\theta_\mu, \theta_\sigma]^{\mathrm{T}}$ is a parameter vector.

Proximal policy optimization

In the present paper, a newly proposed RL algorithm, Proximal Policy Optimization (PPO) (Heess et al. 2017; Schulman et al. 2017), is adopted to achieve a desirable ramp metering policy. The algorithm adopts a classical Actor-Critic (AC) framework which combines the strong points of policy-based and value-based methods as shown in Section 2.2 and imports the Kullback – Leibler (KL) divergence to adjust the update rate of the policy function automatically for good adaptability.

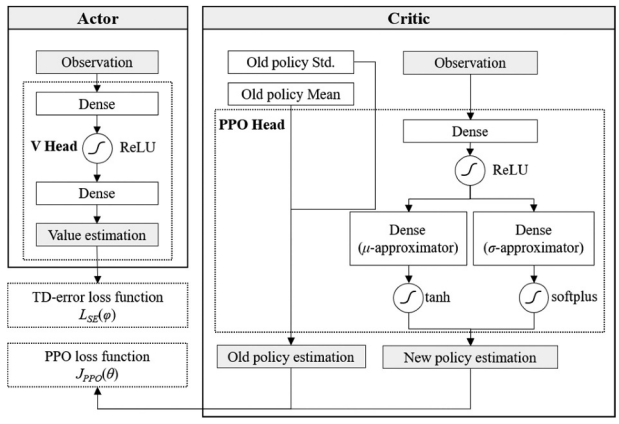

Figure 2. Network structure used in the algorithm.

The backend neural network used in this paper is demonstrated in Figure 2. Actor network is a value function approximator structured as a simple fully connected neural network. It takes a state-action pair $\left(o_t^{(i)}, a_t^{(i)}\right)$ as input and estimates its value. The loss function is the square error:

$$L_{SE}(\varphi) = -\sum_{t=1}^{T}\left(\sum_{t'>t}\gamma^{t'-t} r_{t'}^{(i)} - V_\varphi^\pi\left(\mathbf{o}_t^{(i)}\right)\right)^2 \quad (13)$$

where φ is parameters of value function. Critic network takes a state-action pair $\left(o_t^{(i)}, a_t^{(i)}\right)$ as input and estimates μ and σ, respectively. The update rule employs of a specially designed PPO error function, which will be introduced below. Adam (Kingma and Ba 2014), a first-order gradient-based optimization approach is used for weight update of the neural networks.

The following tricks are embedded in the algorithm:

(1) *Advantage calculation* The term *advantage* refers to the difference value between the action value Q and the state value V (Wang, de Freitas, and Lanctot 2016). Since we cannot determine the Q value directly while training, we can use the return, i.e. the discounted sum of rewards, as an estimation of Q. Thus, the advantage is:

$$\widehat{A}_t = \sum_t \gamma^t r_t^{(i)} - V_\varphi^\pi\left(o_t^{(i)}\right) \quad (14)$$

The advantage allows the agent to determine how much better they turned out to be than expected. We hope the value is neither too large nor too small to make the training process smooth.

(2) *PPO loss function* The actor network of PPO adopts a specified loss function, denoted as:

$$J_{PPO}(\theta) = \sum_{t=1}^{T} \frac{\pi_\theta\left(a_t^{(i)}|o_t^{(i)}\right)}{\pi_{old}\left(a_t^{(i)}|o_t^{(i)}\right)}\widehat{A}_t - \lambda KL[\pi_{old}|\pi_\theta] \quad (15)$$

where$\lambda KL[\pi_{old}|\pi_\theta]$is a regularization term, in which $KL[\pi_{old}|\pi_\theta]$ represents the KL divergence between the policies π_{old} and π_θ defined as:

$$KL[\pi_{old}|\pi_\theta] = \int_{-\infty}^{\infty} \pi_{old} \log \frac{\pi_{old}}{\pi_\theta} d\theta \quad (16)$$

which reflects the similarity between the two probability distributions; λ is the adaptive coefficient. The regularization term will limit the update size of the policy function to avoid divergence (Heess et al. 2017).

(3) *Distributed implementation* For good performance and high training efficiency, the algorithm can be implemented in

a distributed manner. Data collection and gradient calculation are distributed over workers, while global parameters of actor and critic network are updated by the chief processing. Distributed implementation was proved to be more effective in continuous action spaces and have better convergence properties (Mnih et al. 2016).

Description of the algorithm

In this subsection we combine all things together to give a complete description of the ramp-metering algorithm based on multi-agent DRL. The algorithm encompasses two stages, i.e. training and inference, as shown in Figure 3. In the training stage, the RL agents learn a metering policy through interaction with the environment. In the inference stage, agents observe the environment and then take proper actions given the learned policy. Inference only use the μ-approximator to make decisions. The agents can share the delayed reward with others for collaboration in both of the two stages. In addition, pre-trained networks can be transferred to other freeway ramp metering scenarios with only slight modification, e.g. changing the number of input nodes. The following algorithm box shows the complete algorithm. It should be noted that although the training process requires multicore computation platform to achieve distributed training, the inference stage needs far less computing resource, which makes the deployment of algorithm tractable.

