方策勾配を用いた将棋の局面評価関数の教師付学習：静止探索の導入と AdaGrad の適用

古根村光[†1]　山本一将[†2]　森岡祐一　五十嵐治一[†1]

概要：我々はこれまでにコンピュータ将棋における局面評価関数の学習のために，「方策勾配を用いた教師付学習法」を提案してきた．本論文では，静止探索の導入により学習精度を高めるとともに，AdaGrad の手法を適用することにより学習速度を速めることを試みた．また，本手法では局面評価関数に含まれるパラメータによる方策関数の勾配ベクトルを計算して用いる．この勾配ベクトルは方策勾配法や TD 学習などの強化学習にも用いられている．本論文では，提案した教師付学習法とこれらの強化学習法との同時学習が可能であることを論じた．

キーワード：コンピュータ将棋，教師付学習，方策勾配，静止探索，AdaGrad

Policy Gradient Supervised Learning of Positional Evaluation Function in Shogi : Using Quiescence Search and AdaGrad

HIKARU KONEMURA[†1]　KAZUMASA YAMAMOTO[†2]
YUICHI MORIOKA　HARUKAZU IGARASHI[†1]

Abstract: We proposed a method of supervised learning using policy gradient to learn positional evaluation functions in *shogi*. We used quiescence search and AdaGrad to increase the accuracy and accelerate the learning speed. The proposed algorithm uses gradient vectors in the space of the parameters included in positional evaluation functions. These gradient vectors are also used in reinforcement learning such as policy gradient algorithms and TD learning. The proposed supervised learning method can be used with these reinforcement learning methods.

Keywords: Computer shogi, Supervised learning, Policy gradient, Quiescence search, AdaGrad

1. 研究背景と目的

近年のコンピュータ将棋の棋力向上は目覚ましく，プロ棋士以上の強さとなっている．この棋力向上の要因の一つとして機械学習による局面評価関数の学習の成功が挙げられる．その代表例が「Bonanza メソッド」[1]と呼ばれる教師付学習である．しかし，コンピュータ将棋がプロ棋士の棋力を超えてしまった現在では，従来のようにプロ棋士の棋譜を教師データとする教師付学習だけでは不十分である．そのための候補の一つとして強化学習が考えられる．強化学習による学習の試みとして，TD 法などの価値ベースの学習法の適用例[2][3]や，報酬のマルコフ性が必須でない方策ベースの方策勾配法の適用も提案されている[4][5].

また，強化学習の適用では，価値ベースと方策ベースの学習を VAPS アルゴリズム[6]により統一的に扱う融合方式も提案されている[7]．さらに，この融合方式において，教師付学習を同一の枠組みで行う「方策勾配を用いた教師付学習法」が提案され，実際に将棋に適用した実験の報告もある[8]．しかし，研究[8]では教師データとの最善手一致率が向上して定跡形は学習できたが，学習前のプログラムに勝つことは無く，棋力の向上には至らなかった．

本研究では，この研究[8]の問題点を解決し，棋力を向上させることを目的とする．このために学習時に静止探索を導入し，学習システムの方策の精度向上を図った．また，局面ごとの逐次処理であるオンライン学習から並列処理が可能であるミニバッチ学習[9]に変更して学習の高速化を図った．さらに，学習の収束を促すために AdaGrad[10]による学習率の自動調整を行った．

本研究で用いた「方策勾配を用いた教師付学習法」では局面評価関数に含まれるパラメータによる方策関数の勾配ベクトルを計算して用いる．この勾配ベクトルは TD 法や方策勾配法などの強化学習にも用いることができる．本論文では，強化学習との融合方式についても論ずる．

2. 関連研究

2.1 Bonanza メソッドによる教師付学習

「Bonanza」は 2005 年に保木により公開されたコンピュ

†1 芝浦工業大工学部情報工学科
Shibaura Institute of Technology
†2 株式会社コスモ・ウェブ
Cosmoweb Co., Ltd.

