深層強化学習による車両移動経路と信号機の同時最適化

大橋　耕也†　　幸島　匡宏††　　堤田　恭太††　　松林　達史††　　戸田　浩之††

† 東京工業大学 情報理工学院 数理・計算科学系　〒 226–8502 神奈川県横浜市緑区長津田町 4259 G5-19
†† 日本電信電話株式会社 NTT サービスエボリューション研究所　〒 239–0847 神奈川県横須賀市光の丘 1-1
E-mail: †ohashi.k.aj@m.titech.ac.jp,
††{kohjima.masahiro,tsutsumida.kyota,matsubayashi.tatsushi,toda.hiroyuki}@lab.ntt.co.jp

あらまし　都市の交通渋滞は国内外の多くの都市で社会問題となっている．しかしながら, 道路の拡幅のような高コストな解決策の採用は難しいため, 混雑の状況や予報を利用者に情報版等を用いて通知して迂回させたり, 信号機の制御を高度に行なうことで交通渋滞を緩和する ITS (Intelligent Transport Systems) を積極的に活用する取り組みが広く行われている．そこで本研究では, 近年その有効性が示されている深層強化学習を用いて, 車両の移動経路と信号機を同時に制御する技術を提案する．交通シミュレーションを用いた実験を通して得た定量的・定性的結果について報告する．

キーワード　交通渋滞, 交通制御, 深層強化学習

1. はじめに

近年, 都市の交通渋滞が深刻化しており, 国土交通省による平成 24 年度プローブデータを用いた試算では, 乗車時間の約 4 割は渋滞に巻き込まれていると報告されている (注1)．交通渋滞は, 人々の生活を不便にするだけではなく, 流通の遅れによる労働力の損失や観光渋滞による経済効果の減少を引き起こし膨大な経済損失をもたらす．したがって, 交通渋滞は解決されるべき重要な社会問題のひとつである．

交通渋滞の緩和には様々なアプローチが存在するが, 道路の拡幅工事のような高コストな解決策の採用は難しいため, 混雑の状況や予報を利用者に情報版等を用いて通知して迂回させたり, 信号機などの交通システムの制御を最適化することで交通渋滞を緩和する試みが実施され, 成果を挙げている [10]．機械学習の様々なアプローチを利用した検討も進められる中 [12], 深層強化学習を用いて, 道路状況に応じて信号機の進行方向を適応的に変更させる, 適応信号制御の有効性がシミュレーション上で確認されている [2], [11]．しかし, 信号機のみでは制御の自由度が小さく, ボトルネックの交通容量を超える車両の集中や, 異なる方向から同時に車両が到着した場合への対応が困難である．将来, 自動運転車が広く普及することで, 信号機などの交通を制御するインフラと車両がより協調的に動作することで混雑が緩和することが期待される．

そこで本研究では, 信号機だけではなく車両, 特に車両の移動経路を同時に制御する手法を提案する．信号機と車両を同時に制御することで, ある特定の道路だけに車両が集中することを避けたり, 信号により停止することなくスムーズに交差点を通過できる経路を車両に案内することで, より効果的な渋滞緩和が可能になると考えられる．

提案手法は深層強化学習を用いて, 信号機と車両移動経路を同時に制御する学習者 (Agent) の行動ルール (方策) を学習する．ナイーブな学習者の定義では, この学習者の行動空間は信号機の数と車両の数に対して指数的に増大し, 方策の学習は困難である．そこで我々は, ある区間内に存在する車両に対して同一の経路指示を行う, 経路指示機と呼ぶ仮想機械を導入し, 車両数に依存しない行動空間の定義を可能とした．これにより, 本問題を現実的なサイズの問題として定式化することが可能になる．

提案手法の有効性の検証のため, 交通シミュレータ SUMO (Simulation of Urban MObility) [3] を用いた実験を実施した．本実験設定のもとで, 提案手法は信号機のみを制御する従来技術と比較して, 平均待ち時間を削減する効果があることを示す結果を得た．本結果は, 信号機と車両の移動経路を同時に制御することの有効性を示唆する結果といえ, 今後の交通システムの技術発展の方向性の土台になりうるものである．

本論文は次のような構成から成る．§ 2 で深層強化学習の概要を述べ, § 3 では, 提案手法の定式化を行う．§ 4 で, 実験による提案手法の有効性を定量的・定性的に述べ, § 5 でまとめる．