Algorithm: Ramp Metering based on MAPPO Framework

I. Training Stage
Initialization: Initialize the actor and critic network with random parameters;
For each episode **do**:
  Reload the ramp metering environment;
  Run policy $\pi_\theta^{(i)}$ for T timesteps, collecting $\left(o_t^{(i)}, a_t^{(i)}, r_{t+1}^{(i)}\right)$;
  Estimate advantage $\widehat{A}_t = \sum_{t'>t} \gamma^{t'-t} r_{t'}^{(i)} - V_\varphi^\pi\left(o_t^{(i)}\right)$;
  $\pi_{old}^{(i)} \leftarrow \pi_\theta^{(i)}$;
  While not done **do**:
    $J_{PPO}(\theta) = \sum_{t=1}^{T} \frac{\pi_\theta\left(a_t^{(i)}|o_t^{(i)}\right)}{\pi_{old}\left(a_t^{(i)}|o_t^{(i)}\right)} \widehat{A}_t - \lambda KL[\pi_{old}|\pi_\theta]$;
    Update θ by Adam optimization algorithm;
  While not done **do**:
    $L_{SE}(\varphi) = -\sum_{t=1}^{T} \left(\sum_{t'>t} \gamma^{t'-t} r_{t'}^{(i)} - V_\varphi^\pi\left(o_t^{(i)}\right)\right)^2$;
    Update φ by Adam optimization algorithm;
  If $KL[\pi_{old}|\pi_\theta] > KL_{\max}$ **then**:
    $\lambda \leftarrow a\lambda$;
  Else:
    $\lambda \leftarrow \lambda/a$;
II. Inference Stage
Initialization: Load the parameters of the critic (policy) network;
Loop forever:
  Observe the state $o_t^{(i)}$;
  Append the reward(s) of other agents to the state: $o_t^{(i)} \leftarrow o_t^{(i)} \cup r_t^{(j)}$;
  Output the metering action $a_t^{(i)}$ using the μ-approximator;
  Calculate the reward $r_{t+1}^{(j)}$;

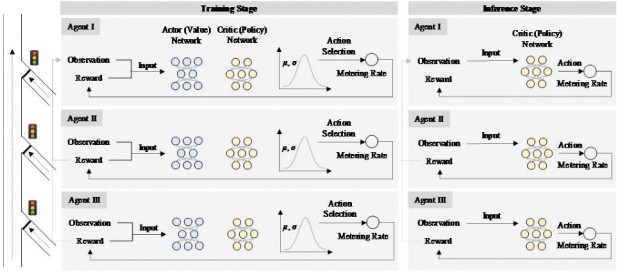

Figure 3. Schematic view of the system architecture.

Simulation experiments

To evaluate the performance of the proposed algorithm, simulation experiments with various freeway network and traffic demand scenarios were conducted based on a customized simulation platform.

Simulation platform

The proposed DRL-based ramp metering algorithm reaches good performance by training neural network policies, which requires repeated interaction with the environment. We built up a simulation platform named ST4RLRM (an acronym for the Simulation Testbed for Reinforcement Learning Ramp Metering). The work was mainly inspired by a computational framework for deep RL and traffic control experiments *Flow* (Wu et al. 2018). An open-source road traffic simulator SUMO (Simulation of Urban Mobility) (Lopez et al. 2018) was adopted as the simulation backend. SUMO provides a TCP-based client/server communication protocol TraCI (Traffic Control Interface) to achieve online interaction with the simulation. In addition, SUMO supports parallel simulation in multiprocessing behavior, which makes distributed RL algorithm (e.g. DPPO) tractable. Ramp metering logic was implemented in Python script language and TensorFlow machine learning framework.

The components and workflow of ST4RLRM platform are demonstrated in Figure 4. Given the traffic demand profile based on the historical observation, as well as the freeway network, the backend traffic simulator will run the simulation and record specified measurements (i.e. occupancy) as input of ramp metering algorithms. Algorithmic modules then take the measurements to determine the metering rate in real time. The algorithm library contains a set of ramp metering algorithms for convenient test and comparison. As to the training of the proposed PPO-based algorithm, SUMO has the capability to run multiple simulation instances in parallel. A well-configured algorithm will be taken into implementation. During the implementation, the training of the PPO agents can still be in progress in back end. The only difference lies in that the measurements should be collected from the realistic traffic environment. The feature makes self-improvement of the metering agent possible.

All of our following experiments were carried out on a computing server with one i7-8700K CPU (3.7 GHz, 6 Cores). There is no need of dedicated GPU for simulation or training.

