マルチエージェント強化学習とインテリジェント・ドライビング・アルゴリズム*

A Multiagent Reinforcement Learning as an Intellegent Driving Algorithm

宮城俊彦**
By Toshihiko MIYAGI

1. はじめに

ドライバーが走行経験を経るに従い、ナビゲーションシステムが道路区間の走行情報を自動的に更新し、自律的な学習過程を通してより最適な経路情報をドライバーに伝達できるようなシステムについて考える。現在の混雑情報システムは、ドライバーにとっては外部化された情報であり、必ずしも必要な情報ばかりとは限らない。また、都市内での起終点走行時間情報を提供するためには限界がある。本研究が想定するガイダンス・システムは、ドライバーの経験を内部化し、過去の走行経験をナビゲーションシステムが記憶し、学習することにより、最適な経路をドライバーの要請に応じて情報提供できる仕組みである。そのために必要な最短経路探索と自律的学習アルゴリズム開発のための基本的モデルを考える。ただし、この小稿ではその学習アルゴリズムの具体的な手法を示すことが主目的ではなく、その考え方の延長線上でネットワーク均衡問題を議論するめの導入として位置づけている。

現在、交通量配分等で利用される最短経路アルゴリズムは、Bellman[1)]の最適性原理を解くために開発されてきた。一方、Haward[2)]は、ノード間の推移確率を導入した確率的最小費用経路(Stchastic Shortest Path:SSP)問題を導入し、これをマルコフ決定過程として解く方法を提案した。推移確率が１かゼロの値しかとらない場合には、SSPは通常の最小費経路探索問題になる。マルコフ決定過程は一般に時間集合 $t \in T$、状態集合 $s,s' \in S$、行動集合 $a \in A$、推移確率 $p_t(s,a,s')$、利得（あるいはコスト） $c_t(s,a,s')$ によって特定化される。状態間の推移確率が与えられるならばマルコフ決定過程(Markov Decision Process:MDP)は方策選択の多様な問題に応用することができ、多くの場合、不動点定理によって解に収束することが知られている[3)]。

Bertsekas and Tuitsiklis[4,5)]は、有限時間のMDPとしてSPPの特性を明らかにするとともに，推移確率が未知の場合、SSP問題が機械学習に置き換えられること、機械学習の収束性は、結局、確率近似理論の収束性の議論になることを示した。確率近似の収束性については、同期アルゴリズムについては、Kushner and Clark[6)]によって、また、非同期アルゴリズムについてはBorker and Soumyanath[7)]によって証明されている。

本研究は、まず、上述した自律型経路学習過程のモデルとして機械学習の一種であるＱ学習[8)]あるいは相対Ｑ学習を利用したアプローチを採用する。次に、この考え方を交通量配分問題に適用できるように拡張する。そのためには、いくつかの変更が必要である。まず、最小費用経路問題はノード系列あるいはノードリンク系列の選択であるため、これを経路選択への変更する必要がある。次に、有限時間問題を無限反復問題に置き換えることが必要である。最後に、単一学習者を複数学習者とする必要がある。

経路選択を前提にした繰り返しゲームについて、Miyagi[9,10,11)] は、ゲーム理論における確率的仮想プレイの理論[12,13)]を応用したアプローチを提案している。そして、過去の走行経験に基づく知識の集積と、それに基づく経路選択過程の繰り返しが、Nash均衡を実現することを明らかにしている。この場合、パフォーマンス関数は非連続でもよく、また、非対称のヤコビ行列であったもよい。走行経験による評価値分布と先験分布の凸結合として情報を更新するため、その収束性は確率近似理論に基づく。

本研究の狙いは、マルチエージェント強化学習理論から導かれるネットワーク均衡問題と確率的仮想プレイに基づくネットワーク均衡モデルが同じ結論に至ることを示すことにある。また、新たに、非同期アルゴリズムを前提に時間スケールの異なる学習パラメータを導入したモデルについても検討する。

