リカレントネット内の感度調整によって学習を行う強化学習

大分大学　○徳丸侑輝　柴田克成

Reinforcement Learning that Learns to Adjust Sensitivities in Recurrent Neural Networks

Yuuki Tokumaru and Katsunari Shibata, Oita University

Abstract: In this paper, we propose a new type of reinforcement learning in which dynamics of a recurrent neural network is trained directly according to the TD error. In order to learn the dynamics, we use "sensitivities", each of which is the magnitude of gradient vector of its output with respect to the input and expresses how much the neighbors of the present state diverge or converge in each neuron. When the TD error is positive, each neuron reduces its sensitivity to make the dynamics converge, and when it is negative, the neuron increases the sensitivity to make the dynamics diverge. We confirmed that an agent with a recurrent network learned two kinds of simple goal reaching task by the proposed learning method as a first report though the learning process should be analyzed more in detail.

1　序論

われわれの研究室では、多層のニューラルネット (NN) に対し、報酬と罰のみで学習を行う強化学習を組み合わせることで、人間が書くプログラムでは表現ができないレベルにも対応した、人間が持つ様々な機能が創発するのではないかと考え、研究を行っている。これまで、この手法による記憶や予測、認識とみられる機能の創発を確認している [1-3]。また近年は、深層強化学習という形でこのような考え方の有用性が示されている [4-5]。

現在は、人間が持つ高次の機能とされる思考の創発に向けた研究を行っている。思考についての定義は難しいが、本研究室では少なくとも "時間的に状態が次々と意味のある形で移り変わっていく、状態遷移を持ったダイナミクス" が必要と考えている。そして、このようなダイナミクスが創発するため、リカレント NN 内部にカオス性を持たせ、これを探索として用いると共に、学習によってそれを意味のある状態遷移にすることで、思考と呼べるようなダイナミクスに変化するのではないかという仮説を立てた [6]。しかし、この手法では内部のカオスダイナミクスを探索に利用しているため、探索成分を陽に取り出すことが困難であり、従来の強化学習をそのまま適用できないという問題点があった。本研究室ではこの解決策として、因果トレースという方法を用いた新しい強化学習を提案したが [6]、所望のダイナミクスを得るために重要な、フィードバック部の学習ができなかった一方で、カオス NN を用いた学習において、適度なカオス性を持つことが重要な鍵となることがわかった。

そこで先行研究では、カオスの特徴である初期値鋭敏性を基に各ニューロンにその状態の近傍の発散 · 収束の度合いを見る "感度" という指標を定義し、この指標を用いた学習によって、リカレント NN におけるカオスダイナミクスの生成、増大が可能であることを確認した [7]。その後、われわれはこの指標を状況の良し悪しによって学習を通じて上げ下げすることで、内部のダイナミクスそのものを直接学習することが可能となるのではないかと考えた。そして、強化学習によってこのような "ダイナミクスの学習" を行い、評価が良ければより収束して再現性を強化し、悪ければより発散して探索を強化するという全く新しい学習方法を考えた。

本研究では思考の創発に向けたさらなる基盤研究として、リカレントニューラルネットに対し、"感度" の指標に基づいた新たな強化学習の方法を提案し、それを入力方式を変えた 2 つの簡単なゴール到達タスクを学習させることで、まずは提案手法が強化学習として働くことを確認した。

2　学習方法

エージェントの処理は行動を生成する Actor 部と、その状態に対する評価を行う Critic 部の 2 つのニューラルネットで構成される。Critic は "TD 学習" により状態価値 $V(S_t)$ を出力し、次式 (1) の TD 誤差を計算する。

$$\hat{r}_t = r_{t+1} + \gamma \cdot V(S_{t+1}) - V(S_t) \tag{1}$$

$\hat{r}_t$: 時刻 t における TD 誤差
r_{t+1} : 時刻 $t+1$(次の時刻) で受ける報酬 or 罰
γ : 割引率

Actor のネットワークの各ニューロンには以下の式 (2) に示す指標 "感度" を導入する。"感度" はカオスの特徴である初期値鋭敏性を基にした、ニューロンの入力の変化量に対する出力の変化量の比の最大値であり、個々のニューロンにおける微小な信号変化の拡大 · 縮小を表すローカルな指標として用いられる。ただし、各ニューロンのフォワード計算は $u = \boldsymbol{w} \cdot \boldsymbol{x} = \sum_i w_i x_i$ と内部状態