ータ将棋ソフトである[1][11][12]. Bonanza メソッドと呼ばれる Bonanza の局面評価関数の学習手法は，プロ棋士の棋譜を用いてコンピュータ将棋における大規模な局面評価関数パラメータの機械学習に成功した手法として知られている．現在，Bonanza 6.0 が最新版として公開されている．

Bonanza 6.0 の局面評価関数パラメータは駒価値及び KKP，KPP と呼ばれる三駒の位置関係に対するパラメータである．KKP は双方の玉とその他の駒一つ，KPP は片方の玉と玉以外の駒二つを表す．文献[1]ではこれらのパラメータを学習した際の学習則が述べられている．まずソフトの探索による最善手を棋譜中の着手と同じにすることを考える．ソフトの最善手と棋譜中の着手の一致度を測る目的関数を以下のように設計した．

$$J(P,v)=\sum_{p\in P}\sum_{m=2}^{M_p}T_p[\xi(p_m,v)-\xi(p_1,v)] \tag{1}$$

Pは棋譜中に現れる局面集合，M_pは局面pの合法手数，p_1は棋譜の着手後の局面でありpの子局面，p_mはm番目の合法手を指した直後の子局面，$\xi(p_m,v)$はp_m以降を Minimax 探索した際の評価値である．ここで関数$T_p(x)$をステップ関数とすれば，目的関数$J(P,v)$は全局面中において棋譜中の指し手よりも高評価をした指し手の総数となる．文献[1]では，$T_p(x)$をシグモイド関数にし，(1)に正則化項を追加した関数が目的関数として採用された．なお，学習の際には評価値を得るための Minimax 探索では深さ 2，3 程度の探索が行われた．

2.2 方策勾配を用いた教師付学習法の試み

方策勾配を用いた教師付学習[7]を将棋に適用した大串らの研究がある[8]. 文献では学習時に最小化する誤差関数をカルバック・ライブラー情報量（Kullback-Leibler divergence）により以下のように定義する．

$$\delta_{KLD}(S;\pi^*,\pi)\equiv\sum_{s\in S}\sum_{a\in A(s)}\pi^*(a|s)\ln\left\{\frac{\pi^*(a|s)}{\pi(a|s;\omega)}\right\} \tag{2}$$

ただし，π^*とπはそれぞれ教師と学習システムの着手決定方策を表す確率分布である．また，$A(s)$は局面sにおける合法手集合，ωは局面評価関数中のパラメータである．

ここでπを，局面sで指し手aを選択した際の最善応手手順（以下 PV）の leaf 局面s^*の静的評価値$E_s(s^*|a,s;\omega)$を用いた Boltzmann 分布で表す．

$$\pi(a|s;\omega)=exp\left(\frac{E_s(s^*|a,s;\omega)}{T}\right)/Z \tag{3}$$

$$Z\equiv\sum_{x\in A(s)}exp\left(\frac{E_s(s^*|x,s;\omega)}{T}\right) \tag{4}$$

ただし，Tは温度パラメータである．

(1)の最小化に勾配法を用いると次の学習則を得る[5][6].

$$\begin{aligned}\Delta\omega &= -\varepsilon\nabla_\omega\delta_{KLD}(\sigma;\pi^*,\pi)\\ &=\varepsilon\sum_{s\in\sigma}\sum_{a\in A(s)}\pi^*(a|s)\nabla_\omega\ln\pi(a|s;\omega)\\ &=\frac{\varepsilon}{T}\sum_{s\in S}\sum_{a\in A(s)}\{\pi^*(a|s)-\pi(a|s;\omega)\}\nabla_\omega E_s(s^*|a,s;\omega)\end{aligned} \tag{5}$$

(5)におけるεは学習係数であり，$\varepsilon>0$である．

Bonanza メソッドと本手法の目的関数を比較すると，Bonanza メソッドでは着手ごとに評価値同士を比較してソフトの評価の良し悪しを判断しているが，本手法では選択確率同士を比較している点が異なる．カルバック・ライブラー情報量は，確率分布間の近さの尺度として情報理論ではしばしば用いられている．教師データとして確率分布が与えられている場合には，(1)よりも(2)の方がそれらの情報をすべて利用できるという点でメリットがある[a].

