

転移学習における方策再利用法の技法と比較

河野 仁 *1

Comparison of policy reusing mechanisms in transfer reinforcement learning

Hitoshi Kono

In recent year, the robot systems with learning algorithms are deployed in the real world situation, for example, automatic driving car, warehouse robots and so on. Especially, reinforcement learning (RL) don't need the supervised data, and RL can explore the optimal solution by it self. However, the robot or virtual agent (called agent) which is implemented RL, can be only acquired a optimal or near optimal solution that is focused on specific tasks. Hence, a transfer learning frame work was proposed for the past dozen years or so to discuss the application capability of reinforcement learning. This paper introduce methods of transferring policy which focused on Tylor's transfer learning mechanisms, and this paper is discussed difference of effectiveness of policy reusing mechanisms which are proposed by related work.

1 諸言

近年、知能ロボットの実世界応用の研究・開発が活発になされ、学習アルゴリズムが実装されたロボットシステムの研究も多い。さらに、自分で試行錯誤的に解を獲得することができる強化学習は、ヒトによる教示や大量の訓練データを用意する必要がなく、メリットが多い [1-4]。しかし、強化学習は多くの学習の繰り返しが必要であることや、学習時間が長いことが課題として指摘される。そのため、過去に獲得した知識（強化学習の方策）を新たな環境やタスクでの学習に再利用し、学習時間の短縮や高パフォーマンスを獲得しやすくする転移学習が提案されている [5]。特に近年では、複数のエージェントが同時に転移学習する方法や、方策の再利用度合いを調整して新たな環境での適応力を向上させる手法など、多岐にわたる関連手法が提案されている [6-9]。これらの手法は様々な環境で有用性を確認されているが、条件設定や手法が細分化され、性能の比較を行うことが難しい。また実装レベルの議論になると、転移学習という大枠の概念は共通理解できていても、技法ともいえる実装の細部が開発者によって異なる可能性が高い。そのため、パラメータの調整による手法への影響が見えにくい。

そこで、本稿では方策再利用法の実装をいくつか議論し、計算機実験の結果から効果の違いを議論する。また、転移を行うことでエージェントが新たな環境に適応できず、本来の転移学習の効果が得られないことを Negative transfer（負の転移）と呼ぶが、仮に負の転移が発現したエージェントが、時間が必要であるとしても再学習により新たな環境に適応することも重要であり、新たな手法を提案せず従来のパラメータ調整に回避可能であるかも議論する。

以下、2 章では方策再利用法の議論に必要な基礎的知識として、強化学習や転移学習などの関連技術を述べる。3

章では、本稿で実施する計算機実験の条件を述べ、4 章では計算機実験による結果と考察を議論する。最後に 5 章では、本稿のまとめと今後の課題を論じる。

2 方策再利用法

本章では、議論の前に必要となる強化学習や転移学習などの基礎的な知識、前提条件に関して以下に述べる。

2.1 強化学習

強化学習は、エージェントが環境から与えられる報酬をもとに、試行錯誤的に行動を繰り返し最適解を獲得するアルゴリズムであり、動物行動心理学などの知見から着想を得ている [10]。本稿では Q 学習を採用し、議論を進める。エージェントは環境の状態集合 S から状態 s を観測でき、それを共に行動集合 A から行動 a を選択し、実行することが可能である。エージェントは目的を達成すると報酬 r が獲得でき、次式により方策 $Q(s, a)$ を獲得する。

$$Q(s, a) \leftarrow Q(s, a) + \alpha \{ r + \gamma \max_{b \in A} Q(s_{t+1}, b) - Q(s_t, a) \} \tag{1}$$