2. 準　　備

本節では, 強化学習の概要と本稿で利用する深層強化学習について説明する．

2.1 マルコフ決定過程 (MDP)

強化学習とは, 学習者 (Agent) が環境 (Environment) との相互作用を通して, 最適な行動ルール (方策) を推定する手法のことを指す．強化学習では, 環境の設定として, マルコフ決定過程 (Markov Decision Process, MDP) が多くの場合利用され, 本稿でもこれを利用する．

マルコフ決定過程は 4 つ組 (S, A, P_M, R) により定義される．S を状態空間, A を行動空間と呼び, それぞれの元 $s \in S$ を状態, $a \in A$ を行動と呼ぶ．$P_M : S \times A \times S \to [0, 1]$ は状態遷移関数と呼ばれ, 状態 s で行動 a を行ったときの次状態 s' への遷移確率を定める．$R : S \times A \times S \to \mathbb{R}$ は報酬関数である．

(注1)：http://www.mlit.go.jp/common/001067075.pdf

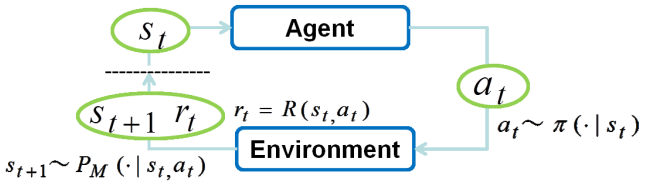


図 1: 学習者 (Agent) と環境 (Environment) の相互作用

報酬関数が状態 s で行動 a を行ったときに得られる報酬を定義している．学習者は，上記の環境の中で将来にわたって得られる報酬の和ができるだけ多くなるように行動を行う．学習者の各状態 s で行う行動 a を選択する確率を定めたものを方策 $\pi : S \times A \to [0, 1]$ と呼ぶ．

2.2 価値関数

方策を 1 つ定めると，学習者は図 1 に示すように環境との相互作用を行うことが可能となる．各時刻 t で，状態 s_t にいる学習者は方策 $\pi(\cdot|s_t)$ に従って行動 a_t を決定する．すると，状態遷移関数と報酬関数に従い，学習者の次時刻の状態 $s_{t+1} \sim P_M(\cdot|s_t, a_t)$ と報酬 $r_t = R(s_t, a_t)$ が決定する．これを繰り返すことで，学習者の状態と行動の履歴が得られる．以後，時刻 0 から T 回遷移を繰り返した状態と行動の履歴 $(s_0, a_0, s_1, a_1, \ldots, s_T)$ を d_T と表記し，これをエピソードと呼ぶ．

ここで価値関数と呼ばれる，方策の良さを表す役割を持つ関数を定義する．価値関数は，状態 s において行動 a を選択し，後は方策 π に従って行動し続けた時の (割引) 報酬和の平均として定義され，以下の式で表される．

$$Q^\pi(s, a) \equiv \lim_{T\to\infty} \mathbb{E}^\pi_{d_T}\left[\sum_{k=0}^{T} \gamma^k R(s_k, a_k, s_{k+1}) \middle| s_0 = s, a_0 = a\right]$$

ただし，$\gamma \in [0, 1)$ は割引率，$\mathbb{E}^\pi_{d_T}[]$ は方策 π でのエピソードの出方に関する平均操作を表す．ある方策 π, π' が任意の $s \in S, a \in A$ で $Q^\pi(s, a) \geqq Q^{\pi'}(s, a)$ を満たすとき，方策 π は π' よりも多くの報酬を学習者にもたらすと期待できるため，これを $\pi \geqq \pi'$ と書くとする．強化学習の目的は，任意の方策 π について，$\pi^* \geqq \pi$ を満たす最適方策 π^* を得ることである．

最適方策はその価値関数 Q^*(最適価値関数と呼ばれる) を用いて，$\pi^*(a|s) = \delta\left(a - \arg\max_{a'} Q^*(s, a')\right)$ と設定することで得られる．最適価値関数は最適ベルマン方程式

$$Q^*(s, a) = \mathbb{E}_{s'}\left[R(s, a, s') + \gamma \max_{a'} Q^*(s', a')\right] \qquad (1)$$