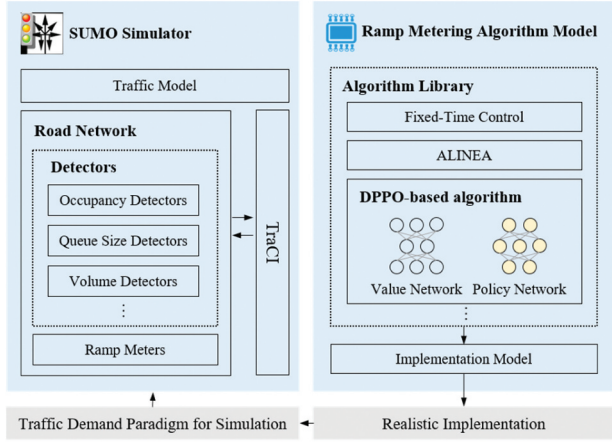

Figure 4. Schematic view of ST4RLRM platform.

Experiment design

Simulation set-up

Various types of freeway network structures and traffic demand profiles were required to verify the effectiveness of the proposed algorithm. We created both single-ramp and multi-ramp simulation scenarios as shown in Figure 5. The single-ramp scenario consisted of a three-lane freeway mainline, approximately 1 km in length, and one entrance ramp. The multi-ramp scenario comprised a three-lane freeway mainline, approximately 1.7 km in length and two entrance ramps, i.e. Ramp I and Ramp II, respectively. The length of all the ramps in the simulation scenarios is approximately 270 m. On the mainline, virtual loop detectors were set every 200–300 m, and queue length detectors were placed on each entrance ramp. Additionally, to obtain accurate spatial-temporal diagrams of speed (see Figure 8), a higher density of virtual loop detectors was utilized, with one placed every 50 m. Each entrance ramp was followed by a 300-m merge area with an additional acceleration lane in comparison to the mainline.

The state was represented by the mainline occupancies, which were monitored by virtual loop detectors near the merging position. The reward was calculated using the bottleneck point's speed profile downstream of the merging position as well as the queue length on the ramp (as shown in Equation (11)). Three types of demand profiles: (i) stationary volume Figure 6(a,d) (ii) flat peak (Figure 6 (b,e)), and (iii) sharp peak (Figure 6(c,f)), were designated to simulate different traffic conditions for mainline (Figure 6(a-c)) and on-ramp (Figure 6(d-f)). The flow generation equation for the flat peak scenario is represented by:

$$q_t = q_{\min} + (q_{\max} - q_{\min}) \cdot \sin(\frac{\pi}{180} \cdot \frac{t}{T}), \tag{17}$$

where t denotes the simulation time step (in seconds), T denotes the time span (in seconds) of each episode, $v_{\min}$ and $v_{\max}$ represent the minimum and maximum volumes of demand profiles, respectively. Similarly, the sharp peak demand profile is derived by the following equation:

$$q_t = q_{\min} + (q_{\max} - q_{\min}) \cdot \frac{1}{(1/\zeta^2) \cdot (t/T - 1/2)^2 + 1}, \tag{18}$$

where ζ denotes the shape factor. Here we set $\zeta = 0.2$.

Stationary volume was applied to simulate the daily homeostatic traffic pattern, while flat peak and sharp peak were used to simulate rush hour and unexpected traffic incidents, respectively. During the training stage, the above three demand profiles were executed consecutively. To avoid overfitting phenomenon, a small random noise is appended to the profile before each training episode.

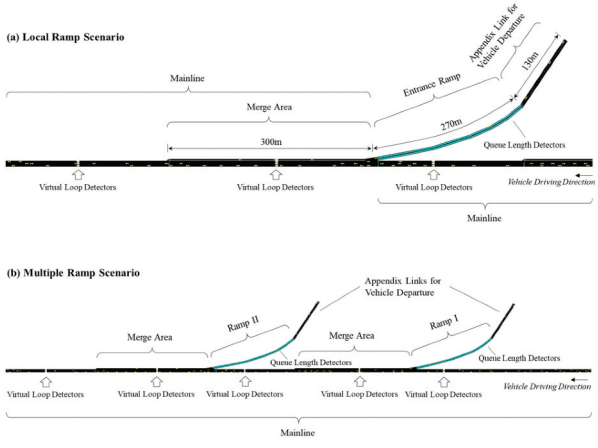

Figure 5. Freeway network used in experiments.

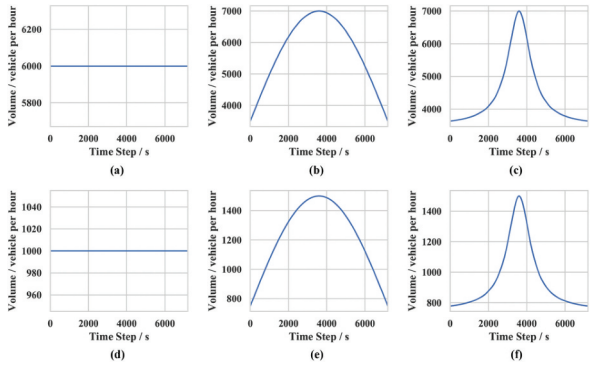

Figure 6. Demand profiles of mainline and ramp.