大串らは，インターネット上で入手可能な 55,800 局の棋譜を教師として Bonanza 6.0 を用いて局面評価関数パラメータの学習を行った[8]．この実験では静止探索などは行わず，対象局面の各合法手を指した局面自体の静的評価値$E_s(s^*|a,s;\omega)$を用いた．また，駒割は学習させず Bonanza 6.0 の値を用いた．教師データに対するオンライン学習を 100 回繰り返し適用した結果，学習に要した時間は約 98 時間，教師の指し手との一致率が約 44.2%まで向上し，対局時には美濃囲いなどの将棋の「型」が確認できたとしている．また，課題として探索を深く行う必要性やパラメータの対称性を利用した計算の高速化を挙げている．

3. 本研究の学習方式

3.1 基本方針

先行研究[8]と同様に，Bonanza 型の局面評価関数のパラメータを学習する．ただし，駒価値は Bonanza 6.0 と同じ値を定数として使用し，KKP と KPP のパラメータを学習する．一般に静止探索などの探索を行えば静的評価値$E_s(s^*|a,s;\omega)$の精度が上がるため，学習効果も高まると思われる．また，対局時には静止探索の末端局面が評価されることが一般的であるため，静止探索を行う環境で学習済みのパラメータを使用するならば学習対象の局面も静止探

a) (1)では棋譜データしか用いないので第 2 候補以下の手に関する教師の情報は存在しない．しかし，教師がソフトである場合は第 2 候補以下の評価も利用できる可能性がある．

索の末端局面であることが望ましい．本研究では 2.2 の学習法に加えて，学習時の探索に静止探索を実装する[b]．さらに AdaGrad による学習係数の自動調整とミニバッチ学習を並列的に行うことにより学習の高速化を試みる．

3.2 探索部と静止探索

本研究では将棋ソフトの「芝浦将棋 Jr.」を用いた．芝浦将棋 Jr.は Bonanza 6.0 の局面評価関数を使用している．今回用いたのは， 2016 年開催の第 4 回電王トーナメントに出場したバージョンである（35 組中 24 位）．

芝浦将棋 Jr.の静止探索では駒を取る手の内，移動先で有利な駒の交換が行われる手を合法手として生成している．ただし被王手時にはこれらの内，王手を回避する手のみが合法手となる．合法手が無いならばその局面の静的評価値を返している．静止探索には $\alpha\beta$ 法を採用している．本研究では評価値の精度を上げるために静止探索を変更し，学習時は被王手時に駒を取る手でなく王手回避手を生成しており，合法手が無いならば詰みを表す定数を返している[c]．

3.3 AdaGrad による学習係数の自動調整

バッチ集合一組の勾配計算をマルチスレッドによる並列処理で行い，集合ごとにパラメータを更新した．勾配計算では AdaGrad により学習係数の自動調整を行った．

実装した学習則について文献[9][10]を基に説明する．i 番目のパラメータ ω_i の t 回目の更新を考える．ミニバッチにおける(5)の勾配和を $\Delta\omega_i^{(t)}$，平均勾配ベクトルを $g^{(t)}$ とする．この成分 $g_i^{(t)}$ を以下の様に定める．

$$g_i^{(t)} \equiv \begin{cases} 0, & \text{if } N_i^{(t)} = 0 \\ \dfrac{\Delta\omega_i^{(t)}}{N_i^{(t)}}, & \text{if } N_i^{(t)} \neq 0 \end{cases} \tag{6}$$

ただし i 番目のパラメータのミニバッチ内での出現回数を $N_i^{(t)}$ とする．このとき，パラメータ $\omega_i^{(t)}$ の更新量は $\Delta\omega_i^{(t)}$ でなく(7)の $v_i^{(t)}$ で与えられる．

$$v_i^{(t)} \equiv \begin{cases} 0, & \text{if } (g_i^{(t)})^2 = 0 \\ \eta \dfrac{g_i^{(t)}}{\sqrt{\sum_{\tau=1}^{t}(g_i^{(\tau)})^2}}, & \text{if } (g_i^{(t)})^2 \neq 0 \end{cases} \tag{7}$$