を満たすことが知られ，上記の関係式を用いて推定が行われる．代表的な手法は Q 学習 [9] と呼ばれる手法であり，状態空間が離散かつ状態数が膨大でなければ良好に動作することが多くの実験で報告されている．しかしながら，本稿の問題のように，状態空間が連続かつ状態数が膨大である問題に適用することは困難である．

2.3 価値関数近似

上記のような問題に対し，価値関数をパラメタを持つ関数で近似するというアプローチが検討され，(パラメタに関して) 線形な関数で近似する方法 [1] [4] やニューラルネットワーク [6] で近似する手法がこれまで提案された．上記に代表される価値関数近似手法の基本的なアイディアは，(最適) 価値関数をパラメタ w を持つ関数 Q_w で近似し，目的関数

$$L(w) = \sum_{(s,a,s')\in D} \left(R(s, a, s') + \gamma \max_{a'} Q_w(s', a') - Q_w(s, a)\right)^2$$

を最小化することで，パラメタ w を学習し，最適方策 π^* を求める，というものである．ただし，$\mathcal{D}$ は学習者と環境の相互作用の履歴であり，遷移前状態 s，行動 a，遷移後状態 s' の組 (s, a, s') の集合として定義される．上記の目的関数はベルマン最適方程式 (1) の右辺と左辺の差を最小化することに相当する．近年では，上記のニューラルネットワークを用いる方法をベースに種々のヒューリスティクスなどを導入した Deep Q-Network (DQN) [5] と呼ばれる手法が，ゲームのプレイ画像そのものを入力とする高次元状態空間であっても，学習が可能であることを報告し注目を集めている．本稿の問題も多くのセンサからの情報を入力空間とする高次元状態空間を扱う問題であり，学習法には DQN のアプローチ，特に既存の DQN を上回る性能が報告されている Double DQN [8] と呼ばれる最新のアプローチを採用した．

3. 提案手法

この章で，本研究で提案する，車両移動経路と信号機の同時制御手法について説明する．提案手法は強化学習のアプローチを利用し，行動空間の定義が本質的なアイディアとなる．

3.1 行動空間の定義と課題

制御対象は信号機と車両移動経路であり，信号機に関する行動空間全体 $\mathbb{A}_{\text{sig}}$ と車両移動経路に関する空間全体が $\mathbb{A}_{\text{veh}}$ と定義されているとする．各行動空間は，例えば $\mathbb{A}_{\text{sig}} = \{$ 南北青, 東西青 $\}$ [(注2)] $\mathbb{A}_{\text{veh}} = \{\text{routeA}, \text{routeB}, \text{routeC}\}$ などが挙げられる．南北青 $\in \mathbb{A}_{\text{sig}}$ が交差点の信号の南北 (上下) 方向を青に，東西 (左右) 方向を赤にする，という行動を表す．また，routeA $\in \mathbb{A}_{\text{veh}}$ が車両をルート A の経路で移動させることを表す．

上記の設定として素朴に信号機の行動空間 $\mathbb{A}_{\text{sig}}$ と車両毎に存在する車両の行動空間 $\mathbb{A}_{\text{veh}}$ の積として行動空間を定義すると，

$$A = \mathbb{A}_{\text{sig}} \times \underbrace{\mathbb{A}_{\text{veh}} \times \mathbb{A}_{\text{veh}} \times \mathbb{A}_{\text{veh}} \times \cdots}_{\text{車両数}}, \qquad (2)$$

と行動空間のサイズが車両数に応じて指数的に大きくなり，現実的に解けるサイズの問題として定式化することができないという課題がある．

3.2 経路指示機の導入

そこで我々は，下記で定義する経路指示機という仮想機械を導入することで，この課題を解決する．

定義: 定められた区間内に存在する全制御対象車両に対して，同一の経路指示を行う機械を経路指示機と呼ぶ．

(注2)：信号機は黄色状態を行動空間に持つことも可能であるが，ここでは黄色状態は別の状態へ遷移する際に一定時間実行される陰な状態として表現することとし，黄色信号にするという行動が存在する必要のない設定を考える．

