強化学習を用いた
データ駆動型の動的混雑課金の最適化手法

佐藤　公洋[1]・瀬尾　亨[2]・布施　孝志[3]

[1]非会員　東京大学大学院　工学系研究科社会基盤学専攻 (〒113-8656 東京都文京区本郷7-3-1)
E-mail:sato@trip.t.u-tokyo.ac.jp

[2]正会員　東京大学大学院助教　工学系研究科社会基盤学専攻 (〒113-8656 東京都文京区本郷7-3-1)
E-mail:seo@civil.t.u-tokyo.ac.jp

[3]正会員　東京大学大学院教授　工学系研究科社会基盤学専攻 (〒113-8656 東京都文京区本郷7-3-1)
E-mail:fuse@civil.t.u-tokyo.ac.jp

交通渋滞を緩和する施策の1つとして，1日の中での交通需要の変動を考慮する動的混雑課金の有用性が提唱されている．また，課金主体と道路利用者の間にある情報の非対称性に対応するため，Trial-and-error型の課金額決定手法が提案されている．本研究では，強化学習を用い，様々な環境に対応可能かつ課金額更新を速やかに行えるTrial-and-error型の動的混雑課金手法を構築する．具体的には，単一ボトルネックモデルおよび複数ボトルネックモデルにおける出発時刻選択問題の課金額をQ学習によって最適化する手法を複数構築した．そして，交通モデル上でのシミュレーションにより，既存研究との比較・環境変化への対応可能性の検証を行った．その結果，時間帯別にほぼ独立に課金額を調整する分散制御型手法が有効であるという知見を得た．

Key Words : *dynamic congestion toll, trial-and-error, reinforcement learning, day-to-day dynamics*

1. はじめに

自動車の交通渋滞問題解決のためのソフト面での対策の1つとして混雑課金が注目されている．特に，1日の中での交通需要の変動を考慮する動的混雑課金の有用性が提唱されている．しかしながら，混雑課金において利用者の出発時刻選択のための経済計算 (例：個人の時間価値の値) を課金主体の側が具体的に把握することは困難であり，情報の非対称性が存在すると言える．このため，最適課金額の即座の決定は難しい．

情報の非対称性に対処するために，観測可能な交通データを用いたTrial-and-error手法が提案されている (Li[1], Ye et al.[2])．本手法は，何らかの混雑課金を課した際の交通状態を観測し，観測データに基づく課金額の調整を試行錯誤的に繰り返し行い，最適課金を発見するものである．Seo and Yin[3], Seo[4]は，Trial-and-error型の動的混雑課金手法を提案している．しかし，これらの手法は，所与の簡潔なルールにより課金額を調整するものであり，課金額調整速度が優れているとはいえない．更に，時間価値，早着コスト，遅着コスト等のパラメータの変化を考慮していない点，単一ボトルネックモデルにおける出発時刻選択モデルの場合のみの検討に留まっている点等で課題が残されている．

一方で，近年，機械学習の分野で強化学習が大きな成果を挙げている．強化学習とは，ある環境内において，エージェントが観測した状態に基づき行動を選択する中で報酬を獲得し，その報酬に基づき最適な行動を学習するものである．詳細が未知の環境においても学習可能である点から，動的混雑課金額決定への応用が期待される．

本研究では，強化学習を用い，異なる環境に対応可能，かつ課金額更新を速やかに行えるTrial-and-error型の動的混雑課金手法を構築する．具体的には，状態を交通データ，行動を課金額の変更とし，待ち時間を減少させる行動に対し良い報酬を与えて強化学習を行う．また，シミュレーションにより提案手法の性質を検証する．

対象とする交通モデルとしては以下の2つを考える．1つ目は，既往研究で用いられている単一ボトルネックモデルにおける出発時刻選択モデルである．これにより，同じ交通モデルを採用したSeo[4]との比較が可能となる．2つ目は，より複雑な複数ボトルネックを有するネットワークにおける出発時刻選択と経路選択のあるモデルである．これにより，モデルフリー性を生かした複雑な状

況への適用可能性の検討が可能となる．

本稿の構成は以下の通り．第2章にて動的混雑課金の先行研究を紹介する．第3章にて本研究で用いる強化学習の概要を紹介する．第4章にて交通モデルの設定を示す．第5章にて強化学習の実装法として中央制御型手法と分散制御型手法の2種類を提案する．第6章にてシミュレーション実験の結果と考察を示す．第7章にて結論と今後の課題をまとめる．

2. 動的混雑課金の先行研究

(1) 単一ボトルネックモデルの設定

本節では，Vickrey[5]により提案された単一ボトルネックモデルを概観する．

朝のラッシュアワーにおける単一ボトルネックモデルを考える．数学記法は以下の通りである．

j：日数
t：1日の中での時刻
μ：単位時間当たりのボトルネック容量
M：1日当たりの総旅行者数
t^*：旅行者の勤務地への希望到着時刻
$a_j(t)$：j日目，時刻tでの居住地からの出発率
$N_j(t)$：j日目，時刻tでの待ち行列台数
$w_j(t)$：j日目，時刻tでの待ち時間
$\tau_j(t)$：j日目，時刻tでの課金額

待ち行列は物理的な長さを持たないPoint Queueであるとし，t^*は全旅行者について等しいとする．また，tは勤務地への到着時刻に基づき定義されるとする．例えば，$a_j(t)$はj日目に勤務地に時刻tに到着する旅行者の出発率である．よって，課金額等は勤務地への到着時刻に依存する形で定義される．

j日目，時刻tにおける待ち行列台数の変化を式(1)のように表す．

$$\frac{\mathrm{d}N_j(t)}{\mathrm{d}t} = \begin{cases} 0 & \left(N_j(t) = 0 \text{ and } a_j(t) < \mu\right) \\ a_j(t) - \mu & (\text{otherwise}) \end{cases} \tag{1}$$

j日目，時刻tにおけるボトルネックでの待ち時間を式(2)のように表す．

$$w_j(t) = \frac{N_j(t)}{\mu} \tag{2}$$

j日目，時刻tにおける1人当たりの旅行者の一般化コストを式(3)のように表す．

$$c_j(t) = \tau_j(t) + \alpha w_j(t) + \begin{cases} \beta(t^* - t) & (t < t^*) \\ \gamma(t - t^*) & (\text{otherwise}) \end{cases} \tag{3}$$

ここで，αは 1 人の旅行者における単位時間当たりの時間価値，βは希望到着時刻と比べて早く勤務地に到着する場合の単位時間あたりのコスト(早着コスト)，γは希望到着時刻と比べて遅く勤務地に到着する場合の単位時間あたりのコスト(遅着コスト)を示す．また，tは勤務地への到着時刻を示す．

ここでは，ピーク時間帯における$c_j(t)$が一定である場合を利用者均衡状態であると考える．この時の$\alpha w_j(t)$の分のコストを混雑課金(j日目，時刻tでの課金額は$\tau_j(t)$で表される)に置き換えられれば，待ち時間が0になり，社会最適が達成されると言える．

