Learning Cooperative Policy among Self-Driving Vehicles for Relieving Traffic Jams

Shota Ishikawa[1], **Sachiyo Arai**[1],
[1] Chiba University 1-33 Yayoi-cho Inage-ku Chiba, Japan 263-8522
aefa3392@chiba-u.jp, arai@tu.chiba-u.ac.jp

Abstract

We propose a novel driving policy which is a velocity control for self-driving vehicles to relieve traffic jams. Although the driving policy in previous research was empirically designed for a given traffic situation, which meant that the driving policy required to be reconfigured for every traffic situation and every change in traffic, we propose a driving policy that is learned by a learning agent via reinforcement learning using the data collected from the self-driving vehicles during simulation. The driving policy is relayed to the smart vehicles, which, in turn, are guided by the driving policy. To test and evaluate our proposed driving policy, we conducted traffic flow simulations with manually driven and self-driving vehicles in several scenarios wherein the two key parameters, vehicle density and self-driving vehicle penetration rate, are assigned different values. Our findings show that a driving policy for self-driving vehicles does relieve traffic jams in conditions such as (1) when the vehicle density is 42 vehicles/km and the penetration of the self-driving vehicle is 10% of the total traffic, and (2) when the vehicle density is 50 vehicles/km and the penetration of the self-driving vehicle is 70% of the total traffic (at which point traffic flow is nearly optimized). In addition, we found that inter-vehicle communication among self-driving vehicles provides real-time traffic information that relieves traffic jam even more effectively.

1 Introduction

Traffic jams cause significant inconvenience and economic costs in terms of fuel usage and time lost. Traffic jams are caused by bottlenecks, which reduce road capacity, and by perturbations associated with manually driven vehicles, which can increase vehicle density. In a previous study, we proposed a driving policy for relieving perturbation-induced traffic jams in traffic situations involving homogeneous vehicles [Ishikawa and Arai, 2015]. In this present study, we propose a driving policy for relieving traffic jams in traffic situations involving heterogeneous vehicles, that is, both manually driven and self-driving vehicles.

The self-driving vehicle is equipped with smart functions, such as an adaptive cruise control (ACC) or cooperative adaptive cruise control (CACC) that can penetrate and potentially influence traffic flow. An ACC-equipped vehicle can automatically detect the leading vehicle and can control velocity using sensor and radar instruments. A CACC-equipped vehicle can receive driving information from the vehicle preceding it via vehicle-to-vehicle (V2V) communication. Some papers have proposed a driving policy of ACC and CACC to relieve traffic jams. For example, Kesting et al. [Kesting *et al.*, 2008] proposed the driving policy of ACC, and Forster et al. [Forster *et al.*, 2014] proposed the driving policy of CACC. Detecting traffic condition, these vehicles drive flexibility and improve traffic flow stability.

However, the current practice of designing a driving policy is challenging as the driving policy must account for any number of traffic situations (road structures, traffic regulations, etc.), consider perturbations induced by manually driven vehicles, and direct and coordinate self-driving vehicles. Designing driving policies requires simulation trial-and-error, is labor intensive, and is time consuming.

We propose the driving policy that is learned by a learner agent via reinforcement learning using data that are collected from the self-driving vehicles. In the proposed approach, a learner agent for the driving policy simultaneously interacts with the all self-driving vehicles in traffic simulation. Collecting driving data of the self-driving vehicles that obey the driving policy, the learner agent learns the driving policy from driving data. After this interaction repeats, the learner agent acquires the driving policy. To validate the proposed approach, we introduce self-driving vehicles equipped with driving policy into traffic jam simulations induced by perturbation of a manually driven vehicle. Several traffic situations having different vehicle densities and self-driving vehicle penetration rates were used in the simulation. The effectiveness of the driving policy on relieving traffic jam was measured based on the amount of increase in traffic flow.

The rest of this paper is organized as follows. In Section 2, we discuss our approach to relieving traffic jams by means of a learner agent that learns and updates the driving policy through data collected from self-driving vehicles. In Section 3, we describe a traffic problem scenario. In Section 4, we propose a framework for learning the driving policy by a learner agent. In Section 5, we describe a Generalized

Nagel–Schreckenberg (GNS) model of traffic flow for manually driven and self-driving vehicles. In Section 6, we describe the traffic simulation experiments conducted based on our proposed approach. In Section 7, we conclude this paper with remarks on future work.

2 Related Work

The proposed approach aims at generating a driving policy with data collected from self-driving vehicles and reinforcement learning of the driving policy by a learner agent. This approach is based on works related to traffic flow control in terms of driving policy and reinforcement learning.