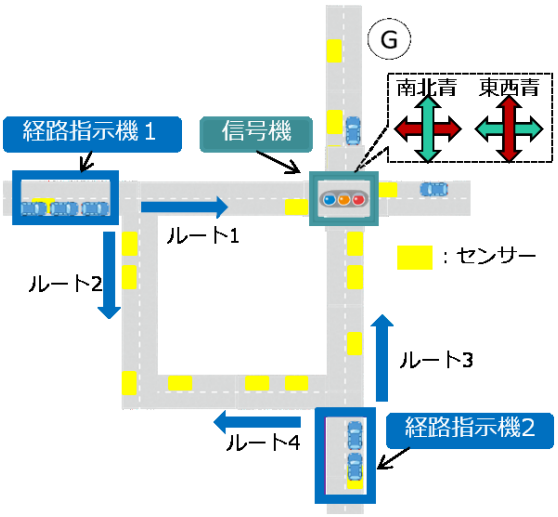
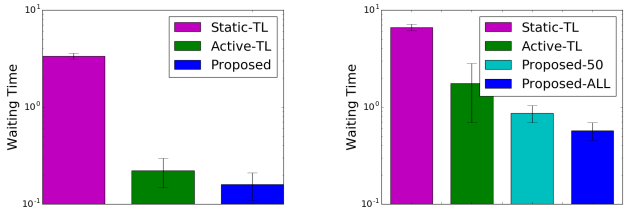


図 2: 制御対象エリアの例

表 1: 問題サイズとニューラルネットワークの設定

	片側 1 車線	片側 2 車線
状態空間の次元数	488	852
行動空間の次元数	8	64
ニューラルネットの素子数	488-300-300-8	852-1000-1000-64



(a) 片側 1 車線　　(b) 片側 2 車線

図 3: 実験結果. 待ち時間が短いほど良い.

なお, ここで “仮想”機械と呼んでいるのは, 通常の信号機と異なり, 対応する物理的実体を必ずしも必要としないためである. たとえば, 信号機のような運転手が視認することができる装置がなくても, 遠隔地に設定された装置が, 定めた区間内に侵入した車両から区間内にいるという情報を受信した際に, その車両に対して進むべき経路を指示する, という手続きを行うことで, 上記で定義した経路指示機の役割を果たすことができる.

上記の議論に基づき, 提案手法の行動空間を信号機の行動空間 $\mathbb{A}_{\text{sig}}$ と経路指示機毎に存在する車両の行動空間 $\mathbb{A}_{\text{veh}}$ の積として定義する. 図 2 で定義される制御対象エリアの場合では, 経路指示機 1 の行動空間は $\mathbb{A}^{(1)}_{\text{veh}} = \{\text{ルート 1}, \text{ルート 2}\}$, 経路指示機 2 の行動空間は $\mathbb{A}^{(2)}_{\text{veh}} = \{\text{ルート 3}, \text{ルート 4}\}$ となる. 経路指示機は 2 つであることから, その行動空間は以下で与えられることとなり,

$$A = \mathbb{A}_{\text{sig}} \times \underbrace{\mathbb{A}^{(1)}_{\text{veh}} \times \mathbb{A}^{(2)}_{\text{veh}}}_{\text{経路指示機の数}}, \qquad (3)$$

全行動数は $2 \times 2 \times 2 = 8$ となる. これにより, 式 (2) の定義では不可能であった, 信号機と移動経路を同時に最適化する問題を現実的に解けるサイズの問題として定式化できた.

3.3 行動可能タイミングの制限

提案技術を現実の交通環境で利用することを想定し, 学習者の行動可能タイミングには制約を設けた. 各経路指示機は毎秒行動の選択が可能であるが, 信号機は安全性及び黄色信号を考慮しなければならない. そこで, 信号機は別の状態へ遷移する際に黄色信号を 5 秒間継続し, またある状態は少なくとも 2 秒間は継続するという制約を設けることとした.(注3) これにより, 現実に則した設定で提案技術を適用することができる.

3.4 状態空間, 報酬関数の例

本手法を適用するうえで, 状態空間, 報酬関数は任意のものが利用できる. ここでは一例として, 後の実験で利用する状態空間と報酬関数の定義について述べる.