(2) Trial-and-error型の動的混雑課金手法

本節では，Seo[4]により提案された動的混雑課金手法を概観する．交通モデルには単一ボトルネックモデルを採用する．交通モデル・課金額更新の概要は**図-1** の通りである．日毎の出発率の変化(day-to-day dynamics)には，replicator dynamics[6]を採用する．出発率は式(4)，(5)に基づいて変化すると仮定する．

$$a_{j+1}(t) = a_j(t) + \delta a_j(t)(\bar{c}_j - c_j(t)) \tag{4}$$

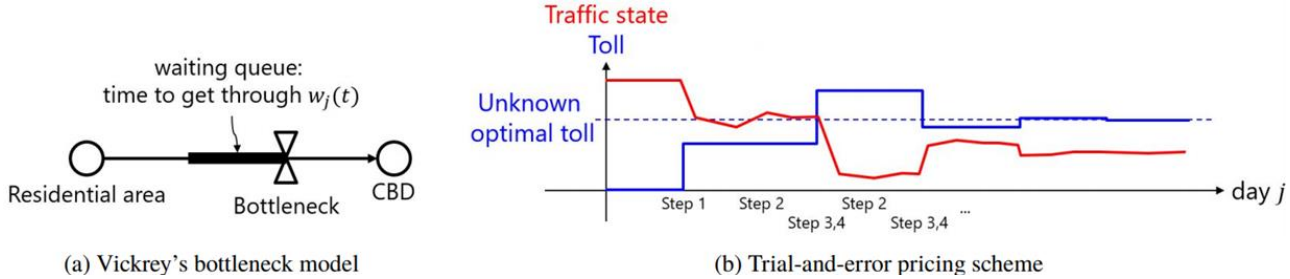
$$\bar{c}_j = \frac{\int a_j(T)\, c_j(T)\mathrm{d}T}{\int a_j(T)\mathrm{d}T} \tag{5}$$

ここで，δは day-to-day dynamics における日毎の変化速度，$\bar{c}_j$はj日目における平均一般化コストを表す．

j日目，時刻tでの課金額は式(6)のように表現する．

$$\tau_j(t) = \tau_j^c(t) + \tau_j^s(t) \tag{6}$$

ここで，$\tau_j^c(t)$はj日目，時刻tにおける課金額の大枠を決定する項，$\tau_j^s(t)$はj日目，時刻tにおける day-to-day dynamics の安定化項である．井料 [7]，Iryo et al.[8]の理論に基づき，$\tau_j^s(t)$を式(7)で定義する．



(a) Vickrey's bottleneck model　　(b) Trial-and-error pricing scheme

図-1 単一ボトルネックモデル・課金額更新の概要[4]

$$\tau_j^s(t) = \sigma_1 \left(\frac{a_{j-1}(t)}{\sigma_2 M} \right)^{\sigma_3} \quad (\sigma_1 > 0, \sigma_2 > 0, \sigma_3 \geq 0) \quad (7)$$

課金額決定のアルゴリズムの詳細は次の通りである．なお，T_pはピーク時間帯を表す.

[Step 0]

日数を$j := 0$とし，出発率の初期状態$a_0(t)$を設定する.

[Step 1]　初期化段階 (Initialization phase)

[Step 1.1]

日数を$j := j + 1$とする．安定化項$\tau_j^s(t)$による課金のみを実施する ($\tau_j^c(t) = 0$).

[Step 1.2]

交通流における$w_j(t)$および$a_j(t)$を観測する.

[Step 1.3]

収束確認を行う．eの設定値に対して，

$\left| w_j(t) - w_{j-1}(t) \right| / w_j(t) < e$なら Step 1.4 に進む．その他の場合，日数を$j := j + 1$とし，Step 1.1 に戻る.

[Step 1.4]

現在の状態を利用者均衡状態 (user equilibrium state) とみなす．日数を$j := 0$にリセットし，Step 2 に進む.

[Step 2]　Trial-and-error 段階 (Trial-and-error phase)

[Step 2.1]

日数を$j := j + 1$とし，$\tau_j^c(t) = \tau_{j-1}^c(t)$と決定する.

[Step 2.2]

課金額を$\tau_j(t) = \tau_j^c(t) + \tau_j^s(t)$として課金を実施する.

[Step 2.3]

交通流における$w_j(t)$および$a_j(t)$を観測する.

[Step 2.4]

収束確認を行う．eの設定値に対して，

$\left| w_j(t) - w_{j-1}(t) \right| / w_j(t) < e$なら Step 2.5 に進む.

そうでない場合，Step 2.1 に戻る.

[Step 2.5]　課金額の更新 (Toll update)

[Step 2.5.1]

もしピーク時間帯内においてボトルネックでの待ち行列が存在する場合 ($w_j(t) > \phi, \forall t \in T_p$)，課金額を$\tau_j^c(t) := \tau_j^c(t) + \hat{\alpha} w_j(t)$と更新し，Step 2.1 に戻る.

[Step 2.5.2]

もしピーク時間帯内のある時刻において交通流が無い場合 ($a_j(t) = 0, \exists t \in T_p$)，$\hat{\alpha}$を$\hat{\alpha} := \hat{\alpha}/2$，課金額を$\tau_j^c(t) := \hat{\alpha} w_0(t)$と再設定する．その後，Step 2.1 に戻る.

[Step 2.5.3]

その他の場合，現在の状態を最適に近い状態であるとみなし，アルゴリズムを終了する.

Seo[4]により提案された手法では，所与の簡潔なルールに基づくアルゴリズムとなっており，時間価値の異なる環境や，単一ボトルネックモデル以外への交通モデルへの対応性が不十分であるという限界がある.

3. 強化学習の概要

(1) Q学習

一般的な強化学習のアルゴリズムの1つとして，Q学習[9]がある．Q学習のアルゴリズムは以下の通りである.

1. 全ての状態，行動に対応するQ値の初期値を設定する.
2. ステップk，状態s_kにおいて行動b_kが選択され，ステップ$k+1$において状態がs_{k+1}に遷移する.
3. 状態s_{k+1}への遷移により，報酬r_{k+1}を獲得する.
4. 式 (8) に基づいて，Q値を更新する.

$$Q(s_k, b_k) \leftarrow Q(s_k, b_k) + \eta_k(s_k, b_k) \{ r_{k+1} + \xi \max Q(s_{k+1}, b_{k+1}) - Q(s_k, b_k) \} \quad (8)$$

ここで，$\eta_k(s_k, b_k)$はステップkでの学習率であり，Q値更新の緩急を決定するパラメータである．また，ξは割引率であり，将来的に得られる報酬の度合いを決定するパラメータである.

Q学習は，一定の条件下で最適状態に収束することが証明されており，簡潔かつ有用な強化学習アルゴリズムとして幅広く用いられている.

(2) ε -greedy法

Q学習における行動の決定方式の1つとして，ε -greedy法[10]がある．内容は次の通りである.

1. εの値を$0 < \varepsilon < 1$の範囲で一意に定める.
2. 一様分布$U[0,1]$に従う乱数uを生成する.
3. 乱数uの値がε以下の場合，ランダムに行動を選択する (探索する).
4. 乱数uの値がεより大きい場合，学習によって得た最善の行動を選択する (利用する).

