深層強化学習エージェントの重み付き結合に関する検討

佐藤件一郎† 幸島 匡宏†† 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 〒 152–8552 東京都目黒区大岡山 2-12-1
†† 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: †sato.k.bo@m.titech.ac.jp, ††{kohjima.masahiro,matsubayashi.tatsushi,toda.hiroyuki}@lab.ntt.co.jp

あらまし 近年強化学習の研究領域において, それぞれが異なるタスクを解く複数の学習済みエージェントを結合することで, 新たなタスクを解くエージェントを構成するアプローチの研究が進められている. 本研究では, 単純平均をとることで学習済みエージェントを結合する既存技術を拡張し, 重み付き和によるエージェントの結合を行う手法を提案する. これにより報酬が学習済みタスクの線形和として定義される新たなタスクの問題を解くことが可能になる. CartPole 制御と適応信号制御という 2 つの実験を通して提案手法の有効性を検証した.

キーワード 強化学習, 深層強化学習, 最大エントロピー強化学習, 構成性, 信号制御

1. はじめに

深層学習のブレイクスルーにより AI 技術が大きく注目されている. 数ある成功の中でも強化学習とよばれる自律的な試行錯誤を行う学習フレームワークと組み合わせた深層強化学習が, ゲーム AI (コンピュータゲーム, 囲碁など) の分野で大きな成果を上げている [1] [2]. この成功を受けて近年では深層強化学習を用いたロボット制御や信号機の適応制御 [3] などへの応用検討が進められている.

深層強化学習には次の 2 つの欠点が存在することが知られている. (i) エージェントと呼ばれる学習者 (例えばロボット) の試行錯誤が必要であるため一般に長い学習時間を必要とする. (ii) 強化学習の学習結果は与えられた環境 (タスク) に依存するため, 環境が変われば基本的にゼロから学習し直しになってしまう. したがって人の目から見れば類似したタスクであっても, 環境が変わる度に学習し直しになり, 多大な時間と労力が必要になってしまう.

この問題意識のもと, ベースとなるタスクを解くエージェント (それぞれ部品タスク, 部品エージェントと呼ぶ) をあらかじめ学習しておき, 部品エージェントを組み合わせることで, 複雑なタスクを解くエージェントを構成するというアプローチが 検討されている [4], [5]. しかしながら, この既存研究では, 単純平均で表現されるタスクを, 部品エージェントの単純平均を用いて構成する場合のみが考察されており, 適用シーンが限定されていた.

そこで本研究では, 部品タスクの重み付き和で表現されるタスクを解くエージェントを, 部品エージェントの重み付き和を用いて構成する方法を提案する. 重み付き和で表現されるタスクには例えば次に示すシューティングゲームや信号制御が挙げられる. シューティングゲームにおいて, ある敵 A を撃ち落とすという部品タスク A を解くエージェント A, ある敵 B を撃ち落とすという部品タスク B を解くエージェント B がすでに得られているとする. このとき, 例えば敵 A を撃ち落とした時に 50 ポイント, 敵 B を撃ち落とした時に 10 ポイントが得られるタスクは, 部品タスク A と部品タスク B の重み付き和として表現される. 同様に信号制御において, 一般車両の待ち時間を短くする部品タスク A を解く部品エージェント A, バスなどの公共車両の待ち時間を短くする部品タスク B を解く部品エージェント B がすでに得られているとする. このとき, 例えば “一般車両の待ち時間 + 公共車両の待ち時間 ×5”を最小化するというタスクは, 部品タスク A と部品タスク B の重み付き和として表現される.

提案手法によって, 上記のような重み付き和で表現されるタスクに対して, 新たにエージェントをゼロから学習しなおすことなく, そのタスクに対応したエージェントを部品エージェントから構成することができるようになる, もしくは, ゼロからの再学習よりも短い時間で学習結果を得ることが可能になる. CartPole 制御と適応信号制御を用いた実験で提案手法の有効性を確かめた.

2. 関連研究

本研究の利用先の 1 つとして想定するものに適応信号制御がある. 強化学習を用いた信号制御の取り組みは古くから存在するものの (e.g. [6]), 深層学習を利用したアプローチの検討が近年多数行われている [3] [7] [8] [9]. これには都市の混雑が深刻化したことで混雑緩和に向けた様々な取り組みが行われていることが背景にあると考えられ [10], (深層学習を用いた) 適応信号制御は, これらの取り組みの延長として利用されていくと想像できる. しかし, 設置箇所 1 つ 1 つでその環境に合わせるために信号機を学習し直すというアプローチが取られるとは考え難い. 我々は学習済みのエージェントを組み合わせることで, 設置箇所に合わせた調整を行うことが有望だろうと考え本研究を実施した.

本研究では, エージェントの結合を考えるために通常の強化学習とは異なる定式化である, 最大エントロピー強化学習 (Maximum Entropy RL, MERL) と呼ばれる定式化を利用する. MERL では, 報酬和を最大化しつつ探索 (Exploration) に長けた方策が得られるように, 方策のエントロピーを目的関数に導

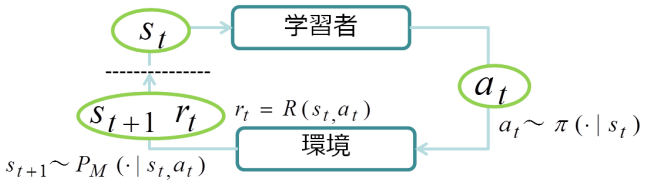


図 1　学習者 (Agent) と環境 (Environment) の相互作用

入した定式化を行う．このような方策や行動に関する正則化項を目的関数に導入した強化学習手法は，これ以外にも過去提案されており，線形可解マルコフ決定過程 (Linearly Solvable MDP) と呼ばれる設定に基づく方法に関する研究が進められてきた [11] [12] [13] [14] [15]．MERL もこの文脈に連なるものとして位置付けられる．

学習済みのエージェントを結合して新たなエージェントを構成できる性質は構成性 (Compositionality) と呼ばれ，上記で述べた新たな定式化に基づく強化学習手法の持つ重要な性質と認識されている [5] [16] [17]．本研究の貢献は，特に連続状態空間・離散行動空間における MERL のエージェントの重み付き結合に関して，その性能を理論と実験の両面から調べたところにある．

3. 準　　備

3.1 マルコフ決定過程 (MDP)