η は学習率のスケールを決める定数である．$(g_i^{(t)})^2$ を判定に利用しているのは，アンダーフローの発生を考慮しているためである．また，AdaGrad の実装により，学習係数 ε や温度パラメータ T は(7)の分母分子で約分される．

3.4 ミニバッチ学習における勾配計算の並列化

本研究ではミニバッチ学習における勾配計算を並列化することで高速化している．具体的な処理について図 1 を用いて述べる．

まず教師データをシャッフルし（①），類似局面が連続しないようにする．この教師データをスレッド数で均等に割り（②），各スレッドに分配する（③）．その後，各スレッドが各々に分配された局面に対して並列に勾配計算を行う．各スレッドで処理を終えた局面数の和がミニバッチサイズに達したとき（④），パラメータの更新のための同期を取るが，このとき各スレッドに残ったデータ数にはバラつきがある．そこで，分配されたデータを処理し終えた際の待機時間を減らすためにミニバッチごとにデータの再分配を行う（⑤～⑦）．この一連の処理を全教師データに対する勾配計算を終えるまで繰り返す（⑧）．

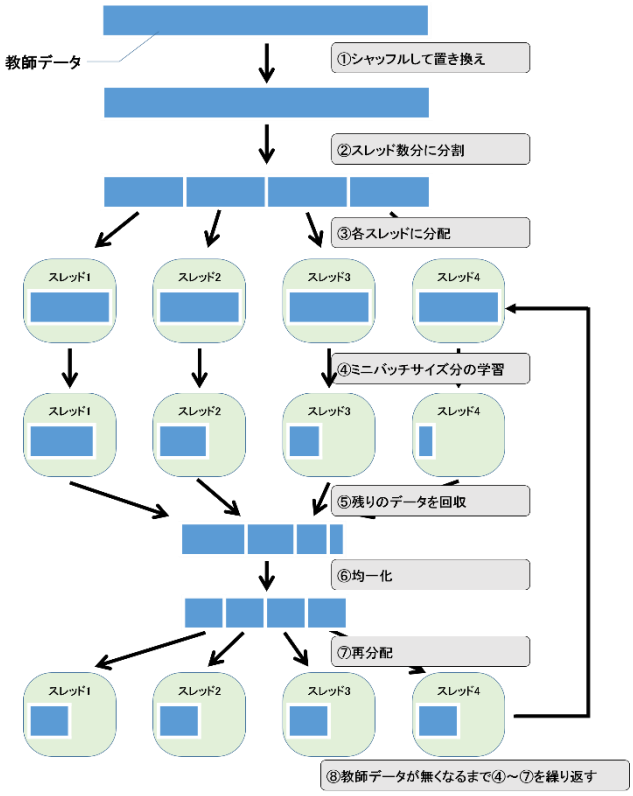

図 1　教師データの扱いと並列処理（4 スレッドの場合）

4. 学習実験

4.1 教師データの作成

先行研究[8]で使用された「2chkifu」[13]に含まれる 55800 局分の棋譜を用いた．局面単位に分割し，各局面 s において指された指し手 a 及び全棋譜中における指し手 a の着手回数 n_a の組を事前に用意し，教師データとして使用した．

教師の着手決定方策を表す π^* は，本研究では先行研究と同じく次の式で定義した．

b) 文献[8]の場合と同じく，読みの深さは 1 のままである．
c) 4 章での学習評価のための対局では，トーナメント出場版をそのまま用いたために王手回避手を生成しない方の静止探索を使用している．

$$\pi^*(a|s) = \frac{\text{局面}s\text{における指し手}a\text{の着手回数}}{\text{局面}s\text{の出現回数}} \quad (8)$$

