離散行動空間における教師なしスキルの獲得手法

海野良介[1,a)]　鶴岡慶雅[2]

概要：深層強化学習は Atari 2600 や囲碁などのゲームにおいて高いパフォーマンスを達成できるようなった．しかし，課題の一つとして学習を行うエージェントに望み通りの行動を取るように報酬関数を設計することが困難であるという点がある．本論文では，エージェントが離散的な行動を取る環境において外部からの報酬を与えられることなく，一貫性のある行動を取るような方策である「スキル」を学習する手法を提案する．本稿では実験として，行動空間が離散的である二次元グリッド空間，MountainCar-v0，Freeway の 3 つのタスクに適用し，多様なスキルの学習であるか，また学習したスキルで階層型強化学習の手法による学習を行った際の性能を検証した．その結果，学習した「スキル」がタスクを達成する上で有用であることがわかった．

Unsupervised Skill Discovery in Discrete Action Spaces

Ryosuke Unno[1,a)]　Yoshimasa Tsuruoka[2]

Abstract: Deep reinforcement learning can now achieve high performance in games such as Atari 2600 and Go. However, one of the challenges in reinforcement learning is to design a reward function that leads the agent to learn a policy with desired actions. In this study, we propose a method for learning skills in discrete action spaces without any external rewards. We conducted experiments on three discrete action tasks, namely, 2D grid space, MountainCar-v0, and Freeway, to confirm that the agent can learn diverse sets of skills. We also applied the learned skills to hierarchical reinforcement learning tasks to measure whether the skills can be used in downstream tasks. As a result, we found that learned skills are useful for solving tasks.

1. はじめに

近年，ニューラルネットワークを活用した深層学習技術の発展にともなって，深層学習の画像処理や自然言語処理の分野への利用が盛んに研究されている．強化学習の分野においても，深層学習と組み合わせることで Atari 2600 や囲碁などのゲームで人間を超えるパフォーマンスを発揮できるようになった [1], [2].

強化学習では環境から受け取る報酬の合計を最大化するようなエージェントの行動を決定する方策を学習する．深層強化学習の課題として，最適な方策を学習するまでに環境からのサンプルを多く必要とし，学習効率が低いということが挙げられる．特に環境から外的な報酬がほとんどの状態において与えられず，報酬が得られるまでに長時間の遷移を必要とするような報酬が疎な環境ではエージェントはランダムに行動することしかせず，偶然報酬が得られるような状態に到達することがなければ最適な方策の学習は困難である．また，学習を促すために目標の途中段階で報酬を与えようとしても，想定外の行動を学習する場合がある．望みどおりの行動をエージェントに学習させるためには適切な報酬の設計が必要であり，それも容易であるとは限らない．

しかし，人間は明示的にそのような報酬を与えられることがなくても新しい環境に直面した際に，その環境を探索し，環境との相互作用を通して環境の仕組みを理解する．そして，その後の活動において有用となる知識や一貫性を

[1] 東京大学工学部電子情報工学科
Department of Information and Communication Engineering, The University of Tokyo
[2] 東京大学大学院情報理工学系研究科電子情報学専攻
Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo
[a)] runno@logos.t.u-tokyo.ac.jp

© 2020 Information Processing Society of Japan

持った行動をスキルとして学ぶことができる．さらにその後に特定の目標が与えられたときに，それまでに学んだ知識やスキルを活用して素早く，効率的に目標を達成することができる．例えば，人間は生まれたばかりの頃は体の関節を軽く動かすといった基本的な動作しかできないが，それらの動作を組み合わせて次第に歩いたり，ものを掴んだりといった行動をスキルとして身につける．そしてこれらの学んだスキルは，ものを拾って別の場所に運ぶというような新たなタスクが与えられたときに，組み合わせることでタスクを達成するのに役立つものとなる．

ここでの人間のようにエージェント外部からの報酬を与えることなく環境を行動する上で有用となる方策である「スキル」を学習させる手法がある [3], [4]．外部からの報酬なしに学習を行うため，モデルがエージェントに対してスキルごとに特有の行動が見られるように内部報酬を生成する．その後，学習したスキルを行動単位として階層型強化学習の手法にしたがって利用する．これらの手法ではスキルを先に学習し，段階を踏んで方策が学習されるため探索が効率化される．また，スキルの学習に外部からの報酬を必要としないので，報酬が疎な環境における学習に利用できると考えられている．