2. SSP問題とQ学習

ノード集合 $\mathbf{N}$、リンク集合 $\mathbf{L}$ で構成されるネットワークを考える。簡便のため、単一ＯＤペアの場合を議論するが、複数ＯＤペアに拡張することは容易である。今、状態集合 $\mathbf{S}$ としてネットワーク上のノード集合を選び、行動集合 $\mathbf{A}$ として流入したノー

---

* 経路選択、交通量配分、交通制御

**正会員　工博　岐阜大学教授　地域科学部
（〒501-1193 岐阜県岐阜市柳戸1-1）E-mail:miyagi@cc.gifu-u.ac.jp, Tel:058-293-3307

ドに隣接するリンクとおく。すなわち、ノード $i \in \mathbf{N}$ に流入したドライバーの取りうる行動は、ノード i を始点とする流出リンク集合であり、これを $\mathbf{A}(i) = \{a_1, a_2, \cdots, a_{m(i)}\} \equiv \{\ell_1, \cdots, \ell_{m(i)}\}$ とおく。また、行動 $a \in \mathbf{A}$ を選択したときのノード間の推移確率を $p(i,a,j), i,j \in \mathbf{N}$ で表す。このとき、ある任意のノード i_0 からノード i に至るSSP問題は、次のMDPとして定式化できる[5]。

$$V^{t+1}(i) = \min_{a \in A(i)} [\sum_{j \in S} p(i,a,j)\{c(i,a,j) + V^t(j)\} - V^t(i_0)] \quad (1)$$

ここに、$V^t(i)$ はステージ t におけるノード i までの最小費用コストである。また、$c(i,a,j)$ はノード i から j へ行動 $a \in \mathbf{A}(i)$ を選択して推移したときのコストである。$c(i,a,j)$ が与えられており、また、推移確率 $p(i,a,j), i,j \in \mathbf{N}$ が１または0で与えられる決定論的な行動では、(1)の問題は通常の最小費用経路を求める問題になる。また、(1)は価値反復法で記述しているが、行動選択は方策反復として定式化する必要がある。本研究ではQファクターに変換し、Qファクターを利用してある行動選択を求めるので、その必要が無い点も留意が必要である。特定の行動集合を指定することは、ノードとそれに付随する行動を指定するため、$\{i_0, a_{i_0} \in A(i_0), i_1, a_{i_1} \in A(i_1), \cdots\}$ という系列を発生させる。したがって、（１）は行動集合として経路集合を指定し、最小費用経路を求める問題として定式化することも可能である。

推移確率とコストが与えられれば、(1)の問題をMDPとして解くことができる。しかし、正確な推移確率を求めることは容易ではない。したがって、この問題を学習よって求める方法について考える。Q学習では、推移確率に代わって次に定義するQファクターを用いる：

$$Q^{t+1}(i,a) = \sum_{j \in S} p(i,a,j)\{c(i,a,j) + V^t(j)\} - V^t(i_0) \quad (2)$$

これより、(1)は次のように書き換えられる。

$$V^{t+1}(i) = \min_{a \in A(i)} Q^{t+1}(i,a) \quad (3)$$

これを用いてQファクターを再定義すると、

$$Q^{t+1}(i,a) = \sum_{j \in S} p(i,a,j)\{c(i,a,j) + \min_{b \in A(j)} Q^t(j,b)\} - V^t(i_0)$$
$$= E[c(i,a,j) + \min_{b \in A(j)} Q^t(j,b)] - V^t(i_0) \quad (4)$$

