社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

因果トレース
- 並列かつ主観的時間スケールの導入による過去の処理の効率的学習 -

柴田　克成†

† 大分大学工学部電気電子工学科　大分市大字旦野原 700 番地
E-mail: †shibata@oita-u.ac.jp

あらまし　途切れることなく続く時間の中で,「主観的時間」の基本概念の下，ニューラルネットで効率よく過去の処理を学習する汎用的な手法として「因果トレース」を提案する。各ニューロンの各結合に対して割り当てられたトレースは，そのニューロンの出力の時間変化の大きさに応じて当該入力を取り込み，出力に変化がないときはその値を保持する。これによって，過去の重要な事象 (イベント) のみをローカルなメモリに記憶し，保持し，現在の強化信号や教師信号から過去の処理に対して，効果的に学習を行うことができる。学習によって，何が重要な事象であるかも獲得され，ニューロン間で時間軸上のどの事象に反応するかの役割分担も促進される。時間軸という観点から見ると，ニューラルネット内部に，並列かつ一様でない主観的な時間スケールが存在することになる。本論文では，この因果トレースが，ニューラルネットを用いた評価値の TD 学習とリカレントネットの教師あり学習の両者に適用できることを示すとともに，適用方法の相違点を示す。さらに，継続時間が長い状態と短い状態が混在する状態価値の TD 学習タスクにおける因果トレースの高い学習性能と学習によるニューロン間での時間軸上の役割分担の促進を示す。
キーワード　因果トレース，ニューラルネット，強化学習，適格度トレース，主観的時間

Causality Traces　- Effective Retrospective Learning by Introducing Parallel and Subjective Time Scales -

Katsunari SHIBATA†

† Oita University, 700 Dannoharu, Oita, JAPAN
E-mail: †shibata@oita-u.ac.jp

Abstract　As a general method for effective retrospective learning in uninterrupted time based on the concept of “subjective time”, “causality trace” is introduced. A trace, which is assigned at each connection in each neuron, takes in the corresponding input signal according to the temporal change in the neuron’s output. This enables to memorize only past important events, to hold them in its local memory, and to learn the past processes effectively. Through learning, the criteria of what is important is acqquired, and the division of roles in the time axis among neurons is promoted. From the viewpoint of time, there are parallel, non-uniform and subjective time scales in the neural network. The causality traces can be applied to value learning with a neural network, and also to the learning of recurrent neural networks though the way of application is a bit different. A new simulation result in a value-learning task shows its effectiveness and the division of roles in the time axis among neurons through learning.
Key words　causality trace, neural network, rreinforcement learning, eligibility trace, subjective time

1. はじめに

われわれは時間の中で生きており，得られるセンサ信号は空間的に膨大であるだけでなく，時間的にも絶え間なく入ってくる。そのような世界に住むわれわれは，現在の報酬等から過去の処理を学習する素晴らしい能力を持っており，この能力を人工の学習システムで実現することが長年研究されてきている。

リカレントネットを用いると，過去の重要な情報の記憶や有用なダイナミクスの生成を，後に与えられる教師信号から学習することができる。しかし，最も良く使われる BPTT [2] では，基本的に過去のすべての状態を保持して過去にさかのぼって学習しなければならず，多くのメモリと計算量が必要となる。も

う一つ良く使われる RTRL [3] では，過去にさかのぼることなく，オンラインで学習できるが，結合の数のオーダー $O(n^2)$ (n: ニューロン数) を越える $O(n^3)$ のメモリ容量と $O(n^4)$ の計算コストが必要となり，局所的なメモリや計算では実現できない。

TD 学習に代表される強化学習 [1] も，現在の報酬や罰から過去の評価や適切な行動の時系列の獲得を可能にする。中でも，適格度トレース (Eligibility Traces) [1] を用いた TD(λ) 学習は，適格度トレースにその時々の情報を一定の割合で取り込み，過去の情報を少しずつ忘却していくことで，少ないメモリと計算量で効率的な過去の処理に対する学習を可能にする。

しかし，われわれ人間は，過去のすべての状態を一様に意識しているわけではなく，重要な事象 (イベント) だけを意識することで効果的な学習を可能にしているように見える。たとえば，1 日かけて飛行機で外国に行ったとき，その日の 1 秒ごとのすべての状態（センサ信号）を覚えているわけではなく，たとえば，乗り継ぎで迷子になった空港でのことを思い出して学習する。そこで筆者は，重要な事象での情報を重点的に記憶することで，効果的に学習することを考えた。時間軸の観点から言い換えると，重要な事象では時間がゆっくり経過し，そうでない場合は速く経過することに相当する。しかし，重要な事象が何であるかをシステムに前もって提供するとそのシステムは柔軟性を失ってしまう。そこで，重要な事象の抽出とそれに基づく柔軟な時間スケールを学習によって獲得するシステムの開発を目指した。時間の基準となる重要の度合いを学習者自身で獲得するという意味から，この時間スケールを「主観的」と呼ぶ。