To prevent traffic jams caused by the perturbation of a manually driven vehicle, the vehicle must be able to maintain an appropriate gap distance between itself and the preceding vehicles to prevent perturbation from propagating downstream to eventually be reflected upstream. Research has been done on the effect of maintaining an appropriate gap between vehicles for relieving traffic jam when one vehicle, all vehicles, or some vehicles are regulated by a driving policy [Kamal *et al.*, 2014; Forster *et al.*, 2012; Papacharalampous *et al.*, 2015].

The driving policy for a manually driven vehicle may include predicting a traffic situation using inter-vehicle communication and recommending that the driver keep an appropriate amount of distance [Knorr *et al.*, 2012]. Kesting et al. and Forster et al. proposed a driving policy for an ACC and CACC self-driving vehicle that adapts to a traffic situation, respectively [Kesting *et al.*, 2008; Forster *et al.*, 2014]. Won et al. proposed fuzzy inference systems that effectively capture the dynamics of traffic jams [Won *et al.*, 2017]. Although these approaches are effective ways of relieving traffic jams, designing a driving policy that anticipates various traffic scenarios is difficult. We propose an approach that uses a learner agent to learn the driving policy in order to cut down on designing the policy.

Research on reinforcement learning for traffic flow optimization includes finding policies dictating how speed limits should be assigned to highway sections [Walraven *et al.*, 2016] and controlling ramp metering devices with Q-learning [Rezaee *et al.*, 2012]. For advanced reinforcement learning approaches, a multi-objective reinforcement learning involves learning the traffic signal policy [Khamis and Gomaa, 2014], and multi-agent reinforcement learning determines the route planning [Zolfpour-Arokhlo *et al.*, 2014]. In contrast, our approach acquires the driving policy of the self-driving vehicles.

3 Traffic Problem Scenario

Figure 1 shows a traffic scenario involving two road-to-vehicle communication infrastructures (R2Vs), N self-driving vehicles, and M manually driven vehicles. The R2Vs, which share information on the number of self-driving or manually driven vehicles passed by it, are installed at the edge of a road section having length L. The R2Vs can calculate the traffic density ρ and the penetration rate of the self-driving vehicle μ of the road section. The upstream R2V sends the driving policy $\pi_{\rho,\mu}$ corresponding to ρ and μ to the self-driving vehicles passed by it.

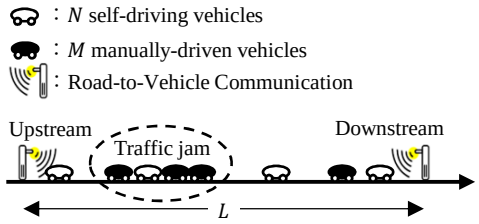

Figure 1: A traffic situation that we assume in this paper.

We propose a solution to relieving traffic jams on the road by instituting driving policy $\pi_{\rho,\mu}$, wherein the self-driving vehicle complies with driving policy $\pi_{\rho,\mu}$; that is, the self-driving vehicle observes a state s, and performs action output a expressed as $\pi_{\rho,\mu}(s) = a$. The state s is a six-dimensional vector $s = (\phi^{\mathrm{vel}}, \phi^{\mathrm{gap}}, \phi^{\mathrm{rel}}, \phi^{\mathrm{c-d}}, \phi^{\mathrm{c-v}}, \phi^{\mathrm{c-g}})$, where ϕ^{vel}, ϕ^{gap}, ϕ^{rel}, $\phi^{\mathrm{c-d}}$, $\phi^{\mathrm{c-v}}$, and $\phi^{\mathrm{c-g}}$ indicate velocity, gap, relative velocity, communication distance between communication partners, communication partner's velocity, and communication partner's gap, respectively. The action a is velocity control. The state s contains the information about preceding vehicle, and the driving policy is cooperative policy to relieving traffic jams.

4 Framework for the Reinforcement Learning of Driving Policy

The reinforcement learning framework shown in Figure 2 comprises the traffic environment and the learner agent.

Environment

The traffic environment comprises self-driving and manually driven vehicles on a road characterized by periodic-boundary conditions. Because the number of vehicles is constant, vehicle density ρ and penetration rate μ are also constant. The learner agent therefore learns the driving policy $\pi_{\rho,\mu}$ by interacting with a traffic environment in which vehicle density ρ and penetration rate μ are constant.

Learner agent

We explain a procedure that the learner agent updates the driving policy whenever time t is updated from t to $t+1$. At time t, the learner agent delivers the driving policy $\pi_{t,\rho,\mu}$ to all self-driving vehicles. Following equation (1), the driving policy outputs randomly selected action with probability ϵ or action a' selected by $\mathrm{argmax}_{a'} Q_{\rho,\mu}(s,a')$ with $1-\epsilon$. Here, the probability $\epsilon = \{\epsilon | 0 \leq \epsilon \leq 1\}$ is a parameter used to explore a new state, and $Q_{\rho,\mu}(s,a)$ is an action value function when the vehicle state and action are, respectively, s and a. After all vehicles drive, at time $t+1$, the self-driving vehicles observe the next state s_{t+1} and receive a reward r_{t+1}. The learner agent then collects the driving data $\zeta_n = \{s_t, a_t, s_{t+1}, r_{t+1}\}$ from the self-driving vehicle.