状態空間: 状態 s は, 環境内に設置したセンサーが取得する情報と信号機の状態により定義する. 具体的に図 2 の場合で説明する. 対象環境内に設置した各センサーは毎秒 (i) 車両の有無の情報 $\mathbb{B} = \{0, 1\}$ と (ii) 車両の速度: $\mathbb{R}^+ = \{x \mid x \geq 0, x \in \mathbb{R}\}$ の 2 つの情報を取得する. また, 対象環境内の信号機の状態 $\mathbb{S}_{\text{sig}}$ も観測する. 図 2 の場合は $\mathbb{S}_{\text{sig}} = \{$ 南北青, 東西青, 南北黄色, 東西黄色 $\}$ の 4 つの状態を取りうる. 以上で定義した量により, 状態空間 S を $S \equiv (\mathbb{B} \times \mathbb{R})^{\#\text{sensor}} \times \mathbb{S}_{\text{sig}}$ により定義する.

報酬関数: 後の実験では, 報酬関数として対象環境内の車両の平均待ち時間の符号反転 $R(s_t, a, s_{t+1}) \equiv -\frac{1}{|V|}\sum_{v \in V} \text{WaitingTime}(s_{t+1}, v)$ を報酬として定義する. ただし, V は対象環境内の車両の集合を, $\text{WaitingTime}(s_{t+1}, v)$ は時刻 $t+1$ 時点で車両 v が速度 0.1m/s 以下であった継続時間をそれぞれ表す.

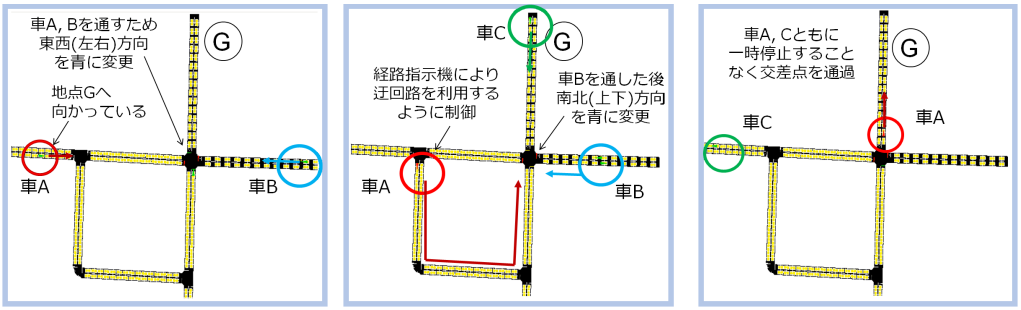
4. 実　　験

4.1 実験設定

本実験では, 交通シミュレータである SUMO [3] を利用する. SUMO は道路ネットワークや信号機, 車両の流量や最高速度等を自由に定義し, シミュレーションが実行可能なソフトウェアである. 道路ネットワークには図 2 で示した交差点 1 つから成る信号機 1 つと経路指示機 2 つが存在するネットワークを利用し, 学習者は, 信号機 1 つと経路指示機 2 つを同時に制御する. なお, 制御対象車両は, 北方向の目的地 (G) を目指す車両に限定した. 状態空間を定義するセンサーは 5m 間隔で設置し, 南北方向と東西方向の交通量は同程度とした. そのうち制御対象車両は全体の 28% である. 実験は片側 1 車線と 2 車線の 2 つの場合で行った. 各設定における, 状態空間, 行動空間のサイズは表 1 の通りである. ニューラルネットワークの活性化関数には ReLU, 各素子数は表 1 の設定を用い, chainerRL (注4) の Dou-

(注3)：これは厳密には semi-markov 決定過程と呼ばれる複数時間ステップにまたがる行動が定義されたマルコフ決定過程を考えていることに相当する [7].

(注4)：https://github.com/chainer/chainerrl



(a) (b) (c)

図 4: (a) 東西方向に交差点を通過したい車 A, 車 B が出現．提案手法は東西方向を青へ変更するよう制御．(b) さらに南北方向に通過したい車 C の登場．車 B はそのまま交差点を通過するが, 車 A は経路を変更するよう制御．車 B の通過後, 南北を青に切り替える．(c) その結果, 車 A, 車 C は一時停止することなくに交差点を通過．

bleDQN で minibatch_size = 100, replay_start_size = 500, update_interval = 1, target_update_interval = 500 と設定し学習を行った．