強化学習とは，学習者であるエージェントが環境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 $(\mathcal{S}, \mathcal{A}, P_M, \mathcal{R})$ により定義される．$\mathcal{S}$ を状態空間，$\mathcal{A}$ を行動空間と呼び，それぞれの元 $s \in \mathcal{S}$ を状態，$a \in \mathcal{A}$ を行動と呼ぶ．$P_M : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ 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．有界な関数である $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ は報酬関数である．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．エージェントは，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．エージェントの各状態 s で行う行動 a を選択する確率を定めたものを方策 $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ と呼ぶ．

方策を 1 つ定めると，エージェントは図 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 にいるエージェントは方策 $\pi(\cdot|s_t)$ 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，エージェントの次時刻の状態 $s_{t+1} \sim P_M(\cdot|s_t, a_t)$ と報酬 $r_t = \mathcal{R}(s_t, a_t)$ が決定する．これを繰り返すことで，エージェント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．

3.2 最大エントロピー強化学習

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．通常の強化学習 [18] において，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．

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

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．価値関数の値は多くの報酬をもたらす方策 π ほど大きくなる．本研究では，最大エントロピー強化学習 [4], [5] の定式化に従い，上記と異なる以下で定義される価値関数を利用する．

$$\begin{aligned} &Q^\pi_{soft}(s, a) \\ &\equiv \mathbb{E}^\pi_{d_T}\left[\sum_{k=0}^{\infty} \gamma^k \{\mathcal{R}(s_k, a_k) + \alpha \mathcal{H}(\pi(\cdot|s_k))\} \middle| s_0 = s, a_0 = a\right] \end{aligned} \tag{1}$$

ただし，α は重みパラメタ，$\mathcal{H}(\pi(\cdot|s_k))$ が状態 s_k にいるときの各行動の選択確率を定める分布 $\{\pi(a_1|s_k), \cdots, \pi(a_{|\mathcal{A}|}|s_k)\}$ のエントロピーを表す．エントロピーは一様分布に近いほど値が多くなるため，常に固定の行動をとる決定的な方策ではない，多様な行動を実行する探索 (Exploration) に長けた方策ほどエントロピー和は大きくなる．よってこの新たに定義した価値関数は報酬和が多くかつ “探索的” な方策ほど値が大きくなる．また，$\alpha = 0$ のときは通常の価値関数と一致する．ある方策 π, π' が任意の $s \in S, a \in A$ で $Q^\pi_{soft}(s, a) \geqq Q^{\pi'}_{soft}(s, a)$ を満たすとき，方策 π は π' よりも多くの報酬とエントロピーをエージェント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．最大エントロピー強化学習の目的は，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^*_{soft}(最適価値関数と呼ぶ) を用いて，

$$\pi^*(a|s) = \exp\Big(\frac{1}{\alpha}\{Q^*_{soft}(s, a) - V^*_{soft}(s)\}\Big), \tag{2}$$

と表現できる．ただし，

$$V^*_{soft}(s) = \alpha \log \sum\nolimits_{a'} \exp\Big(\frac{1}{\alpha} Q^*_{soft}(s, a')\Big).$$

このように最大エントロピー強化学習では，最適方策が確率的な方策として表現される．なお，通常の強化学習と同様，価値関数の推定には，最大エントロピー強化学習におけるベルマン方程式

$$Q^*_{soft}(s, a) = \mathcal{R}(s, a) + \gamma \mathbb{E}_{s' \sim P_M(s'|s,a)}[V^*_{soft}(s')] \tag{3}$$

を利用することで推定することができる．　この式中の Q^*_{soft}, V^*_{soft} を置き換えれば，これは通常の強化学習のベルマン方程式と等しい．状態空間が離散かつ状態数が膨大でなければ，Q 学習 [19] とほぼ同様の手法で価値関数を推定することが可能である．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ため，価値関数を近似する手法を利用する必要がある．

Algorithm 1 Soft Q-learning [4] の離散行動空間版

Input: $\mathcal{D}$: 入力データ, α: 正則化項
Output: $\hat{\theta}$: 価値関数を近似するニューラルネットのパラメタ

for *epoch* $= 1$ to *nepoch* **do**
  for $t = 1$ to T **do**
    %環境との相互作用
    パラメタ $\pi(\cdot|s_t)$ の多項分布に従い行動 a_t を確率的に選択
    行動 a_t を実行し, 環境から報酬 r_t と次状態 s_{t+1} を得る.
    replay memory に履歴保存 $\mathcal{D} \leftarrow \mathcal{D} \cup (s_t, a_t, r_t, s_{t+1})$.
    %価値関数の更新
    replay memory からミニバッチデータ $\mathcal{D}_{mini}$ を取得
    $\mathcal{D}_{mini}$ を用いた損失関数の勾配を計算しパラメタ θ を更新
  end for
end for

3.3 価値関数近似

(通常の強化学習における) 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には, パラメタに関して線形な関数で近似する方法 [20] やニューラルネットワーク [21] で近似する手法がこれまで提案されている. 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 θ を持つ関数 Q^θ で近似し, 目的関数

$$L(\theta) = \sum_{(s,a,s')\in D} \Big(\mathcal{R}(s) + \gamma V^\theta(s') - Q^\theta(s,a)\Big)^2$$

を最小化することで, パラメタ θ 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方程式 (3) の右辺と左辺の差を最小化することに相当する. このニューラルネットワークを用いる方法を基に種々のヒューリスティクスを導入した手法が Deep Q-Network (DQN) [1] である.

最大エントロピー強化学習においてもニューラルネットワークを用いた同様のアプローチが適用でき, Soft Q-Learning と呼ばれるアルゴリズムが提案されている [4]. 我々も部品エージェントの価値関数推定にこのアルゴリズムを利用する. ただし, この文献 [4] では状態空間と行動空間が両方連続空間である状況が考えられており, 価値関数を近似するネットワークと方策を近似するネットワークの両方を推定している. 今回我々は行動空間は離散である状況を考えるため, 方策を近似するネットワークの推定は必ずしも必要ではない. そこでこのアルゴリズムから方策を近似するネットワークの処理部分を除いた, "離散行動空間版の Soft Q-learning" アルゴリズムを利用することとした. 疑似コードを Algorithm 1 に示す.

4. 提案手法

提案手法による学習結果の結合方法を示す.