(3) 行動価値関数の近似

一般的なQ学習では，状態と行動を共に離散値として扱う必要がある．一方で，状態として出発率を離散化して扱う場合，状態数の増大とそれに伴う計算負荷の増加が予想される．そこで，行動価値 (Q値) 関数における状態を連続値として扱うために，ガウスカーネル法による関数近似が提案されている．これは，ガウスカーネル (多変量正規分布) を基底関数として，その組み合わせにより関数近似を行うものである．m番目の基底関数 $(m = 1,2,\cdots,d\ ;\ d$は基底関数の数$)$は式 (9) のように設定される.

$$\varphi_m(\boldsymbol{s},\sigma)=\exp\left(-\frac{\|f(\boldsymbol{s})-\boldsymbol{p}_m\|^2}{2\sigma^2}\right) \tag{9}$$

ここで，σは多変量正規分布の分散を示す．$\boldsymbol{p}_m$は状態空間におけるm番目の基底関数の座標を定めるものであり，各要素は離散値で設定される．関数$f(\boldsymbol{s})$は，$\boldsymbol{p}_m$と$\boldsymbol{s}$の各要素の値を調整するための関数である．

強化学習は，基底関数の係数となるパラメータベクトルの更新により行う．l番目の行動ベクトル$\boldsymbol{b}_l$ $(l=1,2,\cdots,c\,;c$は行動の選択肢の数$)$ に対応するパラメータベクトルを式(10)のように設定する．

$$\boldsymbol{\theta}^{\boldsymbol{b}_l}\in\mathbb{R}^d \tag{10}$$

l番目の行動ベクトル$\boldsymbol{b}_l$に対応する行動価値関数を式(11)，(12)のように設定する．

$$Q^{\boldsymbol{b}_l}(\boldsymbol{s})=\boldsymbol{\theta}^{\boldsymbol{b}_l{}^T}\boldsymbol{\varphi}(\boldsymbol{s},\sigma) \tag{11}$$

$$\boldsymbol{\varphi}(\boldsymbol{s},\sigma)=\begin{pmatrix}\varphi_1(\boldsymbol{s},\sigma)\\ \varphi_2(\boldsymbol{s},\sigma)\\ \vdots\\ \varphi_d(\boldsymbol{s},\sigma)\end{pmatrix} \tag{12}$$

本手法では，行動の選択肢と同じ数だけパラメータベクトルと行動価値関数をおく．パラメータベクトルの更新式を式(13)のように設定する．

$$\boldsymbol{\theta}^{\boldsymbol{b}_l}\leftarrow\boldsymbol{\theta}^{\boldsymbol{b}_l}+\eta\{r_{k+1}+\xi\max_{\boldsymbol{b}_{k+1}}Q^{\boldsymbol{b}_{k+1}}(\boldsymbol{s}_{k+1})-Q^{\boldsymbol{b}_l}(\boldsymbol{s}_k)\}\boldsymbol{\varphi}(\boldsymbol{s}_k,\sigma_u) \tag{13}$$

ここで，ηは学習率，ξは割引率，kはステップであり，σ_uは定数，$\boldsymbol{b}_{k+1}$はステップ$k+1$での行動である．

4. 交通モデルの設定

本研究では，day-to-day dynamicsによって1日の中での出発時刻選択が決定され，within-day dynamicsにより経路選択確率が時々刻々と変化する交通モデルを設定する．単純化のため，単一ODを複数の経路が結び，それぞれの経路にボトルネックが存在するネットワークを考える．すると，経路が1つの場合は単一ボトルネックモデルとなり，経路が2つの場合は図-2に示すような2経路の複数ボトルネックモデルとなる．

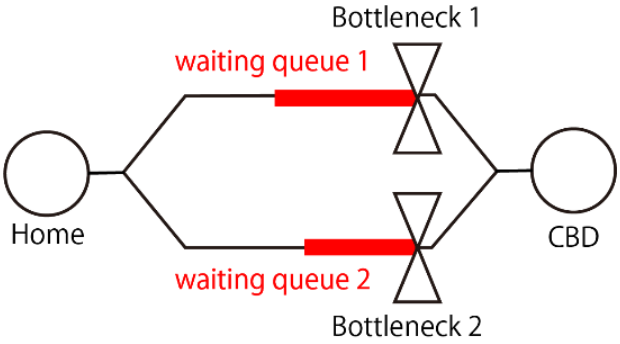


図-2　複数ボトルネックモデルの概要

2章で紹介した単一ボトルネックモデルにおける数学記法を，複数ボトルネックモデルに合わせて以下のように拡張する $(i=1,2)$．

μ_i：ボトルネックiの単位時間当たりの容量

$a_{j,i}(t)$: j日目にボトルネックiを通る旅行者の居住地からの出発率

$N_{j,i}(t)$: j日目，ボトルネックiにおける待ち行列台数

$w_{j,i}(t)$: j日目，ボトルネックiにおける待ち時間

$\tau_{j,i}(t)$: j日目，ボトルネックiにおける課金額

その他の数学記法については，単一ボトルネックモデルにおける記法と同一である．待ち行列は物理的な長さを持たないPoint Queueであるとし，t^*は全旅行者について等しいとする．また，tは勤務地への到着時刻に基づき定義されるとする．例えば，$a_j(t)$はj日目に勤務地に時刻tに到着する旅行者の出発率である．よって，課金額等は勤務地への到着時刻に依存する形で定義される．

更に，簡単のため，以下の設定を置く．

- 居住地から各ボトルネックモデルまでの自由旅行時間は，時刻によらず一定かつ全ボトルネックで等しいと設定する．
- ボトルネック以外の道路における自由旅行時間は，いずれの経路を選択した場合においても，時刻によらず一定かつ全経路で等しいと設定する．

j日目，時刻tにおけるボトルネックi $(i=1,2)$での待ち行列台数の変化を式(14)のように表す．

$$\frac{\mathrm{d}N_{j,i}(t)}{\mathrm{d}t}=\begin{cases}0 & (N_{j,i}(t)=0 \text{ and } a_{j,i}(t)<\mu_i)\\ a_j(t)-\mu_i & (\text{otherwise})\end{cases} \tag{14}$$

j日目，時刻tにおけるボトルネックi $(i=1,2)$での待ち時間を式(15)のように表す．

$$w_{j,i}(t)=\frac{N_{j,i}(t)}{\mu} \tag{15}$$

j日目，時刻tにおけるボトルネックi $(i=1,2)$での1人当たりの旅行者の一般化コストを式(16)のように表す．

$$c_{j,i}(t)=\tau_{j,i}(t)+\alpha w_{j,i}(t)+\begin{cases}\beta(t^*-t)\\ \gamma(t-t^*)\end{cases} \tag{16}$$

ここで，αは1人の旅行者における単位時間当たりの時間価値，βは希望到着時刻と比べて早く勤務地に到着する場合の単位時間あたりのコスト(早着コスト)，γは希望到着時刻と比べて遅く勤務地に到着する場合の単位時間あたりのコスト(遅着コスト)を示す．また，tは勤務地への到着時刻を示す．

日毎の出発率の変化(day-to-day dynamics)には，replicator dynamics[6]を採用する．出発率は式(17)～(19)に基づいて変化すると仮定する．

$$A_{j+1}(t)=A_j(t)+\delta A_j(t)(\bar{C}_j-C_j(t)) \tag{17}$$

$$A_j(t)=\sum_i a_{j,i}(t)\quad(i=1,2) \tag{18}$$

$$\bar{C}_j = \frac{\int A_j(T)\, C_j(T)\mathrm{d}T}{\int A_j(T)\mathrm{d}T} \tag{19}$$

ここで，δは day-to-day dynamics における日毎の変化速度を表す．$C_j(t)$は，j日目，時刻tに旅行者が各ボトルネックに向かう際のコストのログサム変数（期待最小コスト関数）として式(20)のように表される．

$$C_j(t) = -\frac{1}{\lambda}\ln\left(\sum_i \exp(-\lambda c_{j,i}(t))\right) \quad (i = 1,2) \tag{20}$$

ここで，λはパラメータである．

次に，経路選択の within-day dynamics を定める．確率効用モデルにロジットモデルを用い，式(21)のようにボトルネックhにおける within-day dynamics を定める$(h, i = 1,2)$．

$$a_{j,h}(t+\Delta t) = A_j(t+\Delta t)\frac{\exp(-c_{j,h}(t))}{\sum_i \exp(-c_{j,i}(t))} \tag{21}$$