u を求め、活性化関数を用いて $o = f(u) = \tanh(u + \theta)$ と出力を求める。θ は各ニューロンのバイアスである。

$$\text{感度 } |\nabla_{\boldsymbol{x}} o| = \sqrt{\sum_i \left(\frac{\partial o}{\partial x_i}\right)^2} = f'(u)|\boldsymbol{w}| \tag{2}$$

x_i ： ニューロンに入る i 番目の入力
o ： ニューロン出力
u ： ニューロンの内部状態
w_i ： i 番目入力に対応する重み値

本研究ではこの “感度” を用いて、状態の良し悪しに合わせて、学習を通じて感度を上げ下げし、ネットワーク内部の状態の変化 (ダイナミクス) の発散 · 収束を調整する。具体的には、Critic ネット出力に基づいて式 (1) によって算出される TD 誤差の正負に合わせて、中間層重み値およびバイアスを更新、学習を行う。ただし、式中の入力に対する重み値 $\boldsymbol{w^{in}}$ とフィードバックの重み値 $\boldsymbol{w^{fb}}$ をまとめて $\boldsymbol{w}$ と表記し、η は学習係数、$\boldsymbol{gsw_t}$ と $gs\theta_t$ は時刻 t での感度の重み値およびバイアスに対する勾配であり、$\overline{\boldsymbol{gsw_t}}$、$\overline{gs\theta_t}$ はそれを出力の変動に応じて更新 · 保持するトレース値である。

● TD 誤差が負、すなわち行動が悪かった場合
感度の勾配情報を保持する

$$\begin{aligned}
\boldsymbol{gsw_t} &= \nabla_{\boldsymbol{w}}|\nabla_{\boldsymbol{x}} o_t| \\
&= \frac{1-o_t^2}{|\boldsymbol{w}|}\left(\boldsymbol{w} - 2o_t|\boldsymbol{w}|^2\boldsymbol{x}\right) && (3)\\
\overline{\boldsymbol{gsw_t}} &= \left(1 - \frac{|o_t - o_{t-1}|}{2}\right)\overline{\boldsymbol{gsw_{t-1}}} + \frac{|o_t - o_{t-1}|}{2}\boldsymbol{gsw_t} && (4)\\
gs\theta_t &= -2(1-o_t^2)o_t|\boldsymbol{w}| && (5)\\
\overline{gs\theta_t} &= \left(1 - \frac{|o_t - o_{t-1}|}{2}\right)\overline{gs\theta_{t-1}} + \frac{|o_t - o_{t-1}|}{2}gs\theta_t && (6)
\end{aligned}$$

● TD 誤差が正、即ち行動が良かった場合
感度の勾配のうち、同一入力時に出力が変化しない成分を保持
※出力の維持拘束 = 内部状態を変化させない、すなわち $(\boldsymbol{w} + \Delta\boldsymbol{w}) \cdot \boldsymbol{x} = \boldsymbol{w} \cdot \boldsymbol{x}$ より $\Delta\boldsymbol{w} \perp \boldsymbol{x}$ となるため、重み値の更新量から入力ベクトル $\boldsymbol{x}$ の方向の成分を引いて直交成分のみを取り出す。

$$\begin{aligned}
\boldsymbol{gsw_t} &= \nabla_{\boldsymbol{w}}|\nabla_{\boldsymbol{x}} o_t| - \left\{(\boldsymbol{x} \cdot \nabla_{\boldsymbol{w}}|\nabla_{\boldsymbol{x}} o_t|)\frac{\boldsymbol{x}}{|\boldsymbol{x}|^2}\right\} \\
&= \frac{1-o_t^2}{|\boldsymbol{w}|}\left(\boldsymbol{w} - \frac{\boldsymbol{w}\cdot\boldsymbol{x}}{|\boldsymbol{x}|^2}\boldsymbol{x}\right) && (7)\\
\overline{\boldsymbol{gsw_t}} &= \left(1 - \frac{|o_t - o_{t-1}|}{2}\right)\overline{\boldsymbol{gsw_{t-1}}} + \frac{|o_t - o_{t-1}|}{2}\boldsymbol{gsw_t} && (8)\\
gs\theta_t &= -\frac{(1-o_t^2)\boldsymbol{w}\cdot\boldsymbol{x}}{|\boldsymbol{w}||\boldsymbol{x}|^2} && (9)\\
\overline{gs\theta_t} &= \left(1 - \frac{|o_t - o_{t-1}|}{2}\right)\overline{gs\theta_{t-1}} + \frac{|o_t - o_{t-1}|}{2}gs\theta_t && (10)
\end{aligned}$$