と書ける。すなわち、Qファクタは、確率的な現象の平均値であり、その１つの実現値が

$$[c(i,a,j) + \min_{b \in A(j)} Q^t(j,b)]$$

で与えられると考えるわけである。したがって、その実現値がどのような確率分布に従うのかが重要になる。ブートストラップ法などで平均化する手法を考案すればよく、(1)のように明示的な推移確率を必要としない。あるいは、実際の走行などで実現した値を観測し、それ平均化する手続きを考えればよい。この平均化プロセスはRobins and Munro[14]によって提案され,その後多くの研究者によって改良された次のような確率近似法を利用する。

$$Q^{t+1}(i,a) = Q^t(i,a) + \gamma^t [F(Q^t, \tilde{j}) - Q^t(i,a)]$$
$$F(Q,\tilde{j})(i,b) = c(i,b,\tilde{j}) + \min_{b \in A(\tilde{j})} Q^t(\tilde{j},b) \quad (5)$$

ここで、$\tilde{j}$ は確率法則に従う j の選択を表している。Q学習は、(5) の第2式に示されるように、推移したノード先での最適選択（先読み）を必要とする。また、この平均化プロセスは常微分方程式体系(ODE) $\dot{x} = f(x)$,の形式になっており、その収束性はODEの特性に依存し[15]、学習率パラメータ γ と F が関与する。変換 T を次のように定義する。

$$T(Q)(i,a) = \sum_{j \in S} p(i,a,j) F(Q,j) \quad (6)$$

このとき、

1) $\sum_t \gamma(t) \to \infty,\ \sum_t \gamma^2(t) < \infty$

2)変換 T はLipsitz連続である。

3)変換 T は非拡大(non-expansive)

ならば、(5)は大域的、漸近的安定均衡に収束する[7]。

(5)はすべてのノードでのQファクターを同時に更新するので同期アルゴリズムと呼ばれる。しかし、我々が対象にしている最小費用経路問題では、ドライバーは一回のステージで到達しているのはひとつのノードなので、このノードのQファクターが更新できるのみである。これは非同期アルゴリズムと呼ばれ（ゲーム理論では、これを強化学習と呼んでいる）、次のように表される。

$$Q^{t+1}(i,a) = Q^t(i,a) + \gamma(\nu(t,i))[F(Q^t, \tilde{j}) - Q^t(i,a)] I(i \in S^t) \quad (7)$$

ここに $I(i \in S^t)$ は指示関数であり、（）で示される条件が成立しているとき1,そうでないときは0の値をとる。(7)の収束条件は若干ややこしくスペースをとるので省略する。このように、インテリジェント・ドライビングシステムでは、システムが自動的にQファクターを記憶し、走行のたびごとに（７）によって自動的にQファクターを更新していけばよい。このとき、$c(i,a,j)$ は確率的に変動する値と仮定することができ、多くの場合、前述の収束条件を満足する。Q学習では、

$$d(i) = \arg\min_{b \in A(i)} Q(i,b) \quad (8)$$

によって、最適選択が与えられる。

3. マルチ･エージェント強化学習

ところで、すべてのドライバーがこのようなインテリジェント・ドライビング・システムを装備していたならばどうなるであろうか。この場合、$c(i,a,j)$ は、各々のドライバーのとる行動に左右される。また、推移確率もそうである。したがって、Qファクターも他のドラーバーの行動に影響を受け

ることになる。すなわち、すべてのドライバーが（5）あるいは（7）に示した更新方策に従うならば、行動 a をベクトル $\mathbf{a}=[a^1,\cdots a^M]$ に変える必要がある。ここに M はすべてのドライバーの数。このとき、上記の問題は、マルチ・エージェント強化学習問題になる。このようなマルチエージェントが競合する状況の下で、各エージェントが最大利得あるいは最小費用を得ようと行動する場合に実現する均衡はNash均衡として知られている。Hu and Welman[16]は、Nash均衡を実現するQ学習を提案している。

