多様なロボットの挙動を学習する深層強化学習

○長 隆之（九工大/理研/ホンダ）

1. 序論

近年発展が著しい深層強化学習は様々なアプリケーションにおいて社会的な注目を集める成果を挙げている．囲碁のトップ棋士を破った AI に用いられたことで [5]，強化学習はいわゆる AI の中でも注目度の高い分野の一つとなっている．その一方で，深層強化学習による学習のプロセスは確率的なものであり，通常，学習過程で用いられる乱数発生器のシード値によって得られる方策の挙動が異なる．強化学習において最適な方策が複数存在しうることは強化学習の研究の初期から知られている [6]．そして，既存の深層強化学習の手法の多くは，そのうちの一つを確率的に学習することに加え，局所解に陥ってしまうことも多い．その結果，学習の実行時に用いる乱数の違いによって，異なる挙動を学習してしまう，ということが起きていると考えられる．しかし，そのように実行するごとに毎回異なる挙動が得られるようでは，ユーザーから信頼される「AI」とはなり得ない．

この問題に対処する方法の一つは，報酬関数を最大化する方策が一つに限定されるように，報酬関数を注意深く設計するというアプローチである．しかし，報酬関数の設計は往々にして困難であることが知られており，幅広いアプリケーションで用いられるようなシステムにおいてユーザーがそれぞれのタスクについて報酬関数を厳密に設計するということは難しい．

そこで本研究では，多様な解を同時に学習する深層強化学習アルゴリズムを提案する．多様な挙動を同時に学習することが可能になれば，確率的に学習結果が変わる要素を完全に取り除くことはできなくとも，得られた解の中に望ましい挙動が含まれる可能性を高めることができる．提案手法では，状態変数に加えて潜在変数を入力の一部とする方策を訓練し，潜在変数の値を介して，挙動のタイプを変更することのできる方策を得る．これにより，連続的に挙動のタイプを調整することが可能になり，ユーザーが好みの挙動を直感的に選択することができる．本研究では，OpenAI Gym [1] の歩行タスクおよび，Assistive Gym [2] のマニピュレーションタスクにおいて多様な挙動を学習できることを確認した．

2. 問題設定

以下に強化学習の問題設定について述べる．マルコフ決定過程 (以下，MDP) は， 状態空間 $\mathcal{S}$， 行動空間 $\mathcal{A}$, 状態遷移確率 $\mathcal{P}(\boldsymbol{s}_{t+1}|\boldsymbol{s}_t, \boldsymbol{a}_t)$， 報酬関数 $r(\boldsymbol{s}, \boldsymbol{a})$, 割引率 γ， 初期状態の確率密度 $d(\boldsymbol{s}_0)$ で与えられる配列 $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma, d)$ で与えられる．方策 $\pi(\boldsymbol{a}|\boldsymbol{s}) : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ は状態 $\boldsymbol{s}$ が与えられた際の行動 $\boldsymbol{a}$ の条件付き確率密度関数として定義される．また累積報酬は $R_t = \sum_{k=t}^{T} \gamma^{k-t} r(\boldsymbol{s}_k, \boldsymbol{a}_k)$ として定義される．強化学習の目標は，MDP として表現される環境下における試行錯誤を通じ，累積報酬の期待値 $\mathbb{E}[R_0|\pi]$ を最大化する方策を発見することである．

強化学習には，大きく分けて方策オン型と方策オフ型の２種類がある．本研究では，方策オフ型のアルゴリズムについて論じる．すわなち，方策 π は，リプレイバッファ $\mathcal{B}$ に蓄積された，試行錯誤によって得られた状態，行動および報酬のデータに基づいて訓練される．

3. 提案手法

本研究では，多様な挙動を一つの方策モデルで表現するため，以下のような形式の方策を考える．

$$\pi(\boldsymbol{a}|\boldsymbol{s}) = \int \pi(\boldsymbol{a}|\boldsymbol{s}, \boldsymbol{z}) p(\boldsymbol{z}) \mathrm{d}\boldsymbol{z} \tag{1}$$