4.1 方策の構成法

報酬関数のみ異なる 2 つの MDP, MDP-1 $(\mathcal{S}, \mathcal{A}, P_M, \mathcal{R}_1, \gamma)$ と MDP-2 $(\mathcal{S}, \mathcal{A}, P_M, \mathcal{R}_2, \gamma)$ を考え, この MDP における最大エントロピー強化学習の最適価値関数 (式 (1)) を Q_1^*, Q_2^*, 式 (2) で定義されるその方策を π_1, π_2 と書く. 今, このそれぞれの MDP に対応するタスクはすでに学習されており, Q_1^*, Q_2^* については既知であるとする. これらを用いて, 重み付き平均で定義される報酬 $\mathcal{R}_3 = \beta_1\mathcal{R}_1 + \beta_2\mathcal{R}_2$ (ただし、$\beta_1 \geqq 0, \beta_2 \geqq 0, \beta_1 + \beta_2 = 1$) を持つ目標となる MDP-3 $(\mathcal{S}, \mathcal{A}, P_M, \mathcal{R}_3, \gamma)$ の方策を構成することを考える.

本研究では, MDP-3 を解くエージェントとして, 次のように価値関数の重み付き平均を取ることで構成して得た方策を用いることを提案する.

$$\pi_\Sigma(a|s) = \exp\Big(\frac{1}{\alpha}\big\{Q_\Sigma(s,a) - V_\Sigma(s)\big\}\Big). \tag{4}$$

ただし,

$$Q_\Sigma = \beta_1 Q_1^* + \beta_2 Q_2^*, \tag{5}$$

$$V_\Sigma(s) = \alpha \log \sum\nolimits_{a'\in\mathcal{A}} \exp\Big(\frac{1}{\alpha} Q_\Sigma(s,a')\Big). \tag{6}$$

方策 π_Σ は Q_Σ を MDP-3 の最適価値関数 Q_3^* とみなして, 式 (2) に代入して得た方策であると考えればよい.

上記の構成が有効であることを示すためには, 方策 π_Σ の MDP-3 における価値関数 $Q_3^{\pi_\Sigma}$ と Q_3^* が近い値をとることが示せればよい (注1). 次節でその理論解析を行う.

4.2 理論解析

まず未知の MDP-3 における最適価値関数 Q_3^* と Q_Σ の関係性を示す補題を導く.

Lemma 1. *MDP-3* $(\mathcal{S}, \mathcal{A}, P_M, \mathcal{R}_3, \gamma)$ の最適価値関数 Q_3^* として有界で $\sum_{a'\in\mathcal{A}} \exp\big(\frac{1}{\alpha}Q_3^*(\cdot,a')\big) < \infty$ を満たすものが存在するとき, *4.1* 節で定義した Q_Σ は次の関係を満たす.

$$Q_\Sigma(s,a) \geqq Q_3^*(s,a) \geqq Q_\Sigma(s,a) - C^*(s,a)$$

ただし, $C^*(s,a)$ は以下の再帰方程式の不動点である.

$$C(s,a) = \gamma \mathbb{E}_{s'\sim P_M(\cdot|s,a)} \Big[\alpha\beta_2 \mathcal{D}_{\beta_1}(\pi_1(\cdot|s')||\pi_2(\cdot|s')) + \max_{a'\in A} C(s',a')\Big]$$

ただし, $\mathcal{D}_{\beta_1}(\pi_1(\cdot|s')||\pi_2(\cdot|s'))$ は以下で定義されるオーダー β_1 の Rényi ダイバージェンスである.

$$\mathcal{D}_{\beta_1}(\pi_1(\cdot|s')||\pi_2(\cdot|s')) \equiv \frac{1}{\beta_1 - 1}\log \sum_{a'\in\mathcal{A}} \pi_1^*(a'|s')^{\beta_1}\pi_2^*(a'|s')^{\beta_2}.$$

証明は Appendix に示す. この再帰方程式は Rényi ダイバージェンスを報酬としたベルマン方程式と見ることもできる. よって π_1 と π_2 の Rényi ダイバージェンスが小さければ, その不動点 $C^*(s',a')$ の値も小さくなり $Q_3^*(s,a)$ と $Q_\Sigma(s,a)$ は近い値をとることをこの補題は示している.

この補題を用いると下記の定理が導ける.

(注1) : Q_Σ と $Q_3^{\pi_\Sigma}$ の違いには注意する. Q_Σ は式 (5) で便宜的に定義しただけのものであって, これから導かれる方策 π_Σ を用いた場合の "本当" の価値関数 $Q_3^{\pi_\Sigma}$ とは一般に一致しない.

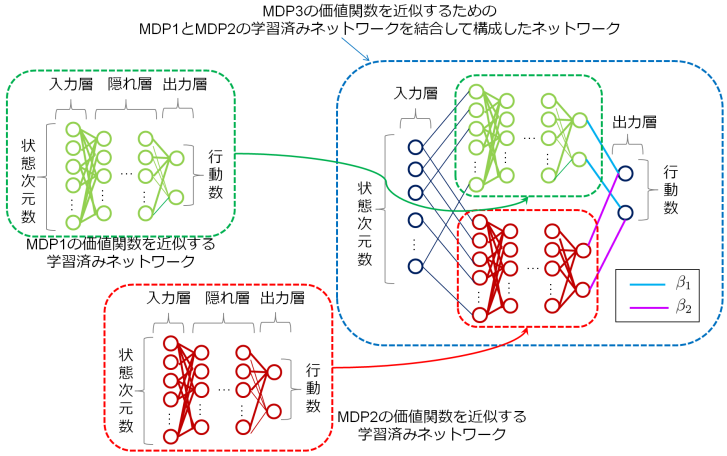


図 2　提案手法による新たなタスクを解くネットワークの構成

Theorem 1. *Lemma 1* と同様の条件下で Q_Σ に基づいた方策 π_Σ の *MDP-3* における価値関数を $Q_3^{\pi_\Sigma}$ とすると, $Q_3^{\pi_\Sigma}$ と最適価値関数 Q_3^* は

$$Q_3^{\pi_\Sigma}(s,a) \geqq Q_3^*(s,a) - D^*(s,a)$$

の関係を満たす. ただし, $D^*(s,a)$ は以下の再帰方程式の不動点である.