実際の更新は、各トレース値に TD 誤差 $\hat{r}$ および学習係数 η との積をとることで、TD 誤差が負の時は感度を上げてダイナミクスを発散させ、TD 誤差が正の時は感度を下げてダイナミクスを収束させる。実際の更新式を次式 (11)、(12) に示す。

$$\begin{aligned}
\Delta\boldsymbol{w} &= -\eta\hat{r}\cdot\overline{\boldsymbol{gsw_t}} && (11)\\
\Delta\theta &= -\eta\hat{r}\cdot\overline{gs\theta_t} && (12)
\end{aligned}$$

Actor ネットの出力層重み値は次式 (13) より更新する。

$$\Delta\boldsymbol{w_t} = \eta\hat{r}\boldsymbol{o_t^{out}}(\boldsymbol{o_t^{hidden}})^{\mathsf{T}} \tag{13}$$

Critic ネットは通常の 3 層の NN であり、(14) に示す教師信号 T_t^{out} を基に誤差逆伝播法より重み値の更新を行う。

$$T_t^{out} = \hat{r}_t + V(S_t) = r_{t+1} + \gamma\cdot V(S_{t+1}) \tag{14}$$

3 シミュレーション

3.1 設定

ここでは新しく提案した強化学習の方法で本当に学習ができるかどうかを確認するため、10×10 のフィールド上で、Agent がランダムなスタート位置からゴールに向かうことを目的としたゴール到達タスクを、センサ信号を変えた 2 通りのタスクで行う。タスクと使用するネットワークを Fig. 1 に、センサ信号の違いを簡単に示したものを Fig. 2 に示す。

Fig. 1 中の 2 つの Actor 出力はそれぞれ横 (x) 方向、縦 (y) 方向の移動量を表し、移動可能範囲が半径 1.0 の円になるように移動方向は変えずに大きさを調整した値にしたがって Agent が移動する。

ゴールは半径 1.0 とし、フィールド上の中心に固定し、Agent は 1 辺が 1.0 の正方形で、そのスタート位置は毎試行、ゴール · 壁からの距離がともに 1.0 以上のランダムな位置に設置する。外部からの乱数は入れず、内部のダイナミクスによる探索成分のみで Agent を探索させ、Agent の中心がゴール内に入ったときに報酬 0.8 を獲得する。加えて、フィールドの端の部分には壁を設置し、ここに Agent のいずれか一辺が接する、あるいはそれを超えるような行動をした場合、該当する壁から垂直方向に 0.25 離れたところに再設置してタスクを続行させる。また、タスク 1 では壁にぶつかった際、罰 -0.1 を与えた。

Agent がゴールに到達する、あるいはステップ上限の 100 ステップ通過するまでを 1 試行とし、20,000 試行の学習を行う。そして、Actor ネット中間層重み値 (入力 · 相互結合) およびバイアスの学習を前述の提案手法を用いることで、内部のダイナミクスの調整だけでゴールに到達できるようになるかどうか、入力のセンサ信号が異なっても学習が可能かを確認する。ただし、Fig. 2 におけるタスク 1 は、重み値学習促進の面から各入力を 4 倍して入れる処理を行なった。