提案手法 (Proposed/Proposed-ALL) の比較手法には, 静的な信号 (Static-TL) と, 信号機のみの制御を行う手法 (Active-TL) を利用した．静的な信号とは, 対象環境の状況とは無関係にあらかじめ決められたサイクルを繰り返す制御方法であり, 今回は SUMO のデフォルト値のまま, 各状態は 31 秒とした．信号機のみの制御とは, 深層強化学習により探索された, 信号機の制御方法であり, 提案手法のニューラルネットワークの構造と同一のものを用いた．また, 片側 2 車線の実験では, 車両が提案手法による経路指示に必ずしも従わず, 確率 50%で従う設定 (Proposed-50) での検証もおこなった．

4.2 実験結果

定量評価: 図 3 に実験結果を示す．図 3a3b に示すように, 片側 1 車線と片側 2 車線の両方の設定で提案手法による制御は信号機のみの制御に比べて平均待ち時間が短いことが分かる．この結果により, 信号機と車両の移動経路を同時に制御することで既存技術よりも優れた交通制御を実現できることが示された．また, 片側 2 車線の場合には 1 車線の場合よりも提案手法と既存手法の差が大きく, 車両が必ず提案手法の経路指示に従う Proposed-ALL, 50%で従う Proposed-50, Active-TL の順に優れた結果となっている．これは, 片側 2 車線の方が片側 1 車線の場合よりも経路指示に関する行動空間が広く制御の自由度が増したことと, 指示に従う車両が多いほど提案手法の効果が大きいためであると考えられる．

定性評価: 図 4 に提案手法によって学習された制御の例を示す．この図からわかるように, 提案技術による制御では, 車 A と車 C のように異なる方向から同タイミングに車両が信号に到着することが予期される場合に, 車両の経路を変更することによって一時停止することなく車両を通過させている．このような制御は信号のみの制御では不可能である．したがって, 提案技術は信号と経路を組み合わせた制御を行うことによって, 信号機のみ制御する従来法よりも待ち時間を減少させていると考えられる．

5. まとめ

本研究では, 信号機と車両の経路選択の同時制御システムを提案し, 交通シミュレータを用いた実験によってその有効性を検証した．自動運転車の普及が予想される現在, 未来の交通システムの議論にも本稿の結果は有用であると考えられる．今後の展望として, 複数の交差点間で協力しより効率的な交通制御システムを構築することが挙げられる．

文献

[1] Justin A Boyan. Technical update: Least-squares temporal difference learning. *Machine Learning*, Vol. 49, No. 2, pp. 233–246, 2002.

[2] Wade Genders and Saiedeh Razavi. Using a deep reinforcement learning agent for traffic signal control. *CoRR*, Vol. abs/1611.01142, , 2016.

[3] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. Recent development and applications of sumo-simulation of urban mobility. *Int. J. On Adv. in Sys. and Measurements*, Vol. 5, No. 3&4, pp. 128–138, 2012.

[4] Michail G Lagoudakis and Ronald Parr. Least-squares policy iteration. *JMLR*, Vol. 4, No. Dec, pp. 1107–1149, 2003.

[5] Volodymyr Mnih, et al. Human-level control through deep reinforcement learning. *Nature*, Vol. 518, No. 7540, pp. 529–533, 2015.

[6] Martin Riedmiller. Neural fitted q iteration-first experiences with a data efficient neural reinforcement learning method. In *ECML*, Vol. 3720, pp. 317–328. Springer, 2005.

[7] Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, Vol. 112, No. 1-2, pp. 181–211, 1999.

[8] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *AAAI*, pp. 2094–2100, 2016.

[9] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, Vol. 8, No. 3-4, pp. 279–292, 1992.

[10] 株式会社 NTT データ. 中国・貴陽市において、ビッグデータを活用した「渋滞予測・信号制御シミュレーション」の実証実験で渋滞緩和効果を確認. `http://www.nttdata.com/jp/ja/news/release/2016/053101.html`, 2016.

[11] 佐藤季久恵, 高屋英知, 小川亮, 芦原佑太, 栗原聡. Deep q-network を用いた交通信号制御システムの提案. In *JSAI*, 2017.

[12] 伊藤秀剛, 堤田恭太, 松林達史, 戸田浩之. ベイズ最適化を用いた交通流信号制御の最適化. In *IBIS*, 2017.