過去に多くの研究で，学習における「時間」に焦点を当てている。その中の多くは，ペースメーカークロックや減衰メモリのように，時間を測るメカニズムを直接議論してきており，時間軸でのウェーバーの法則の説明が一つの目標となっている [5] [6] [7]。Nakahara らは，TD モデルに内部時間を導入することで，報酬が得られるまでの時間によって，二者択一の選択が反転する現象の説明を試みている [8]。しかし，評価値の学習を異なった時間スケール間で変換することに重点が置かれており，内部の時間スケールの形成方法には言及されていない。Daw らは，部分観測セミマルコフ過程での TD 学習則に基づいたドーパミン応答のモデルを提案した [9]。セミマルコフ過程は，一定間隔でない事象駆動型の状態遷移の枠組みを提供する。しかしながら，重要さの指標は事象かどうかの 2 値であり，学習者が事象を発見する方法も示されていない。また，Yamashita らは，異なる時定数の 2 つのサブネットからなる MTRNN を提案し，大きな時定数のサブネットにおいてより抽象的な状態を表現するダイナミクスが形成されることを示した [10]。しかしながら，時定数は各ニューロンにおいて定数であり，大きな時定数のサブネットでは状態変化は常にゆっくりであるため，長時間事象と短時間事象の両者を効率的に扱うことは難しい。

著者は，上記の考え方に基づいたリカレントネットの教師あり学習法 [11] [12] とニューラルネットを用いた評価値の TD(Temporal Difference) 学習法 [13] [14] をすでに提案した。両学習方法は少し異なるが，ニューラルネットを用い，過去の重要な事象を効果的に記憶して学習に利用する基本メカニズムは共通である。本稿では，両者に共通した基本となる手法を「因果トレース」と呼び，ニューラルネットで過去の処理を効率的に学習する汎用的な手法として提案する。そして，これを一般的に定式化し，両学習での共通点と相違点を明確にする。また，継続時間が長い状態と短い状態が混在する状態価値の TD 学習タスクにおいて，適格度トレースの場合と比較して，因果トレースの優位性を示すとともに，時間軸におけるニューロン間の役割分担が学習によって促進されることを示す。

2. 因果トレース (Causality Traces)

2.1 因果トレースの基本概念

著者は，図 1(a) のように，ほぼ毎日の通勤経路について,「交差点を右折」「橋を渡る」などの重要な事象は覚えているが，すべてのセンサ信号を覚えているわけではない。そのような記憶のおかげで，逆に効率的に学習できると考えることができる。問題は,「重要な事象」をどうやって定義するかである。「状態遷移」を重要な事象として定義することは容易である。しかし,「状態」を「センサ信号」に置き換えると，センサ信号の数は非常に多く，個々の信号も頻繁に変化する。たとえば，頭を動かすだけでも目からの視覚センサ信号は大きく変化する。しかし，状態の定義を与えるとシステムは柔軟性を失ってしまう。

そこで，図 1 の (b) のように,「状態遷移」を各ニューロンの出力の時間変化と定義する。「因果トレース」というメモリを各ニューロンの各結合部に配置する。そして，ニューロンの出力の時間変化の大きさにしたがって当該入力信号を取り込み，出力が変化しないときはその値を保持する。強化信号または教師信号が与えられると，その値を使って，過去にさかのぼることなく過去の処理の学習を行う。学習を通して，個々のニューロンは，学習に不要な事象には反応せず，重要な事象のみ表現するようになることで，因果トレースが遠い過去の重要な事象の情報を保持できるようになり，今度はそのトレースによってより遠い過去の処理を学習できるようになるという相乗効果が期待される。また，学習が進むことで，ニューロン間で異なった時間の事象に反応するような役割分担も期待される。

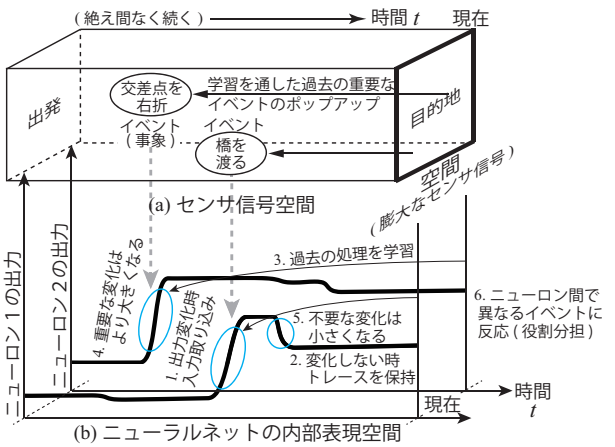


図 1　時間軸に広がりを持つセンサ信号空間とニューラルネットの内部表現空間。各ニューロンの出力の時間変化で重要な事象を定義し，因果トレースはその大きさに応じて入力を取り込む。