まず、ゲーム論に関連した若干の用語を定義しておこう。

プレイヤー k の混合戦略を $\pi^k \in \Delta(A^k)$、プレイヤー全体の同時混合戦略：$\boldsymbol{\pi}=(\pi^1,\cdots,\pi^M)\in\Delta(\mathbf{A})$ とおく。プレイヤー k 以外の同時混合戦略は、$\boldsymbol{\pi}^{-k}=(\pi^1,\cdots,\pi^{k-1},\pi^{k+1},\cdots\pi^M)$ と表記する。Hu and Welman は次のような Nash-Q 関数を定義し、Q学習が Nash 均衡を導くことを証明している。

$$\begin{aligned} &Q_{t+1}^k(s,a^1,\cdots a^M)=(1-\gamma_t)Q_t^k(s,a^1,\cdots a^M)\\ &\quad+\gamma_t[c_t^k+\alpha\pi^1(s')\pi^2(s')\cdots\pi^M(s')Q_t^k(s')] \\ &k=1,\cdots,M. \end{aligned} \tag{9}$$

このアイデアをネットワーク均衡問題に応用する。

4. 下半連続パフォーマンス関数の下での利用者均衡

以下では、伝統的交通量配分理論と同様、リンクコストがリンクパフォーマンスで与えられる場合のマルチエージェント学習問題について考える。ただし、リンク・パフォーマンス関数は、非連続関数が扱えるようにするため下半連続関数を想定する。**2.** で定義した行動集合は経路で与えられるものと仮定して議論を進める。

$\mathbf{h}=(h_1,\cdots,h_p,\cdots)$：経路フローベクトル

$\mathbf{f}=(f_1,\cdots,f_a,\cdots f_L)$：リンクフローベクトル

$c_a(\mathbf{f})$：リンク a の所要時間（コスト）

$u_p(\mathbf{h})=\sum_{a\in A}\delta_{ap}c_a(\mathbf{f}(\mathbf{h}))$：経路コスト

ただし、以下の関係が成立している.

$$\begin{aligned} f_a&=\sum_{p\in\mathbf{P}}\delta_{ap}h_p\\ \sum_{p\in\mathbf{P}}h_p&=M \end{aligned} \tag{10}$$

次に、利用者 k が経路 $p\in\mathbf{P}$ を選択する確率を $\{\pi_p^k\}$ と置くとき、$\{\pi_p^k\}$ は次の関係を満足する.

$$\begin{aligned} \sum_{p\in\mathbf{P}}\pi_p^k&=1\\ \sum_{k\in\mathbf{M}}\pi_p^k&=h_p\ \forall\ p\in\mathbf{P} \end{aligned} \tag{11}$$

したがって、経路選択確率は次に示される単体上の点として定義される.

$$\Theta\in\mathbb{S}^{M-1}=\{\Theta\in\Re_+^M\,|\pi_p^k\ge 0,\sum_{p\in\mathbf{P}}\pi_p^k=1\}$$

式(10)より、リンクコストそして経路コストは経路選択確率の関数になる.

さて、よく知られたWardrop均衡は次式で定義できる.

$$\begin{aligned} &For\ r\in\mathbf{P}\\ &h_r>0\Rightarrow u_r(\mathbf{h})\le u_s(\mathbf{h}),\ s\in\bar{\mathbf{P}}_k\ \ for\ all\ k\in\mathbf{M} \end{aligned} \tag{12}$$

一方，Dafermos[17]は(12)で定義される均衡よりも広義のネットワーク均衡概念を提案した. Bernstein and Smith[18]は下半連続のパフォーマンス関数を前提にDafermos均衡を次のように再定義している.

$$\begin{aligned} &For\ h_r>0, r\ne s\in\mathbf{P},\\ &u_r(\mathbf{h})\le\liminf_{\varepsilon\downarrow 0}\{u_s(\mathbf{h}-\varepsilon\mathbf{1}_r+\varepsilon\mathbf{1}_s)\} \end{aligned} \tag{13}$$

ここに，$\mathbf{1}_r$ はパスフローと同じ次元を持ち，r 番目要素が1で，その他の要素が0のベクトルである.