$$\pi_{\rho,\mu}(s) = \begin{cases} \text{random select } a & \epsilon \\ \mathrm{argmax}_{a'} Q_{\rho,\mu}(s,a') & 1-\epsilon \end{cases} \quad (1)$$

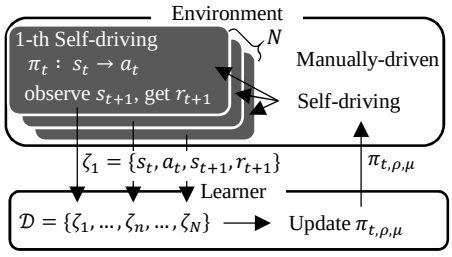

Figure 2: Dynamic interaction between the learner agent and traffic environment within the reinforcement learning of the driving policy.

Algorithm 1 Update action value Q at t, ρ, μ

Input $\mathcal{D} = \{\zeta_n | n \in N\}, \zeta_n = \{s_t, a_t, s_{t+1}, r_{t+1}\}$

1: $Q^{\text{new}}_{\rho,\mu} \leftarrow Q_{\rho,\mu}$

2: **for** $n \leftarrow 1$ *to* N **do**

3: $s_t, a_t, s_{t+1}, r_{t+1} \in \zeta_n$

4: $Q^{\text{new}}_{\rho,\mu}(s_t, a_t)$
$\leftarrow (1-\alpha) Q_{\rho,\mu}(s_t, a_t)$
$+ \alpha(r_{t+1} + \gamma \text{argmax}_{a'} Q_{\rho,\mu}(s_{t+1}, a'))$

5: **end for**

6: $Q_{\rho,\mu} \leftarrow Q^{\text{new}}_{\rho,\mu}$

Following Algorithm 1, the learner agent updates the driving policy using dataset $\mathcal{D} = \{\zeta_n | n \in N\}$. First, the learner agent inserts an action value $Q_{\rho,\mu}$ into a $Q^{\text{new}}_{\rho,\mu}$. Second, the learner agent updates the $Q^{\text{new}}_{\rho,\mu}$ N times. The index n of the most upstream self-driving vehicle is 1 and this index is incremented by 1 from upstream to downstream. The $Q^{\text{new}}_{\rho,\mu}$ is updated by the equation at line 4 in Algorithm 1. Finally, the learner agent inserts the $Q^{\text{new}}_{\rho,\mu}$ into the action value $Q_{\rho,\mu}$.

The equation in line 4 in Algorithm 1 is based on *Q-learning* [Sutton and Barto, 1998]. Here, α is the learning rate, and γ is the discount factor. The learning rate α is a parameter indicating, in degrees, the update of the action value, and the discount factor γ is a parameter that determines the current value of a reward expected to be obtained in the future. The self-driving vehicle accepts the reward according to its own state. The learning agent determines the driving policy that maximizes the action value that is the sum of rewards r discounted by γ at each time t.

5 Simulation Modeling

In this study, we used a modified Generalized Nagel–Schreckenberg (GNS) model [Ishikawa and Arai, 2015] [1]. The $NaSch$ model [Nagel and Schreckenberg, 1992], which is the basic cellular automaton for the description of traffic flow, can model the perturbation of each vehicle. The GNS is used to model the number of communication partners n_i^{com} and the maximum communication distance d_i^{com}.

5.1 Terminology

Figure 3 shows a notation of the GNS. The cellular automaton model reproduces the traffic flow which is characterized by a series of cells that indicate whether a vehicle occupies or does not occupy the cell. Vehicle $i+1$ is ahead of vehicle i, as the vehicle index is incremented by one. x_i, g_i, $v_i(t)$, and $v^{\text{rel}}(t)$ indicate the coordinate, gap, velocity, and relative velocity, respectively. The self-driving vehicle i (white car) is able to communicate with the preceding white $i+2$ (white car) within the given maximum communication distance d_i^{com}. i^{com}, d_i, $g_{i^{\text{com}}}$, and $v_{i^{\text{com}}}(t)$ indicate the index of the communication partner, the communication distance, the gap that the communication partner possesses, and the velocity of the communication partner, respectively.

[1] The point of modification and driving rule are provided in the appendix

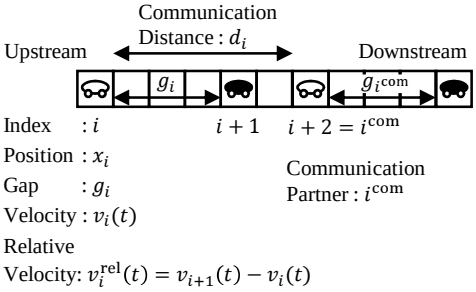

Figure 3: A notation of the Generalized Nagel–Schreckenberg model

Table 1: GNS parameters for the manually driven vehicle, ACC self-driving vehicle, and CACC self-driving vehicle.