As shown in Figure 6, each simulation episode for training and inference spans 2 h. The control period of 1 min equates to a total of 120 control periods in each episode. We adopted the one-car-per-green passing mode, wherein each cycle comprises a 3-s green phase followed by a variable-duration red phase. The start of a new control period coincides with the end of the last cycle.

We utilized the default Krauss model in SUMO as the car-following model while also enabling the sub-lane model to simulate more precise lateral behavior of vehicles. The lateral resolution was set at 0.25 m.

Tested algorithms

Besides the proposed algorithm, other metering strategies are tested in our simulation experiments. In the scenario of a local regulator, the commonly applied ramp metering algo-rithm ALINEA is used as the benchmark. The form of ALINEA regulator is shown in Equation 22:

$$\tilde{q}_t^{(i)} = \tilde{q}_{t-1}^{(i)} + K_R[\hat{o}_k^{(i)} - \bar{o}_{k,t}^{(i)}], \tag{19}$$

where $\tilde{q}_t^{(i)}$ represents the on-ramp outflow ordered by the regulator in the t-th timestep, K_R is the metering gain factor, and $\hat{o}_k^{(i)}$ is the critical occupancy of the downstream freeway. Dividing both sides of Equation 22 simultaneously by the capacity of the on-rampc$^{(i)}$ yields:

$$MR_t^{(i)} = MR_{t-1}^{(i)} + \hat{K}_R[\hat{o}_k^{(i)} - \bar{o}_{k,t}^{(i)}], \tag{20}$$

where MR is the metering rate of the ramp as mentioned previously.

In multi-ramp scenario, the coordinated feedback control algorithm HERO is utilized as a benchmark (Papageorgiou, Hadj-Salem, and Middelham 1997). The HERO algorithm is a rule-based approach for coordinated ramp metering in large-scale applications. It integrates local ALINEA regulators and assigns the task of reducing queues at on-ramps to upstream on-ramps when the queue exceeds a pre-set threshold. To implement HERO strategy, the ALINEA regulator is applied to each on-ramp while the following queue-control policy is activated:

$$\hat{q}_t^{(i)} = -\frac{1}{T_c}\left[w_{\max}^{(i)} - w_t^{(i)}\right] + d_{t-1}^{(i)}, \tag{21}$$

where T_c is the time span of the current control period, $w_t^{(i)}$ is the queue length of the on-ramp i, $w_{\max}^{(i)}$ is the maximum admissible queue length of the on-ramp i, and $d_{t-1}^{(i)}$is the arriving ramp demand. The regulated ramp flow is then:

$$\tilde{q}'^{(i)}_t = \max\{\tilde{q}_t^{(i)}, \hat{q}_t^{(i)}\}, \tag{22}$$

and the metering rate is easily calculated as $MR_t^{(i)} = \widetilde{q}_t^{(i)}/c^{(i)}$.

Grid search has been done to find the best parameter combination of the aforementioned algorithms. In all of the experiments we set the critical occupancy $\widehat{o}$ to 0.18 and the metering gain $\widehat{K}_R$ to 0.35. For coordinated metering scenario, Ramp II (in Figure 5) was selected as the queue-control ramp, and $w_{\max}$ was set to 60 m.

In addition, no-control and fixed-time control are also tested for comparison. The proper metering rates of fixed-time control are determined by enumeration experiment in different scenarios.

Experiment result

Single-ramp scenario

We finished the training process of single-ramp scenario when the episode number reached 1000. The elapsed time is less than 6 h in our training platform. The training process is presented in Figure 7. It can be seen that the agent made poor choices in the initial stage, but then made rapid progress soon; following a rising process the return tends to convergence after 400 episodes. The average speed at the bottleneck point increased about 8.6% and the average queue size on the ramp decreased a lot from the initial worst condition.

To ensure the visibility of trends in the time series curves and to minimize sharp fluctuations, the rolling average strategy with a step size of 10 episodes is employed, alongside a light-colored error band in Figure 7. The width of the error bands was set as 1.645σ (i.e. 90% confidence interval), where σ is the standard deviation of the rolling average interval. The same treatment is also implemented in Figures 8 and 9.

The policy network is thrown into utilization after training process. Four types of ramp metering approaches, i.e. (i) no-control, (ii) fixed-time control, (iii) ALINEA, and (iv) the proposed algorithm, are tested in three demand scenarios. The results are presented in Table 1. No-control strategy always leads to the minimum queue size on ramp but gets the lowest mainstream speed. Fixed-time strategy ameliorates main-stream traffic somehow, but the effect is limited. The performance of ALINEA is unstable. In stationary volume scenario, ALINEA reaches the highest speed but produces overlong queue size. In flat peak scenario, ALINEA achieves worse performance than fixed-time control. The proposed algorithm achieves fairly desirable performance and gets the highest return under all of the demand profiles.