2.2 因果トレースの基本的な定式化

因果トレース c は，リカレントネットのフィードバック結合も含めたニューラルネット内の各結合に一つ配置し，

$$\frac{dc_{j,i}}{dt} = \frac{|do_j|}{dt}(in_i - c_{j,i}) \tag{1}$$

と，出力 (結果) の時間変化に影響を及ぼした入力の情報 (原因) を取り込んで保持する。ここで，o: ニューロンの出力，in: ニューロンへの入力，j: 注目ニューロンの番号，i: 注目ニューロンへの入力の番号とする。両辺から dt を消すことで，

$$dc_{j,i} = |do_j|(in_i - c_{j,i}) \tag{2}$$

と書き換えられる。これは，トレースが外部の時間軸の取り方によらないことを意味する。

一方，強化学習 [1] で使われる適格度トレース (Eligibility Trace) は，ニューラルネットにおいて過去の情報を保持し，過去の処理の学習をするための汎用的な方法として拡張できる。適格度トレース e は，一次遅れ系として一定の割合で入力信号を取り込んで保持するものであり，その時間発展の式は，

$$\tau\frac{de_{j,i}}{dt} = in_i - e_{j,i} \tag{3}$$

と書くことができる。ここで，τ は時定数である。式 (1) と式 (3) を比較すると，因果トレースでは，式 (1) 中の $1/\frac{|dO_j|}{dt} = \frac{dt}{|dO_j|}$ が，定数ではないものの，式 (3) 中の τ と同様に時定数として働いていると見ることができる。つまり，ニューロンの出力が大きく変化したときは時間が早く経過し，出力に変化が小さい時は時間がゆっくり経過することを意味する。ニューロンの出力 o はニューラルネットの外部から与えられたものではなく，外部入力を用いてニューラルネットの内部で生成されたものである。したがって，この時間スケールを「主観的」と呼ぶ。

図 2 は，因果トレースの時間変化を，デモ用に作成した入出力で，適格度トレースの場合と比較したものである。ニューロンの出力 (a) が $t = T_2$ と $t = T_3$ で大きく変化すると，因果トレース (c) は入力 (b) に大きく近づいている。一方，適格度トレース (d)(e) は出力 (a) に関係なく変化し，τ が大きい場合 (d) はゆっくりと，τ が小さい場合 (e) は素早く入力 (b) に追従していることがわかる。また，短い時間間隔で出力が大きく変化している T_3 と T_4 の間の入力の影響によって，T_5 における値が，適格度トレースの場合は τ の値によらずに大きな値に，因果トレースの場合は小さな値になっており，両者の性質の違いを見ることができる。

2.3 評価値の TD 学習における因果トレース [13] [14]

状態または行動価値を出力とする階層型ニューラルネットを用いた評価値の TD 学習において，因果トレースは適格度トレースの代わりに使うことができる。ここでは，簡単のため，離散時間でステップ幅を $\Delta t = 1$ とし，静的ニューロンモデル

$$o_{j,t}^{(l)} = f(u_{j,t}^{(l)}), \quad u_{j,t}^{(l)} = \sum_i w_{j,i}^{(l)} o_{i,t}^{(l-1)} \tag{4}$$

を使った階層型ニューラルネットを使って説明する。ここで，$o_{j,t}^{(l)}$ と $u_{j,t}^{(l)}$ は，時刻 t での第 l 層の j 番目のニューロンの出力と内部状態を示し，$o_{i,t}^{(l-1)}$ は一つ下の層からの i 番目の入力を示す。$f()$ は出力関数を表し，ここではシグモイド関数を用いる。現在の時刻 t での TD 誤差

$$\hat{r}_t = r_{t+1} + \gamma o_{1,t+1}^{(\mathrm{out})} - o_{1,t}^{(\mathrm{out})} \tag{5}$$