	Manual	ACC	CACC
Perturbation p	$[0, 1]$	-	-
Policy action p^{pol}	-	$[0, 1]$	$[0, 1]$
Com. number n_i^{com}	-	-	$[1, \infty)$
Com. distance d_i^{com}	-	$[1, \infty)$	$[1, \infty)$

Road model

The GNS reproduces the road section along length L. The road section contains the perturbation section along length $l (0 \leq l \leq L)$ in which the manually driven vehicle decelerates at probability p. The occurrence of a traffic jam is due to the deceleration of the manually driven vehicle within the perturbation section [Sugiyama *et al.*, 2008], which corresponds to a sag or tunnel in the real world environment.

Vehicle model

The GNS parameters for the manually driven vehicle, ACC self-driving vehicle, and CACC self-driving vehicle are shown in Table 1. The GNS parameters are set at a probability of perturbation p, a probability of driving policy p^{pol}, the number of communication partners n_i^{com}, and the maximum communication distance d_i^{com}. The manually driven vehicle decelerates with probability p in the perturbation section, but the self-driving vehicle does not decelerate. The policy-driven self-driving vehicle decelerates at probability p^{pol} with velocity control on any section of the road. The CACC self-driving vehicle has $1 \leq n_i^{\text{com}}$ communication partners, and the ACC or CACC self-driving vehicle has a maximum communication distance of $1 \leq d_i^{\text{com}}$.

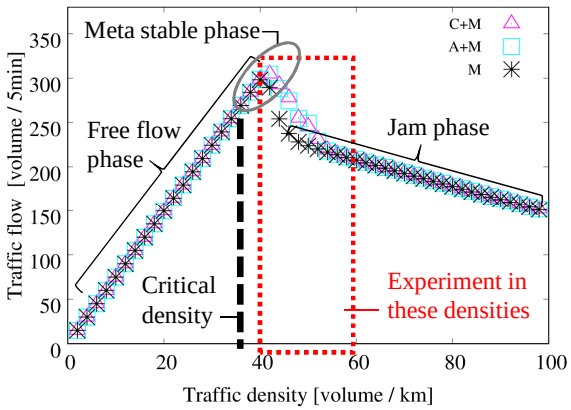

Figure 4: A fundamental diagram of Generalized Nagel-Schreckenberg model.

5.2 Fundamental diagram of a GNS model

Generally, traffic flow analysis focuses on the relationship between traffic flow and vehicle density as shown in Figure 4 using a GNS model diagram comprising traffic flow plots of C+M (CACC self-driving and manually driven vehicles), A+M (ACC self-driving and manually-driven vehicles), and M (manually driven vehicles). Traffic flow as represented by the number of vehicles passing through a measurement point per 5 min is a function of vehicle density, as represented by the number of vehicles per km.

The penetration rate of the self-driving vehicle is 30%. In addition, the diagram shows the free-flow phase, and there is a positive linear relationship between traffic flow and vehicle density. In the jam phase, there is a negative linear relationship between traffic flow and vehicle density. The intersection of the free-flow and jam phases is called "critical density." In the meta-stable phase, traffic flow is as high as in the free-flow phase even when vehicle density is greater than critical density.

For this study, we assume that the effect of relieving a traffic jam is greater as the traffic flow becomes larger than the traffic flow of the jam phase. The plots show that the free-flow phase transitions to the jam phase at 40 vehicles/km. We evaluated the effectiveness of the driving policy in vehicle density ranging from 40 to 60 vehicles/km (red dashed line).

6 Experiment

6.1 Experimental setting

Experimental procedure

A trial of experiment excuses two steps and each step consists of some episodes. Before an episode of simulation starts, we initialize the road by orienting the vehicles randomly and moving the vehicles around 1000 simulation times. We then execute a learning step, in which vehicles move around for a total of 1000 episodes (10,000 simulation times per episode), to be followed by an evaluation step in which vehicle move 100 episodes. We repeated this experiment 10 times and averaged the results.

Table 2: Values of the state vector elements.