$$D(s,a)=\gamma \mathbb{E}_{s'\sim P_M(\cdot|s,a)}\left[\mathbb{E}_{a'\sim\pi_\Sigma(\cdot|s')}\left[C^*(s',a')+D(s',a')\right]\right]$$

証明は Appendix に示す. 補題と同様この再帰方程式は $C^*(s',a')$ を報酬としたベルマン方程式と見ることができ, $C^*(s',a')$ が小さければ, $D^*(s,a)$ も小さくなる. 最適価値関数の定義から $Q_3^*(s,a) \geqq Q_3^{\pi_\Sigma}(s,a)$ が常に成り立つことを考えれば, $D^*(s,a)$ が小さければ, $Q_3^*(s,a)$ と $Q_3^{\pi_\Sigma}(s,a)$ は近い値をとるということをこの定理は示している. これらの結果から, 部品エージェントの方策 π_1 と π_2 の Rényi ダイバージェンスが大きくない状況においては, 提案する構成法は有効であると期待できる.

なお, $\beta_1 = \beta_2 = 1/2$ のときこの補題と定理は文献 [5] 中のそれらと一致し, 単純平均を重み付き和に拡張したものとなっている.

4.3　ニューラルネットワークによる価値関数近似を用いた場合の Q_Σ の具体的構成法

最大エントロピー強化学習において Q 関数を DQN [1] と同様にニューラルネットワークにより近似した場合, どのように Q_Σ を構成すればよいかを示す.

MDP-1 $(\mathcal{S},\mathcal{A},P_M,\mathcal{R}_1,\gamma)$ と MDP-2 $(\mathcal{S},\mathcal{A},P_M,\mathcal{R}_2,\gamma)$ における学習済み価値関数をそれぞれ $Q_1^{\theta_1}, Q_2^{\theta_2}$ とする. ただし, θ_1,θ_2 はネットワークのパラメータである.

$$Q_\Sigma^{\theta_1,\theta_2} \equiv \beta_1 Q_1^{\theta_1} + \beta_2 Q_2^{\theta_2}$$

が成り立つように Q_Σ を構成するには, 図 2 のようにネットワークを構築すればよい. Q_Σ は, 入力層の値そのままを $Q_1^{\theta_1}, Q_2^{\theta_2}$ の入力層にそのまま渡し, それぞれの出力の値を β_1,β_2 で重み付き和をとって出力としている. こうすることで, $Q_1^{\theta_1}, Q_2^{\theta_2}$ さえ既知であれば, 再学習などの必要なく Q_Σ を構成することができる. 後の実験では, この方法により Q_Σ を構成した. なお, もしさらなる性能改善のために再学習をさせたい場合にも, Q_Σ を初期値として学習を行えば, $Q_1^{\theta_1}, Q_2^{\theta_2}$ を学習する際に用いたアルゴリズムをそのまま適用することができる.

5.　実　　験

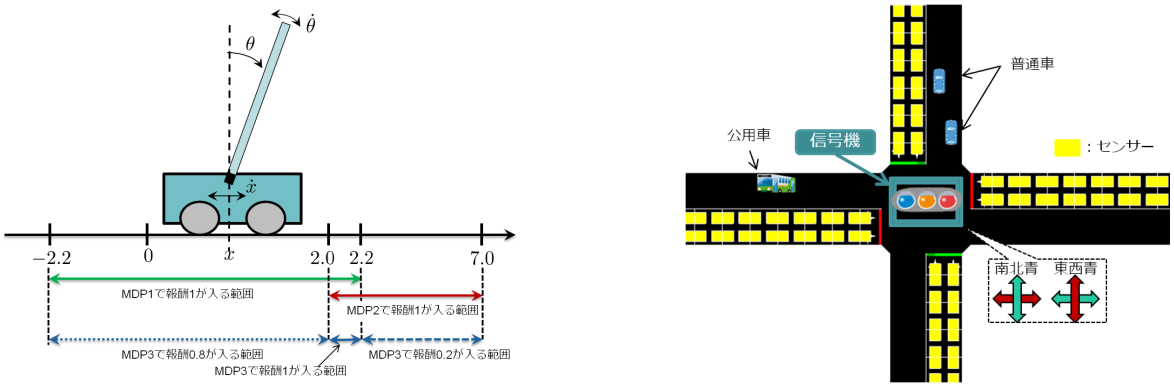
提案アプローチによって構成した Agent の有効性を実験を通して確認する.

5.1　実験設定

Cart Pole: CartPole 制御とはカートに接続されているポールを倒さないようにカートを左右に動かし制御する強化学習のベンチマーク問題である. 強化学習の環境を提供するオープンソースのライブラリ OpenAI gym (注2) の CartPole の環境をベースに報酬関数の設定の異なる 3 つの MDP を作成した. 作成した MDP のを図 3(a) に示す. 3 つの MDP の状態空間 $\mathcal{S}$, 行動空間 $\mathcal{A}$, 状態遷移関数 P_M は共通であり, 各状態 $s \in \mathcal{S}$ は, 位置 x, 速度 $\dot{x}$, 角度 θ, 角速度 $\dot{\theta}$ から成る 4 次元ベクトル, 行動空間 $\mathcal{A}$ は右に行く $(a = \text{Right})$ か左に行く $(a = \text{Leftt})$ の 2 つの行動から成る. 報酬関数は, それぞれの MDP で下記の定義のものを用いる. MDP1 の報酬 $\mathcal{R}_1$ はポールが立っていてかつカートが区間 $[-2.2, 2.2]$ にいるならば報酬 1 を与える関数, MDP2 の報酬 $\mathcal{R}_2$ はポールが立っていてかつ区間 $[2.0, 7.0]$ でポールが立っていれば報酬 1 を与える関数として定義した. MDP3 の報酬 $\mathcal{R}_3$ は, $\beta_1 = 0.8, \beta_2 = 0.2$ として, $\mathcal{R}_3 := \beta_1\mathcal{R}_1 + \beta_2\mathcal{R}_2$ と定義した. 報酬関数 $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3$ を使い MDP-1 $(\mathcal{S},\mathcal{A},P_M,\mathcal{R}_1,\gamma)$,MDP-2 $(\mathcal{S},\mathcal{A},P_M,\mathcal{R}_2,\gamma)$, MDP-3 $(\mathcal{S},\mathcal{A},P_M,\mathcal{R}_3,\gamma)$ を構成した. 割引率は $\gamma = 0.99$ とした.