5. 交通ネットワークにおけるNash均衡

この論文では、混合戦略を用いて次のように利用者均衡を定義しよう。

$$\tilde{\pi}_r^k>0\rightarrow u_r^k(\tilde{\boldsymbol{\pi}})\le u_s^k(\pi^k,\tilde{\boldsymbol{\pi}}^{-k}) \tag{14}$$

紙面の都合上、詳細は割愛するが、上式はドライバー k の混合戦略は、他のドライバーの戦略が与えられた場合の経路コストに対し、純粋戦略をとる場合を端点とするシンプレックス上の点として定義でき、その内部に(13)を満足する解があることを示している。したがって、まず、混合戦略と純粋戦略の組み合わせを表す表記法を導入する。

$(a^k,\boldsymbol{\pi}^{-k})$：$i$ 以外のプレイヤーが同時混合戦略を行使するときに、プレイヤー i がとる純粋戦略。

$u^k(a^k,\boldsymbol{\pi}^{-k})$：$(a^k,\boldsymbol{\pi}^{-k})$ の行動空間でのプレイヤー i の期待コスト

すなわち、$u^k(a^k,\boldsymbol{\pi}^{-k})$ は、k 以外のドライバーがの混合戦略が与えられたとき、ドライバー k が純粋戦略である経路選択したときのドライバー k が知覚するコストを表している。(14)は、これらの純粋戦略の組み合わせとしての混合戦略に最小費用経路選択が存在することを意味している。無論、純粋戦略が解になる場合もある。通常、一人のドライバーの経路変更が経路コストに与える影響は無視できるほど小さいので、Nash 均衡と Wardrop 均衡は一致する。しかし、個々のドライバーの変更が無視できないような状況あるいはある属性を持ったドライバー集団を k と考えるときには、その集団の経路変更行動は無視できないものになる。Nash 均衡は、次式で与えられる。

$$u^k(\tilde{\boldsymbol{\pi}})=\min_{a^k\in A^k}\ u^k(a^k,\tilde{\boldsymbol{\pi}}^{-k}) \tag{15}$$

Fundenberg and Levine は確率錯乱項をもつ利得関数を導入し、これを最大にするようにプレイヤー k の混合戦略が求めている。このとき、最適対応は、

次式で与えられる。

$$\beta^k(\boldsymbol{\pi}^{-k}) = \arg\min_{\pi^k}\{u^k(\pi^k,\boldsymbol{\pi}^{-k}) + \lambda\eta^k(\pi^k)\}$$
$$= \arg\min_{\pi^k}\{\sum_{a^k\in A^k}\pi^k(a^k)u^k(a^k,\boldsymbol{\pi}^{-k}) + \lambda\eta^k(\pi^k)\}$$

確率錯乱項を表す関数として

$$\eta^i(\pi^i) = \sum_{\pi^i}\pi^i(a^i)\log\,\pi^i(a^i)$$

を仮定すると、よく知られたロジット選択公式が得られる。均衡状態は、次の不動点問題になる。

$$\tilde{\pi}^k = \beta^i(\boldsymbol{\pi}^{-k})$$

$u^k(a^k,\boldsymbol{\pi}^{-k})$ は相手の行動の推測を前提にしたプレイヤー k の期待コストである。繰り返しゲームでは、同じゲームを反復して行うため、ゲームの履歴を通して学習を行っていると仮定することは自然である。このとき、プレイヤー k は $u^k(a^k,\boldsymbol{\pi}^{-k})$ ではなく、これを 1 つの実現値とし、これまでの経験の集積であるQファクターとの凸結合によって、時点 t での期待コストの推定値 $Q_t^k = Q(a_t^k)$ を見積もり直し、それに基づいて行動を行う。このときの最適対応は、次式で表わされ,

$$\beta^k(Q) = \arg\min_{\pi^k}\{\sum_{a^k\in A^k}\pi^k(a^k)Q(a^k) + \lambda\eta^k(\pi^k)\}$$