Element	Value range
Slow	$0 \leq v_i(t) \leq 1$
Middle	$1 < v_i(t) \leq 3$
Fast	$3 < v_i(t) \leq v^{\mathrm{limit}}$
Next	$0 \leq g_i \leq 1$
Short	$1 < g_i \leq 4$
Long	$4 < g_i \leq d_i^{\mathrm{com}}$
Not in	$d_i^{\mathrm{com}} < g_i$
Depart	$v_i^{\mathrm{rel}}(t) \leq -2$
Track	$-2 < v_i^{\mathrm{rel}}(t) \leq 1$
Approach	$1 < v_i^{\mathrm{rel}}(t)$
Near	$0 \leq d_i \leq 6$
Far	$6 < d_i \leq d_i^{\mathrm{com}}$
Disconnected	$d_i^{\mathrm{com}} < d_i$ or $n_i^{\mathrm{com}} = 0$

Road and vehicle setting

We evaluate the proposed driving policy using a road model under periodic-boundary condition, which is the same condition as the learning step. Compared with the open-boundary condition in which vehicle density may change because of inflow rate, vehicle density is constant under the periodic-boundary condition in order to evaluate the effect of driving policy on velocity without the confounding factor of inflow rate. The experimental conditions for road and vehicle are as follows:

- a time t = 1 s
- 1 cell = 5 m
- single-lane road under periodic-boundary condition
- limited velocity 5 cell/time = 90 km/h
- road length L = 100 cells
- road where perturbation occurs l = 5 cells
- perturbation probability p = 0.2
- maximum communication distance d_i^{com} = 20
- the number of communication partners n_i^{com} = 1

Learning setting

The probability of exploration is $\epsilon = 0.01$ from 1 to 500 episodes, and $\epsilon = 0$ from 501 to 1100 episodes, learning rate is $\alpha = 0.01$ from 1 to 1000 episodes, and $\alpha = 0$ from 1001 to 1100 episodes, and discount factor is $\gamma = 0.9$.

The elements of the six-dimensional vector of state $s = (\phi^{\mathrm{vel}}, \phi^{\mathrm{gap}}, \phi^{\mathrm{rel}}, \phi^{\mathrm{c-d}}, \phi^{\mathrm{c-v}}, \phi^{\mathrm{c-g}})$ are listed as follows:

- ϕ^{vel} ={slow, middle, fast}
- ϕ^{gap} ={next, short, long, not in}
- ϕ^{rel} ={depart, track, approach, not in}
- $\phi^{\mathrm{c-d}}$ ={near, far, disconnected}
- $\phi^{\mathrm{c-v}}$ ={slow, middle, fast, disconnected}
- $\phi^{\mathrm{c-g}}$ ={next, short, long, not in, disconnected}

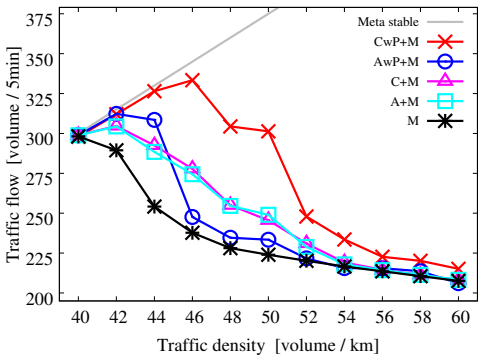

Figure 5: The fundamental diagram of the GNS model with a penetration rate μ of 30%.

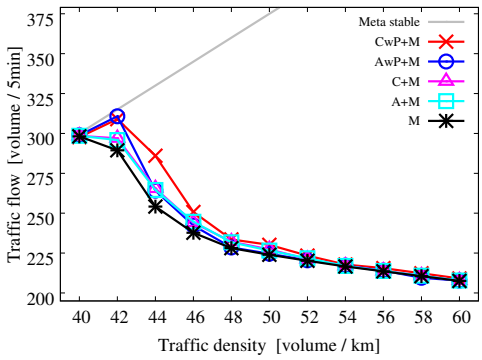

Figure 6: The fundamental diagram of the GNS model with a penetration rate μ of 10%.

Table 2 lists the details of the elements.

The action a is $p^{\mathrm{pol}} = 0$ or $p^{\mathrm{pol}} = 1$.

Equation (2) determines the penalty as r. The self-driving vehicle accepts penalty when any of the following three conditions is satisfied; the first condition is when the self-driving vehicle stops; the second condition is when the self-driving vehicle has a gap larger than 7 cells; and the third condition is when the self-driving vehicle has an absolute value of relative speed more than 1 cell/time.

$$r_t = \begin{cases} -1 & v_i(t) = 0 \text{ or } g_i > 7 \text{ or } |v_i^{\mathrm{rel}}(t)| > 1 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

6.2 Experimental results

Figure 5 shows a fundamental diagram of GNS model with a penetration rate μ of 30%. Plots of the traffic flow for CwP+M (CACC self-driving with policy and manually driven vehicles) and AwP+M(ACC self-driving vehicle with policy and manually driven vehicles) indicate that both CwP+M and AwP+M relieve the traffic jam until vehicle density 44 vehicles/km. CwP+M traffic flow is greater than AwP+M traffic flow. Note that the meta-stable traffic flow (gray line) is the optimal traffic flow when all vehicles maintain limited velocity.