ここで，$\boldsymbol{z}$ は潜在変数である．本研究では，この潜在変数の値を変えると異なるタイプの最適な行動が出力されるよう，方策を訓練する．これを実現するため，本研究では，累積報酬の期待値に加え，状態および行動のペアと潜在変数の間の相互情報量 $I_\pi(\boldsymbol{s}, \boldsymbol{a}; \boldsymbol{z})$ を最大化する．すなわち，以下のような問題を解くことによって，方策を訓練する．

$$\max_{\pi} \left(\mathbb{E}[R|\pi] + I_\pi(\boldsymbol{s}, \boldsymbol{a}; \boldsymbol{z}) \right) \tag{2}$$

このとき，相互情報量は以下のように定義される．

$$I_\pi(\boldsymbol{s}, \boldsymbol{a}; \boldsymbol{z}) = \iiint p_\pi(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z}) \times \log \left(\frac{p_\pi(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z})}{p_\pi(\boldsymbol{s}, \boldsymbol{a}) p(\boldsymbol{z})} \right) \mathrm{d}\boldsymbol{z} \mathrm{d}\boldsymbol{a} \mathrm{d}\boldsymbol{s}, \tag{3}$$

ここで，$p_\pi(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z}) = \beta(\boldsymbol{s}, \boldsymbol{z}) \pi(\boldsymbol{a}|\boldsymbol{s}, \boldsymbol{z})$ かつ $p_\pi(\boldsymbol{s}, \boldsymbol{a}) = \int p_\pi(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z}) \mathrm{d}\boldsymbol{z}$ であり，$\beta(\boldsymbol{s}, \boldsymbol{z})$ はリプレイバッファ内の状態 $\boldsymbol{s}$ と潜在変数 $\boldsymbol{z}$ の同時確率密度である．しかし，相互情報量 $I_\pi(\boldsymbol{s}, \boldsymbol{a}; \boldsymbol{z})$ の計算には $p_\pi(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z})$ および $p_\pi(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z})$ の密度推定が必要となり，直接計算することが容易ではない．そこで，相互情報量 $I_\pi(\boldsymbol{s}, \boldsymbol{a}; \boldsymbol{z})$ を直接最大化するのではなく，その変分下限を最大化する．ここで，密度 $p(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a})$ を近似するモデル $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a})$ を導入し，$\boldsymbol{\phi}$ をそのパラメータとする．すると，相互情報量 $I_\pi(\boldsymbol{s}, \boldsymbol{a}; \boldsymbol{z})$ の変分下限は以下のように得られる．

$$\begin{aligned} I_\pi(\boldsymbol{s}, \boldsymbol{a}; \boldsymbol{z}) &= H(\boldsymbol{z}) - H(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a}) \\ &= \mathbb{E}_{(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z}) \sim p_\pi} \left[\log p(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a}) \right] + H(\boldsymbol{z}) \\ &= \mathbb{E}_{(\boldsymbol{s}, \boldsymbol{a}) \sim \beta(\boldsymbol{s}, \boldsymbol{a})} \left[D_{\mathrm{KL}} \left(p(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a}) || q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a}) \right) \right] \\ &\quad + \mathbb{E}_{(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z}) \sim p_\pi} \left[\log q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a}) \right] + H(\boldsymbol{z}) \\ &\geq \mathbb{E}_{(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z}) \sim p_\pi} \left[\log q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a}) \right] + H(\boldsymbol{z}) \end{aligned} \tag{4}$$

ここで，$H(\boldsymbol{z})$ は潜在変数 $\boldsymbol{z}$ のエントロピーである．最適な密度モデル $q^*_{\boldsymbol{\phi}} = \max_{q_{\boldsymbol{\phi}}} \mathbb{E}[\log q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{a})] + H(\boldsymbol{z})$ が得

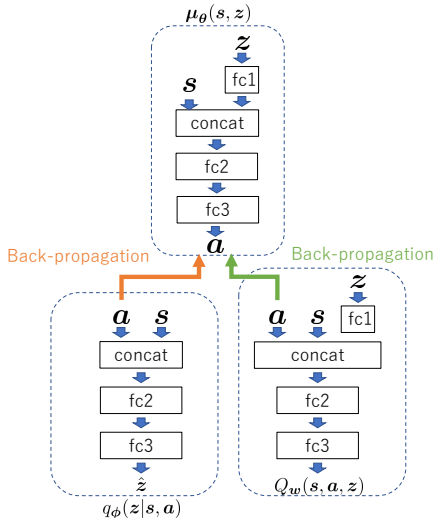

図 1　提案手法でのニューラルネットワークの構造.