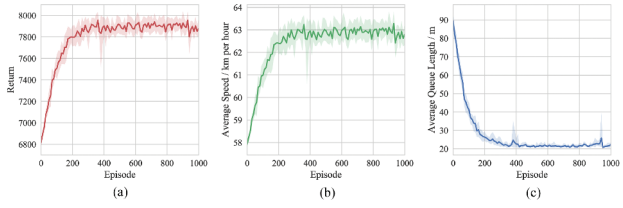

Figure 7. Log visualization of the training process.

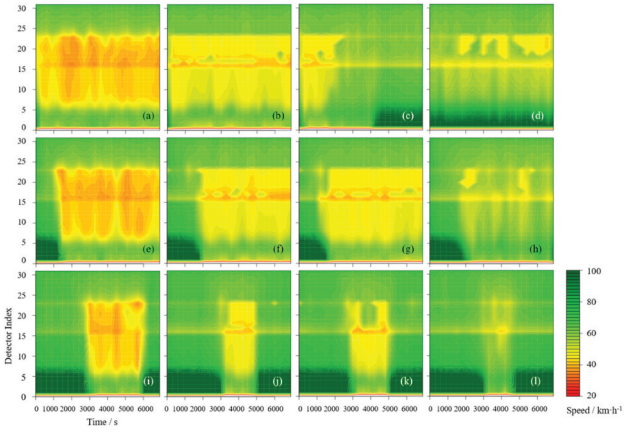

Figure 8. Space-time diagram of speed under different metering strategies.

Shown in Figure 8 is the space-time diagram of mainstream speed under different metering strategies. Inset (a-d), (e-h), and (i-l) show the results of stationary volume, flat peak, and sharp peak demand scenarios embedded with no-control, fixed-time control, ALINEA, and the proposed algorithm, respectively. It is clear that the proposed algorithm can decrease the duration as well as the spatial range of mainline congestion in a variety of demand scenarios. Since the policy network had not been trained in all of these scenarios specifically, the result also presents the scalability and robustness of the algorithm.

In Figure 9, queue length variations are compared for various ramp metering strategies under stationary volume, flat peak, and sharp peak traffic demand profiles. The no-control strategy results in almost no queues throughout the control period, but this reduces travel speeds in the bottleneck section downstream of the mainline, as shown in Figure 8. Fix-time strategy performs similarly under all three demand profiles and does not adapt to traffic flow variation due to the lack of dynamic regulation capabilities. The ALINEA and proposed algorithms do not vary much in stationary volume demand profile, but exhibit significant differences in flat peak and sharp peak scenarios, with the proposed algorithm forming earlier and dissipating faster than ALINEA in the former, and generating a higher peak queue length that dissipates faster and improves the bottleneck section's speed accordingly in the latter, preventing congestion on the mainstream of the freeway.

As previously observed, the ALINEA controller exhibits some time delay in responding to congestion, which is consistent with previous studies (Kan et al. 2016). The time delay is primarily due to the distance between the ramp flow change and the impact on flow dynamics of the downstream bottleneck. Furthermore, when the traffic demand increases, congestion downstream becomes more severe, and control errors tend to accumulate, leading to suboptimal control and longer queues on the ramp. For this reason, ALINEA is comparatively efficient in managing scenarios involving stationary volume demand profile, but its effectiveness slightly diminishes in scenarios marked by significant fluctuations (e.g. flat peak and sharp peak scenarios in Table 1) in traffic demand. In contrast, the proposed algorithm demonstrates greater sensitivity to the occurrence of congestion and can effectively manage the time delay between the ramp flow change and the corresponding flow change of downstream bottleneck, resulting in improved control outcomes.

Multi-ramp scenario

In the coordinated ramp metering scenario, we compared the proposed method with HERO strategy based on the double-ramp network in Figure 5(b) in flat peak demand profile. Key performance metrics including mainline speed in downstream bottleneck,

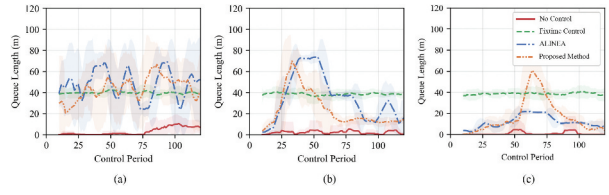

Figure 9. Time-varying queue lengths under different metering strategies.

Table 1. Comparison of metering strategies.