これらの教師なしでスキルを学習し，階層型強化学習の下位方策として利用する研究は連続行動空間上で行われており，離散行動空間上では我々の知る限り行われていない．

本研究では，行動が離散的になるゲームのような環境において多様かつその後の行動において有用となるスキルを外部からの報酬を与えられることなく学習し，そのスキルを下位方策として階層型強化学習の手法に適用した際の性能を評価することを目的とする．

2. 背景

2.1 強化学習

強化学習は機械学習の手法の一つである．他の機械学習の手法である教師あり学習で教師データが入力として与えられるのとは異なり，強化学習では環境が与えられ，環境との相互作用を通してデータを収集して学習を行う．強化学習の問題設定においてはマルコフ性に従う環境であるマルコフ決定過程 (MDP) を仮定する．マルコフ性とは遷移先の状態が直前の状態とその時点での行動のみによって依存するような性質である．MDP は以下の構成要素を持つ．

- 状態の有限集合：$\mathcal{S}$
- 行動の有限集合：$\mathcal{A}$
- 状態遷移確率 $p(s'|s,a)$：$\mathcal{S}\times\mathcal{A}\times\mathcal{S}\to[0,1]$
- 報酬関数 $r(s,a)$：$\mathcal{S}\times\mathcal{A}\to\mathbb{R}$

図 1 に示すように，エージェントと環境が存在する．時間 t においてエージェントは状態 s_t を環境から受け取り，それに対して方策 $a_t\sim\pi(a_t|s_t)$ にしたがって行動 a_t を選択する．その結果としてエージェントは報酬 r_t と次状態

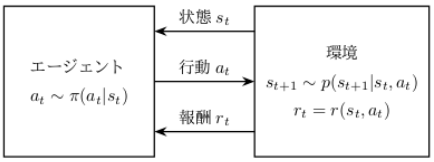

図 1: 強化学習におけるエージェントと環境の相互作用

s_{t+1} を新たに受け取る．この一連の操作を繰り返し，環境から受け取る累積報酬 R を最大化するような方策を求めることが強化学習におけるエージェントの目的である．これは以下の式で表される値を最大化するものである．

$$R=\sum_{t=0}^{\infty}\gamma^t r_t$$

このとき γ $(0\leq\gamma<1)$ は割引率と呼ばれ，将来的に受け取る可能性のある報酬をどの程度重要視するかを決定する定数であり，通常 1 に近い値を用いる．

2.1.1 Q 学習

Q 学習 [5] は強化学習の代表的な手法である．

ある状態 s において行動 a を取り，その後方策 π にしたがって行動を行った際に期待される累積報酬を返す関数を $Q^\pi(s,a)$ とする．この関数を Q 関数と呼び，出力される値を Q 値という．Q 関数は以下のように定式化される．

$$Q^\pi(s,a)=\mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t|s_0=s)\right]$$

Q 関数は最適な方策を π^* としたときに以下の Bellman 方程式と呼ばれる再帰的な関係式として表せる．

$$Q^{\pi^*}(s,a)=\sum_{s'}p(s'|s,a)(r(s,a)+\gamma\max_{a'}Q^{\pi^*}(s',a'))$$

Q 学習では $Q^{\pi^*}(s,a)$ を関数 $Q(s,a)$ を用いて近似し，学習を行う．学習の段階では以下の式にしたがって，Q 値の更新が行われる．

$$Q(s_t,a_t)\leftarrow Q(s_t,a_t)+\alpha(r(s_t,a_t)+\gamma\max_a Q(s_{t+1,a})-Q(s_t,a_t))$$

すなわち，$Q(s_t,a_t)$ を

$$r(s_t,a_t)+\gamma\max_a Q(s_{t+1},a)$$

に近づけるようにして学習が行われる．このとき，α は学習率と呼ばれるパラメータである．

2.1.2 Deep Q-Network

Deep Q-Network (DQN) [6] では Q 関数をニューラルネットワークで近似することによって Q 学習を行う．ここで，ニューラルネットワークの重みをパラメータ θ としたときに Q 関数は $Q_\theta(s,a)$ と近似される．学習は以下の式で表される誤差関数を最小化するようにして進められる．

© 2020 Information Processing Society of Japan

$$y_t = r(s_t, a_t) + \gamma \max_a Q_\theta(s_{t+1}, a)$$

$$L(\theta) = \mathbb{E}\left[\left(y_t - Q_\theta(s_{t+1}, a_t)\right)^2\right]$$

