LC学習:モデルに基づく段階的平均報酬強化学習手法

山口 智浩* 誉田 太朗† 天正 新二郎‡

LC-Learning：In-Stages Model-Based Average Reward Reinforcement Learning

Tomohiro YAMAGUCHI, Taro KONDA and Shinjiro TENSYO

This paper presents a novel model-based average reward reinforcement learning method to compute a bias-optimal policy in a cyclic domain. Most previous methods culculate the policy updating the utilities with successive approximation. However, they never acquire the optimal policy due to some finite error. In addition, a theoretical proof of the policy convergence is also a difficult task. All methods deriving from the Bellman equations contain these defects potentially. Addressing these problems, we show a key notion that only the stationary cycle in a unichain policy is concerned to calculate a maximized average reward. Following this idea, we present a new straightforward method called LC-learning. It performs much better than the Prioritized Sweeping well known as an effective discounted method. Furthermore, it filters a more selected policy, bias-optimal one, with additional necessary-and-sufficient cost.

1 はじめに

強化学習は，エージェントが環境において獲得できる報酬を最大化する政策を求めることを目指す．この環境は，しばしばマルコフ決定過程として表現される[1][8]．一般的に，強化学習の手法として，各規則の効用値として割引期待報酬和を最大化するQ学習が知られている[10]．しかしながら割引型手法は，「最適政策を必ず獲得するためには非常に大きな回数の反復計算を必要とし」，逆に「反復回数を減らそうとすれば，最適政策の獲得が保証されなくなってしまう」という根本的なジレンマを抱えている．この問題を解決するために提案された優先掃き出し法[5]は，割引型手法の計算コストの改善には成功したが，獲得政策の最適性と計算効率とのバランスは，パラメータに大きく依存しており，適切な学習性能を発揮するにはそれらの調整を必要とする．

平均報酬強化学習法は，効用値の計算に割引率を用いないのが最大の特徴である．R学習は最初の非割引型の強化学習手法であり[7]，割引型のQ学習よりも高速な最適政策の獲得が確かめられている[2]．この平均報酬法という枠組みは近年広く研究され，MDPモデルに基づく手法としてa Model-based Bias-Optimality Algorithm[2]やH学習[9]が，MDPモデルに依らない手法としてa Model-Free Bias-Optimality Algorithm[4]などが提案されている．しかしながら，これらの手法が必ず最適政策を獲得するという論理的証明が困難なため，学習性能は実験的に評価されてきた．これらの政策収束における複雑さと曖昧さの原因は，MDPモデルの有無に関わらず，最適性基準を表す方程式の解を逐次近似計算で求めている点にある．以上より，既存の平均報酬強化学習手法は，学習の最適性と計算効率との両立が実現されていない．

これに対し，本論文で我々の提案する**LC学習**は，エルゴディックMDPにおける単鎖政策の特性を考慮し，上述の問題を解決する．単鎖政策は，**定常サイクル**と**遷移パス**によって構成されるが，本手法ではこれらを分割統治する．本論文では，まず既存の平均報酬法の最適性を，定常サイクルと遷移パスそれぞれの最適性へと変換することで論理的な基礎を築き，学習の最適性を保証するLC学習のアルゴリズムを示す．次に,本手法の計算効率を,割引型の最速手法の一つである優先掃き出し法との学習実験により比較し，本手法の有効性を論理的，実験的に明らかにする．

* 情報工学科 † 現京都大学情報工学科
‡ 専攻科 電子情報工学専攻

注1）第29回知能システムシンポジウム
2002年3月28日，口頭発表論文を加筆修正

2　政策の最適性基準

平均報酬強化学習の獲得政策を評価するために，2つの最適性基準が定義されている[3]．本章では，これらを定常サイクルと遷移パスを示す要素へと分離する．

2.1　平均報酬最適性

無限ステップにおける，平均報酬ρを最大化する政策を，平均報酬最適な政策と呼ぶ．ρは以下のように定義される[3]．

$$\rho^{\pi}(s) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=0}^{N-1} r_t^{\pi}(s) \tag{1}$$

ここで，$\rho^{\pi}(s)$は状態sから出発したエージェントが，政策πに従って行動したときの平均報酬を，$r_t^{\pi}(s)$は状態sから出発したエージェントが，政策πに従って行動したときにt時間後に得られる報酬の値を示す．

本研究では，**定常サイクル**を考慮することにより，式(1)を以下のように変形する．

$$\rho^{\pi}(s) = \frac{1}{l} \sum_{T=0}^{l-1} r_T^{\pi}(s') \tag{2}$$

ここでlは定常サイクルの長さであり，状態s'はエージェントが定常サイクルに到達したときの状態を示す．時間Tはエージェントが定常サイクルに到達したときに0に初期化される．