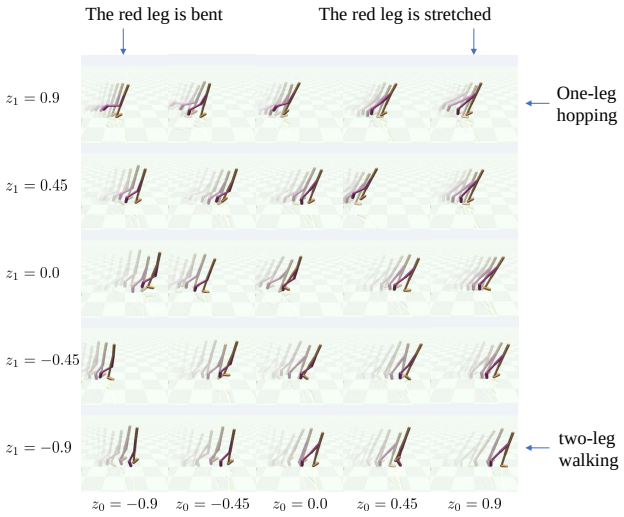

図 2 OpenAI Gym の Walker2d において学習された多様な歩行の例. それぞれのコマがそれぞれ異なる潜在変数の値により得られた歩行の挙動を示す.

られたとき，式 (4) の変分下限は相互情報量 $I_\pi(\boldsymbol{s},\boldsymbol{a};\boldsymbol{z})$ と等しくなる [3]．よって，式 (2) にある相互情報量は $\mathbb{E}[\log q_\phi(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a})]$ に置き換えることができ，式 (2) によって定式化された問題は以下のように置き換えることができる．

$$\max_{\pi,\phi} \left(\mathbb{E}[R|\pi] + \mathbb{E}[\log q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a})]\right). \tag{5}$$

本研究では，twin delayed deep determinitsic policy gradient [4] を拡張することによって式 (5) の問題を解く．提案手法では，潜在変数 $\boldsymbol{z}$ と状態 $\boldsymbol{s}$ を入力として行動 $\boldsymbol{a}$ を出力する決定論的な方策 $\mu_{\boldsymbol{\theta}}(\boldsymbol{s},\boldsymbol{z})$ を訓練する．ここで $\boldsymbol{\theta}$ は方策のパラメータである．また，潜在変数 $\boldsymbol{z}$ が与えられ，状態 $\boldsymbol{s}$ から行動 $\boldsymbol{a}$ をとり，その後方策 π に従った際に得られる累積報酬の期待値を $Q^\pi(\boldsymbol{s},\boldsymbol{a},\boldsymbol{z})$ とする．このとき，以下の目的関数を最大化するように方策のパラメータ $\boldsymbol{\theta}$，および密度モデルのパラメータ $\boldsymbol{\phi}$ を訓練する．

$$\mathcal{J}(\boldsymbol{\theta},\boldsymbol{\phi}) = \mathcal{J}_Q(\boldsymbol{\theta}) + \mathcal{J}_{\text{info}}(\boldsymbol{\theta},\boldsymbol{\phi}) \tag{6}$$

ここで，$\mathcal{J}_Q(\boldsymbol{\theta})$ は推定された Q 関数 $Q_{\boldsymbol{w}}(\boldsymbol{s},\boldsymbol{a},\boldsymbol{z})$ を用いて以下のように与えられる．

$$\mathcal{J}_Q(\boldsymbol{\theta}) = \mathbb{E}_{(\boldsymbol{s},\boldsymbol{z})\sim\mathcal{B}}\left[Q_{\boldsymbol{w}}(\boldsymbol{s},\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{s},\boldsymbol{z}),\boldsymbol{z})\right]. \tag{7}$$

また，式 (6) の第 2 項 $\mathcal{J}_{\text{info}}(\boldsymbol{\theta},\boldsymbol{\phi})$ は以下で与えられる．

$$\mathcal{J}_{\text{info}}(\boldsymbol{\theta},\boldsymbol{\phi}) = \mathbb{E}_{(\boldsymbol{s},\boldsymbol{z})\sim\mathcal{B}}\left[\log q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{s},\boldsymbol{z}))\right] \tag{8}$$