この式の微分は以下のように与えられ，この勾配を誤差逆伝播することによってパラメータを更新し，学習を行う．

$$\nabla_\theta L(\theta) = \mathbb{E}\left[\left(y_t - Q_\theta(s_{t+1}, a_t)\right) \nabla_\theta Q_\theta(s_t, a_t)\right]$$

DQN では，Q 値の推定にランダムノイズが含まれている状態で Q 値が最大となる行動を次状態の行動とすることによって，Q 値が過大評価されてしまうという課題があった．それに対して Double Deep Q-Network (DDQN) [7] では，行動を決める際に使用する Q 関数と，状態と行動から Q 値を求める際に使用する Q 関数を二つ用意することでこれを抑制した．この場合，誤差関数は二つの Q 関数 $Q_\theta, Q_{\theta'}$ を用いることで，以下のように変更される．

$$y_t = r(s_t, a_t) + \gamma Q_\theta(s_{t+1}, \arg\max_a Q_{\theta'}(s_{t+1}, a))$$

$$L(\theta) = \mathbb{E}\left[\left(y_t - Q_\theta(s_t, a_t)\right)^2\right]$$

2.1.3 最大エントロピー強化学習

通常の強化学習では以下のように表される累積期待報酬を最大化させるような方策 π を学習する．

$$\pi = \arg\max_\pi \sum_t \mathbb{E}_\pi[r(s_t, a_t)]$$

学習を進めている間は，行動選択時に Q 学習であれば Q 値の大きい行動の優先して選択する．これにより初期の段階で偶然 Q 値が大きな値となった行動を優先して取るようになり，方策が局所最適解に収束してしまう可能性がある．そのため，通常はエージェントが未知の状態を探索することを促すために行動に何らかのランダム性を持たせる手法を用いる．最大エントロピー強化学習の問題設定においては，以下のようにエントロピー項を導入したものを学習する方策として利用する．

$$\pi = \arg\max_\pi \sum_t \mathbb{E}_\pi[r(s_t, a_t) + \alpha \mathcal{H}(\pi(\cdot|s_t))]$$

$\mathcal{H}(\cdot|s)$ は状態が与えられたときの行動確率のエントロピーを表す．このエントロピー項を追加することで，エージェントの方策で特定の行動を取る確率が高くなって行動確率の分布が尖ってしまうことを抑制し，行動確率の分布を一様分布に近づける作用が働く．そのことにより，行動にランダム性を付与され，エージェントの探索を促すことができる．ここで α はエントロピー項の重みを決定するハイパーパラメータである．

最大エントロピー強化学習の考え方を用いた Q 学習である Soft Q-learning [8] においては，ターゲットとなる Q 値は以下のように表される．

$$\hat{Q}(s_t, a_t) = r_t(s_t, a_t) + \gamma\alpha \log \sum_{a'} \exp\left(\frac{1}{\alpha} Q(s_{t+1}, a')\right)$$

これにより，Q 関数のパラメータ更新のための誤差関数が以下のように定められる．

$$y_t = r_t + \gamma\alpha \log \sum_a \exp\left(\frac{1}{\alpha} Q_{\text{soft}}(s_t, a)\right)$$

$$L(\theta) = \mathbb{E}\left[\left(y_t - Q(s_t, a_t)\right)^2\right]$$

Haarnoja [9] らは最大エントロピー強化学習おけるエントロピー項の重み α を学習パラメータとした．そのために，以下のようなエントロピー制約付きの収益最大化問題として定式化した．

$$\max \mathbb{E}_\pi\left[\sum_{t=0} r(s_t, a_t)\right]$$

$$\text{s.t.} \quad \mathbb{E}_\pi[-\log \pi(a_t|s_t)] \geq \hat{\mathcal{H}}$$

$\hat{\mathcal{H}}$ は目標となる行動確率のエントロピーである．ここでパラメータ α の更新式は以下のように表される．

$$J(\alpha) = \mathbb{E}_\pi\left[-\alpha \log \pi(a|s) - \alpha\hat{\mathcal{H}}\right]$$

行動確率のエントロピーが報酬に対して寄与する重みを学習することによって，最適な方策が定まっていないときはよりランダム性の高い行動をとることで，環境の探索を進める．逆に方策の学習が進み最適な方策に近づき取るべき行動が定まってきた場合には決定論的に行動を選択するようになる．

2.2 階層型強化学習

強化学習において報酬が疎な環境といった学習が難しい環境に対応するための手法として，階層型強化学習が挙げられる．階層型強化学習ではモデルを上位方策と下位方策に分割する．上位方策は長い時間スケールで目標状態に到達する計画を行い，下位方策に対して何らか入力を与える．下位方策は上位方策からの入力をもとに実際に環境に対して取る行動を選択する方策を学習する．直接方策を学習するのでは達成困難なタスクを時間スケールの短いサブタスクに階層化・細分化することで，報酬が疎な環境においても方策を効率的に学習できるようになる [10], [11].