ここで、 α は学習率で ($0 < \alpha \leq 1$)、 γ は割引率 ($0 \leq \gamma \leq 1$) である。方策 $Q(s, a)$ は、基本的にすべての状態 s とすべての行動 a における行動価値が記された Look-up-table で構成されており、Q-テーブルと呼ばれる。観測可能な状態数が増加するほど $Q(s, a)$ を保存するコンピュータ内のメモリスペースが指数関数的に増大する「次元の呪い」が発生する。そのため、 $Q(s, a)$ を関数近似により表現する方法が多く提案されているが、近似誤差による学習への影響など、不確定な要素が増えるため本稿では取り扱わない。

方策により行動を選択するための手法として、greedy 法や soft-max 法など様々な方法があるが、本稿では確率的

*1 東京工芸大学工学部電子機械学科

な行動選択を実現する、次式に示す Boltzmann 分布をもとにした soft-max 法を採用する。

$$P(a|s) = \frac{\exp\{Q(s,a)/T\}}{\sum_b \exp\{Q(s,b)/T\}} \quad (2)$$

ここで、 T は温度定数と呼ばれ、行動価値の高い行動の選択のしやすさを調整するパラメータであり $0 < T < 1$ とする。

2.2 転移学習

転移学習は、転移元 (Source task) のエージェント (広義の転移学習の場合ヒトも含む) が学習した知識を、転移先 (Target task) のエージェントが再利用するフレームワークである。転移学習という用語自体は発達心理学などの分野で使われているが、工学分野全般における転移学習は、神宮による文献 [11] が詳しい。近年は機械学習をはじめ強化学習でも応用され成果を得ている [12]。

強化学習における転移学習では、強化学習エージェントが Source task にて学習を行い方策を獲得する。その後、同一もしくは類似環境である Target task において、同一もしくは類似エージェントが Source task で獲得された方策を再利用し、Target task での解の獲得速度を速めたりすることが可能である。

転移学習における方策の再利用方法として、代表的であり直感的な理解をしやすい方法は知識複写と知識修正であると考えられる [13]。知識複写は次式により形式的に定義できる。

$$Q_t(s,a) \leftarrow Q_s(s,a) \quad (3)$$

ここで、 $Q_s(s,a)$ は Source task から転移された方策であり、 $Q_t(s,a)$ は Target task にてエージェントが使用する方策であり、さらに新たな環境で学習した行動価値もここに更新される。注意が必要なのは、他の文献では明示的に Q_t や Q_s と区別しない場合が多い。

知識修正は、Target task で方策を再利用する際、再利用しやすくするように Source task と Target task 間における環境情報や行動の対応を定義してから方策再利用する手法である。この対応付け処理を Inter-task mapping (ITM) と呼ぶ。式 (3) とは異なり、次式にて定義できる。

$$Q_c(s,a) \leftarrow Q_t(s,a) + Q_s(s,a) \quad (4)$$

ここで、Target task のエージェントが使用する方策は $Q_c(s,a)$ であり、河野らは統合方策と呼んでいる [14]。これは方策再利用前に、 $Q_t(s,a)$ の初期値と再利用方策 $Q_s(s,a)$ を統合している。Target task にて学習した行動価値は $Q_t(s,a)$ に更新され、行動選択は $Q_c(s,a)$ を用いて実行される。すなわち、 $Q_s(s,a)$ は常に参照されるだけの方策となり、Target task での行動価値更新などの処理は実行されない。

ITM を考慮すると、式 (4) は次式のように定義される。

$$Q_c(s,a) \leftarrow Q_t(s,a) + Q_s(\chi_s(s), \chi_a(a)) \quad (5)$$

関数 $\chi_x(\cdot)$ は、Target task における S と A のそれぞれの元を、Source task の S と A のそれぞれの元に対応関係を記述する手法である。すなわち、Source task と Target task のエージェントにおける身体性の違いとその対応関係を定義している。これにより、異なるエージェント間での転移学習が効果的となる場合がある。