次の表 1、2 に、今回用いた各パラメータを示す。なお、表中の “{1}、{2}” はタスクごとに異なるパラメータ

を用いているところで、それぞれタスク 1 、タスク 2 で用いた値であることを示している。

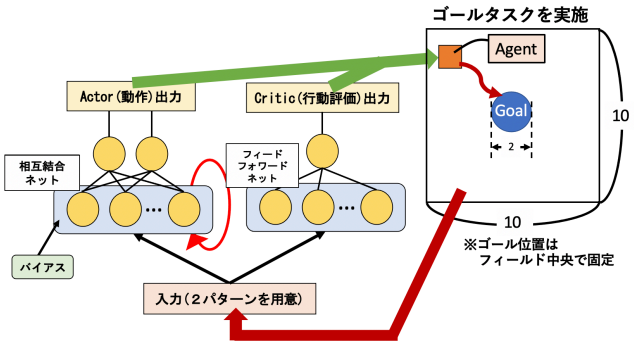

Fig. 1: Task and networks

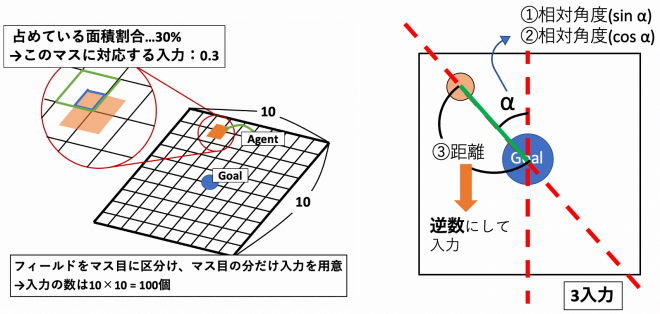

Fig. 2: Different type of sensor inputs employed in each task

Table 1: Parameters for simulation

名前		設定値	
		Actor	Critic
中間層ニューロン		{1}100 {2}50	60
割引率 γ		両タスクとも 0.85	
壁の罰		{1}-0.1 {2}0.0	
重み値の初期値	入力 w^{in}	{1}[-0.4,0.4]	[-0.5,0.5]
		{2}[-0.5,0.5]	
	バイアス θ	[-0.1,0.1]	-
	相互結合 w^{fb}	{1}[-0.4,0.4]	-
		{2}[-0.5,0.5]	
	出力層 w^{out}	[-0.2,0.2]	[0,0]
学習係数	入力 η^{in}	0.025	0.15
	バイアス η^{bias}	0.000075	-
	相互結合 η^{mut}	{1}0.0002	-
		{2}0.000225	
	出力 η^{out}	0.001	0.06

3.2 シミュレーション結果

Fig.3~Fig.5 にタスク 1 の、Fig.6~Fig.8 にタスク 2 の結果として, それぞれ学習曲線、学習後の 8 地点からのエージェントの軌道、学習時における感度の変化を示す。なお、両タスクは共に乱数系列 10 系列で行い、同様の結果となることを確認した。また、Fig.3、5 および Fig.6、8 の赤線は毎試行の値、青線はそれを 100 試行平均したものをプロットした。

●タスク 1 の場合

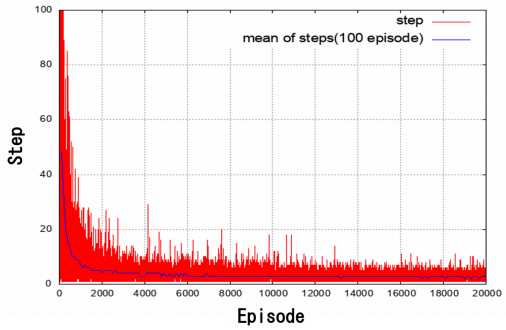

Fig. 3: Learning curve that shows the number of steps to reach the goal (Task 1)

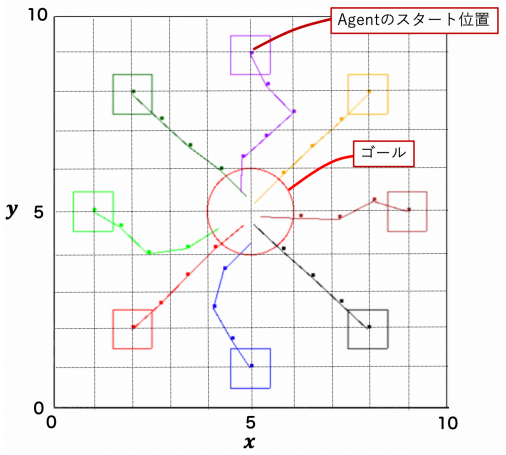

Fig. 4: Agent's behaviors in test trials after learning (Task 1)

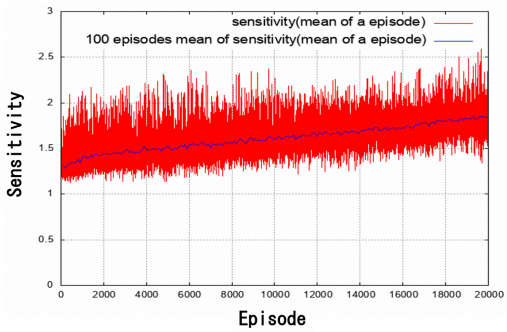

Fig. 5: Change of the average sensitivity during learning (Task 1)

●タスク 2 の場合

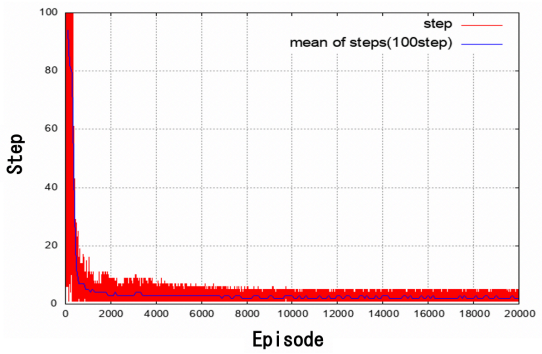

Fig. 6: Learning curve that shows the number of steps to reach the goal (Task 2)