2.3 スキル学習

階層的強化学習においてモデルは通常環境から与えられる報酬を利用して学習が行われるが，外部からの報酬を用いずに下位方策をスキルとして学習する手法がある．このスキル学習の手法の一つに情報理論に基づいた報酬を与えるものがある．この問題設定では学習される方策 $\pi(a|s, z)$ が状態だけではなく使用するスキル z に依存するようにする．個々の選択されたスキル z にエージェントが観測する

© 2020 Information Processing Society of Japan

状態 s が依存するようにそれら二つの事象間の相互情報量が高くなるように学習を進める．これにより，エージェントは選択されたスキルに応じて，そのスキル特有の状態に到達するような行動を取るようになる．

そのために．環境の状態と選択されたスキルの相互情報量をもとに内部報酬として与え，強化学習の手法を用いて報酬を最大化するような方策を求める．ここで状態と選択されたスキルの相互情報量 $\mathcal{I}(s,z)$ について，以下のような変形を施す．

$$\begin{aligned}\mathcal{I}(s,z) &= \mathcal{H}(z|s) - \mathcal{H}(z) \\ &= \mathbb{E}_{s,z\sim p(s,z)}[\log p(z|s)] - \mathbb{E}_{z\sim p(z)}[\log p(z)] \\ &\geq \mathbb{E}_{s,z\sim p(s,z)}[\log q_\phi(z|s)] - \mathbb{E}_{z\sim p(z)}[\log p(z)]\end{aligned}$$

$p(z|s)$ をニューラルネットワークで $q_\phi(z|s)$ と近似して，カルバック・ライブラー・ダイバージェンスの非負性を利用することで変分下限とする．そして，報酬を以下のように設定し，これを強化学習の手法を用いて最大化する．

$$r_t = \mathbb{E}[\log q_\phi(z|s)] - \mathbb{E}[\log p(z)] \tag{1}$$

$q_\phi(z|s)$ の設計方法はいくつかあり，Gregor らが提案した Variational Intrinsic Control (VIC) [12] という手法ではスキルを利用した際のエージェントの初期状態と終端状態を利用して，式 (2) で表される内部報酬を決定しそれを最大化させた．Achiam らの Variational Option Discovery Algorithms (VALOR) [3] においては，エージェントがスキルを利用して行動する中で観測した状態すべてを利用して式 (3) で表される内部報酬を決定した．Eysenbach らの Diversity is all you need (DIAYN) [4] では，式 (4) に示すように，各時間ステップの状態をタイムステップごとに利用して報酬を生成した．この手法では MuJoCo [13] といったロボットの運動シミュレーション環境において，エージェントを前進させるだけではなく，後退や跳躍するといった多様なスキルを外部報酬なく学習に成功した．

$$r_t = \log q(z|s_0, s_T) \quad \text{(VIC)} \tag{2}$$

$$r_t = \log q(z|s_0, s_1, \cdots, s_T) \quad \text{(VALOR)} \tag{3}$$

$$r_t = \log q(z|s_t) - p(z) \quad \text{(DIAYN)} \tag{4}$$

3. 提案手法

我々はゲームのような離散的な行動空間においてスキルを学習する手法を考案した．図 2 に提案手法のモデルの概念図，Algorithm 1 に提案手法の疑似コードを示す．スキル学習を行う際に利用する内部報酬の生成には DIAYN の手法を利用している．識別器を用いて，エージェントが到達した状態から選択されたスキルの予測確率を求め，式 (1) にしたがって報酬を生成している．既存の手法である