Demand Profile	Metering Strategy	Measures		
		Speed (km/h)	Queue length (m)	Return
Stationary Volume	No-control	38.62	4.57	4579.59
	Fixed-time	47.84	39.49	5266.78
	ALINEA	48.57	50.74	5339.52
	Proposed algorithm	50.06	48.90	5420.40
Flat Peak	No-control	43.08	4.45	5116.61
	Fixed-time	51.49	37.28	5731.62
	ALINEA	49.89	38.74	5522.56
	Proposed algorithm	55.71	28.32	6345.36
Sharp Peak	No-control	54.66	1.47	6541.25
	Fixed-time	59.76	38.37	6711.21
	ALINEA	57.54	11.09	6771.72
	Proposed algorithm	60.97	29.34	6964.29

ramp queue length, and reward value were recorded for each ramp during the 2-h episode. The results are presented in Figure 10(a,b,c) respectively. In terms of mainline speed, both of the metering strategies maintain the mainline speed at a relatively desirable level (approximately 50 ~ 60 km/h. However, a significant plunge in mainline speed is observed with an increase in volume when using the HERO metering strategy. The proposed method outperforms HERO by better controlling the ramp queue length, while HERO generates a longer queue at Ramp I due to the queue control policy implemented in Ramp II. Since HERO utilizes the ALINEA strategy at the local level, it inherits its drawbacks, leading to significant time delay in awareness of the changes of traffic flow dynamics. Overall, the proposed method demonstrates a 2.04% higher average reward value compared with HERO strategy.

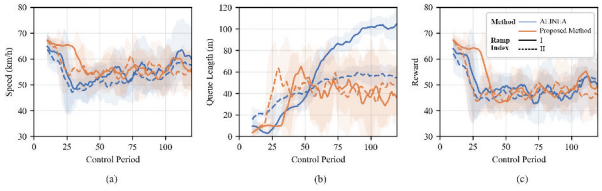

Figure 10. Performance of HERO and proposed algorithm in coordinated ramp metering scenario.

Conclusion

The advancement of reinforcement learning empowers the accurate control of stochastic systems. In this paper, we proposed a novel ramp metering algorithm, in which the multi-agent deep reinforcement learning technique is the primary enabler. The algorithm works in a model-free fashion: the agents take the profile occupancies as input and output the metering rates directly (end to end), without the requirement to pre-assume a freeway traffic flow model. The metering approach is modeled as a policy neural network. Minimal fine-tuning is required when the implementation environment is changed. Hence, the algorithm provides favorable flexibility for time-varying demand and migrating implementation.

We developed a SUMO-based platform to implement the algorithm. A set of simulation experiments demonstrated that the DRL-based ramp metering method is able to automatically smooth traffic flow, thus postponing the occurrence and mitigating the impact of congestion at the bottleneck. In the local ramp metering experiments, the performance of the proposed algorithm is 1.5%, 14.9%, and 2.8% higher than the state-of-the-practice ramp metering algorithm ALINEA in the stationary, flat peak and sharp peak demand scenarios, respectively. In the coordinated ramp metering scenario, the proposed DRL-based algorithm also presents the capability of keeping mainstream speed at a desirable level and preventing the spread of queuing on the ramp, especially during the occurrence of long-lasting congestion.

Further research on the designation of reward function, ramp coordination mechanism, and neural network model architecture is a promising area to explore. Although this work utilized a relatively straightforward reward function, it is important to investigate the effects of various reward definitions in the context of ramp metering for future studies. Additionally, the form of cooperation in multi-agent scenarios warrants investigation. Finally, to improve control system performance, we recommend exploring the use of recurrent neural networks with predictive mechanisms for forward-looking control.

Acknowledgments

This paper is an extension of a shorter conference version that appeared in Deng et al. (2019). The authors gratefully acknowledge the financial supports by Shandong Technology and Business University under grant number BS202305.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

The work was supported by the Shandong Technology and Business University [BS202305].

ORCID

Fuwen Deng http://orcid.org/0000-0002-1150-682X

References

Brockman, G. Cheung, V. Pettersson, L. Schneider, J. Schulman, J. Tang, J. Zaremba, W. 2016. "Openai Gym." *ArXiv Preprint*. arXiv:1606.01540

Camacho, E. F., and C. A. Bordons. 1995. *Model Predictive Control in the Process Industry*. London: Springer.

Chen, C.-I., J. B. Cruz Jr, and J. G. Paquet. 1974. "Entrance Ramp Control for Travel-Rate Maximization in Expressways." *Transportation Research* 8 (6): 503–508. https://doi.org/10.1016/0041-1647(74)90026-4.

Deng, F., Jin, J., Shen, Yu, and Du, Y. "Advanced Self-Improving Ramp Metering Algorithm Based on Multi-Agent Deep Reinforcement Learning," *International Conference on Intelligent Transportation Systems (ITSC)* Auckland, New Zealand, 2019.

Fares, A. and W. Gomaa. "Freeway Ramp-Metering Control Based on Reinforcement Learning," *11th IEEE International Conference on Control & Automation (ICCA)* Singapore, 2014, pp. 1226–1231.

Han, Y., M. Ramezani, A. Hegyi, Y. Yuan, and S. Hoogendoorn. 2020. "Hierarchical Ramp Metering in Freeways: An Aggregated Modeling and Control Approach." *Transportation Research Part C: Emerging Technologies* 110 (Jan): 1–19. https://doi.org/10.1016/j.trc.2019.09.023.