2.3 他の方策再利用メカニズム

強化学習における方策の再利用には、前述のとおり Tyalor らの方法が基本形であると考えられる。しかし、近年では方策再利用時に再利用知識における学習済みの行動価値が高く、Target task での環境適応性能に悪影響を及ぼすという指摘もある [7,9,15]。そのため、次式のような再利用度合いを調整する方法が提案されている。

$$Q_c(s,a) \leftarrow Q_t(s,a) + \zeta Q_s(s,a) \quad (6)$$

式 (6) では、ITM の記述を省略している。 ζ の値を調整することで、行動価値の高さを調整し Target task における探索を促す効果がある。 ζ は定数を与える方法や学習進度に応じて値を減少させる手法などがあるが、本稿では定数項として扱うこととする [14]。

3 方策再利用法の比較条件

本稿では、いくつか提案されている転移学習の知識再利用法の具体的なメカニズムにおいて効果の比較を行う。比較の指標としては以下に示す項目を評価する。

- ・ 転移学習としての効果
- ・ Target task における環境適応性

転移学習の基本的な効果が得られていることを評価するために、「正の転移」、「負の転移」、「零の転移」3つの用語を定義する。正の転移は、エージェントが方策を再利用することで新たな環境において最適解の獲得の高速化が行える状態である。負の転移は、エージェントが方策を再利用することで、環境に適応できず解の獲得に長時間かかるような状態である。零の転移は、方策を再利用しても Target task における学習に、なんら影響のないような状態である。

すなわち、転移学習の効果が得られたと判断できる結果は、正の転移が発生している状態である。また、Target task における環境適応性能を評価する基準としては、デッドロックを考慮する。方策を再利用し、その方策に依存した意思決定を行うことで、新たな環境での行動を行えなくなる状態をデッドロックとする。

また、実験結果の比較するための数値として、学習曲線を採用する。学習曲線は横軸に学習の繰り返し回数、縦軸にパフォーマンスを取る曲線のグラフである。これによ

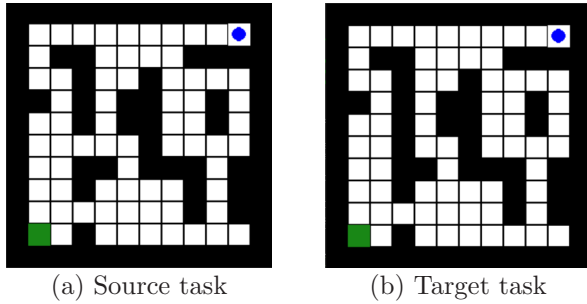


Fig. 1 Environmental settings of source task and target task

り、エージェントが学習を繰り返すことでどれくらいのパフォーマンスを獲得できているかの確認が可能である。学習曲線の取得は、Source task や Target task での学習をそれぞれ 10 回の施行し、平均としたものを学習曲線として描画する。

3.1 環境

本稿では、強化学習タスクとして単純な最短経路問題を採用し、結果の直感的な理解を助ける。また、環境としては図 1 に示すグリッドワールドを設定する。図 1 中の青い円がエージェントであり、初期位置は固定とする。緑色で塗りつぶされたグリッドがゴールであり、白いグリッドはエージェントが移動可能な通路である。

図 1 における環境の違いは、障害物の配置である。Target task では、Source task の最短経路に障害物が配置されており、Source task の最短経路が再現せず、迂回しなければならないように設定されている。

また、エージェントは本実験では Source task と Target task で同一エージェントを採用する。これにより、知識修正の場合は ITM の調整具合が結果に影響しない。すなわち、本稿において Source task と Target task において異なる要素は、環境の障害物配置だけである。

なお、エージェントはグリッド上の上下左右と停止の計 5 種類の行動が選択可能であるとする。

3.2 学習パラメータ

強化学習のための学習パラメータとして、全実験で学習率 $\alpha = 0.1$ 、割引率 $\gamma = 0.99$ 、ゴール報酬 $r_g = 1.0$ とする。また、転移学習のパラメータとして転移率 ζ は、環境に応じて適宜変更し、事前に調整された負の転移にならないパラメータに設定する。