提案手法で用いられたニューラルネットワークの構造の概要を図 1 に示す．方策の訓練に当たっては，相互情報量に基づく目的関数に起因する勾配と，累積報酬の期待値に基づく目的関数に起因する勾配の二つが用いられる．また，実装に当たっては，式 (4) に基づいて $H(\boldsymbol{z})$ を大きくするため，潜在変数の事前分布 $p(\boldsymbol{z})$ には一様分布を用いた．また，潜在変数の値は各エピソードの最初に確率的にサンプリングされ，一つのエピソードの間では固定される．

4. シミュレーションにおける評価

4.1 OpenAI Gym を用いた評価

提案手法の有効性を確認するため，シミュレーションにおける学習の評価を行った．最初の評価では，OpenAI Gym に用意されている Walker2d のタスクを用いた．ここで，無数の解が存在するようなタスクを設定するため，報酬関数を修正した．具体的には，元の実装では学習エージェントが速く走れば走るほど高い報酬が得られる設定となっているが，本研究では，ある一定以上の速度になれば，それ以上速く走っても報酬が高くならないような報酬関数にした．この評価において，潜在変数は連続な 2 次元の変数とした．

提案手法を 3,000,000 ステップの試行錯誤による訓練ののちに得られた挙動を図 2 に示す．図 2 において，それぞれのコマがそれぞれ異なる潜在変数の値により得られた歩行の挙動を示す．横軸の変化が潜在変数の 1 次元目の値の変化に対応し，縦軸の変化が潜在変数の 2 次元目の値の変化に対応する．図 2 の通り，潜在変数の値を変えると，異なるタイプの歩行挙動が得られることが分かる．潜在変数の 1 次元目の値を-0.9 から 0.9 に連続的に変えると，歩行中の赤い足の曲げ具合が変化する．また，潜在変数の 2 次元目の値を-0.9 から 0.9 に連続的に変えると，2 本足での歩行からケンケンへと歩き方が変化する．以上の結果から，提案手法によって多様な歩行挙動を同時に得ることができたといえる．

このような歩行のタスクにおいて，意図的にケンケンのような歩き方をするように報酬関数を設計するのは，専門知識のないユーザーには困難であることが予想される．しかし，アプリケーションによっては，ユーザーが様々な歩き方を求めることも考えられ，提案手法はユーザーに挙動のタイプを選ばせる，編集させる，といったことを実現する手がかりになると考えられる．