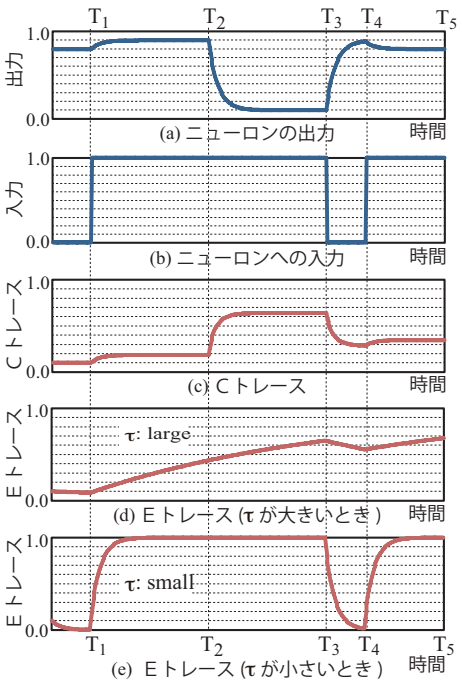
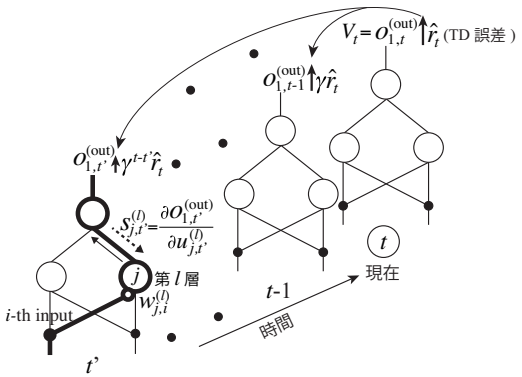
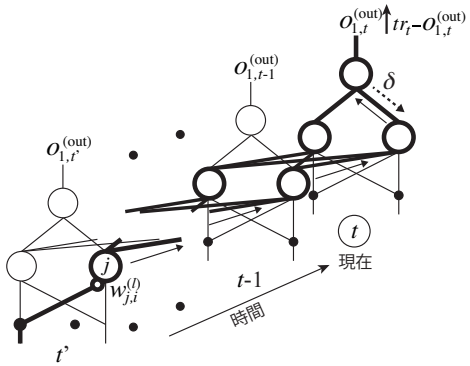


図 2　適格度トレースと因果トレースの時間変化の比較



(a) 階層型ニューラルネットでの評価値の TD 学習



(b) リカレントネットの教師あり学習

図 3　階層型ニューラルネットでの評価の TD 学習とリカレントネットの教師あり学習 (BPTT) の違い。太線は時刻 t' での $w_{j,i}^{(l)}$ が出力に与える影響

から，図 3(a) に示すように，現在の評価値 $o_{1,t}^{(\mathrm{out})}$ だけでなく，過去の評価値 $o_{1,t'}^{(\mathrm{out})}(t'=0,1,..,t)$ を，割引 TD 誤差 $\gamma^{t-t'}\hat{r}_t$ で同時に学習する。ここで，r は与えられる報酬，$\gamma(0 \leqq \gamma < 1)$ は割引率である。現在の時刻 t で，過去の出力 $o_{1,t'}^{(\mathrm{out})}$ を学習するため，過去の内部状態 $u_{j,t'}^{(l)}$ の出力ニューロン $o_{1,t'}^{(\mathrm{out})}$ への感度である $s_{j,t'}^{(l)} = \frac{\partial o_{1,t'}^{(\mathrm{out})}}{\partial u_{j,t'}^{(l)}}$ をトレースの中に含めなければならない。したがって，適格度トレースの更新式は, [4] を参考に，

$$e_{j,i,t}^{(l)} = \gamma\lambda e_{j,i,t-1}^{(l)} + (1-\lambda)s_{j,t}^{(l)}o_{i,t}^{(l-1)} \tag{6}$$

と表される。ここで，$\lambda = \left(1-\frac{1}{\tau}\right)$ である。感度 $s_{j,t}^{(l)}$ は，

$$s_{j,t}^{(l)} = \sum_k \frac{\partial o_{1,t}^{(\mathrm{out})}}{\partial u_{k,t}^{(l+1)}} \frac{\partial u_{k,t}^{(l+1)}}{\partial o_{j,t}^{(l)}} \frac{do_{j,t}^{(l)}}{du_{j,t}^{(l)}} = \sum_k s_k^{(l+1)} w_{k,j}^{(l+1)} f'(u_{j,t}^{(l)}) \tag{7}$$

と展開することで，出力ニューロンからの逆伝播で計算できる。式 (6) の λ を $1-|\Delta o_{j,t}^{(l)}|$ に置き換えることで，因果トレースは

$$c_{j,i,t}^{(l)} = \gamma(1-|\Delta o_{j,t}^{(l)}|)c_{j,i,t-1}^{(l)} + |\Delta o_{j,t}^{(l)}|s_{j,t}^{(l)}o_{i,t}^{(l-1)} \tag{8}$$

と計算する。ここで，$\Delta o_t = o_t - o_{t-1}$ である。時刻 t での各重み値の更新は，トレースの値を用いて，

$$\Delta w_{j,i,t}^{(l)} = \eta\hat{r}_t(e_{j,i,t}^{(l)} \text{ or } c_{j,i,t}^{(l)}) \tag{9}$$