Figure 6 shows a fundamental diagram of the GNS model with a penetration rate μ of 10%. CwP+M and AwP+M successfully relieve the traffic jam for a vehicle density of 42 vehicles/km.

Figure 7 shows a fundamental diagram of the GNS model with a penetration rate μ is 70%. CwP+M achieves not only the highest but also near optimum traffic flow among all of the experiments up to a vehicle density of 60 vehicles/km.

Figure 8 shows traffic flow as a function of the penetration rate of self-driving vehicles. The traffic flow of C+M and A+M increases as the penetration rate climbs, but the traffic flow of CwP+M and AwP+M does not, which is to say that increasing the number of self-driving vehicles with a driving policy does not necessarily increase traffic flow.

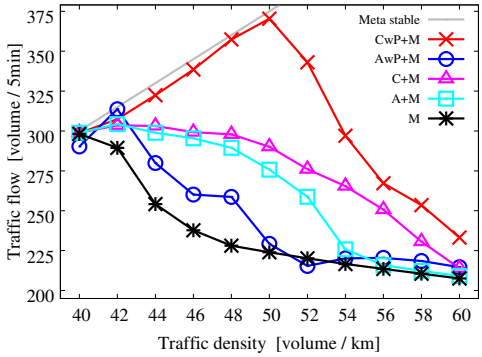

Figure 7: The fundamental diagram of the GNS model with a penetration rate μ of 70%.

6.3 Discussion

Measuring the effect of a driving policy for self-driving vehicles on relieving traffic jams

Table 3 shows the traffic volume and the average number of vehicles that stop per time unit in a traffic scenario having a vehicle density of 44 vehicles/km with 30% penetration rate for self-driving vehicles. The number of stopped vehicles decreases with increasing traffic flow, i.e., relieving traffic jams. There are two reasons for these results: one, a column of stopped vehicles is prevented from forming, and two, the column of stopped vehicles is dissolved quickly. When a column of stopped vehicles is formed because of a traffic jam, vehicles stop/start frequently. When a self-driving vehicle is introduced to the column, it accepts the stop penalty as it moves through the column as expressed in equation (2). The learner agent then learns the driving policy for preventing from forming the column, and for solving the column quickly. Consequently, the time during which the column exists on the road decreases, and all vehicles can smoothly drive without stopping.

The effect of inter-vehicle communication among self-driving vehicles on vehicle behavior

The difference between AwP+M and CwP+M is the number of communication parameters n_i^{com} and states s.

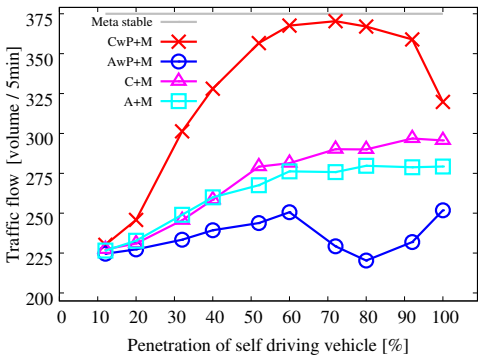

Figure 8: Traffic flow as a function of the penetration of the self-driving vehicle in case of a vehicle density ρ of 50 vehicles/km.

Table 3: The traffic flow and the average number of vehicles that stop per time step in case of 44 vehicles/km vehicle density and 30% penetration rate.

	Traffic flow [volume / 5min]	The number of stop [volume / time step]
M	254.1	2.1
A+M	288.4	1.1
C+M	292.4	1.0
AwP+M	308.5	2.7×10^{-1}
CwP+M	326.5	1.9×10^{-2}

The difference between A+M and C+M is the number of communication parameters n_i^{com}. A+M and C+M traffic flow increases with the increase in penetration rate of the self-driving vehicle. Owing to the characteristics of GNS, the self-driving vehicle equipped with CACC has more opportunity to observe the leading vehicle as the penetration rate of the self-driving vehicle increases. If the self-driving vehicle observes the leading vehicle, the self-driving vehicle cuts needless deceleration.

However, the traffic flow difference between AwP+M and CwP+M is larger than the traffic flow difference between A+M and C+M. We so consider that the states s affects relieving traffic jam. In case of AwP+M, the features $\phi^{\mathrm{c-d}}$, $\phi^{\mathrm{c-v}}$, $\phi^{\mathrm{c-g}}$ become "disconnected" constantly. In contrast, the features of CwP+M become a communication partner's information. Hence, observing the communication partner's information significantly increases the effectiveness of the driving policy for the purpose of relieving traffic jams.

7 Conclusion

We proposed a driving policy for self-driving vehicles to help relieve traffic jams. A learner agent learned the driving policy, which was done via reinforcement learning with the data collected from the self-driving vehicle, which, in turn, were used to update the driving policy. This approach to developing a driving policy reduced the amount of time and labor that go toward designing driving policies for various traffic situations or changes in traffic situations. Our traffic flow simulation experiments under periodic-boundary conditions confirm that the use of the driving policy helps relieve traffic jams. Increased penetration rate of self-driving vehicles further reduces traffic jams and enhances traffic flow.