Han, Y., M. Wang, L. Li, C. Roncoli, J. Gao, and P. Liu. 2022. "A Physics-Informed Reinforcement Learning-Based Strategy for Local and Coordinated Ramp Metering." *Transportation Research Part C: Emerging Technologies* 137 (Apr): 103584. https://doi.org/10.1016/j.trc.2022.103584.

Heess, N., TB, D. Sriram, S. Lemmon, J. Merel, J. Wayne, G. Tassa, Y. Erez, T. Wang, Z. Eslami, SM 2017. "Emergence of Locomotion Behaviours in Rich Environments." *ArXiv Preprint* arXiv:1707.02286.

Hegyi, A., B. De Schutter, and H. Hellendoorn. 2005. "Model Predictive Control for Optimal Coordination of Ramp Metering and Variable Speed Limits." *Transportation Research Part C: Emerging Technologies* 13 (3): 185–209. https://doi.org/10.1016/j.trc.2004.08.001.

Jacobson, L., K. Henry, and O. Mehyar. 1989. "Real-Time Metering Algorithm for Centralized Control." *Transportation Research Record*, 1732:20–32.

Jin, J., and X. Ma. 2015. "Adaptive Group-Based Signal Control by Reinforcement Learning." *Transportation Research Procedia* 10 (July): 207–216. https://doi.org/10.1016/j.trpro.2015.09.070.

Kan, Y., Y. Wang, M. Papageorgiou, and I. Papamichail. 2016. "Local Ramp Metering with Distant Downstream Bottlenecks: A Comparative Study." *Transportation Research Part C: Emerging Technologies* 62 (9): 149–170. https://doi.org/10.1016/j.trc.2015.08.016.

Khamis, M. A., and W. Gomaa. 2014. "Adaptive Multi-Objective Reinforcement Learning with Hybrid Exploration for Traffic Signal Control Based on Cooperative Multi-Agent Framework." *Engineering Applications of Artificial Intelligence* 29:134–151. https://doi.org/10.1016/j.engappai.2014.01.007.

Kingma, D., and J. Ba. 2014. "Adam: A Method for Stochastic Optimization." *ArXiv Preprint* 1412:6980.

Li, L., Y. Lv, and F.-Y. Wang. 2016. "Traffic Signal Timing via Deep Reinforcement Learning." *IEEE/CAA Journal of Automatica Sinica* 3 (3): 247–254.

Lipp, L., L. Corcoran, and G. Hickman. 1991. "Benefits of Central Computer Control for Denver Ramp-Metering System." *Transportation Research Record*, 1320:3–6.

Liu, B., Y. Tang, Y. Ji, Y. Shen, Y. Du, and V. L. Knoop. 2021. "A Deep Reinforcement Learning Approach for Ramp Metering Based on Traffic Video Data." *Journal of Advanced Transportation* 2021 (Oct): 1–13. https://doi.org/10.1155/2021/6669028.

Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Hilbrich, R., Rummel, J., Wagner, P. "Microscopic Traffic Simulation Using SUMO," *International Conference on Intelligent Transportation Systems (ITSC)* Maui, Hawaii, USA, 2018.

Markos Papageorgiou, J.-M. B., and H. Hadj-Salem. 1990. "ALINEA a Local Feedback Control Law for on Ramp Metering." *Transportation Research Reccord* 1320:58–64.

Mnih, V. "Asynchronous Methods for Deep Reinforcement Learning," *International Conference on Machine Learning* New York, USA. PMLR, 2016.

Mnih, V., K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, et al. 2015. "Human-Level Control Through Deep Reinforcement Learning." *Nature* 518 (7540): 529–533. https://doi.org/10.1038/nature14236.

Mosharafian, S., S. Afzali, and J. Mohammadpour Velni. 2022. "Leveraging Autonomous Vehicles in Mixed-Autonomy Traffic Networks with Reinforcement Learning-Controlled Intersections." *Transportation Letters* 1–8. https://doi.org/10.1080/19427867.2022.2146302.

Papageorgiou, M. 1980. "A New Approach to Time-Of-Day Control Based on a Dynamic Freeway Traffic Model." *Transportation Research Part B: Methodological* 14 (4): 349–360. https://doi.org/10.1016/0191-2615(80)90015-6.

Papageorgiou, M., J. M. Blosseville, and H. Hadj-Salem. 1990. "Modelling and Real-Time Control of Traffic Flow on the Southern Part of Boulevard Peripherique in Paris: Part II: Coordinated On-Ramp Metering." *Transportation Research Part A* 24 (5): 361–370. https://doi.org/10.1016/0191-2607(90)90048-B.

Papageorgiou, M., H. Hadj-Salem, and F. Middelham. 1997. "ALINEA Local Ramp Metering: Summary of Field Results." *Transportaion Research Record* 1603 (970032): 90–98. https://doi.org/10.3141/1603-12.