Source task における学習エピソード数は 500 とし、Target task における学習エピソード数も 500 とする。行動選択に用いる soft-max 法の温度定数 T は 0.05 とする。再利用される方策を除く、全ての新たに生成される Q-テーブルの初期値は 0 とする。

3.3 比較に用いる転移学習の技法

本稿の実験では、以下に示す転移学習の実装を行った。

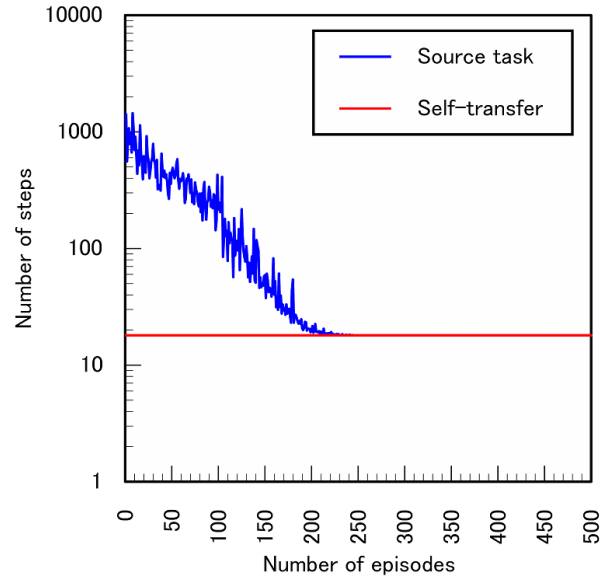


Fig. 2 Learning curve in source task and self-transfer

- ・ 知識複写
- ・ 知識修正
- ・ 転移率を導入した知識修正

前述のとおり、知識複写は再利用する方策を Target task にて学習する方策に一度コピーし、それを用いて意思決定や行動価値の更新を行う。知識修正において、本稿では Source task と Target task において同一エージェントを採用していることから、式 (4) と同等である。

4 実験結果と議論

4.1 Source task における強化学習と方策の生成

最初に、Source task にて正常に方策の獲得がなされているか評価を行う。Source task を実行して得られた学習曲線を図 2 に示す。

図中の学習曲線は横軸にエピソード数、縦軸にゴールまでに要したステップ数としている。Source task では試行錯誤の後、250 エピソード付近で最短経路の 18 ステップに収束していることが見て取れる。

さらに、Source task で獲得した方策が正確に保存されているか確認するために、Source task と同環境でエージェントに方策を再利用させた。その結果が図 2 の Self-transfer である。この結果から、学習開始から最短経路でゴールに到達することが可能となっているため、理想的な正の転移が発現し、方策も正しく保存されていることがわかる。

4.2 知識複写

方策複写により転移学習した場合の学習曲線を図 3 に示す。本実験条件から、方策を再利用するエージェントは Target task で学習を行っている。

知識複写では、学習初期が高いステップ数となっており、

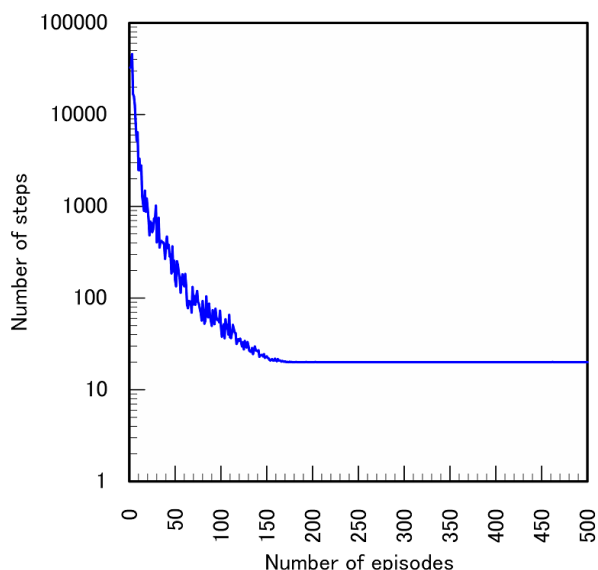


Fig. 3 Learning curve in condition with policy copy

負の転移が発現していると考えられる。しかし、エージェントは Source task の方策を基に行動し、最短経路に配置された障害物に衝突しておりステップ数の増加が発生しているが、強化学習の割引率の効果により、徐々に障害物への衝突が減り Target task の最短経路が獲得出来ている。