次のロジット式が得る[11]。

$$\beta^k(a_{t+1}^k) = \frac{\exp[Q(a_{t+1}^k)/\lambda]}{\sum_{a_{t+1}^k}\exp[Q(a_{t+1}^k)/\lambda]} \qquad (16)$$

この式は,Nash-Q 関数を利用する点を除けば,.Miyagi[9~10]が提案するモデルと同じである.また,強化学習理論におけるGibbsサンプリングを行動論的に誘導したモデルとなっている.このように，本研究で提案する方法は、(8)に示したQ学習の最適選択とは異なる結果を与える。さらに,本研究では、Miyagi[11] とは異なり,時間スケールの異なる確率近似式[19]を用いる方法を検討する。すなわち、早く変化する項として(9)に指示関数で重み付けた経路評価値を用い、遅れて変化するパラメータとして経路選択確率を選ぶ。

$$\pi_p^k(t+1) = \pi_p^k(t) + \gamma_2(t)(\beta_p^i(t) - \pi_p^k(t)) \qquad (17)$$

この方法の収束性は,Borker の確率近似理論に基づく.

参考文献

1) Bellamn, R. (1958): On a routing problem, *Quart. Appl. Math.,* XVI, 87-90.
2) Howard, R.A. (1960): *Dynamic Programming and markov Processes*, Wiley,NY.
3) Puterman, M.L. (1994): *Markov Decision Process*, John Wiley&Sons.
4) Bertsekas, D.P. and Tsitsiklis, J.N. (1991):An analysis of stochastic shortest path problems, Math. of Opns.Res.,16,580-595.
5) Bertsekas, D.P. and Tsitsiklis, J.N. (1996):Nuro-Dynamic Programming, Athena Scientific.
6) Kushner, H.J., and D.S. Clark (1978): *Stochastic Approximation Methods for Constrained and Unconstarained Systems,* Springer and Verlag.
7) Borker, V.S., and K. Soumyanath (1998): An analog parallel scheme for fixed point computation-Part I:Theory. *IEEE Transactions on Circuit and Systems*, Vo.44.
8) Watkins, C.J.C.H. and P. Dayne(1992): Q-learning, *Machine Learning*,3,279-292.
9) Miyagi, T. (2004a): A modeling of route choice behaviour in transportation networks: An approach from reinforcement leaning, *Urban Transport X*, WIT press, UK, pp.235-244.
10) Miyagi, T. (2004b): A reinforcement learning model with endogenously determined learning-efficiency parameters, *The Proceedins of CIS/SIS Conference*, Keio University.
11) Miyagi, T.(2005): A Stochastic fictitious plays, reinforcement learning and user equilibrium, A paper submitted to 'Mathematics in Transport', University College of London.
12) Fundenber, D., and Kreps, D. (1993): Lerning mixed equilibria, *Games Econ. Behavior*, 5, 320-367.
13) Fundenberg, D., and Levine, D. (1998): *Theory of Learning in Games*, MIT Press.
14) Robbins, H., and Monro, S. (1951): A stochastic approximation method, *Ann. Math.Statist.* 22, pp.400-407.
15) Benaim, M. and M.W. Hirsh(1999): Mixed eqilibria and dynamical systems arsing from fictitious play in purturbated games, *Gamnes and Economic Behavior* 29, pp.36-72.
16) Hu, J., and M.P. Wellman(1998): Multiagent reinforcement learning: Theoretical framework and an algorithm, In *15th International Conference on Machine Learning*,242-250, Madison,WI.
17) Dafermos, S.,(1968): *Traffic Assignment and Resource Allocation in Transportation Networks*, Ph.D Thesis, The Johns Hopkins University.
18) Bernstein, D., and T.E. Smith, (1994): Equilibrium for networks with lower semicontinuous costs: With an application to congesting pricing, *Transportation Science*, 28(3),pp. 221-235.
19) Borker, V. S. (1997): Stochastic approximation with two timescales, *Systems Control Lett*. 29, 291-294.