と計算する。ここで，η は学習係数である。

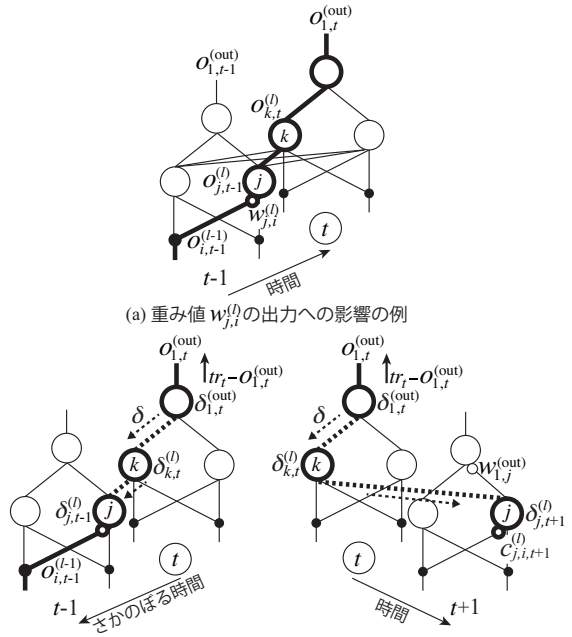
2.4 リカレントネットの教師あり学習における因果トレース [11] [12]

ここでは，例として，中間層ニューロンが互いにフィードバック結合を有するエルマンネットを用いて説明を行う。同じくリカレントネットのオンライン学習が可能な RTRL とは異なり，適格度トレースや因果トレースでは，それぞれの結合重み値が単にその結合が存在するニューロンへの局所的な影響を保持すれば良いので，メモリ容量や計算コストは重み値の数のオーダー $O(n^2)$ となる。図 3(b) のように，リカレントネットでは，過去の入力信号が，フィードバック結合を通して現在の中間層ニューロンの出力に影響し，さらに，静的マッピングを通して現在のネットワークの出力に影響する。したがって，中間層ニューロンだけに因果トレースを採用し，出力ニューロンは通常の BP 法で重み値の更新を行う。前節の評価値の学習の場合とは違い，過去のネットワークの出力を学習するのではなく，現在のネットワークの出力を教師に近づけるために過去の処理に対して学習を行う。よって，過去の出力ニューロンへの感度 s を保持する必要はなく、当該ニューロンへの感度を示す出力関数の微係数 $f'(u_j^{(l)})$ だけ保持すれば良い。したがって，各中間層ニューロンの各因果トレースは，

$$c_{j,i,t}^{(l)} = (1-|\Delta o_{j,t}^{(l)}|)c_{j,i,t-1}^{(l)} + |\Delta o_{j,t}^{(l)}|f'(u_{j,t}^{(l)})o_{i,t}^{(l-1)} \tag{10}$$

と，$f'(u_j^{(l)})$ と入力信号の掛け算したものを取り込んで保持する。フィードバック結合上にある因果トレースの更新は，N を入力信号の数とし，i が N より大きいときに，前のステップの中間層ニューロンからの入力であることを表すとすると，

$$c_{j,i,t}^{(l)} = (1-|\Delta o_{j,t}^{(l)}|)c_{j,i,t-1}^{(l)} + |\Delta o_{j,t}^{(l)}|f'(u_{j,t}^{(l)})o_{i-N,t-1}^{(l)} \tag{11}$$



(b) (a) の場合に対応する重み値の学習のための誤差信号の逆伝播

図 4　BPTT(b-1) と因果トレースを用いた学習 (b-2) での誤差信号の逆伝播の違い

と計算する。そして，各中間層ニューロンの各重み値は，現在の時刻 t において出力側から伝播してくる誤差信号 δ を用いて

$$\Delta w_{j,i,t}^{(l)} = \eta\delta_{j,t}^{(l)}c_{j,i,t}^{(l)} \tag{12}$$

と更新される。BPTT での伝播誤差信号 δ は，時刻 t での教師信号 tr_t との二乗誤差 $E = \frac{1}{2}(tr_t - O_{1,t}^{(out)})^2$ を用いて

$$\delta = -\frac{\partial E_t}{\partial u_{j,t'}^{(l)}} = (tr_t - o_{1,t}^{(out)})\frac{\partial o_{1,t}^{(out)}}{\partial u_{j,t'}^{(l)}} \tag{13}$$

と表され，過去に逆伝搬することで計算するが，ここでは，

$$\delta_{j,t}^{(l)} = (tr_t - o_{1,t}^{(\mathrm{out})})f'(u_{1,t}^{(\mathrm{out})})w_{1,j}^{(\mathrm{out})} + \sum_k v_{k,j,t-1}^{(l)}\delta_{k,t-1}^{(l)} \tag{14}$$

と前向きに計算していく。ここで，$v_{k,j}^{(l)}$ は，フィードバック結合の重み値 $w_{k,j+N}^{(l)}$ とニューロンの出力が変化したときの過去の $f'(u_k^{(l)})$ の値を保持したものであり，因果トレースと似た形で，

$$v_{k,j,t}^{(l)} = (1-|\Delta o_{k,t}^{(l)}|)v_{k,j,t-1}^{(l)} + |\Delta o_{k,t}^{(l)}|w_{k,j+N}^{(l)}f'(u_{k,t}^{(l)}) \tag{15}$$