5. 制御手法の構築

本研究では，強化学習の実装法として中央制御型手法と分散制御型手法の2種類を提案する． 中央制御型手法とは1日のピーク時間帯全体を考慮する手法であり，分散制御型手法とは各時間帯で学習を行い，各々の学習成果を共有して学習を進行させる手法である．両者を比較すると，分散制御型手法で各時間帯間の協調がとれた場合は，各時間帯で学習可能な分散制御型手法の方が早く学習を進行させられると想定される．

強化学習の枠組みには，3章で紹介したQ学習， ε - greedy法，行動価値関数の近似を用いる．

(1) 中央制御型手法

中央制御型手法では， 1日のピーク時間帯における全ての交通量データを状態，全ての課金額変更を行動として設定する．中央制御型手法の長所としては，ピーク時間帯内の時刻間相関への対応可能性が想定される．一方で，短所としては，ピーク時間帯の長期化に伴うパラメータ数の指数的な増大が想定される．

状態は「出発率ヒストグラム」とし，ベクトルとして扱う．ピーク時間帯をn等分し(nは自然数)，ステップkにおける状態ベクトル$\boldsymbol{s}_k$を式(22)のように設定する．

$$\boldsymbol{s}_k = \begin{pmatrix} a_{k,1} \\ a_{k,2} \\ \vdots \\ a_{k,n} \end{pmatrix} \tag{22}$$

ここで，$a_{k,i}$ $(i = 1,2,\cdots,n)$はステップk，時間区域iにおける出発率の平均を表す．

$\tau_j^c(t)$を**図-3** のような区分線形関数として設定し，ステップkにおける行動ベクトル$\boldsymbol{b}_k$を式(23)のように定義する．

$$\boldsymbol{b}_k = \begin{pmatrix} b_{k,1} \\ b_{k,2} \\ \vdots \\ b_{k,N+1} \end{pmatrix} \tag{23}$$

ここで，Nは$\tau_j^c(t)$の時間区分数(自然数)である．また，各区分における時間幅は全て等しいものとし，$b_{k,i}$ $(i = 1,2,\cdots,N+1)$は離散値で設定する．これより，課金額の更新は式(24)，(25)のようになる．

$$\boldsymbol{\tau}_B^c \leftarrow \boldsymbol{\tau}_B^c + \boldsymbol{b}_k \tag{24}$$

$$\boldsymbol{\tau}_B^c = \begin{pmatrix} \tau_{B,1}^c \\ \tau_{B,2}^c \\ \vdots \\ \tau_{B,N+1}^c \end{pmatrix} \tag{25}$$

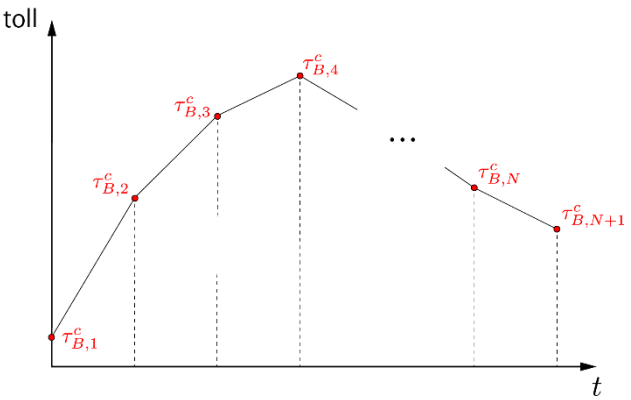


図-3 $\tau_j^c(t)$を表現する区分線形関数の概要

ステップkにおける報酬r_kは式(26)のように定義する．

$$r_k = \max\left(\frac{1}{G}\sinh(\psi_0 - \psi_k), -1\right) \tag{26}$$

ここで，ψ_0は$\tau_j^c(t) = 0$の場合のピーク時間帯内待ち時間総和，ψ_kはステップkにおけるピーク時間帯内待ち時間総和を表す．また，待ち時間総和の減少に対する反応の鋭敏性を高めるため，報酬設定に双曲線関数を用いる．但し，過度な負の報酬による学習の不安定化を回避するため，r_kの最小値は−1に設定する．

課金額の更新，および強化学習によるパラメータベクトルの更新は「day-to-day dynamicsが収束した段階」で行うものとする．収束条件を式(27)のように設定する．

$$\left|w_j(t) - w_{j-1}(t)\right| / w_j(t) < e \tag{27}$$

また，「$\tau_j^c(t) = 0$，かつ$\tau_j^s(t)$によって day-to-day dynamics が収束している状態」を初期状態とし，ステップ$k = 0$からスタートして課金額を更新する度にkを1増やすものとする．式(13)における学習率η，割引率ξについては，本研究では$\eta = 0.1$，$\xi = 0.9$と設定する．kは課金の変更試行回数とする．

(2) 単一ボトルネックモデルに対する分散制御型手法

分散制御型手法では，1日のピーク時間帯における一時間帯での交通流データを状態，一時間帯での課金額変更を行動として設定する．分散制御型手法の長所としては，状態，行動の数が少ないため，探索時間の短縮，パラメータ数の減少に繋がることが想定される．一方，短所としては，各時刻で課金額の更新を行うことによる課金額分布の平滑性の欠如や，特定の時刻での過剰課金による学習の妨害が想定される．

状態は出発率，待ち時間，課金額の安定化項の3つで設定する．ステップkでの状態ベクトル$\boldsymbol{s}_k$を式 (28) のように設定する．

$$\boldsymbol{s}_k(t) = \begin{pmatrix} a_k(t) \\ w_k(t) \\ \tau_k^s(t) \end{pmatrix} \tag{28}$$

ステップk，時刻tにおける行動を$b_k(t)$と表し，離散値で設定する．課金額の更新は式(29)のようになる．

$$\tau_{k+1}^c(t) = \tau_k^c(t) + b_k(t) \tag{29}$$

ステップk，時刻tにおける報酬$r_k(t)$を式 (30) のように設定する．

$$r_k(t) = \begin{cases} \min\left(\dfrac{1}{|\mu - a_k(t)|}, 10\right) & (|\mu - a_k(t)| \neq 0) \\ 10 & (|\mu - a_k(t)| = 0) \end{cases} \tag{30}$$

ある時刻において待ち時間が 0 である場合，ほとんどの場合には出発率がボトルネック容量以下である．よって分散制御型では，報酬を中央制御型に倣って待ち時間の大小で設定すると，出発率がボトルネック容量に比べて著しく小さくなる事態が頻発し，学習が適切に進行しないことが予想される．よって分散制御型では，各時刻において出発率とボトルネック容量の差が小さい程大きな値をとるような報酬設定を行った．但し，逆数の使用に伴う報酬の過大化を回避するため，報酬の最大値を 10 に設定する．

課金額の更新，および強化学習によるパラメータベクトルの更新のタイミングは中央制御型手法と同一とする．式(13)における学習率η，割引率ξについても，中央制御型手法と同様に$\eta = 0.1$，$\xi = 0.9$と設定する．kは課金の変更試行回数とする．

(3) 複数ボトルネックモデルに対する分散制御型手法

複数ボトルネックモデルでは，中央制御型手法を適用しようとすると，観測する状態の次元の増加によってパラメータ数が指数的に増大し，計算負荷が大きくなる．そこで，複数ボトルネックモデルには分散制御型手法を拡張して適用するものとする．