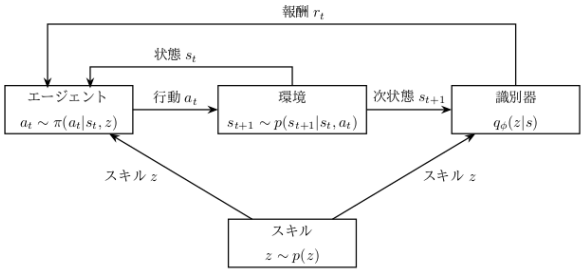

図 2: 提案手法のモデル

Algorithm 1 離散行動空間におけるスキル学習

```
while not converged do
  Sample skill z ~ p(z) and initial state s_0 ~ p_0(s)
  for t = 0, ··· , max_episode_steps do
    Sample action a_t ~ π_θ(a_t|s_t, z)
    Step environment s_{t+1} ~ p(s_{t+1}|s_t, a_t)
    Compute q_φ(z|s_{t+1}) with discriminator
    Set skill reward r_t = log q_φ(z|s_{t+1}) − log p(z)
    Update policy (θ) to maximize r_t with Soft Q-learning
    Update discriminator (φ) with Adam
  end for
end while
```

DIAYN においては生成された内部報酬を最大化するようにして各スキルの方策を学習するのに，最大エントロピー強化学習の手法を利用した Soft Actor-Critic (SAC) [9] を用いている．しかし，これは連続行動空間においてしか利用することができない．そのため，提案手法では SAC と同様に最大エントロピー強化学習の考え方を用いていて離散行動空間に適用可能な Soft Q-learning を使用している．

4. 実験

4.1 実験概要

本実験では, 既存のスキル学習の手法である DIAYN の報酬関数の手法を用いる．そして, Soft Q-learning を行い, 二次元グリッド空間および Open AI Gym の MountainCar-v0 というエージェントが離散的な行動を取る環境に適用することで，多様なスキルの学習が行えているかどうかを確認した．また，Atari の Freeway 環境においてもスキルを学習し，それらの学習されたスキルを用いて階層型強化学習を行うことで，スキルがその後のタスクにおいて有用であるかの検証を行った．

4.2 実験設定

4.2.1 環境

二次元グリッド空間

二次元グリッド空間の環境ではエージェントの初期状態が座標 (0,0) で与えられる．エージェントは各タイムステップごとに上，下，右，左，右上，右下，左上，左下の 8 つの移動方向からいずれか一つを選択し，その方向に移動することができる．ただし，20 %の確率で選択した移

© 2020 Information Processing Society of Japan

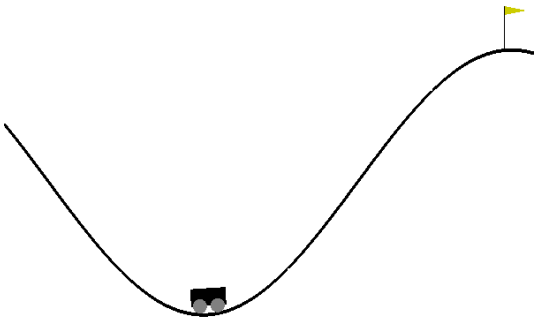

図 3: MoutainCar-v0 のプレイ画面

動方向に関わらずランダムな方向へ移動してしまう．そして，移動後のエージェントの座標が次の状態として環境から与えられる．x 方向か y 方向のいずれかの座標の絶対値が 1.0 を超える，またはステップ数が 100 を超えた場合にエピソードが終了する．

MountainCar-v0

MountainCar-v0 は図 3 のような環境である．このタスクの目標は谷底にある滑車を位置 0.5 にある山の頂上まで移動させることである．各時間ステップでエージェントは何もしないか，左右いずれかの方向に力を加えるという 3 つの行動から 1 つを選択して行うことができる．滑車の水平方向の位置とその時点での速度が状態入力として与えられる．滑車が山頂に到達するか，ステップ数が 200 を超えるとエピソードが終了する．初期位置は [-0.6, -0.4] の範囲でランダムに与えられ，初速度は 0 である．滑車は斜面方向に重力による力を受けているため，何もしないという行動を取ると谷底の方向に滑り落ちてしまう．

Freeway

図 4 に Freeway のプレイ画面を示す．このタスクでは画面左下のニワトリを操作して道路を横断させることが目的である．状態入力としてプレイ画面の 84×84 のグレースケール画像が与えられる．各時間ステップでエージェントは前進，後退，停止のいずれかの行動を選択できる．エージェントが道路を移動する車に衝突した場合少し手前の位置に戻される．フレーム数で 8192 フレーム分経過するとエピソードが終了する．本来の環境であればエージェントが道路を 1 回横断するごとに報酬が 1 与えられる．エージェントが道路を渡り切るとスタート位置に戻される．ゴールに到達しなければ報酬が全く与えられないため，報酬が疎な環境である．