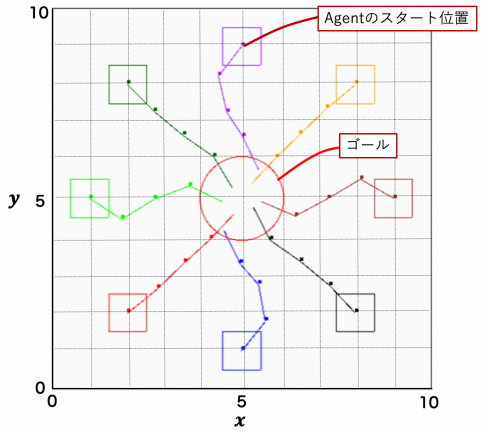

Fig. 7: Agent's behaviors in test trials after learning (Task 2)

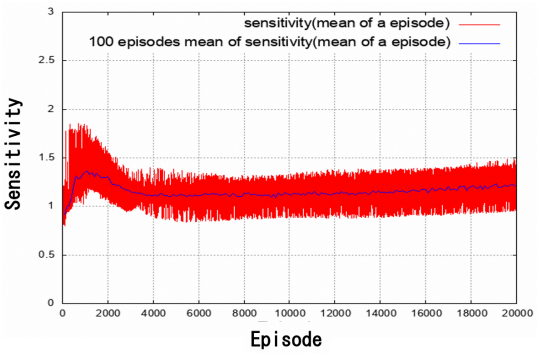

Fig. 8: Change of the average sensitivity during learning (Task 2)

両タスクの学習曲線 (Fig.3、6) や学習後のテスト試行における Agent の動き (Fig.4、7) から、両タスク共に学習し、少ない回数でゴールに向かうようになっていることがわかる。

次にそのときの感度のニューロン平均 (Fig.5、8) を見てみると、両タスク共に感度の学習機会 100 回平均 (各図の青線) が、学習初期から一貫して 1 を上回っていること、加えて学習を重ねるにつれて少しずつ増加し続けていることが確認できる。“感度” のニューロン単位の平均が 1 を上回るとネットワークにカオス性があるとみることができる。学習を重ねるにつれカオス性が減少し、収束していくという想定とは異なっており、より詳しい解析が必要である。

4 結論

本研究では、各ニューロンにおけるローカルな指標である “感度” を基にして、ダイナミクスを学習するという新しい強化学習法について提案した。そして、この学習法による簡単なタスクを強化学習によって学習可能であることを確認した。一方で、想定していた感度の減少は見られず、今後学習過程を詳細に解析する必要がある。

今後は、本学習がどのように進行しているか、ダイナミクスの変化などを見ていきたい。また、従来のカオスベース強化学習で学習できなかったリカレント部の重み値の学習の確認のため、記憶を要するタスクが学習可能かの調査を行なっていきたい。

謝辞

本研究は JSPS 科研費 (15K00360, 20K11993) および栢森情報科学技術振興財団研究助成金の補助を受けた。

参考文献

[1] K. Shibata, T. Kawano: Learning of Action Generation from Raw Camera Images in a Real-World-like Environment by Simple Coupling of Reinforcement Learning and a Neural Network, Proc. of ICONIP, Vol. 5506, pp. 755-762 (2009)

[2] K. Shibata, H. Utsunomiya: Discovery of Pattern Meaning from Delayed Rewards by Reinforcement Learning with a Recurrent Neural Network, Proc. of IJCNN, pp. 1445-1452 (2011)

[3] 柴田克成, 後藤健太: 予測を要して連続動作を含む柔軟な行動の Actor-Q 学習による獲得, FAN2013 講演論文集, pp. 86-91 (2013)

[4] V. Mnih, et al.: Human-level control through deep reinforcement learning; Nature, Vol.518, pp. 529-533 (2015)

[5] D. Silver, et al.: Mastering the game of Go with deep neural networks and tree search, Nature, Vol.529, pp. 484-489 (2016)

[6] 柴田克成: 深層学習が示唆する end-to-end 強化学習に基づく機能創発アプローチの重要性と思考の創発に向けたカオスニューラルネットを用いた新しい強化学習, 認知科学, Vol. 24, No.1, pp. 96-117 (2017)

[7] 徳丸侑輝, 柴田克成: リカレントネットにおける感度調整学習でのカオスダイナミクスの生成と維持, 第 38 回計測自動制御学会九州支部学術講演会予稿集, pp. 75-78 (2019)