まず，単一ボトルネックモデルにおけるj日目，時刻tでの課金額の式を式(31)～(33)のように拡張する．

$$\boldsymbol{\tau}_j(t) = \boldsymbol{\tau}_j^c(t) + \boldsymbol{\tau}_j^s(t) \tag{31}$$

$$\boldsymbol{\tau}_j^c(t) = \begin{pmatrix} \tau_{j,1}^c(t) \\ \tau_{j,2}^c(t) \end{pmatrix} \tag{32}$$

$$\boldsymbol{\tau}_j^s(t) = \begin{pmatrix} \tau_{j,1}^s(t) \\ \tau_{j,2}^s(t) \end{pmatrix} \tag{33}$$

ここで，$\boldsymbol{\tau}_j^c(t)$はj日目，時刻tにおける課金額の大枠を決定する項，$\boldsymbol{\tau}_j^s(t)$はj日目，時刻tにおけるday-to-day dynamicsの安定化項である．本研究では，式 (34) のような設定を置く．

$$\tau_{j,1}^s(t) = \tau_{j,2}^s(t) \tag{34}$$

これにより，経路選択の within-day dynamics による待ち時間分布の不安定な変動を抑えながら，day-to-day dynamics を収束させる．

経路選択の within-day dynamics を考慮するため，状態に「1 つ前の時刻における出発率」，報酬に「1 つ後の時刻における出発率とボトルネック容量の差」を組み込む設定とする．これにより，行動決定の際に 1 つ前の時刻からの影響，行動評価の際に 1 つ後の時刻への影響を考慮することが可能になる．単一ボトルネックモデルの場合と同様にガウスカーネルによる関数近似を行うため，ステップkにおける状態，関数近似における基底関数，基底関数の座標を定める$\boldsymbol{p}$を式 (35) ～ (37) のように設定する．

$$\begin{aligned} \boldsymbol{s}_k(t) &= \begin{pmatrix} s_{1,1} & s_{2,1} \\ s_{1,2} & s_{2,2} \end{pmatrix} \\ &= \begin{pmatrix} a_{k,1}(t-1)/D_1 & a_{k,2}(t)/D_2 \\ a_{k,1}(t-1)/D_1 & a_{k,2}(t)/D_2 \end{pmatrix} \end{aligned} \tag{35}$$

$$\varphi_m(\boldsymbol{s}, \sigma) = \exp\left(-\frac{\|\boldsymbol{s} - \boldsymbol{p}_m\|_F^2}{2\sigma^2}\right) \tag{36}$$

$$\boldsymbol{p} = \begin{pmatrix} p_{1,1} & p_{2,1} \\ p_{1,2} & p_{2,2} \end{pmatrix} = \begin{pmatrix} P_{1,1}/D_1 & P_{1,2}/D_2 \\ P_{2,1}/D_1 & P_{2,2}/D_2 \end{pmatrix} \tag{37}$$

$$P_{j,i} \in \{\mu_j - v_{1,j}D_j, \mu_j - (v_{1,j}-1)D_j \cdots, \mu_j + v_{2,j}D_j\} \quad (j, i = 1,2)$$

$\|\boldsymbol{H}\|_F$は行列$\boldsymbol{H}$のフロベニウスノルムであり，式 (38) のように定義される．

$$\|\boldsymbol{H}\|_F = \sqrt{\sum_{i,j} h_{i,j}^2} \quad \left(h_{i,j}: \boldsymbol{H}\text{の}i\text{行}j\text{列の要素}\right) \tag{38}$$

なお，tがピーク時間帯の開始時刻である場合については，式(39)のように状態を処理する．

$$s_{1,1} = 0, \ s_{2,1} = 0 \tag{39}$$

次に，ステップk，時刻tにおける行動ベクトル$\boldsymbol{b}_k(t)$を式(40)のように設定する．

$$\boldsymbol{b}_k(t) = \begin{pmatrix} b_{k,1}(t) \\ b_{k,2}(t) \end{pmatrix} \tag{40}$$

$b_{k,i}(t)$ $(i = 1,2)$は，ボトルネックiにおけるステップk，時刻tでの相対的な課金額変更を表す．よって，課金額変更は式(41)のようになる．

$$\boldsymbol{\tau}_{k+1}^c(t) = \boldsymbol{\tau}_k^c(t) + \boldsymbol{b}_k(t) \tag{41}$$

ステップk，時刻tにおける報酬$r_k(t)$を式(42)，(43)のように設定する．

$$r_k(t)=\begin{cases}\min\left(1/\zeta_k(t),\ 100\right) & (\zeta_k(t)\neq 0)\\ 100 & (\zeta_k(t)=0)\end{cases} \quad (42)$$

$$\zeta_k(t)=\sum_i\left|\mu_i-a_{k,i}(t)\right|+\sum_i\left|\mu_i-a_{k,i}(t+1)\right| \quad (43)$$

$$(i=1,2)$$

なお，tがピーク時間帯の終了時刻である場合については，式(44)のように$\zeta_k(t)$を処理する．

$$\zeta_k(t)=2\left(\left|\mu_1-a_{k,1}(t)\right|+\left|\mu_2-a_{k,2}(t)\right|\right) \quad (44)$$

(4) アルゴリズムのまとめ

以下に，本研究で提案したアルゴリズムの流れを示す．フローチャートは**図-4**の通りである．状態，行動，報酬の設定は中央制御型手法，単一ボトルネックモデルに対する分散制御型手法，複数ボトルネックモデルに対する分散制御型手法で異なるが，アルゴリズムは共通である．

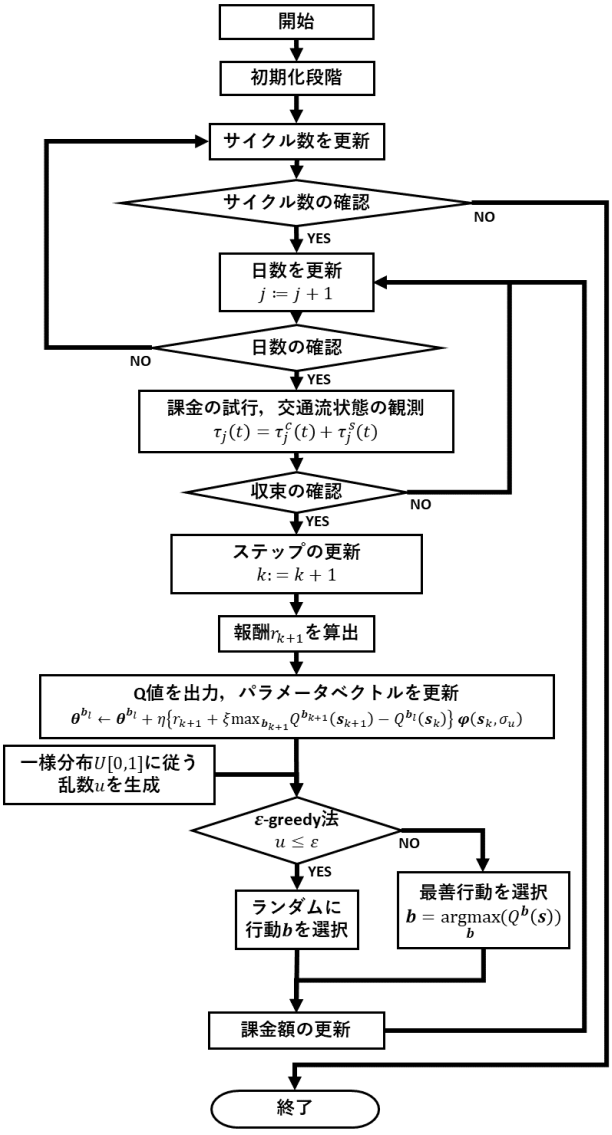


図-4 アルゴリズムのフローチャート

[Step 0]