と逐次計算していく。

誤差信号 δ の伝播方法には，BPTT 法と以下の 2 つの大きな違いがある。まず一つは，δ に当該ニューロンの $f'(u_j^{(l)})$ の値を含まないことである。過去の入力信号が当該ニューロンに影響を与えたのは現在ではなく，過去であるため，因果トレースが式 (10) のように過去の $f'(u_j^{(l)})$ の値を保持しているからである。2 つ目の大きな違いは，BPTT と違って，フィードバック結合を伝播する際に，時間をさかのぼらないことである。Fig. 4(a) のように，時刻 $t-1$ での重み値 $w_{j,i}^{(l)}$ が，時刻 t での k 番目の中間層ニューロンを通して現在の出力 $o_{1,t}^{(\mathrm{out})}$ に与えた影響に基づいて重み値 $w_{j,i}^{(l)}$ を更新することを考える。BPTT では，図 4(b-1) のように，誤差信号 $\delta_{j,t-1}^{(l)}$ を現在の誤差信号 $\delta_{k,t}^{(l)}$ か

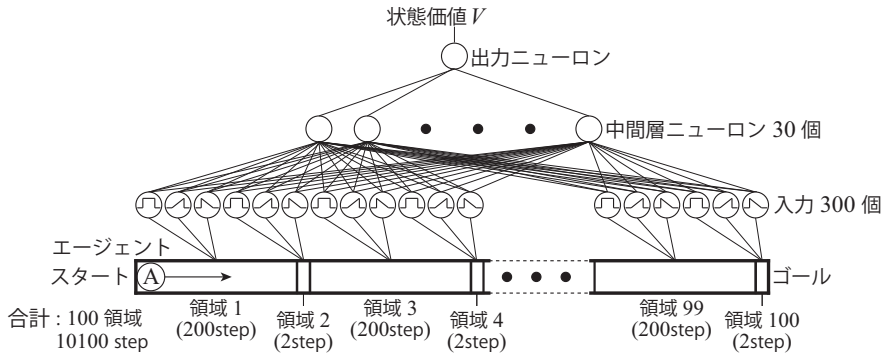


図 5　不均一な領域よりなる一次元タスクフィールドと状態価値を計算するニューラルネット

ら計算し，それと入力信号 $o_{i,t-1}^{(t-1)}$ の掛け算に基づいて重み値が更新される。これをさらに，時刻 $t-2$，$t-3$ と一つずつ時刻をさかのぼって学習していく。しかし，因果トレースを用いた学習方法では，図 4(b-2) のように，誤差信号がフィードバック結合を通って逆向きに伝播するものの，誤差信号 δ の伝播に使う v も因果トレース c も，1 結合に対し 1 つのメモリに過去の情報を圧縮して保持したものであり，伝播によって出力の計算と一緒に時間が進むため，毎ステップ時間をさかのぼって学習することなくリアルタイムで学習できる。ただ，このアルゴリズムはまだ完全に固まっておらず, [11] [12] とも異なる部分があり，今後さらに洗練させていく余地がある。適格度トレースの場合は，$|\Delta o_{j,t}^{(l)}|$ を $(1-\lambda)$ に置き換えれば同様な議論ができる。

3. シミュレーション

この節では，因果トレースの有効性を示すための，状態価値 (critic) を階層型ニューラルネットで学習させるシミュレーションを紹介する。図 5 のように，100 個の領域に分割された一次元のフィールドがある。エージェントは 200 ステップかけて奇数番目の領域を通過し，2 ステップで偶数番目の領域を通過することができる。エージェントは 300 個の入力信号を受け取り，それぞれの信号は，100 個の領域のうちの一つの領域だけで 0 でない値を取る。入力信号は 3 つのタイプに分類でき，そのうち一つは，その領域中で一定の値 1.0 をとる。それ以外の 2 つは，その領域中で，値が 0.0 から 1.0 まで一定の割合で連続的に上昇するか，逆に，1.0 から 0.0 まで連続的に減少する。中間層ニューロン数は 30 個である。出力層はニューロンは 1 個で，状態価値 (critic) を TD 学習に基づいて学習する。ここでは，通常の強化学習とは違い，探索 (試行錯誤) は行わず，エージェントは常にゴールの方向である右に向かって一定の割合で移動する。エージェントは，10,100 ステップ目にフィールドの右端であるゴールに到達し，1.0 の報酬を得る。割引率 γ は，最初のステップの理想的な状態価値が 0.2 になるように設定した。