Target task への環境適応は可能となっているが、最短経路を獲得するまでに再利用する方策が悪影響を及ぼしていると考えられる。ただし、環境によっては知識複写の方法でも正の転移が得られる可能性があると考えられる。

4.3 方策修正

方策修正では、エージェントが Target task において最短経路を妨害する障害物に衝突し、デッドロックした。このデッドロックにより、ゴールへの到達ができなため、報酬も受け取れず方策の改善、すなわち再学習も行えない。学習曲線は得られていないため、グラフは示さない。

そこで、デッドロックを防止するために、エージェントが障害物に衝突した場合、負の報酬 ($R_c = -1.0$) を与える設定としたところ、方策改善がなされ、図 4 に示す学習曲線が得られた。図 4 の Without transfer の曲線は、比較のために行った Target task の環境を一から強化学習した学習曲線である。

これにより、Source task で獲得された方策を再利用することで、Target task の最短経路も獲得しやすくなっている正の転移が発現している。

しかし、本実験により方策修正の手法では方策の転移時に注意しなければいけない知見が得られた。知識修正は再利用する方策に対して行動価値の更新を行わないため、Target task にて最短経路を獲得したとしても、学習進捗が浅い場合は学習途中の方策を保存することとなる。その場合の学習曲線を図 5 に示す。

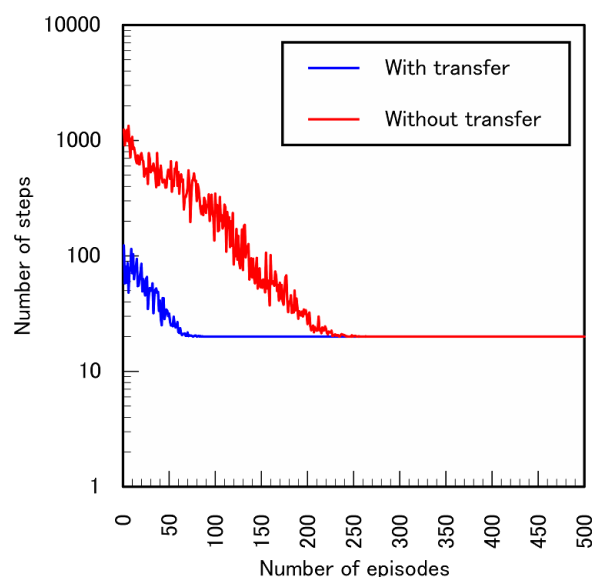
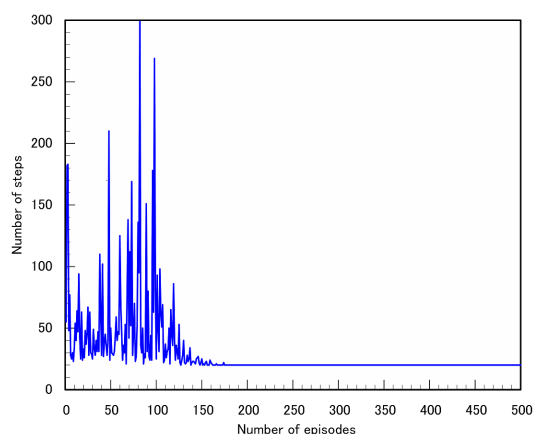
Fig. 4 Learning curve in target task with $\zeta = 1.0$, $R_c = -1.0$ 

Fig. 5 A kind of the case of negative transfer caused by 100 iteration.