教師データを解析したところ，着手が一つしかない局面は全体の約 98%に及んだ．これは中終盤の局面は他の棋譜中に出現することが稀であることが原因であると思われる．

4.2 実験条件

使用ソフトは 3.2 で述べた芝浦将棋 Jr.である．ただし，駒価値は Bonanza 6.0 の値で固定してある．root 局面での合法手生成の後，各々に対して最大深さ 6 の静止探索を行う．全教師データの学習を 1 エポックとし，ミニバッチサイズは 10 万局面とする．ただし 1 エポック内の最終更新の際はミニバッチサイズに満たなくてもパラメータの更新を行う．学習率ηは最終エポックの学習が終わるまでに初期値にリセットされることはない．各エポックの開始時に乱数により教師データを提示する順番をシャッフルする．学習時のコンピュータのスペックは，CPU が Intel® Xeon® E5-2603 v3 1.60GHz（6 コア×2CPU），RAM が 32GB，OS が Windows 7 Professional 64bit である．

4.3 教師データとの一致率の検証

本実験では$T = 5$，$\eta = 5$と設定し，4.2 の条件の下で 100 エポックの学習を行った[d]．学習後，エポックごとに全教師データとの最善手一致率を求めた．また，学習後のパラメータと Bonanza 6.0 のパラメータをそれぞれ使用した芝浦将棋 Jr.同士で 500 局の対局を行い，勝率を求めた．

全教師データに対する最善手一致率をエポックごとに出力した(図 2). ただし同一局面で複数の手が教師にある場合は，最も多く指された手を最善手とした．0 エポックでは約 27.1%だった一致率が 20 エポックでは約 38.3%となり約 11.2 ポイント上昇した．100 エポックでは一致率が約 43.3%となり，0 エポックから約 16.2 ポイント上昇した．この一致率の上昇は定跡形のパターンを学習した分だと考えられる．実際，駒価値だけでは現れにくい定跡形が学習後の対局(次での評価実験)には現れていた．なお，学習前（ 0 エポック）の一致率が 27%と高いが，教師データで最善手が駒を取る手である場合が多く（約 3 割），駒価値と静止探索だけで正解手を選択できたためと思われる．

次に，芝浦将棋 Jr.の探索プログラムを用いて Bonanza 6.0 のパラメータとの対局を行い，20 エポックと 100 エポックのパラメータを評価した(表 1). 対局時は通常の探索深さを 6，静止探索の深さを 30 とした．256 手続いた場合と連続王手でない千日手を引分けとした．

表 1 では 20 エポックで 19.2%だった勝率が 100 エポックで 28.2%に上昇している．後者の対局中の棋譜を見ると，初手から 7 六歩などの定跡にある手を指し，矢倉囲いなどの定跡形もしばしば現れた．このように，一致率や勝率

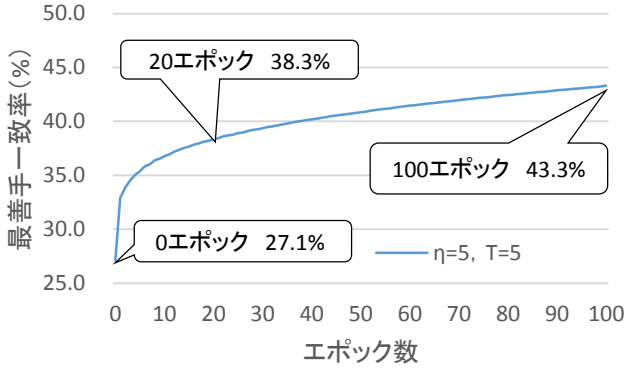

図 2　$\eta = 5$，$T = 5$における最善手一致率の推移

表 1　Bonanza パラメータとの対局（500 局，$\eta = T = 5$）

エポック数	勝	引分	負	勝率
0	11	9	480	2.2%
20	96	10	394	19.2%
100	141	10	349	28.2%

の上昇，対局時における定跡形の出現から学習が適切に行われたことが分かる．

4.4 学習の温度依存性の検証

温度パラメータTによって学習結果がどのように変化するかを調べた．$T = 40, 60, 80, 100$の場合の一致率と対局結果を図 3，表 2 に示す．なお本実験では$\eta = 20$とした[e]. 本実験も 4.2 の条件の下で 20 エポックの学習を行った．学習後，4.3 と同様の評価実験を行った．