日数を$j:=0$と設定し，旅行者の出発率の初期状態$a_0(t)$を設定する．

[Step 1] 初期化段階 (Initialization phase)

[Step 1.1]

日数を$j:=j+1$と更新する．安定化項$\tau_j^s(t)$による課金のみを実施する$(\tau_j^c(t)=0)$．

[Step 1.2]

交通流における$w_j(t)$および$a_j(t)$を観測する．

[Step 1.3]

収束確認を行う．eの設定値に対して，$\left|w_j(t)-w_{j-1}(t)\right|/w_j(t)<e$ならStep 1.4に進む．その他の場合，日数を$j:=j+1$に設定し，Step 1.1 に戻る．

[Step 1.4]

現在の状態を利用者均衡状態 (user equilibrium state) とみなす．日数を$j:=0$にリセットし，Step 2 に進む．

[Step 2] Trial-and-error 段階 (Trial-and-error phase)

[Step 2.1]

サイクル数に 1 を足す．サイクル数が設定値以下なら Step 2.2 に進む．サイクル数が設定値より大きい場合はアルゴリズムを終了する．

[Step 2.2]

日数を$j:=j+1$と更新する．日数が設定値以下なら Step 2.3 に進む．日数が設定値より大きい場合は，$j:=0$とリセットして Step 2.1 に戻る．

[Step 2.3]

$\tau_j^c(t)=\tau_{j-1}^c(t)$と決定し，課金額を$\tau_j(t)=\tau_j^c(t)+\tau_j^s(t)$として課金を実施する．

[Step 2.4]

交通流における$w_j(t)$および$a_j(t)$を観測する．

[Step 2.5]

収束確認を行う．eの設定値に対して，$\left|w_j(t)-w_{j-1}(t)\right|/w_j(t)<e$なら，ステップを$k:=k+1$と更新し，Step 2.6 に進む．そうでない場合，Step 2.2 に戻る．

[Step 2.6] 強化学習，課金額の更新 (Toll update)

[Step 2.6.1] 報酬の算出

観測した状態に基づき，報酬を算出する．

[Step 2.6.2] Q 値の計算，パラメータベクトルの更新

day-to-day dynamics の収束前に観測した交通流データと実際に試行した行動から$Q^{\boldsymbol{b}_k}(\boldsymbol{s}_k)$，および収束後に観測した交通流データから

$\max_{\boldsymbol{b}_{k+1}}Q^{\boldsymbol{b}_{k+1}}(\boldsymbol{s}_{k+1})$を計算し，報酬と合わせてパラメータベクトルを更新する．

[Step 2.6.3] 乱数の生成

一様分布$U[0,1]$に従う乱数uを生成する．

[Step 2.6.4]　ε -greedy 法による探索・利用の選択

uがεの設定値以下なら Step 2.6.5 に進み，εの設定値より大きければ Step 2.6.6 に進む．

[Step 2.6.5]　行動の探索

行動の選択肢の中から，ランダムに行動を選択し，次の行動とする．その後，Step 2.6.7 に進む．

[Step 2.6.6]　行動の利用

day-to-day dynamics の収束後に観測した交通流データから，行動の全選択肢における$Q^{\boldsymbol{b}_{k+1}}(\boldsymbol{s}_{k+1})$を計算する．最も Q 値が大きくなる$\boldsymbol{b}_{k+1}$を最善行動とみなし，次の行動とする．その後，Step 2.6.7 に進む．

[Step 2.6.7]　課金額の更新

Step 2.6.5 または Step 2.6.6 で選択した行動に基づき，課金額を更新する．その後，Step2.2 に戻る．

(5)　考察

本手法を現実の交通システムに適用するにあたっては，複数のアプローチが考えられる．最も素朴なアプローチは，本手法を直接現実の交通システムに適用するというものである．この場合．Q学習が初期状態から実時間で進行するため，非常に長い実時間を要する可能性がある．これは，複数回にわたり交通のday-to-day dynamicsが収束するのを待つ必要があるためである．

より高度なアプローチとして，現実の交通システムを模した計算機を用意し，計算機上のシミュレーションで本手法を繰り返し適用し，その学習結果を初期状態とした本手法を現実の交通システムに適用する方法が考えられる．この場合，学習のほとんどは計算機上で進展するため要する実時間が短くてすむと予想される．また，計算機上に用意した交通システムと現実のそれは完全に一致する必要はなく，それらの間の差異は本手法の現実における学習過程で吸収されると予想される．これは機械学習における転移学習に相当する．

6.　実験結果と考察

(1)　実験の目的

開発した制御手法の性質をシミュレーション実験により確認する．具体的には以下の内容を確認する．

・単一ボトルネックモデルにおいて，最適なエージェントパラメータを設定した上で，中央制御型手法・分散制御型手法それぞれで強化学習を行い，学習済みエージェントの待ち時間減少性能を比較することで，分散制御型手法の優位性を検証する．
・時間価値が異なる環境に上記の学習済みエージェントを適用し，待ち時間減少性能をSeo[4]による既存手法と合わせて比較することで，計算機上と現実で交通システムが完全に一致しなくとも，現実における学習過程で両者の差異を吸収できるという可能性を示す．
・複数ボトルネックモデルにおいて，最適なエージェントパラメータを設定した上で，分散制御型手法で強化学習を行い，学習済みエージェントの待ち時間減少性能を分析することで，経路選択のある交通モデルへの対応可能性を検証する．

(2)　単一ボトルネックモデルでのシミュレーション

まず，中央制御型手法と分散制御型手法のそれぞれを用いてε，eの値について幾つかのパターンで学習を行い，中央制御型手法では$\varepsilon = 0.4$，$e = 0.05$，分散制御型手法では$\varepsilon = 0.2$，$e = 0.05$を最適と決定した．なお，学習時の各種数値の設定は以下の通りである．

交通モデルの数値設定

$$\mu = 20,\ \alpha = 1.0,\ \beta = 0.45,\ \gamma = 1.2,\ \delta = 1.0$$
$$t \in \mathbb{N},\ 1 \le t \le 40,\ t^* = 30,\ \Delta t = 1$$

課金額の安定化項の数値設定

$$\sigma_1 = 200,\ \sigma_2 = 100,\ \sigma_3 = 2,\ e = 0.02$$

強化学習の数値設定

・中央制御型

状態・行動・報酬

$$n = 5,\ N = 4,\ b_{k,l} \in \{-0.05, 0, 0.01, 0.05, 0.1\}$$
$$G = 10000$$

関数近似に用いる基底関数

$$D = 2.5,\ v_1 = 4,\ v_2 = 4,\ \sigma = 6,\ \sigma_u = 1.5$$

・分散制御型

行動

$$b_k = 0.01z\ \ (z \in \mathbb{Z},\ -10 \le z \le 15)$$

関数近似に用いる基底関数

$$D_a = 2,\ D_w = 0.02,\ D_\tau = 0.05$$
$$v_1 = 10,\ v_2 = 20,\ v_3 = 20,\ v_4 = 10,\ v_5 = 10$$
$$\sigma = 3,\ \sigma_u = 0.6$$

また，学習の際は1000日を1サイクルとし，1000日経過後に「$\tau_j^c(t) = 0$，かつ$\tau_j^s(t)$によってday-to-day dynamicsが収束している状態」に戻って学習を行うものとした(学習結果，即ち行動価値関数におけるパラメータベクトルの更新結果は引き継ぐ)．