タスクは, [13] [14] で行ったものと似ているが，ここでは，因果トレースの利点をより明確に示すために，領域の大きさに大きな差を付けた。さらに，入力-中間層間の初期結合重み値を変化させてみて，小さな初期重み値が良いことを確認した。そこで，適格度トレースの場合は，-1.0 から 1.0 の間の乱数の代わりに，すべての初期重み値を 0.0 とした。しかし，因果トレースの場合に初期重み値を 0.0 にしてしまうと，中間層ニューロンの出力が変化しなくなって，因果トレースも入力の値を全く取り込まなくなってしまうので，-0.1 から 0.1 までの小さな乱数を初期重み値として使用した。それでも，中間層ニューロンの出力の変化は依然小さいので，出力の変化 ΔO は，過去に経験したニューロンの出力の値域によって正規化した。中間層-出力層間の結合重み値は, [13] [14] のときと同様に，-1.0 から 1.0 の間の乱数とした。また，中間層と出力層の学習係数の最適化のために，両者を別々に，2 倍 2 倍と変化させていき，50 回目の試行の際の後述する大域的な誤差と局所的な誤差の積の，乱数系列を変えた 10 回分の平均が最小になるものを見つけて用いた。以下のグラフでは，1 ステップ学習 (トレースを使わない通常の学習，つまり，$\lambda = 0.0$)，適格度トレースのいくつかの λ の場合，それに因果トレースの場合の結果を示す。

図 6 は 2 種類の誤差の学習曲線を示す。(a) は，全 10,100 ステップにおける理想評価値との差の絶対値の平均であり，評価値の曲線と理想値との大域的な近さを表す。一方 (b) は，TD 誤差の絶対値の平均であり，主に隣の領域との評価値の滑らかさという局所的な誤差を表す。図 7 は，4 つの場合の，30 試行学習後の各状態に対する状態価値 (critic) を示す。1 ステップの場合 (a) や小さな λ の適格度トレースの場合 (b) は，ゴール状態からの評価の広がりがゆっくりであり，図 6(a) より，理想値からの差の減り方が遅いことがわかる。しかしながら，評価の曲線はあまり変動しておらず，図 6 (b) から，TD 誤差はあまり大きくなっていないことがわかる。一方，図 7 (c) の大きな λ での適格度トレースの場合では，評価の大まかな形は理想値に近いものの，周期的なパルス列が見られる。偶数領域ではエージェントは 2 ステップの短い時間しか滞在せず，奇数領域と比べて，入力の取り込みが小さい。そのため，図 6 (a) のように，出力は理想値に早い段階で近づくが，TD 誤差の方は，学習開始後すぐに大きくなる。学習前に TD 誤差が小さいのは，入力-中間層間の結合重み値がすべて 0.0 で，入力が常に一定値となるためである。因果トレースの場合は，図 7(d) のように，評価値はとても滑らかでほとんど理想曲線と一致しているように見える。学習前は，入力-中間層間の重み値が 0 ではないので，TD 誤差が他の場合より大きいが，図 6(b) のようにすぐに減少し，6(a) のように，大域的な評価の形も，最初は $\lambda = 0.999$ の場合が速いが，15 試行あたりで追い越している。