入力を画像にするにあたってスキップ数 4 のフレームスキップを適用した．すなわち, 1 つの行動を 4 回繰り返し，その間の観測されるプレイ画面 4 フレーム分の観測データをまとめて 1 つの観測データとした．以降，画像入力の環境の学習における 1 ステップではプレイ画面における 4 フレームが経過するものとする．

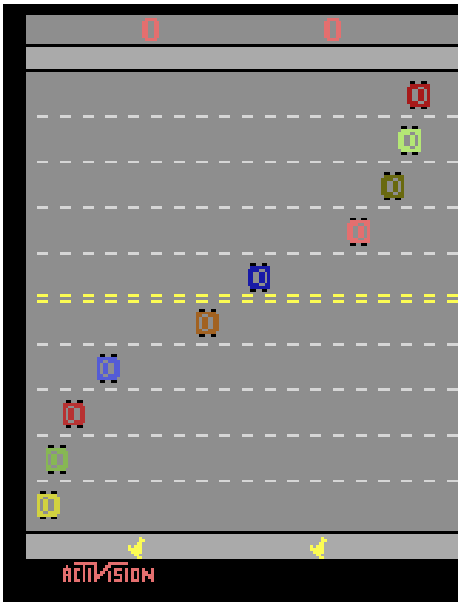

図 4: Freeway のプレイ画面

4.2.2 モデル設計および学習

エージェントのモデルは DDQN をベースとした Soft Q-learning としており，行動にランダム性を持たせるための行動のエントロピー項の寄与を決定する係数 α も学習パラメータとしている．このとき目標となる行動のエントロピーの値は $\hat{\mathcal{H}} = 0.9 \log |\mathcal{A}|$ とした．ここで，$|\mathcal{A}|$ はエージェントが各環境内で取りうる行動の選択肢の数である．エージェントのモデルの Q 関数および内部報酬を生成するための識別器はそれぞれ隠れ層ユニットのユニット数は 128，Freeway 環境のときのみ隠れ層のユニット数を 512 で．3 層のニューラルネットワークで構成し，最終的な出力層を除いて，各層には Relu 関数を活性化関数として適用した．スキルの選択は one-hot 表現としていて，状態入力に結合して入力している．識別器および Soft Q-learning におけるすべての学習可能なパラメータの更新は学習率 0.001 の Adam [14] を用いて行った．

学習するスキルの数は二次元グリッド空間環境と MountainCar-v0 のどちらについても 10 とした．Freeway 環境ではスキルの数は 20 とした．また，各エピソード開始時に使用するスキルを一様分布からランダムサンプルした．

スキルを行動として階層型強化学習をした際は，上位方策としては DDQN を用いた．エージェントは 10 ステップあるいはエージェントが終端状態に到達するまで選択されたスキルの方策に従って行動する．上位方策が受け取る観測する状態はスキルの方策が切り替わるまでに受け取る最後の観測データで，報酬はその間の累積報酬である．このとき，学習されたスキルのパラメータの更新は行われず，上位方策のパラメータのみ更新が行われる．

© 2020 Information Processing Society of Japan

4.3 実験結果

二次元グリッド空間

二次元グリッド空間の環境において計 500 エピソード分学習した結果を図 5 に示す．それぞれ色分けされた線がエージェントがスタート地点である原点から各スキルの方策にしたがって環境中を行動した際の移動の遷移を表している．

各エージェントの軌跡がスキルごとに二次元グリッド空間内を散らばるようにして移動していることから，スキルを多様性を持った形で学習ができていることがわかる．

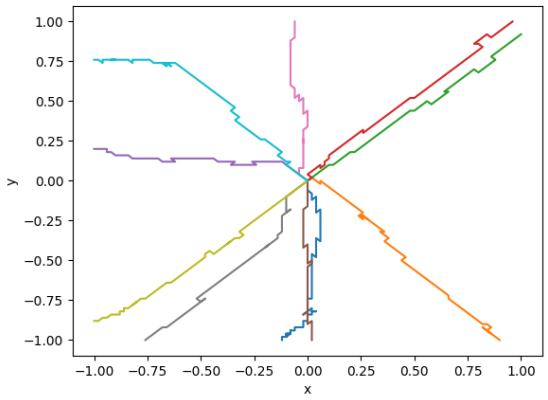

図 5: 二次元グリッド空間内におけるエージェントの移動の軌跡

MountainCar-v0

また，MountainCar-v0 の環境において，計 300 エピソード分学習した結果を図 6 に示す．学習後，エージェントがそれぞれのスキルの方策にしたがって行動した際の水平方向の座標の変化を表している．位置 0.5 にある点線は山頂の位置を表している．