式(2)は，平均報酬最適な政策を求めるには，平均獲得報酬が最大の定常サイクルのみを求めれば十分であり，遷移パスを考慮する必要がないことを示している．

2.2　バイアス報酬最適性

平均報酬最適性では，定常サイクルに含まれない状態の行動を決定することができない．この政策決定を行うための最適性基準がバイアス報酬最適性である．この政策をバイアス報酬最適な政策と呼ぶ．これはバイアス値$V^{\pi}(s)$を最大化する．$V^{\pi}(s)$は以下のように定義される[3]．

$$V^{\pi}(s) = \lim_{N\to\infty} \sum_{t=0}^{N-1} (r_t^{\pi}(s) - \rho^{\pi}) \tag{3}$$

本研究では，**定常サイクル**と**遷移パス**を考慮することにより，式(3)を以下のように変形する．

$$V^{\pi}(s) = V_P^{\pi}(s) + V_C^{\pi}(s') \tag{4}$$

$V_P^{\pi}(s)$を**パスバイアス値**と呼び，式(5)で表す．これは，ある遷移パス中の状態で獲得されるバイアス値を示している．

$$V_P^{\pi}(s) = \sum_{t=0}^{n-1} (r_t^{\pi}(s) - \rho^{\pi}) \tag{5}$$

$V_C^{\pi}(s')$を**サイクルバイアス値**と呼び，式(6)で表す．これは，定常サイクル中の状態におけるバイアス値に対応する．

$$V_C^{\pi}(s') = \frac{1}{l} \left(\sum_{T=0}^{l-1} (r_T^{\pi}(s')(l-T)) - \frac{l(l+1)\rho^{\pi}}{2} \right) \tag{6}$$

これは，バイアス値は定常サイクル中の状態によって変化し，それぞれが個別に算出可能であるという事を示している．

3　LC-Learning

式(3)，(4)，(5)，(6)を基にして，LC学習のアルゴリズムを提案する．この手法は以下の手順でBias-Optimal政策を効率良く求める．

3.1　アルゴリズム

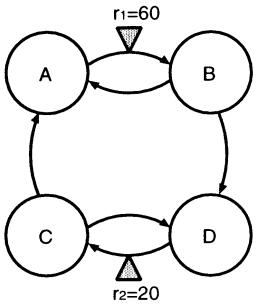

図1：The MDP with two rewards

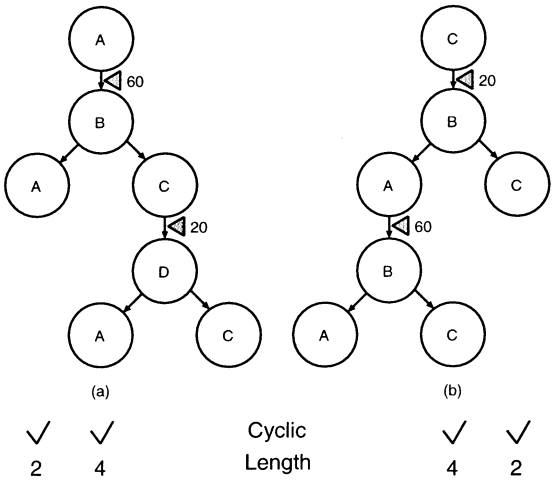

図2：The tree strctures converted from the MDP

1. 全ての報酬獲得サイクルの検出

全ての報酬獲得規則を根として，MDPを木構造に変形することで，全ての定常サイクルを求める．例を挙げると，図1に示した4状態6規則2報酬のMDPは，図2の様に2つの木構造へと変換され，図3に示されるように，3つの報酬獲得サイクルが検出されている．

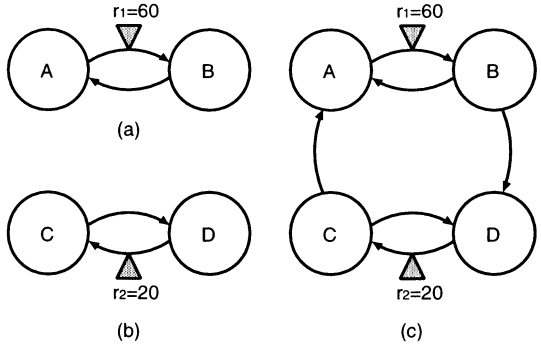

図3：The stationary cycles in the MDP

2. 最適サイクルの決定

検出された全ての報酬獲得サイクルの中で，最大の平均報酬を持つサイクルを式(2)に従い求める．このサイクルを**最適サイクル**と呼ぶ．図3においては，(a)が最適サイクルとなる．

3. サイクルバイアス値の評価

最適サイクルに含まれる全ての状態について，サイクルバイアス値を式(6)に従い求める．

4. 遷移パスにおけるバイアス値の計算と行動の決定

最適サイクルにおけるサイクルバイアス値を，Backward Induction[6]を応用することによって伝播させ，遷移パス上の行動を決定する．これは式(5)を基に以下の手順で行われる．