図 5 では、Target task での学習エピソード数を 100 回に制限し、そこで獲得できた方策を同環境同エージェントへ Self-transfer した時の学習曲線である。Target task では再利用方策が存在したため、見せかけの最適解獲得が学習曲線から見て取れるが、実際には行動価値の更新が十分に行えておらず、学習初期は環境の探索を行う傾向が発現した。本実験では正の転移が発現しているが、このような状態は Target task での学習進捗によっては零の転移となるため、注意が必要であると考えられる。

4.4 転移率を用いた知識修正

転移率を用いた知識修正では、あらかじめ再利用する方策を ζ で割引いておくことで、環境適応性能を向上させる方法である。本実験では $\zeta = 0.1$ とした。これにより得られた学習曲線を図 6 に示す。図 6 においても、図 4 と同

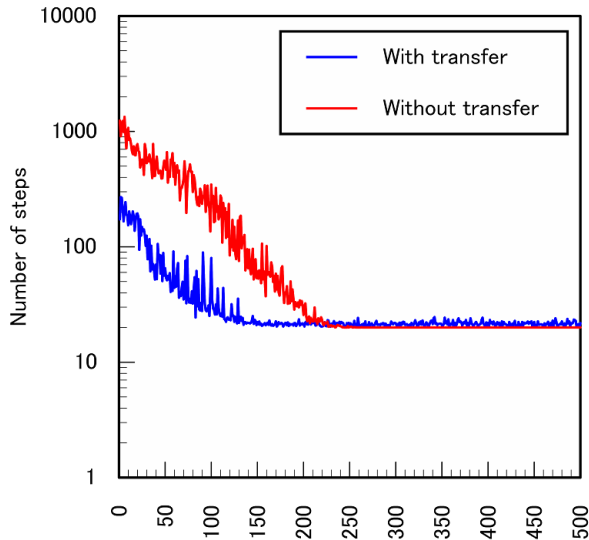


Fig. 6 Learning curve in target task with $\zeta = 0.1$, $R_c = 0$

様に Without transfer は Target task の環境において一から強化学習を行った場合の学習曲線である。

転移率をあらかじめ調整することで，“負の報酬を導入した知識修正”と同様の正の転移が発現した。しかし、今回の ζ のパラメータ設定では図 4 程の高速な解の獲得とならず、さらには最短経路という静的環境にも関わらず解が一意に収束しなかった。学習曲線は 10 回の学習で得られた曲線の平均であるため、それぞれの学習曲線を比較すると、10 回中 7 回は最短経路へ収束する学習曲線であり、残りの 3 回は多少の振動を伴う、準収束であった。

今回は ζ を定数として実験を行い、また値も低めに設定したため、正の転移は発現しているが、図 4 程のパフォーマンスの獲得はできなかった。

5 結言

本稿は、Tyalor らの提案した強化学習における転移学習をもとに、いくつかの方策再利用法を議論し、計算機実験からそれらの効果を検証した。

実験結果から、知識複写では負の転移が発現し、Target task の最適解探索に時間を要した。しかし、時間は要するが環境適応は出来ているといえる。

知識修正では、Target task における Source task の最短経路を妨害する障害物にエージェントが衝突し続けてデッドロックし、明らかな負の転移となった。しかし、手法を修正し障害物衝突時はエージェントに負の報酬を与えることで、デッドロックを回避し、さらには正の転移が発現した。また、本実験条件では正の転移が発現したが、Target task にて最短経路が発現したとしても、学習を早めに打ち切った場合にはその方策を転移しても行動価値が十分に高くなっていない可能性があり、正の転移が得られにくいこ

とが明らかとなった。

転移率を導入した知識修正では、手法に修正を加えることなく正の転移が発現したが、負の報酬を導入した知識修正の手法の結果ほど、Target task における解の獲得スピード、最短経路への収束が表れず、転移率 ζ の微調整が必要であると考えられる。