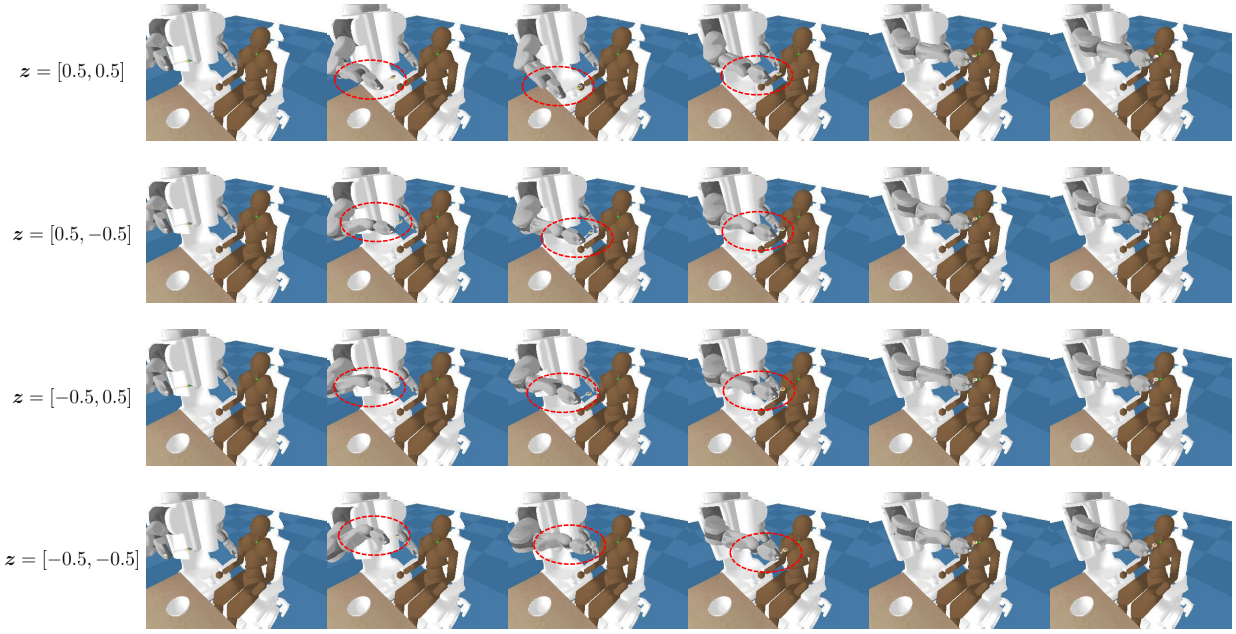

図 3　Assitive Gym の FeedingPR2_v1 において学習された多様な挙動の例.

4.2　Assistive Gym を用いた評価

歩行タスクだけでなく，マニピュレーションタスクに対しても提案手法が有効であるかどうかを確認するため，オープンソースのシミュレーション環境である Assistive Gym [2] を用いた評価を行った．ここでは，食事介助動作を模擬したタスクを PR2 で行う FeedingPR2_v1 と呼ばれる環境を用いた．先程の評価と同様，潜在変数は連続な 2 次元の変数とした．

得られた挙動の例を図 3 に示す．図 3 では，横軸に沿って時間が経過していくように図を配置しており，各行が異なる潜在変数の値に対応している．最終的にスプーンを持っていく位置は潜在変数の値を変えても大きな違いは生じないが，最終的な位置にアプローチする際，アームが経由する位置の高さやアームの姿勢が潜在変数の値によって変わることが分かる．この結果から，提案手法がマニピュレーションタスクにおいても，多様な挙動を同時に得るうえで有効であることが示唆された．

また，この Assistive　Gym では，ロボットによる介護動作に携わる研究者によって，人間が好むであろう動作を実現するように報酬関数が設計されている．その Assistive Gym においても多様な動作が学習されるということは，望ましい挙動を一意に決めるように報酬関数を決めることは難しいということを暗示している．実際，被介護者が，図 3 の一番上のパターンのような，下からスプーンが近づいてくる動きを好むこともあれば図 3 の一番下のパターンのような，上あるいは横からスプーンが近づいてくる動きを好む場合もあると考えられる．そして，数時間ときには数日の学習を要する深層強化学習の性質上，それぞれの被介護者の好みに合わせて報酬関数を設定し，動きを学習しなおすことは現実的ではない．提案手法のようなアプローチで無数の挙動を事前に用意しておき，ユーザーの要望に応じて挙動を調節できるようにしておくことは，深層強化学習を様々なアプリケーションに応用していくうえで重要なことであると考える．

5.　結論

本研究では，多様な最適解を同時に学習する深層強化学習のアルゴリズムを構築した．これに当たり，潜在変数と状態変数を入力として行動を出力するニューラルネットワークを，相互情報量の変分下限に基づいて訓練することを提案した．また，提案手法は歩行タスクおよび介護タスクにおいて評価され，多様な挙動を同時に学習できることが確認された．

参　考　文　献

[1] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba: “Openai gym,” arXiv, arXiv:1606.01540, 2016.

[2] Z. Erickson, V. Gangaram, A. Kapusta, C. K. Liu, and C. C. Kemp. “Assistive gym: A physics simulation framework for assistive robotics,” IEEE International Conference on Robotics and Automation (ICRA), 2020.

[3] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, : “An introduction to variational methods for graphical models,” Machine Learning, Vol. 37, pp. 183–233, 1999.

[4] S. Fujimoto, H. van Hoof, and D. Meger: “Addressing function approximation error in actor-critic methods,” International Conference on Machine Learning, pp. 1587–1596, PMLR, 2018.

[5] J. Schrittwieser, I. Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, E. Lockhart, D. Hassabis, T. Graepel, T. Lillicrap, D. Silver: “Mastering Atari, Go, chess and shogi by　planning with a learned model” Nature, Vol. 588, pp. 604–609, 2020.

[6] R. S. Sutton and A. G. Barto: “Reinforcement Learning: An Introduction,” MIT Press, second edition, 2018.