MountainCar-v0 においても，スキルごとにエージェントが違った動きをしていることがわかる．エージェントの動きは大きく分けて 4 つに分類することができる．谷底付近でほぼ静止するもの，谷底の右側の斜面で振動するもの，谷底の左側の斜面で振動するもの，谷底を中心として大きく振動するものに分けられた．このうち，谷底を中心として大きく振動するもの中には一部峠を越えるものもある (水色と黄色の線)．そのため，MountainCar-v0 の環境における本来の目標である峠を越えて滑車を道下の頂上に導くという課題を達成する上でも，有用となるようなスキルを学ぶことができているといえる．

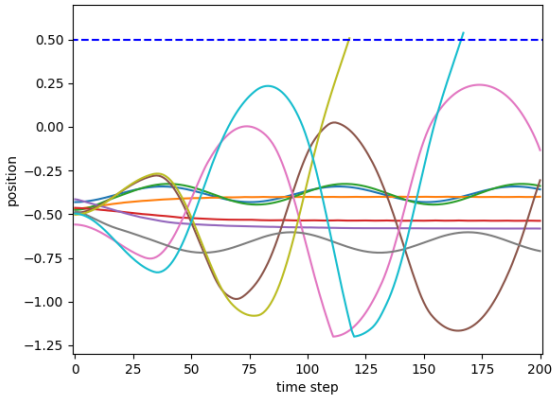

図 6: MountainCar-v0 における滑車の位置の時間変化

Freeway

図 7 に Freeway において，2M ステップ分のスキルの学習を行い，その後，学習したスキルを用いて 100 エピソード分の学習を階層型強化学習で行った際の結果を示す．階層化を行わず DDQN を用いて学習を行った場合，スキルをランダムに初期化した場合と提案手法によって初期化した場合についてそれぞれ 5 回ずつ学習を行った．ここで，DDQN においてはパラメータの更新頻度が階層化した場合と比べて 10 倍程度多くなっている．図中の実線は 1 エピソード間で取得した累積報酬の移動時平均の中央値を，色付きの領域は標準偏差を表している．提案手法で初期化した場合に高い割合で，報酬を得られるような方策を学習できている．

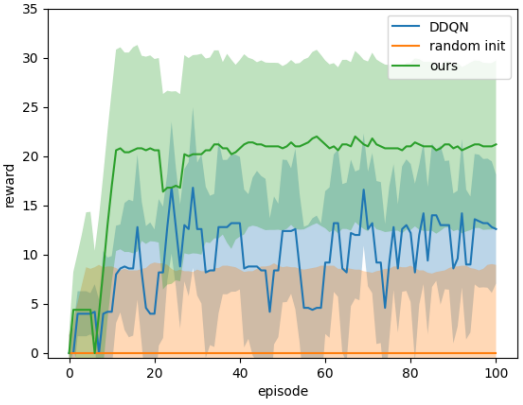

図 7: Freeway における学習結果

5. 考察

二次元グリッド空間と MountainCar-v0 といった状態入力が比較的単純な環境においては多様なスキルの学習を行えた．一方で，Freeway といった画像入力で状態空間が複雑な環境においてはスキルの学習においては報酬が得られるような方策が得られているものの，その全てが局所解に陥ってしまっていた．ここでの局所解とは，Freeway の環境で常に前進をすることである程度の報酬が得られてしまう状態のことである．つまり，エージェントの動きのみに注目してしまっていて道路を通過する車の位置などのエージェント以外の情報が考慮されるようなスキルが得られていないということである．これは Freeway 環境では状態空間が広くなるため，識別器による報酬の生成が難しくなる