最大エントロピー強化学習によって MDP1 と解く部品 Agent 1, MDP2 を解く部品 Agent 2 を作成し, 提案アプローチによって部品 Agent1 と部品 Agent2 を組み合わせることにより新たなエージェント Proposed を構成した. 実験において MDP-3 におけるの部品 Agent1, 部品 Agent2, Proposed のパフォーマンスを比較した. なお, 使用したニューラルネットワークの詳

(注2)：https://gym.openai.com/



(a) CartPole 制御　　(b) 交通信号制御

図 3　実 験 環 境

表 1　各問題とニューラルネットワークの設定

	CartPole	信号制御
状態空間の次元数	4	385
行動空間の次元数	2	2
ネットワークの素子数	4-100-100-2	385-100-2
活性化関数	tanh	relu+batch normalization

細は表 1 に記述した.

適応信号制御: 適応信号制御とは, 車の待ち時間減少などを行うために道路などに設置されているセンサーなどの情報をもとに適応的に交通信号を変化させる問題である. 本実験ではオープンソースの交通シミュレータ Simulation of Urban MObility (SUMO) [22] を利用した. 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 3(b) に示す 2 車線の十字路を考えた. 東西南北それぞれの道路には毎秒おきに車両の有無とその車両が公用車か否かを観測するセンサーが 5m 間隔で 48 個設置してある. 東西南北から一定の頻度でランダムに生成される普通車と公用車の待ち時間が小さくなるよう交差点の信号機を制御する. CartPole の実験と同様に報酬関数を変えることにより, 3 つの MDP をつくる. 3 つの MDP の状態空間 $\mathcal{S}$, 行動空間 $\mathcal{A}$, 状態遷移関数 P_M は共通であり, 状態空間 $\mathcal{S}$ は, 該当地点の車の有無 (有: 1, 無: 0) とそれが公用車か否か (公用車: 1, 非公用車: 0) を返すセンサー 192 個分と現在の信号機の状態 (青赤, 赤青) を表す 385 次元ベクトルを要素に持つ集合, 行動空間 $\mathcal{A}$ は青赤に変える, 赤青に変えるの 2 つの行動をもつ要素数 2 の集合である. $\mathcal{R}_1$ は前ステップから現在のステップに至るまでの全車両の待ち時間の合計値の-1 倍, $\mathcal{R}_2$ は前ステップから現在のステップに至るまでの全公用車の待ち時間の合計値の-1 倍, $\mathcal{R}_3$ は $\mathcal{R}_3 := \frac{1}{N}\mathcal{R}_1 + \frac{N-1}{N}\mathcal{R}_2$ とする. Agent 1, 部品 Agent 2 と呼ぶことにする. 提案アプローチによって部品 Agent1 と部品 Agent2 を組み合わせることにより新たなエージェント Proposed をつくり, まず $N = 4$ のとき, 重み付き和の比率を変化させたときパフォーマンスがどのように変化するかを計測することで, 構成したエージェントが部品エージェントよりも優れているかを確認する. さらに信号機としての有効性を確認するため, MDP-3 における N を $2, 3, 4, 9$ と変化させ, 重み付き和の値も同じ値を用いた場合のパフォーマンスを固定信号のパフォーマンスと比較する.

5.2　実 験 結 果

定量評価: 図 4(a) に Cartpole 制御の定量評価の結果を示す. この図より Proposed が最も高い報酬和を獲得したことがわかる. これによって, 提案する方策の構成法で部品 Agent をそのまま用いるよりも優れた方策を作成しうることが確認できた.

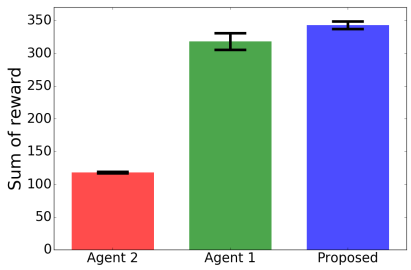
図 4(b) に適応信号制御の定量評価の結果を示す. 結合せずに部品エージェントそのままを利用した設定 $\beta_1 = 0.0, 1.0$ よりも $\beta_1 \approx 1/N = 0.25$ 付近の設定したときに重み付きで構成したエージェントの方が優れたパフォーマンスを示していることが確認できる. したがって提案する構成法の有効性が確認できた.

定性評価: CartPole 制御の各 Agent の挙動を図 5, 6, 7 に示す. 図 5, 6 により提案手法の部品となる Agent1, 2 は報酬が得られる範囲内で移動しつつポールを倒れさせない方策を学習できている. さらに図 7 に注目すると上記 Agent を提案手法により結合して構成した Agent 3 は, 最も得られる報酬が大きい位置まで移動し, そこに留まる方策となっていることが分かる. これが Agent 3 が最も高い報酬和を獲得できている理由であると考えられる.

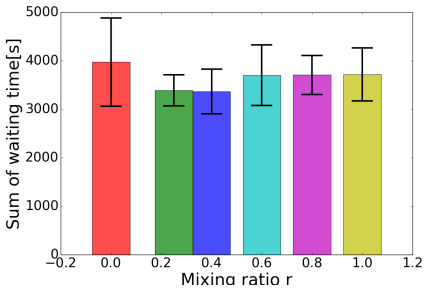
次に結合して構成した適応信号制御と固定信号との比較結果を図 8 を示す. $N = 2, 3, 4, 9$ のいずれの場合においても, Proposed がおおそよ 4 割ほど固定信号 (Baseline) の性能を上回っていることが分かる. これにより結合して構成した適応信号制御が固定信号よりも優れた性能を持つことも確認できた.