There are two issues that we intend to address in future studies: first, we intend to design a reward function and state feature to increase traffic flow with 100% penetration rate of the self-driving vehicle. Second, we plan to evaluate traffic flow using a road under an open-boundary condition which enables inflow, thereby changing vehicle density.

A Generalized Nagel–Schreckenberg Model

We used a modified GNS model [Ishikawa and Arai, 2015] for modeling traffic flow. In the unmodified version of the model, the number of communication parameters n_i^{com} is common for all vehicles. However, to more accurately model traffic flow where manually driven and self-driving vehicles are present, the GNS model was modified to be able to set the number of communication parameters n_i^{com} and the maximum communication distance d_i^{com} for individual vehicles.

A.1 GNS for vehicle i

At time t, all vehicles determine the next velocity simultaneously using Algorithm A·1. We explain Algorithm A·1 below.

Determine velocity: Vehicle i calculates the vehicle $i^{\mathrm{head}} \leftarrow i + n_i^{\mathrm{com}}$, which is the leading vehicle with respect to maximum communication and maximum communication distance $x^{\max} \leftarrow x_i(t) + d_i^{\mathrm{com}}$. Following Algorithm A·2 $MaxV$, vehicle i determines the velocity for the next time increment: $v_i(t+1)$.

Decelerate: In case of the manually driven vehicle, in which d_i^{com} is 0, the velocity of vehicle i becomes $v_i(t+1) \leftarrow \max(0, v_i(t+1) - 1)$ with perturbation probability p within the perturbation section of the road. For the self-driving vehicle, the velocity of vehicle i becomes $v_i(t+1) \leftarrow \max(0, v_i(t+1) - 1)$ with driving policy probability p^{pol}.

Move: Vehicle i determines the next time coordinate $x_i(t+1) \leftarrow x_i(t) + v_i(t+1)$.

A.2 MaxV

We explain the $MaxV$ that is showed at Algorithm A·2.

Accelerate: Vehicle i sets its own velocity $v_i(t+1) \leftarrow \min(v_i(t)+1, v^{\mathrm{limit}})$. If vehicle i has an adequate gap for velocity $v_i(t+a)$ after acceleration, vehicle i completes $MaxV$.

Adjust the number of communications: Vehicle i modifies vehicle i^{head} in accordance with front vehicle $i+1$'s number of communication parameters n_{i+1}^{com}. If vehicle $i+1$ has $n_{i+1}^{\mathrm{com}} > 0$ and satisfies $i^{\mathrm{head}} - (i+1) > n_{i+1}^{\mathrm{com}}$, then i^{head} becomes $i^{\mathrm{head}} - n_{i+1}^{\mathrm{com}}$. If vehicle $i+1$ has $n_{i+1}^{\mathrm{com}} == 0$, which has no communication ability, then i^{head} becomes i.

Communicate: If front vehicle $i+1$ exists behind i^{head} and within $x^{\max}$, then vehicle i calculates the predicted front vehicle's velocity v_{i+1}^{pred} by applying $MaxV$. This is in case of communication with front vehicle $i+1$.

Maximize velocity: In case of no communication, vehicle i determines the predicted front vehicle's velocity $v_{i+1}^{\mathrm{pred}} \leftarrow \max(0, \min(v_{i+1}(t), v^{\mathrm{limit}} - 1, g_{i+1} - 1))$, even if the perturbation probability $p = 1$ is taken into account.

Algorithm A·1 GNS for vehicle i

Determine velocity

$i^{\mathrm{head}} \leftarrow i + n_i^{\mathrm{com}}$

$x^{\mathrm{head}} \leftarrow x_i(t) + d_i^{\mathrm{com}}$

$v_i(t+1) \leftarrow MaxV(i, i^{\mathrm{head}}, x^{\mathrm{head}})$

Decelerate

$v_i(t+1) \leftarrow \max(0, v_i(t+1) - 1)$ probability p or p^{pol}

Move

$x_i(t+1) \leftarrow x_i(t) + v_i(t+1)$

Algorithm A·2 $MaxV(i, i^{\mathrm{head}}, x^{\mathrm{head}})$

Accelerate

$v_i(t+1) \leftarrow \min(v_i(t)+1, v^{\mathrm{limit}})$

if $v_i(t+1) \leq g_i$

  return $v_i(t+1)$

end if

Adjust the number of communications

if $n_{i+1}^{\mathrm{com}} > 0$ and $i^{\mathrm{head}} - (i+1) > n_{i+1}^{\mathrm{com}}$