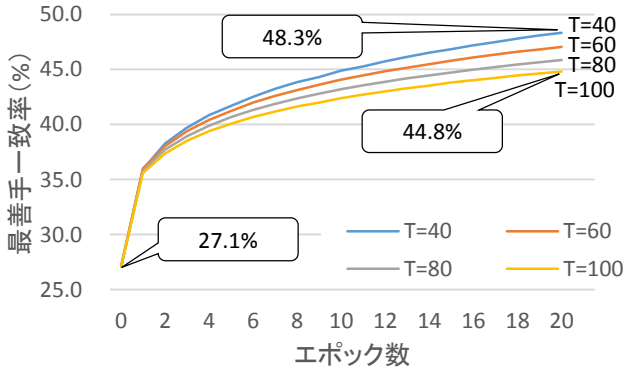

図 3　Tを変化させた際の最善手一致率の推移（$\eta = 20$）

表 2　Bonanza パラメータとの対局（500 局，$\eta = 20$）

T	勝	引分	負	勝率
100	179	8	313	35.8%
80	208	8	284	41.6%
60	227	16	257	45.4%
40	146	6	348	29.2%

d）1 エポック当りの学習時間は約 1.5 時間だった．

e）$\eta = 5,10,15,20,25$で学習を行ったが，$\eta = 20$で最も高い勝率を得た．

図 3 では，グラフの概形は同じで，Tの値が小さいほど一致率が高くなる傾向がみられる．最善手一致率では$T = 40$としたときに最も良い結果を得られたが，表 2 では勝率が最も良かった温度は$T = 60$である．一方，$T = 40$ では勝率が著しく低い．以上の結果から，最善手一致率の高さが必ずしも棋力の高さに結び付くわけではない．また，学習時の温度が勝率に大きな影響を与えているので，温度の値を適切に調整する必要があると考えられる．

さらに，本実験で最も高い勝率を得た $T = 60$ $(\eta = 20)$ で 100 エポックまでの学習を試み，同様の評価実験を行った．一致率は 100 エポック終了時点で 53.8%，勝率は 47.6% であった．

4.5 学習時における静止探索の効果

本研究で学習時に導入した静止探索の有効性を検証するための実験を行った．本実験も 4.2 の条件の下で 20 エポックの学習を行った．学習率ηは$\eta = 20$とし，温度パラメータTは 4.4 で最も高い勝率を得た$T = 60$とした．学習後，4.3 と同様の評価実験を行った．

一致率と対局結果を図 4，表 3 に示す．学習時に静止探索ありでは 20 エポックで一致率が約 19.9 ポイント上昇しているが，静止探索無しでは約 12.1 ポイントの上昇にとどまった．これに対し，勝率における静止探索の有無による差はもっと大きかった．すなわち，20 エポック学習後の一致率の差は 7.8 ポイントに過ぎないが，勝率の差は 40 ポイントを超えている．したがって，学習時の静止探索は，教師データとの一致率の向上よりは棋力向上に大きく寄与していることがわかった．