© 2020 Information Processing Society of Japan

ためである．

また，学習されたスキルにおいて前進しているエージェントが後退をはじめるといったスキル内での行動の変化がほぼ見られなかった．これは Freeway 環境の二次元グリッド空間と MountainCar-v0 との環境の違いが表れているものとも考えられる．二次元グリッド空間や MountainCar-v0 は異なる状態へ向かうことによりスキルごとに違った行動が得られ，高い報酬を得ることができる．一方で，ゲームという環境においては新しい状態へと向かうだけではなく元の状態に戻ってくることも必要である．本手法では，次状態を識別することにより報酬を生成しているため，エージェントはスキルごとに異なる状態へと探索を進めようとする．MountainCar-v0 では位置が振動するなどして同じ状態を行き来しているが，そのような場合であっても速度の状態が異なるため，識別器には違う状態としてみなされる．Freeway の環境においても一度横断して戻ってくる場合でも画像に変化はあるものの，それらの状態が直接与えられる環境に比べて変化が捉えにくいため，前進と後退を組み合わせたスキルが見られなかったと予想される．

6. 関連研究

階層型強化学習においてのスキルは階層の下位に位置する方策であるスキルとスキルを行動として選択して環境内での行動を決定する上位の方策を同時に学習する方法である Option Critic [10] によって学習されてきた．この学習では外部から報酬を受け取りそれを最大化するため，報酬の設計が必要である．また，スキルと上位方策を同時に学習を進めるため，学習の際に上位方策によって有用な方策ばかりが選択されて，その環境では使えないスキルは選択されず学習が進まない．

そこでスキルを独立に学習するために情報理論に基づいたの報酬を与えて外部から報酬を与えずにスキルを学習する手法が用いられた．そこではエージェントが到達する状態とスキルの相互情報量の最大化を目的関数とする報酬設計を行っている．その中でも相互情報量の計算において状態が与えられたときのスキルのエントロピーを用いるものと [3], [4], [12] とスキルと現在の状態が与えられたときの次状態のエントロピーを用いるもの [15] がある．これらの手法は MuJoCo といった連続値制御のタスクで適用された．

離散行動空間上での教師なし学習では Hansen らの Variational Intrinsic Sucessor Features (VISR) [16] があり，方策を教師なしで Pre-train し，Few-shot での学習を成功させている．

7. おわりに

本稿では状態入力が単純な環境において各スキルが学習可能であるかを検証し，多様かつ実際に環境で行動する上でも有効なスキルの獲得を達成した．また，Freeway の画像という複雑な状態入力の際にも報酬を獲得するのに有用なスキルを学習できたが，最適に近い方策の学習が困難であるという結果が得られた．画像入力による広大な状態空間をスキルの学習に必要なものに制限することやゲームの探索に適した報酬の与え方に改善することが今後の課題として挙げられる．

参考文献

[1] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with deep reinforcement learning. In *Neural Information Processing Systems Deep Learning Workshop*, 2013.

[2] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, L Robert Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy P. Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of Go without human knowledge. *Nature*, Vol. 550, pp. 354–359, 2017.

[3] Joshua Achiam, Harrison Edwards, Dario Amodei, and Pieter Abbeel. Variational option discovery algorithms. *ArXiv*, Vol. abs/1807.10299, , 2018.

[4] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*, 2019.

[5] Chris Watkins and P. Dayan. Q-learning. *Machine Learning*, Vol. 8, pp. 279–292, 1992.

[6] V. Mnih, K. Kavukcuoglu, D. Silver, Andrei A. Rusu, J. Veness, Marc G. Bellemare, A. Graves, Martin A. Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, S. Petersen, C. Beattie, A. Sadik, Ioannis Antonoglou, H. King, D. Kumaran, Daan Wierstra, S. Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, Vol. 518, pp. 529–533, 2015.

[7] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[8] Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International Conference on Learning Representations*, 2017.

[9] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

[10] Pierre-Luc Bacon, Jean Harb, and Doina Precup. The option-critic architecture. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[11] Ofir Nachum, Shixiang S Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 3303–3313, 2018.

[12] Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. Variational intrinsic control. In *International Conference on Learning Representations Workshop*, 2017.

[13] Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. *IEEE/RSJ International Conference on Intelligent Robots and Sys-*

© 2020 Information Processing Society of Japan

tems, pp. 5026–5033, 2012.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] Archit Sharma, Shane Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised skill discovery. In *International Conference on Learning Representations*, 2020.

[16] S. Hansen, W. Dabney, André Barreto, T. Wiele, David Warde-Farley, and V. Mnih. Fast task inference with variational intrinsic successor features. In *International Conference on Learning Representations*, 2020.

© 2020 Information Processing Society of Japan