そして，中央制御型手法では300サイクル，分散制御型手法では30サイクルを1セットとし，上記のε，eの数値設定で3セット学習を行い，中央制御型手法と分散制御型手法の学習結果の比較を行った．分散制御型手法で

は各時刻で学習する分，学習回数は多く確保できる．そのため，1セットにおけるサイクル数を中央制御型より少なく設定した．以下に，各手法の各セットにおける学習済みエージェントを学習時と同一条件の環境に適用した際のピーク時間帯内待ち時間総和の推移を**図-5・図-6**として示す．1st，2nd，3rdはそれぞれ1回目，2回目，3回目のセットでの結果を表す．

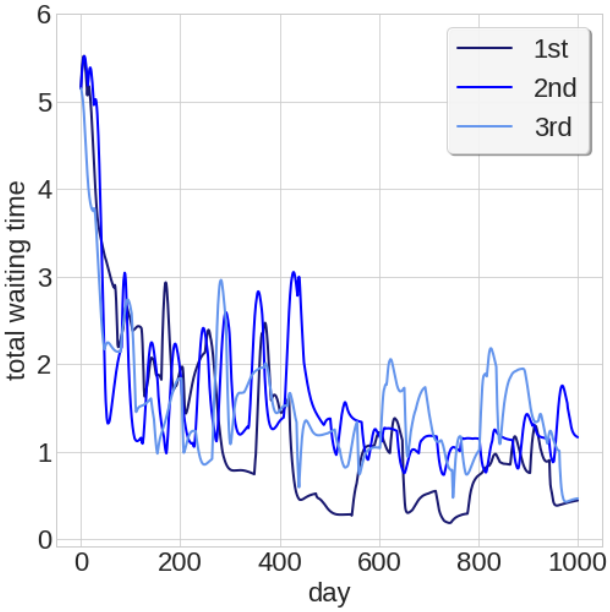


図-5　中央制御型手法による学習済みエージェント適用時の待ち時間総和の推移

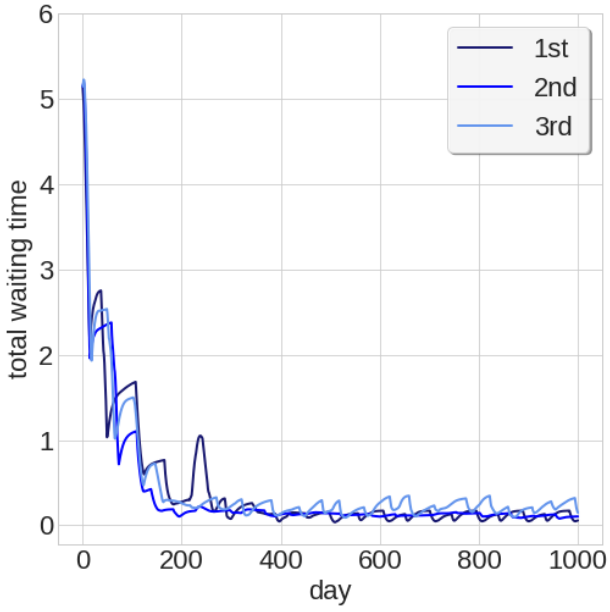


図-6　単一ボトルネックモデルに対する分散制御型手法による学習済みエージェント適用時の待ち時間総和の推移

図-5・図-6より，分散制御型手法の方が速やかにボトルネックでの待ち時間を減少させていることが分かる．

分散制御型手法の場合における待ち時間$w_j(t)$の分布を**図-7**，課金額の大枠を決定する項$\tau_j^c(t)$の分布を**図-8**として示す．**図-8**より，課金額分布が日数の経過に伴って全体として上下していることが分かる．しかし，$\tau_j^c(t)$の分布における最小値を分布全体から引く操作を行うことで，$\tau_j^c(t)$の分布の乱高下は容易に回避可能である．

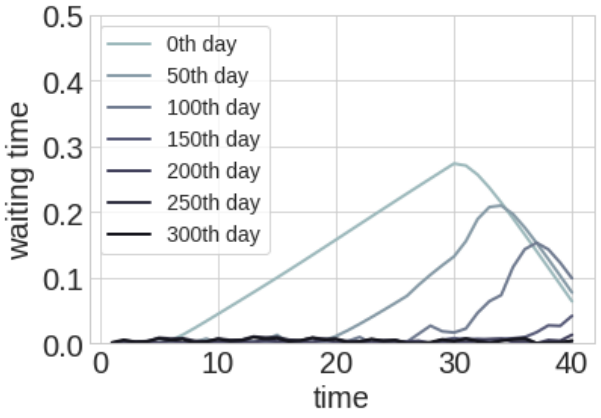


図-7　単一ボトルネックモデルに対する分散制御型手法による学習済みエージェント適用時の待ち時間$w_j(t)$の分布の推移

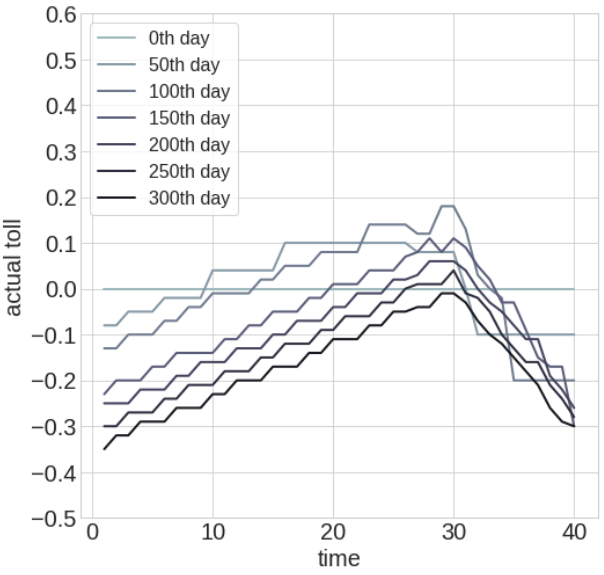
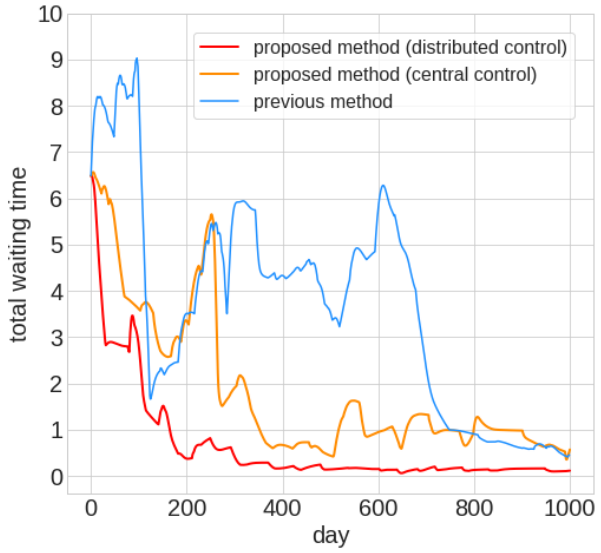
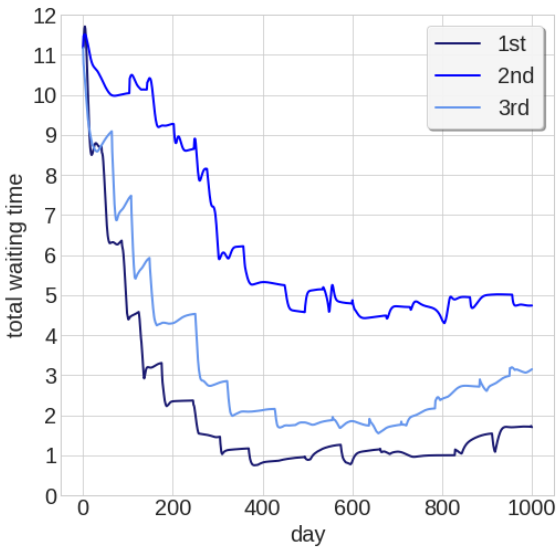