6.　ま　と　め

本研究では部品 Agent の学習結果を重み付き和で組み合わせることで新たな Agent を構成する手法を提案した. 理論解析により提案手法により構成した方策の価値関数と最適価値関数の関係を示す定理を導出し, CartPole 制御と適応信号制御の実験により提案手法の有効性を確認した. 今後の展望としては, 3 つ以上の部品 Agent を結合する場合の検討が挙げられる. その際には構成したネットワークのサイズが結合する個数に応じて大きくなるため, 蒸留 [23] などの利用により, 結合して構成し



(a) CartPole 制御の実験結果．値が大きいほど良い．



(b) 交通信号制御の実験結果．値が小さいほど良い．

図 4　(a) CartPole 制御と (b) 適応信号制御の実験結果．横軸はエージェントの結合時に用いたパラメタ $\beta_1 = 0.0, 0.25, 0.4, 0.6, 0.8, 1.0$, 縦軸は各 Agent を MDP-3 $(1/N = 0.25)$ で 10 回実行したときの各エピソードの報酬和/待ち時間和の平均と分散を示す．

たネットワークの入出力を再現するコンパクトなネットワークを作るアプローチが必要となる可能性があると考えられる．

Appendix

Lemma 1 の証明を示す．

Proof.　任意の正の整数 k に対して $Q^{(k)}, C^{(k)}$ をそれぞれ次のような漸化式によって定める．

$$Q^{(k)}(s,a) = \begin{cases} k=0: \ Q_\Sigma(s,a) \\ k \geqq 1: \\ \mathcal{R}_3 + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \Big[\\ \qquad \alpha \log \sum_{a' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q^{(k-1)}(s',a')\right) \Big] \end{cases}$$

$$C^{(k)}(s,a) = \begin{cases} k=0: \ 0 \\ k \geqq 1: \ \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\alpha\beta_2 \mathcal{D}_{\beta_1}(\pi_1(\cdot|s') || \pi_2(\cdot|s')) \right. \\ \qquad \left. + \max_{a' \in A} C^{(k-1)}(s',a') \right] \end{cases}$$

Lemma 1 の条件下で $Q^{(k)}(s,a), C^{(k)}(s,a)$ が $k \to \infty$ のとき，それぞれ $Q_3^*(s,a), C^*(s,a)$ に収束することは [4] に同様の証明がある．よって，任意の k に関して $Q_\Sigma(s,a) \geqq Q^{(k)}(s,a) \geqq Q_\Sigma(s,a) - C^{(k)}(s,a)$ を示せば十分．帰納法を用いて $Q^{(k)}(s,a) \geqq Q_\Sigma(s,a) - C^{(k)}(s,a)$ を示す．$k=0$ のときは定義より明らか．$Q^{(k)}(s,a) \geqq Q_\Sigma(s,a) - C^{(k)}(s,a)$ が成立すると仮定すると，

$$\begin{aligned} & Q^{(k+1)}(s,a) \\ &= \mathcal{R}_3 + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\alpha \log \sum_{a' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q^{(k)}(s',a')\right) \right] \\ &\geqq \mathcal{R}_3 + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\alpha \log \sum_{a' \in \mathcal{A}} \exp \frac{1}{\alpha} \left(Q_\Sigma(s',a') \right. \right. \\ &\qquad \left. \left. - C^{(k)}(s',a') \right) \right] \\ &\geqq \mathcal{R}_3 + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\alpha \log \sum_{a' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_\Sigma(s',a')\right) \right. \\ &\qquad \left. - \max_{a' \in A} C^{(k)}(s',a') \right] \\ &= \mathcal{R}_3 - \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\max_{a' \in A} C^{(k)}(s',a') \right] \\ &\quad + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\alpha \log \sum_{a' \in \mathcal{A}} \left(\frac{\exp(\frac{1}{\alpha} Q_1(s',a'))}{\sum_{a'' \in \mathcal{A}} \exp(\frac{1}{\alpha} Q_1(s',a''))} \right)^{\beta_1} \right. \\ &\qquad \left(\frac{\exp(\frac{1}{\alpha} Q_2(s',a'))}{\sum_{a'' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_2(s',a'')\right)} \right)^{\beta_2} \\ &\qquad \left(\sum_{a'' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_1(s',a'')\right) \right)^{\beta_1} \\ &\qquad \left. \left(\sum_{a'' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_2(s',a'')\right) \right)^{\beta_2} \right] \end{aligned}$$

ここで，$\pi_1(a'|s') = \frac{\exp(\frac{1}{\alpha} Q_1(s',a'))}{\sum_{a'' \in \mathcal{A}} \exp(\frac{1}{\alpha} Q_1(s',a''))}$，$\pi_2(a'|s') = \frac{\exp(\frac{1}{\alpha} Q_2(s',a'))}{\sum_{a'' \in \mathcal{A}} \exp(\frac{1}{\alpha} Q_2(s',a''))}$, $\mathcal{D}_{\beta_1}(\pi_1(\cdot|s') || \pi_2(\cdot|s')) \equiv \frac{1}{\beta_1 - 1} \log \sum_{a' \in \mathcal{A}} \pi_1^*(a'|s')^{\beta_1} \pi_2^*(a'|s')^{\beta_2}$ に注意すると，

$$\begin{aligned} & Q^{(k+1)}(s,a) \\ &\geqq \mathcal{R}_3 \\ &\quad - \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\max_{a' \in A} C^{(k)}(s',a') + \alpha\beta_2 \mathcal{D}_{\beta_1}(\pi_1(\cdot|s') || \pi_2(\cdot|s')) \right] \\ &\quad + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\alpha\beta_1 \log \left(\sum_{a'' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_1(s',a'')\right) \right) \right. \\ &\quad \left. + \alpha\beta_2 \log \left(\sum_{a'' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_2(s',a'')\right) \right) \right] \\ &= Q_\Sigma(s,a) - C^{(k+1)}(s,a) \end{aligned}$$

よって，任意の k に関して $Q^{(k)}(s,a) \geqq Q_\Sigma(s,a) - C^{(k)}(s,a)$ が成立することがわかった．

同様に $Q_\Sigma(s,a) \geqq Q^{(k)}(s,a)$ が成立することを数学的帰納法により示す．$k=0$ のときは定義より明らか．$Q_\Sigma(s,a) \geqq Q^{(k)}(s,a)$ が成立すると仮定すると，

$$\begin{aligned} & Q^{(k+1)}(s,a) \\ &= \mathcal{R}_3 + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\alpha \log \sum_{a' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q^{(k)}(s',a')\right) \right] \end{aligned}$$

$$\leqq \mathcal{R}_3 + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[\alpha \log \sum_{a' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_\Sigma(s', a') \right) \right]$$