(a) 最適サイクルに含まれる全ての状態をS_yとし，それ以外の状態をS_xとする．

(b) 一つの行動で遷移可能な全ての状態がS_yに含まれているS_xの要素xについて，

$$V^{\pi^*}(x) = \max_{y \in s_y}(V^{\pi^*}(y) + r_{xa}) - \rho^{\pi^*}$$

この時の行動aを状態xにおける政策とし，S_xからxを取り除き，S_yに加える．ここでπ^*は最適政策を表す．

(c) S_xが空ならば終了．そうでなければ(a)へ．

3.2 計算量の改善と見積もり

全ての報酬獲得サイクルを検出するために要する計算量を削減するために，木展開に以下の2つの枝狩り手法を適用する．

・定常サイクルの多重検知防止

図2では，同一の定常サイクルが重複して検出されている．これは，報酬獲得規則に順序をつけて展開を行い，既に展開した報酬獲得規則で展開を止めることで実現できる．

・平均報酬の上限値予測に基づく木展開の打ち切り

ある節において展開を続けた場合の平均報酬の最大値を予測し，その値がそれまでに得られた定常サイクルの最大の平均報酬を越えなければ，その節における展開を止めることができる．図2においては，(b)における最初の状態Cにおいて展開打ち切りが可能である．

これらの手法を実装したLC学習の計算量を推定する．LC学習の最悪ケースの一つと考えられる，全ての状態が直接通信している完全グラフのようなMDPにおいて，規則数を変化させた場合(表1)，報酬数を変化させた場合(表2)について計算時間と計算空間を測定した．共に線形性が確認でき，計算量を$O(|r||S \times A|)$と見積もることができる．ここで$|S|$は状態数，$|A|$は行動数，$|r|$は報酬数，$|S \times A|$は規則数に対応する．また*Steps*は計算にかかった時間を，*Max queue*は必要としたメモリ空間を表している．

表1：The result varing the number of the rules

$\|S\|$	$\|r\|$	$\|S \times A\|$	Steps	Max queue
4	2	16	29	2
6	2	36	151	13
8	2	64	409	33

表2：The result varing the number of the rewards

$\|S\|$	$\|r\|$	$\|S \times A\|$	Steps	Max queue
8	2	64	409	33
8	4	64	1467	122
8	6	64	4533	361

3.3 優先掃き出し法との比較学習実験

LC学習の性能を測定するために，優先掃き出し法(Prioritized Sweeping)との比較実験を行う．優先掃き出し法は，割引型手法において最も高速な手法の一つである．実験環境は図4で示される観光バス問題である．これは信号もしくは横断歩道を状態，道路を行動，ある観光地へ行きたい観光客の人数を報酬とし，最も効率よいプランを探索する問題である．これは直ちにMDPへと変換され，$|S \times A| = 112$，$|r| = 4$である．優先掃き出し法の割引率γは一般的な0.99，0.90とした．まずLC学習のMDPモデルの参照回数と優先掃き出し法の効用値の更新回数を比較した結果を表3に示す．どちらの割引率に対しても，LC学習が良い性能を示していることが分かる．次に図5に，時間経過と共に獲得された定常サイクルの平均報酬の値を示す．このタスクでは，合計8つのサイクルが検出された．568ステップ目で平均報酬1.1667の最適サイクルが検出され，最大値予測による打ち切りにより，程なく展開が終了しているのが分かる．最後に図6に，展開待ち