図-8　単一ボトルネックモデルに対する分散制御型手法による学習済みエージェント適用時の$\tau_j^c(t)$の分布の推移

また，異なる環境への対応可能性を検証するため，時間価値が異なる環境に学習済みエージェントを適用した．具体的には，単一ボトルネックモデルにおいて$\alpha = 1.0$を$\alpha = 0.5$と変更し，学習済みエージェントの適用結果を中央制御型，分散制御型，Seo[4]による既存手法で比較した．各手法での結果を**図-9**として示す．**図-9**より，中央制御型手法・分散制御型手法共に，既存手法よりも速やかにボトルネックでの待ち時間を減少させられていることが分かる．



図–9 時間価値が異なる環境に対する学習済みエージェント適用時の待ち時間総和の推移



図–10 複数ボトルネックモデルに対する分散制御型手法による学習済みエージェント適用時の待ち時間総和の推移

(3) 複数ボトルネックモデルでのシミュレーション

単一ボトルネックモデルの場合と同様に，まずε，eの値について幾つかのパターンで学習を行い，$\varepsilon = 0.2$，$e = 0.02$を最適と決定した．なお，学習時の各種数値の設定は以下の通りである．

交通モデルの数値設定

$$\mu_1 = 20,\ \mu_2 = 15,\ \alpha = 1.0,\ \beta = 0.45,\ \gamma = 1.2,$$
$$\delta = 1.0,\ t \in \mathbb{N},\ 1 \le t \le 40,\ t^* = 30,\ \Delta t = 1$$

課金額の安定化項の数値設定

$$\sigma_1 = 200,\ \sigma_2 = 100,\ \sigma_3 = 2,\ e = 0.02$$

強化学習の数値設定

行動

$$b_{k,i} = \{-0.05, 0, 0.01, 0.05, 0.1\}\ \ (i = 1,2)$$

関数近似に用いる基底関数

$$D_1 = 4,\ D_2 = 3,\ v_{1,1} = 5,\ v_{2,1} = 5,$$
$$v_{1,2} = 5,\ v_{2,2} = 5,\ \sigma = 3,\ \sigma_u = 0.6$$

サイクル，セットの設定については単一ボトルネックモデルの場合と同様である．そして，30サイクルを1セットとし，上記のε，eの数値設定で3セット学習を行った．以下に，各手法の各セットにおける学習済みエージェントを学習時と同一条件の環境に適用した際のピーク時間帯内待ち時間総和の推移を**図–10**として示す．

(4) 考察

単一ボトルネックモデルにおける中央制御型手法と分散制御型手法を比較すると，分散制御型手法の方が優れた待ち時間減少性能を有していると言える．これは，分散制御型手法では各時刻で学習するために学習回数を多く確保できること，および状態・行動のパターンが少ないことによる行動探索の容易さに起因すると考えられる．

複数ボトルネックモデルにおける拡張された分散制御型手法も，ある程度待ち時間を減少できた．しかし，単一ボトルネックモデルの場合と比較し，減少速度は遅く，完全に待ち時間をゼロにするには至らなかった．これについては，現在の行動と報酬の定義では各時間帯間の協調が不十分であることが原因であると推測される．

7. 結論と今後の課題

本研究では，day-to-day dynamicsを考慮した出発時刻選択問題に対する動的混雑課金手法の開発を行った．本手法の特徴は，Trial-and-errorによる課金額更新によって，時間価値等が異なる環境に適用可能，かつ課金額更新を速やかに行える点にある．Trial-and-error手法は，強化学習に基づくものとした．これにより，交通モデルフリー，即ち様々なネットワークに対して適用可能な，交通データにより駆動する動的混雑課金が可能となる．具体的には，単一ボトルネックモデルに対する中央制御型手法・分散制御型手法，複数ボトルネックモデルに対する分散制御型手法の3手法を構築した．

検証の結果，単一ボトルネックモデルについては，分散制御型手法が中央制御型手法に対する優位性を持つこと，および，中央制御型手法・分散制御型手法共に，時間価値の異なる環境への対応性が既存手法より優れていることを確認した．複数ボトルネックモデルにおける検証では，提案手法が待ち時間減少性能を有することを確認した．今後の拡張により，交通モデルフリーな動的混雑課金手法の構築が可能となると考えられる．

今後の課題としては，行動(課金額の変更)をボトルネック別に行うことによる強化学習における状態・行動の次元の削減などがあり，現在研究を進めている．

謝辞：本研究は国土交通省新道路技術会議の研究課題「学習型モニタリング・交通流動予測に基づく観光渋滞マネジメントについての研究開発」の助成を受けた．ここに謝意を表します．

参考文献

1) M.Z.F. Li. The role of speed‐flow relationship in congestion pricing implementation with an application to singapore. Transportation Research Part B: Methodological，Vol.36，No.8，pp.731‐754，2002.
2) H. Ye， H. Yang， and Z. Tan. Learning marginal-cost pricing via a trial-and-error procedure with day-to-day flow dynamics. Transportation Research Part B: Methodological，Vol.81，pp.794‐807，2015. ISTTT 21 for the year 2015.
3) T. Seo and Y. Yin. Optimal pricing for departure time choice problems with unknown preference and demand: Trial-and-error approach. 2019.
4) T. Seo . Trial-and-error congestion pricing scheme for morning commute problem with day-to-day dynamics. Transportation Research Procedia (The 22nd EURO Working Group on Transportation Meeting，18‐20 September 2019，Barcelona，Spain)，2019.
5) W.S. Vickrey. Congestion theory and transport investment. The American Economic Review， Vol.59， No.2， pp.251‐260，1969.
6) P. Schuster and K. Sigmund. Replicator dynamics. Journal of Theoretical Biology，Vol.100，No.3，pp.533‐538，1983.
7) 井料隆雅．不安定な動的利用者均衡に対する安定化制御．土木計画学研究発表会・講演集，55，2017.
8) T. Iryo, M. Smith, and D. Watling. Stabilisation strategy for unstable transport systems under general evolutionary dynamics. Transportation Research Part B: Methodological，2019.
9) C.J.C.H. Watkins and P. Dayan. Q-learning. Machine Learning，Vol.8，No.3，pp.279‐292，1992.
10) R.S. Sutton and A.G. Barto. Reinforcement Learning: An Introduction. The MIT Press，second edition，2018.

(2020.3.8 受付)

DATA-DRIVEN DYNAMIC CONGESTION TOLL OPTIMIZATION METHODS BASED ON REINFORCEMENT LEARNING

Kimihiro SATO, Toru SEO and Takashi FUSE

As one of the measures to alleviate traffic congestion, the usefulness of dynamic congestion toll is proposed to consider the fluctuation of traffic demand within a day. In addition, a trial-and-error congestion toll method is proposed to deal with the asymmetric information between a toll entity and road users.

In this research, we construct a trial-and-error dynamic congestion toll method with reinforcement learning that can respond to various environments and quickly update the toll. Then, through simulations on traffic models, we compare with existing research and verify the possibility of responding to environmental changes.