$\log \sum_{a' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_\Sigma(s', a') \right)$ の部分を以下のように展開すると,

$$\begin{aligned}
&\log \sum_{a' \in \mathcal{A}} \exp(\frac{1}{\alpha} Q_\Sigma(s', a')) \\
&= \log \left\{ \sum_{a' \in \mathcal{A}} \left(\frac{\exp(\frac{1}{\alpha} Q_1^*(s', a')}{\sum_{a'' \in \mathcal{A}} \exp(\frac{1}{\alpha} Q_1^*(s', a''))} \right)^{\beta_1} \right. \\
&\left(\frac{\exp(\frac{1}{\alpha} Q_2^*(s', a')}{\sum_{a'' \in \mathcal{A}} \exp(\frac{1}{\alpha} Q_2^*(s', a''))} \right)^{\beta_2} \left(\sum_{a'' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_1^*(s', a'') \right) \right)^{\beta_1} \\
&\left. \left(\sum_{a'' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q_2^*(s', a'') \right) \right)^{\beta_2} \right\} \\
&= \log \left\{ \sum_{a' \in \mathcal{A}} \pi_1(a'|s')^{\beta_1} \pi_2(a'|s')^{\beta_2} \right\} + \beta_1 V_1(s') + \beta_2 V_2(s').
\end{aligned}$$

よって, $Q^{(k+1)}(s,a)$ は

$$\begin{aligned}
Q^{(k+1)}(s,a) &\leqq \mathcal{R}_3 + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} [q_1 V_1^*(s') + q_2 V_2^*(s')] \\
&\quad + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} [\alpha \beta_2 \mathcal{D}_{\beta_1}(\pi_1(\cdot|s') || \pi_2(\cdot|s'))] \\
&\leqq Q_\Sigma(s,a)
\end{aligned}$$

ただし, 最後の式は Rényi ダイバージェンス $\mathcal{D}_{\beta_1}(\pi_1(\cdot|s') || \pi_2(\cdot|s'))$ が非負であることを使った. □

次に, Theorem 1 の証明にうつる前に以下の系を先に示す.

Corollary 1. 補題 *1* と同様の条件下で,

$$V_\Sigma(s) \geqq V_3(s) \geqq V_\Sigma(s) - \max_{a \in A} C^*(s,a)$$

Proof. Lemma 1 より,

$$\begin{aligned}
& Q_\Sigma(s,a) \geqq Q_3(s,a) \geqq Q_\Sigma(s,a) - C^*(s,a) \\
\Rightarrow\ & Q_\Sigma(s,a) \geqq Q_3(s,a) \geqq Q_\Sigma(s,a) - \max_{a \in A} C^*(s,a)
\end{aligned}$$

であることがわかり, この式に対し, $\frac{1}{\alpha}$ をかけ, $\alpha \log \sum_{a \in \mathcal{A}} \exp$ をとれば系が得られる. □

これを用いて Theorem 1 の証明を行う.

Proof. 任意の正の整数 k に対して $Q^{(k)}, D^{(k)}$ をそれぞれ次のような漸化式によって定める.

$$Q^{(k)}(s,a) = \begin{cases} k = 0: & Q_\Sigma(s,a) \\ k \geqq 1: & \mathcal{R}_3 + \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} [\alpha \mathcal{H}(\pi_\Sigma(\cdot|s')) \\ & \quad + \mathbb{E}_{a' \sim \pi_\Sigma(\cdot|s')} \left[Q^{(k-1)}(s'|a') \right] \right] \end{cases}$$

$$D^{(k)}(s,a) = \begin{cases} k = 0: & 0 \\ k \geqq 1: & \gamma \mathbb{E}_{s' \sim P_M(\cdot|s,a)} \left[E_{a' \sim \pi_\Sigma(\cdot|s')} [C^*(s',a') \right. \\ & \quad \left. + D^{(k-1)}(s',a') \right] \right] \end{cases}$$

この漸化式は $Q^{(k)}(s,a), D^{(k)}(s,a)$ が不動点になっており, 関数のノルムとして $||Q_1 - Q_2|| \equiv \max_{s,a} |Q_1(s,a) - Q_2(s,a)|$ と定義したとき, 縮小写像になっている. よって, 不動点定理よりそれぞれ $k \to \infty$ のとき, $Q^{\pi_\Sigma}(s,a), D^*(s,a)$ に収束する. 以下, Lemma 1, Corollary 1 を使って [5] で行った式変形と全く同様に定理を導くことができる. □

文　献

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

[2] David Silver, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, Vol. 529, No. 7587, p. 484, 2016.

[3] Wade Genders and Saiedeh Razavi. Using a deep reinforcement learning agent for traffic signal control. *arXiv preprint arXiv:1611.01142*, 2016.

[4] Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. *arXiv preprint arXiv:1702.08165*, 2017.

[5] Tuomas Haarnoja, Vitchyr Pong, Aurick Zhou, Murtaza Dalal, Pieter Abbeel, and Sergey Levine. Composable deep reinforcement learning for robotic manipulation. *arXiv preprint arXiv:1803.06773*, 2018.

[6] Marco Wiering. Multi-agent reinforcement learning for traffic light control. In *ICML*, pp. 1151–1158, 2000.

[7] 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.

[8] 大橋耕也, 幸島匡宏, 堤田恭太, 松林達史, 戸田浩之. 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. In *DEIM*, 2018.

[9] Hua Wei, Guanjie Zheng, Huaxiu Yao, and Zhenhui Li. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *KDD*, pp. 2496–2505. ACM, 2018.

[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] Emanuel Todorov. Linearly-solvable markov decision problems. In *NIPS*, pp. 1369–1376, 2006.

[12] Krishnamurthy Dvijotham and Emanuel Todorov. Inverse optimal control with linearly-solvable MDPs. In *ICML*, pp. 335–342, 2010.

[13] Ang Li and Paul R Schrater. Efficient learning in linearly solvable mdp models. In *IJCAI*, pp. 248–253, 2013.

[14] Masahiro Kohjima, Tatsushi Matsubayashi, and Hiroshi Sawada. Generalized inverse reinforcement learning with linearly solvable mdp. In *ECMLPKDD*, pp. 373–388, 2017.

[15] 松原崇充. 確率最適制御の最近の動向: 確率推論による解法. システム/制御/情報, Vol. 59, No. 10, pp. 369–374, 2015.

[16] Emanuel Todorov. Compositionality of optimal control laws. In *NIPS*, pp. 1856–1864, 2009.

[17] 内部英治. 線形可解マルコフ決定過程を用いた順・逆強化学習. 日本神経回路学会誌, Vol. 23, No. 1, pp. 2–13, 2016.