の節が格納された待ち行列の大きさの時間遷移を示す．567ステップ目で大きさが72で最大となり，817ステップ目で終了するまで減少している．

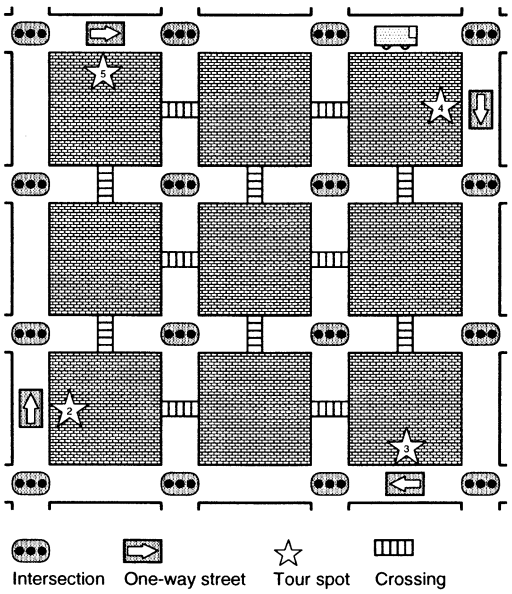

図4：The Sightseeing Bus Tour Problem

表3：The result comparing with Prioritized Sweeping

	γ	Cost of Time
LC 学習	-	817
優先掃き出し法	0.990	22352
	0.900	2192

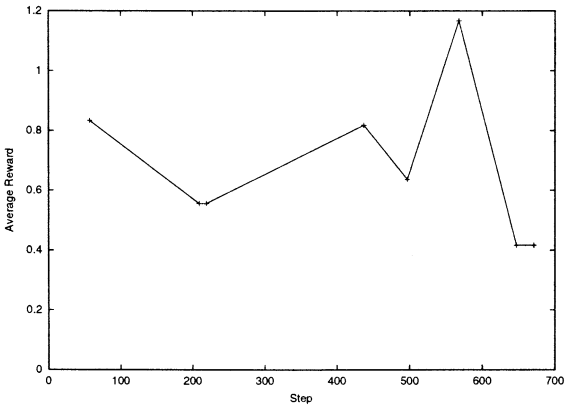

図5：The average rewards of the found cycles

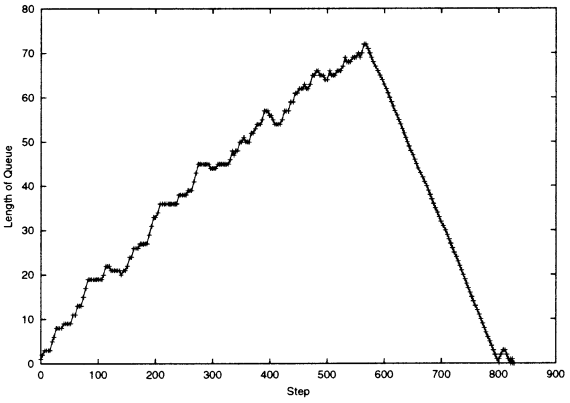

図6：The changes of the queue size

4　結論

本研究では，既存の平均報酬に基づく最適性に即した政策を，厳密かつ効率的に求める新しい平均報酬強化学習法であるLC学習を提案した．その獲得政策の最適性を論理的に明らかにした上で，学習コストを割引型の優先掃き出し法と実験的に比較し，本手法が獲得政策の最適性と計算効率とを両立させた優れた手法であることを示した．この手法の特徴は，単鎖政策は定常サイクルと遷移パスに分けられるという考えに基づき学習を行い，まず最適サイクルを求め，次にそのバイアス値を伝播することによってバイアス報酬最適な政策を獲得する点である.特に本手法は，本来の強化学習が仮定している報酬の少ない環境でより良い性能を示す．

5　今後の課題

LC-Learningに対する今後の課題を以下に示す．

・確率的環境への応用
・他の平均報酬法[2]との比較実験
・オンラインでのモデル同定機能の実装
・報酬が時間と共に変化する動的環境での性能テスト
・報酬の局所分布に対応した階層的手法の適用
・節に優先度をつけた探索手法の適用

参考文献

[1] Leslie P. Kaelbling and Michael L. Littman and Andrew P. Moore: Reinforcement Learning: A Survey, *Journal of Artificial Intelligence Research*, 4, pp.237-285, 1996

[2] Sridhar Mahadevan: An Average-Reward Reinforcement Learning Algorithm for Computing Bias-Optimal Policies, *Proceedings of the Thirteenth AAAI (AAAI-1996)*, pp.875-880, 1996

[3] Sridhar Mahadevan, Average Reward Reinforcement Learning: Foundations, Algorithms, and Empirical Results, *Machine Learning*, 22 (1-3), pp.159-195, 1996

[4] Sridhar Mahadevan, Sensitive-discount optimality: Unifying average-reward and discounted reinforcement learning, *Proceedings of the Thirteenth International Conference on Machine Learning (ICML-1996)*, pp.328-336, 1996

[5] Andrew W. Moore and Christopher G. Atkeson, Prioritized Sweeping: Reinforcement Learning With Less Data and Less Time, *Machine Learning*, 13, pp.103-130, 1993

[6] Martin L. Puterman, *Markov Decision Processes: Discrete Dynamic Stochastic Programming*, John Wiley, pp.92-93, 1994

[7] Anton Schwartz, A Reinforcement Learning Method for Maximizing Undiscounted Rewards, *Proceedings of the Tenth International Conference on Machine Learning (ICML-1993)*, pp.298-305, 1993

[8] Richard S. Sutton and Andrew G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998

[9] Prasad Tadepalli and DoKyeong Ok, Model-Based Average Reward Reinforcement Learning, *Artificial Intelligence*, 100 (1-2), pp.177-223, 1998

[10] Christopher J. Watkins and Peter Dayan, Q-learning. Machine Learning, *Machine Learning* 3 (8), pp.279-292, 1992