Papamichail, I., and M. Papageorgiou. 2008. "Traffic-Responsive Linked Ramp-Metering Control." *IEEE Transaction on Intelligent Transportation Systems* 9 (1): 111–121. https://doi.org/10.1109/TITS.2007.908724.

Schulman, J., F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. 2017. "Proximal Policy Optimization Algorithms." *ArXiv Preprint* arXiv:1707.06347.

Shalev-Shwartz, S., S. Shammah, and A. Shashua. 2016. "Safe, Multi-Agent, Reinforcement Learning for Autonomous Driving." *ArXiv Preprint* arXiv:1610.03295.

Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, et al. 2016. "Mastering the Game of Go with Deep Neural Networks and Tree Search." *Nature* 529 (7587): 484–489. https://doi.org/10.1038/nature16961.

Smaragdis, E., and M. Papageorgiou. 2003. "Series of New Local Ramp Metering Strategies: Emmanouil Smaragdis and Markos Papageorgiou." *Transportation Research Record* 1856 (1): 74–86. https://doi.org/10.3141/1856-08.

Smaragdis, E., M. Papageorgiou, and E. Kosmatopoulos. 2004. "A Flow-Maximizing Adaptive Local Ramp Metering Strategy." *Transportation Research Part B: Methodological* 38 (3): 251–270. https://doi.org/10.1016/S0191-2615(03)00012-2.

Stephanedes, Y. J., "Implementation of On-Line Zone Control Strategies for Optimal Ramp Metering in the Minneapolis Ring Road," *7th International Conference on IET* London, UK, pp. 181–184, 1994.

Stylianopoulou, E., M. Kontorinaki, M. Papageorgiou, and I. Papamichail. 2020. "A Linear-Quadratic-Integral Regulator for Local Ramp Metering in the Case of Distant Downstream Bottlenecks." *Transportation Letters* 12 (10): 723–731. https://doi.org/10.1080/19427867.2019.1700005.

Tabadkani Aval, S. S., and N. Eghbal. 2020. "Feedback-Based Cooperative Ramp Metering for Highway Traffic Flow Control: A Model Predictive Sliding Mode Control Approach." *International Journal of Robust and Nonlinear Control* 30 (18): 8259–8277. https://doi.org/10.1002/rnc.5237.

Taylor, C. J., D. Meldrum, and L. Jacobson. 1998. "Fuzzy Ramp Metering: Design Overview and Simulation Results." *Transportation Research Record* 1634 (1): 10–18. https://doi.org/10.3141/1634-02 .

Van De Panne, M. 2017. "DeepLoco: Dynamic Locomotion Skills Using Hierarchical Deep Reinforcement Learning." *ACM Transactions on Graphics* 36 (4): 41. https://doi.org/10.1145/3072959.3073602.

van de Weg, G. S., A. Hegyi, S. P. Hoogendoorn, and B. De Schutter. 2019. "Efficient Freeway MPC by Parameterization of ALINEA and a Speed-Limited Area." *IEEE Transaction on Intelligent Transportation Systems* 20 (1): 16–29. https://doi.org/10.1109/TITS.2018.2790167.

Wang, C. 1972. "On a Ramp-Flow Assignment Problem." *Transportation Science* 6 (2): 114–130. https://doi.org/10.1287/trsc.6.2.114.

Wang, Z., N. de Freitas, and M. Lanctot, "Dueling Network Architectures for Deep Reinforcement Learning," *International Conference on Machine Learning* New York, USA, PMLR, 2016.

Wang, J. J., and A. D. May. 1973. "Computer Model for Optimal Freeway On-Ramp Control." *Highway Research Record* 469:16–25.

Wang, C., Y. Xu, J. Zhang, and B. Ran. 2022. "Integrated Traffic Control for Freeway Recurrent Bottleneck Based on Deep Reinforcement Learning." *IEEE Transaction on Intelligent Transportation Systems* 23 (9): 15522–15535. https://doi.org/10.1109/TITS.2022.3141730 .

Wattleworth, J. A. 1967. "Peak Period Analysis and Control of a Freeway System with Discussion." *Highway Research Record*, 157:1–21.

Wu, C., Parvate, K., Kheterpal, N., Dickstein, L., Mehta, A., Vinitsky, E., and Bayen, Alexandre M "Framework for Control and Deep Reinforcement Learning in Traffic," *IEEE Conference on Intelligent Transportation Systems (ITSC)* Maui, Hawaii, USA, 2018.

Xu, Q., Z. Liu and Z. Xu. "A Novel Ramp Metering Algorithm Based on Deep Reinforcement Learning," *2nd International Conference on Algorithms, High Performance Computing and Artificial Intelligence (AHPCAI)* Guangzhou, China, 2022, pp. 128–133.

Yang, G., Z. Tian, D. Wang, and H. Xu. 2019. "Queue Length Estimation for a Metered On-Ramp Using Mesoscopic Simulation." *Transportation Letters* 11 (10): 570–579. https://doi.org/10.1080/19427867.2018.1477491.