[18] Richard S Sutton and Andrew G Barto. *Introduction to reinforcement learning*. MIT press Cambridge, 1998.

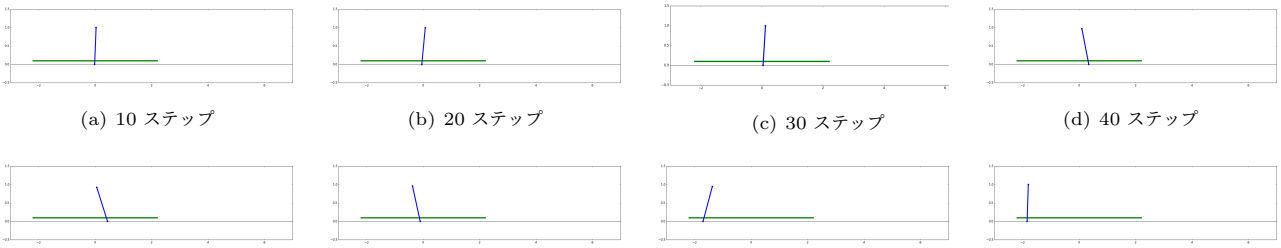
[19] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, Vol. 8, No. 3-4, pp. 279–292, 1992.

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

[21] 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.

[22] 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.

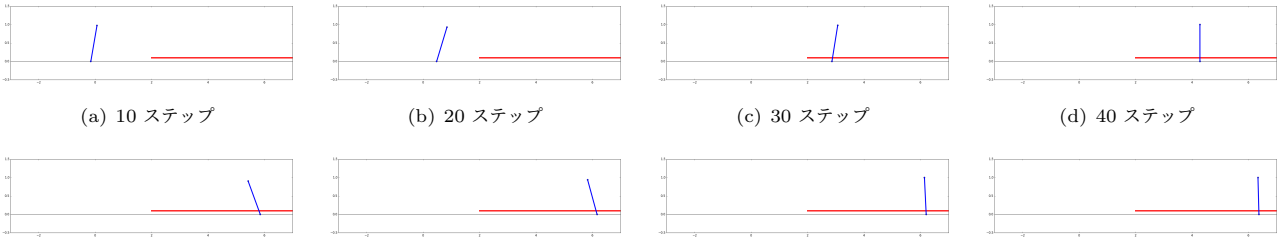
[23] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.



(a) 10 ステップ　(b) 20 ステップ　(c) 30 ステップ　(d) 40 ステップ

(e) 50 ステップ　(f) 70 ステップ　(g) 100 ステップ　(h) 120 ステップ

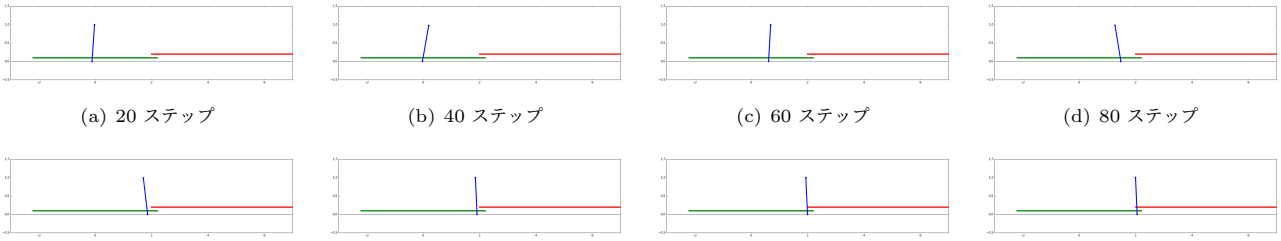
図 5　MDP1 の学習済みネットワークを用いた学習済みエージェントの挙動．緑色の線上の範囲が報酬が得られる位置を表す．初期の状態から停止したままバランスをキープしつつ (〜30 ステップ), バランスを崩しても左に移動することで (〜100 ステップ), 再度バランスを持ち直す動作が学習できている．



(a) 10 ステップ　(b) 20 ステップ　(c) 30 ステップ　(d) 40 ステップ

(e) 50 ステップ　(f) 70 ステップ　(g) 100 ステップ　(h) 120 ステップ

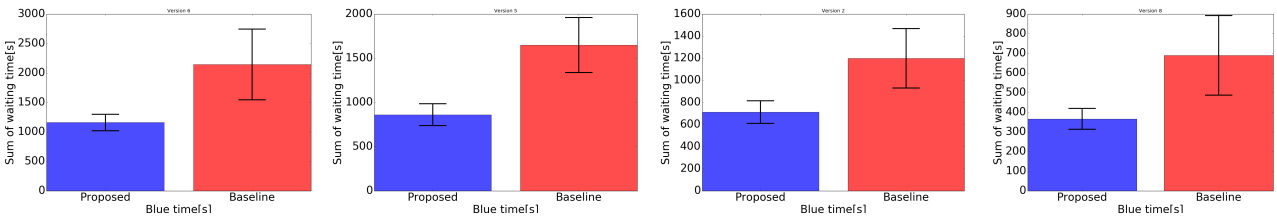
図 6　MDP2 の学習済みネットワークを用いた学習済みエージェントの挙動．赤色の線上の範囲が報酬が得られる位置を表す．ポールをまず右側に傾けてから右へ移動し (〜30 ステップ), ポールを倒さないように注意しつつ減速し (40 ステップ〜), バランスを保って停止している (100 ステップ〜)．このように右へ移動し停止するという動作が学習されている．



(a) 20 ステップ　(b) 40 ステップ　(c) 60 ステップ　(d) 80 ステップ

(e) 100 ステップ　(f) 110 ステップ　(g) 120 ステップ　(h) 130 ステップ

図 7　学習済みネットワークを結合して構成したエージェントの挙動．緑色と赤色の線はそれぞれ図 5, 6 と同じ範囲を表し, MDP3 ではこの 2 つの線の重なる範囲が最も報酬が大きくなる範囲を表す．このエージェントは, この報酬最大の範囲へ移動し, そこで停止する動作を行っていることが分かる．



(a) 比率 2:1　(b) 比率 3:1　(c) 比率 4:1　(d) 比率 9:1

図 8　結合して構成した適応信号制御と固定信号の比較結果．値が小さいほど良い．