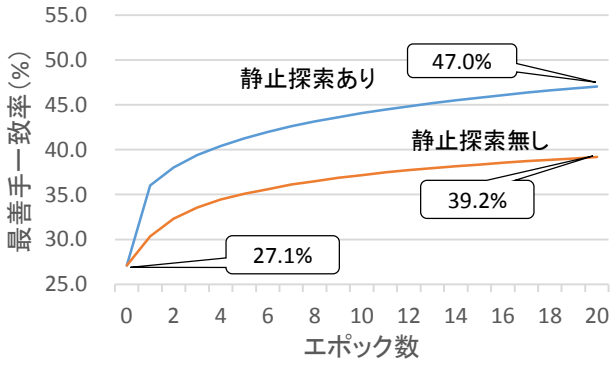

図 4　静止探索の有無と一致率の関係（$\eta = 20, T = 60$）

表 3　Bonanza パラメータとの対局（500 局, $\eta = 20, T = 60$）

学習時の静止探索	勝	引分	負	勝率
あり	227	16	257	45.4%
無し	17	1	482	3.4%

f) 将棋の場合，局面評価関数を 3 駒関係などの特徴量の重みの和で表すことが多い．その場合，勾配は重みそのものであり，計算が容易である．

5. 強化学習との融合についての考察

5.1 方策勾配を用いた教師付学習と強化学習

2 章，3 章で紹介した学習方式は「方策勾配を用いた教師付学習法」[5]であり，教師付学習であった．一方，方策勾配を用いた方策ベースの強化学習法として方策勾配法がある．さらに，この方策ベースの強化学習法と TD 学習などの価値ベースの強化学習法とを融合させた強化学習方式として VAPS アルゴリズム[6]がある．

これらの学習則は，すべて最終的には最善応手手順の leaf 局面s^*の静的評価値の勾配$\nabla E_s(s^*|a, s; \omega)$の値で表すことができる[f]．したがって，教師付き学習と強化学習とを同時に実行することが可能である．本章ではこのことについて詳しく考察する．

まず，文献[7]に従って，次の目的関数 $U(\omega)$を定義する．

$$U(\omega) = E[U_\sigma(\sigma; \omega)] \tag{9}$$

$$\equiv \sum_\sigma P(\sigma; \omega) U_\sigma(\sigma; \omega) \tag{10}$$

ただし，$U_\sigma(\sigma; \omega)$はエピソードσにおける学習の目的関数の値であり，$E[\cdot]$ はエピソードσの出現確率$P(\sigma; \omega)$による期待値操作を表している．ここで，$U_\sigma(\sigma; \omega)$を

$$U_\sigma(\sigma; \omega) \equiv \alpha R(\sigma) - \beta \delta(\sigma; \omega) \tag{11}$$

と定義する．(11)の$R(\sigma)$はエピソードσに対する報酬の総和であり，「エピソード収益」と呼ぶ．通常の強化学習で用いられる収益や割引収益もこれの一部に含めることができる．$\delta(\sigma; \omega)$は TD 誤差のような状態価値関数などが満たすべき条件に関する誤差であり，「エピソード誤差」と呼ぶ．α, β は 2 つの項の重みである．REINFORCE[14]などの方策ベースの方策勾配法は(11)の第 1 項だけを，TD 学習のような価値ベースの強化学習法は第 2 項だけを考えるが，VAPS アルゴリズムでは(11)の第 1 項と第 2 項の両方の項を考える．

(10)において，エピソードσの出現確率$P(\sigma; \omega)$は，プロ棋士の棋譜データベースなど学習システムの方策とは無関係な外部データを用いた学習であればωに依らない．しかし，実際の対局を用いた学習であれば学習エージェントの方策に，したがって ω に依存し，ω による微分を考える際には考慮する必要がある．本章では強化学習との同時実施を意図しているので，教師付学習においても棋譜データベースではなく，学習システム自身が対局して得られた行動列（棋譜）を用いることにする．その際には出現局面に対する正解手が教師データとして必要である．ここではその存在を前提として話を進める．

VAPS アルゴリズムでは(9)の目的関数 $U(\omega)$をωで微分し，その勾配方向へωを更新する．すなわち，

$$\Delta\omega = \varepsilon\nabla_\omega U(\omega) = \varepsilon\nabla_\omega E[U_\sigma(\sigma;\omega)] \quad (12)$$
$$= \varepsilon E[\nabla_\omega U_\sigma(\sigma;\omega) + U_\sigma(\sigma;\omega)\sum_t e_\omega(t)] \quad (13)$$

と表される．ただし，tは離散時刻（手番）を表し，

$$e_\omega(t) \equiv \nabla_\omega \ln\ \pi(a_t; s_t, \omega) \quad (14)$$

である. さらに, 学習をエピソード単位に行うことにして，ωを次のようにエピソード終了時に更新する．

$$\Delta\omega = \varepsilon[\nabla_\omega U_\sigma(\sigma;\omega) + U_\sigma(\sigma;\omega)\sum_t e_\omega(t)] \quad (15)$$