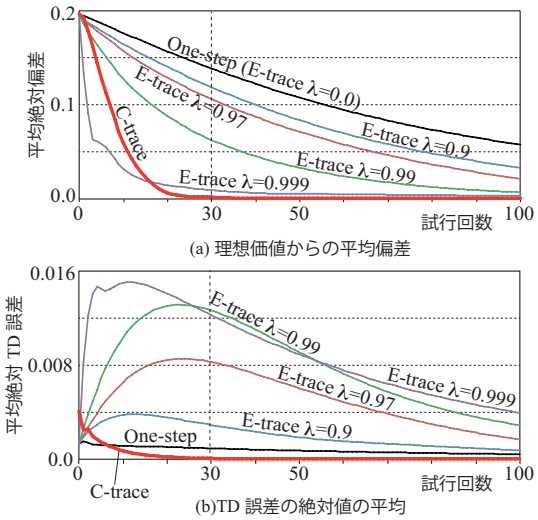


図 6　学習曲線。(a) 大域的な学習進行を表す理想評価値からの平均絶対偏差，(b) 局所的な学習進行を表す TD 誤差の絶対値の平均

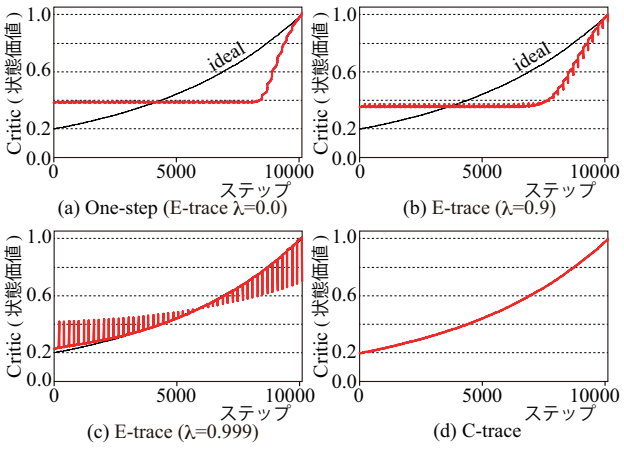


図 7　30 試行の学習後の状態価値の形状の比較

次に，個々の中間層ニューロンが過去の異なった事象に反応するようになるかを確認するため，入力-中間層間の重み値ベクトルの中間層ニューロン間の相関を観察した。出力ニューロンとの結合重み値の絶対値が大きな方から 8 個の中間層ニューロンを取り出し，8 個のうちのすべての 2 ニューロンの組み合わせでの平均相関の学習中の変化を観察した。平等な比較になるように，適格度トレースの場合も，初期の入力-中間層の結合重み値は-0.1 から 0.1 のランダムとした。２つの中間層ニューロン間で出力ニューロンへの重み値の符号が違う場合は，相関係数の符号を反転させた。図 8 に平均相関係数の変化を示す。因果トレースの場合，学習が早いため最初の立ち上がりが早いが，学習が進むと明らかに他のいずれの場合より相関が小さい。この傾向は初期重み値を変えても変化はなかった。適格度トレースでは，すべての中間層ニューロンが λ で決まる一定の割合で同じように入力を取り込み，式 (6) の感度 s だけが異なる。一方，因果トレースでは，式 (8) の Δo がニューロンごとに異なることで，他のニューロンとは違った時間の事象を取り込むことができるためと考えられ，ニューロン間での時間軸上の自律的な役割分担という非常に重要な能力を有することがわかる。

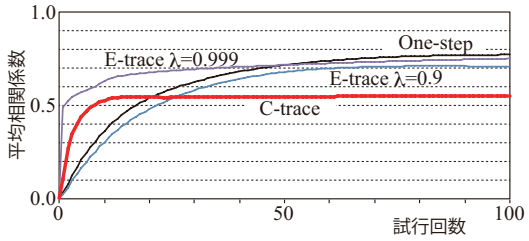


図 8　出力ニューロンとの重み値が大きい 8 個の中間層ニューロン間の入力-中間層間の重み値ベクトルの平均相関係数の学習時の変化

4. 結　　論

本論文では，ニューラルネットにおける汎用的な過去の処理の効率的学習法として「因果トレース」を提案した。トレースは外部の時間スケールに影響されることなく，各ニューロンの各結合において，並列かつ主観的に過去の値を取り込み，保持することで，評価値の TD 学習やリカレントネットの教師あり学習に利用できる。継続時間が長い状態と短い状態が混在するタスクにおける状態価値の学習で，因果トレースは，適格度トレースの場合と比較して，非常に高い学習性能を示し，さらに，時間軸上での役割分担がニューロン間で促進することも示した。

今後は，試行錯誤による行動の学習を含む強化学習における「因果トレース」の能力の検証と，リカレントネットでの教師あり学習と評価値の TD 学習の統一が課題として挙げられる。

文　　献

[1] Sutton, R. S. & Barto, A. G. (1998) *Reinforcement Learning: An Introduction*, A Bradford Book, The MIT Press

[2] Rumelhart, D. E. et al. (1986) Learning Internal Representation by Error Propagation, *Parallel Distributed Processing*, MIT Press, **1**, 318-364

[3] Williams, R.J. & Zipser, D. (1989) A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. *Neural Computation*, **1**, 270-280

[4] Bakker, B. et al. (2003) A Robot that Reinforcement-Learns to Identify and Memorize Important Previous Observations. *Proc. of IROS 2003*, 430-435

[5] Gibbon, J. (1991) *Learn. Motiv.*, **22**: 3-38

[6] Staddon, J.E.R. (2005) *Trends in Cog. Sci.*, **9** (7): 312-314

[7] Buhusi, C.V. et al.(2005) *Nature Reviews Neurosci.*, **6**(10): 755-765

[8] Nakahara, H. & Kaveri, S. (2010) Internal-time temporal difference model for neural value-based decision making. *Neural Computation*, **22**(12): 3062-106.

[9] Daw N.D., Courville A.C. & Touretzky D.S. (2006) Representation and timing in theories of the dopamine system. *Neural Computation*, **17**(7):1637-77

[10] Yamashita, Y. & Tani, J. (2008) Emergence of Functional Hierarchy in a Multiple Timescale Neural Network Model: a Humanoid Robot Experiment. *PLoS Comput. Biol.*, **4**(11).

[11] Shibata, K. et al. (1998) Simple Learning Algorithm for Recurrent Networks to Realize Short-Term Memories. *Proc. of IJCNN '98*, 2367-2372

[12] Samsudin, M.F. et al. (2007) Practical Recurrent Learning (PRL) in the Discrete Time Domain. *Neural Information Processing, LNCS*, **4984**: 228-237

[13] Shibata, K. et al. (2012) Differential Trace in Learning of Value Function with a Neural ... *Proc. of RiTA 2012*, 55-64

[14] 榎修志，柴田克成 (2012) ニューラルネット を用いた価値関数の学習における微分型トレースの提案. SICE SSI2012 講演論文集, 3B1-2.pdf, 396-401