  $i^{\mathrm{head}} \leftarrow i + 1 + n_{i+1}^{\mathrm{com}}$

else if $n_{i+1}^{\mathrm{com}} == 0$

  $i^{\mathrm{head}} \leftarrow i$

end if

Communicate

if $i+1 \leq i^{\mathrm{head}}$ and $x_{i+1} \leq x^{\mathrm{head}}$

  $v_{i+1}^{\mathrm{pred}} \leftarrow \max(0, MaxV(i+1, i^{\mathrm{head}}, x^{\mathrm{head}}) - 1)$

Maximize velocity

else

  $v_{i+1}^{\mathrm{pred}} \leftarrow \max(0, \min(v_{i+1}(t), v^{\mathrm{limit}} - 1, g_{i+1} - 1))$

end if

return $\min(v_i(t+1), v_{i+1}^{\mathrm{pred}} + g_i)$

Finally, $MaxV$ returns $\min(v_i(t+1), v_{i+1}^{\mathrm{pred}} + g_i)$ as the maximum velocity.

References

[Forster *et al.*, 2012] Markus Forster, Raphaël Frank, Mario Gerla, and Thomas Engel. Improving highway traffic through partial velocity synchronization. In *Global Communications Conference (GLOBECOM), 2012 IEEE*, pages 5573–5578. IEEE, 2012.

[Forster *et al.*, 2014] Markus Forster, Raphael Frank, Mario Gerla, and Thomas Engel. A cooperative advanced driver assistance system to mitigate vehicular traffic shock waves. In *INFOCOM, 2014 Proceedings IEEE*, pages 1968–1976. IEEE, 2014.

[Ishikawa and Arai, 2015] Shota Ishikawa and Sachiyo Arai. Evaluating advantage of sharing information among vehicles toward avoiding phantom traffic jam. In *Winter Simulation Conference (WSC), 2015*, pages 300–311. IEEE, 2015.

[Kamal *et al.*, 2014] Md Abdus Samad Kamal, Jun-ichi Imura, Tomohisa Hayakawa, Akira Ohata, and Kazuyuki Aihara. Smart driving of a vehicle using model predictive control for improving traffic flow. *IEEE Transactions on Intelligent Transportation Systems*, 15(2):878–888, 2014.

[Kesting *et al.*, 2008] Arne Kesting, Martin Treiber, Martin Schönhof, and Dirk Helbing. Adaptive cruise control design for active congestion avoidance. *Transportation Research Part C: Emerging Technologies*, 16(6):668–683, 2008.

[Khamis and Gomaa, 2014] Mohamed A Khamis and Walid Gomaa. 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, 2014.

[Knorr *et al.*, 2012] Florian Knorr, Daniel Baselt, Michael Schreckenberg, and Martin Mauve. Reducing traffic jams via vanets. *IEEE Transactions on Vehicular Technology*, 61(8):3490–3498, 2012.

[Nagel and Schreckenberg, 1992] Kai Nagel and Michael Schreckenberg. A cellular automaton model for freeway traffic. *Journal de physique I*, 2(12):2221–2229, 1992.

[Papacharalampous *et al.*, 2015] Alexandros E Papacharalampous, Meng Wang, Victor L Knoop, Bernat Goñi Ros, Toshimichi Takahashi, Ichiro Sakata, Bart van Arem, and Serge P Hoogendoorn. Mitigating congestion at sags with adaptive cruise control systems. In *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, pages 2451–2457. IEEE, 2015.

[Rezaee *et al.*, 2012] Kasra Rezaee, Baher Abdulhai, and Hossam Abdelgawad. Application of reinforcement learning with continuous state space to ramp metering in real-world conditions. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 1590–1595. IEEE, 2012.

[Sugiyama *et al.*, 2008] Yuki Sugiyama, Minoru Fukui, Macoto Kikuchi, Katsuya Hasebe, Akihiro Nakayama, Katsuhiro Nishinari, Shin-ichi Tadaki, and Satoshi Yukawa. Traffic jams without bottlenecks–experimental evidence for the physical mechanism of the formation of a jam. *New journal of physics*, 10(3):033001, 2008.

[Sutton and Barto, 1998] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

[Walraven *et al.*, 2016] Erwin Walraven, Matthijs TJ Spaan, and Bram Bakker. Traffic flow optimization: A reinforcement learning approach. *Engineering Applications of Artificial Intelligence*, 52:203–212, 2016.

[Won *et al.*, 2017] Myounggyu Won, Taejoon Park, and Sang H Son. Toward mitigating phantom jam using vehicle-to-vehicle communication. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1313–1324, 2017.

[Zolfpour-Arokhlo *et al.*, 2014] Mortaza Zolfpour-Arokhlo, Ali Selamat, Siti Zaiton Mohd Hashim, and Hossein Afkhami. Modeling of route planning system based on q value-based dynamic programming with multi-agent reinforcement learning algorithms. *Engineering Applications of Artificial Intelligence*, 29:163–177, 2014.