転移学習のためのパラメータチューニングは環境やエージェントの違いに敏感であり、少なくとも知識複写のような手法では転移学習の効果が表れる場面は少ないと考えられる。単純な知識複写ではエージェントのデッドロックが発生するため、転移率や壁への衝突時の負の報酬が必要であるが、環境やエージェントのヘテロジニティに応じてパラメータを決定する方法論であったり、オートチューニング技術の開発が必要であると考えられる。

謝辞

本研究は、公益財団法人中山隼雄科学技術文化財団、平成 28 年度研究助成 (A-1) による助成を受けて実施された。

参考文献

- [1] R. D. Andrea, “Guest Editorial: A Revolution in the Warehouse: A Retrospective on Kiva Systems and the Grand Challenges Ahead”, IEEE Transactions on Automation Science and Engineering, vol.9, no.4, pp.638–639, 2012.
- [2] M. J. Matarić, “Reinforcement Learning in the Multi-Robot Domain”, Autonomous Robots, vol.4, pp.73–83, 1997.
- [3] E. Yang and D. Gu, “A Survey on Multiagent Reinforcement Learning Towards Multi-Robot Systems”, IEEE 2005 Symposium on Computational Intelligence and Games, 2005.
- [4] J. Kober, J. A. Bagnell, and J. Peters, “Reinforcement learning in robotics: A survey”, The International Journal of Robotics Research, vol.32, no.11, pp.1238–1274, 2013.
- [5] M. E. Taylor, “Transfer in Reinforcement Learning Domains”, Springer-Verlag Berlin Heidelberg, 2009.
- [6] A. Taylor, I. Duparic, E. Galván-López, S. Clarke, and V. Cahill, “Transfer learning in multi-agent systems through parallel transfer”, 30th International Conference on Machine Learning, Atlanta, 2013.
- [7] 田代淳史, 河野仁, 神村明哉, 富田康治, 鈴木剛, “ヘテロジニアス間転移学習のための知識再利用法の検討”, JSME Conference on Robotics and Mechatronics, 2A1-L06(CD-ROM), 2015.

- [8] 岩朝睦美, 久保田直行, “ヘテロジニアスマルチエージェント環境におけるファジィ Q 学習と共通行動の抽出を用いた転移学習”, 日本知能情報ファジィ学会第 31 回ファジィシステムシンポジウム, pp.258–261, 2015.
- [9] H. Kono, Y. Murata, A. Kamimura, K. Tomita, and T. Suzuki, “Knowledge Co-creation Framework: Novel Transfer Learning Method in Heterogeneous Multi-agent Systems”, Distributed Autonomous Robotic Systems, Springer Tracts in Advanced Robotics, vol.112, Springer, 2016.
- [10] R. S. Sutton, A. G. Barto, “Reinforcement Learning: An Introduction”, The MIT Press, 1998.
- [11] 神畠敏弘, “転移学習”, 人工知能学会誌 vol.25, no.4, pp.572–580, 2010.
- [12] A. Lazaric, “Transfer in Reinforcement Learning: a Framework and a Survey”, Reinforcement Learning–State of the art, vol.12, Springer, pp.143–173, 2012.
- [13] 松井藤五郎, “強化学習における知識の転移: 異なるタスクへの適応 (< 特集 > 知識の転移)”, 人工知能学会誌, vol.27, no.4, pp.373–380, 2012.
- [14] 河野仁, 伊藤祐希, 郡司拓朗, 神村明哉, 富田康治, 鈴木剛, “強化学習の方策再利用時におけるステップ単位の方策忘却手法”, ロボティクス・メカトロニクス講演会 2017, 2P1-F06 (CD-ROM), 2017.
- [15] 高野敏明, “同一エージェント間における転移学習を用いた強化学習の高速化に関する研究”, 三重大学, 博士論文, 2013.