さらに，$U_\sigma(\sigma;\omega)$として(11)を(15)へ代入する．

$$\Delta\omega = \varepsilon[-\beta\nabla_\omega \delta(\sigma;\omega) + (\alpha R(\sigma) - \beta\delta(\sigma;\omega))\sum_t e_\omega(t)] \quad (16)$$

(15)の右辺の第 1 項は，エピソードごとの学習の目的関数$U_\sigma(\sigma;\omega)$を直接増大させる勾配方向なので，「勾配項」と呼ぶ．一方，(15)の第 2 項は$U_\sigma(\sigma;\omega)$の値そのものを増大させる方向ではなく，$U_\sigma(\sigma;\omega)$の値が大きいエピソードの生成確率，すなわち，エピソード中に選択した各行動の選択確率を高める方向を表している. 以下では「生成項」と呼ぶ．

(16)において，勾配項だけを考えて，エピソード誤差$\delta(\sigma;\omega)$として状態価値関数の 2 乗誤差δ_{MSE}を用いるのがTD(λ)法であり，Beal & Smith が将棋に適用している[2]．逆に，エピソード収益だけを含む生成項(α=1，β=0)のみを考えるのが方策勾配法である[7].

5.2 指導による教師付学習

将棋に適用されてきた教師付学習は，プロ棋士の棋譜データベースから局面と正解手の組を大量に作成し，学習用の訓練用データとして用いてきた．これは，プロ棋士の棋譜や棋書を読んで形や手筋を模倣する勉強法と言える．一方，将棋ではプロ棋士による指導対局や，実際に自分が指した棋譜をプロに見せて講評してもらうことも勉強の一方法である．学習システムに自由に行動選択を行なわせ，その選択に対して正解行動を示すことにより行う教師付学習を「指導による教師付学習」と呼ぶことにする．

この場合, エピソード誤差として(2)のδ_{KLD}を考えて, (16)の学習則で勾配項だけを用いれば良い. ただし, この場合, (2)の教師データの集合 S はエピソード（局）に出現した局面の集合 σ に置き換える必要がある．また，学習システムが対局した後に，出現局面において正解手と思われる指し手を用意する必要がある．これを人間が行うのは質と量ともに限界がある．棋力が高い将棋のオープンソフトを利用することが必要であろう．

さらに，指導による教師付学習は強化学習と同時に行うことができる．その際には，(16)において，エピソード収益$R(\sigma)$を考慮する（方策勾配法），エピソード誤差としてδ_{KLD}とδ_{MSE}の両方の和を取るなどの形で考慮すれば良い（前者が指導による教師付学習，後者が TD 法）[g].

この場合，(5)と(14)から分かるように，指導による教師付学習と方策勾配法に必要なのは，どちらも方策関数の対数微分ベクトルである．この勾配ベクトルは，(5)に示したように，方策関数として(3)の Boltzmann 分布を用いると，局面評価関数の勾配ベクトルの計算に帰着する．また，TD(λ)法[15]では状態価値関数の勾配ベクトルが必要となるが，Beal & Smith の方法[2]を用いると，これも局面評価関数の勾配ベクトルの計算に帰着する[h]．従って，本節で述べた方法を用いれば，対局での出現局面において各合法手の PV leaf 局面に含まれる特徴量パラメータに関する局面評価関数の勾配ベクトルを計算すれば，上記３つの学習が同時に行うことが可能である．

6. まとめ

本研究では，教師と学習システムとの行動選択確率を一致させるための「方策勾配を用いた教師付学習」を将棋に適用し，棋力の向上を確認することを目的として学習実験を行った．学習実験の際，先行研究で課題として挙げられていた探索の導入として静止探索を導入した．また，学習の高速化を目的として，局面ごとの逐次処理であるオンライン学習でなく並列化が容易であるミニバッチ学習の採用し，AdaGrad を採用して学習率調整による収束促進を図った．今回の実験では全教師データとの最善手一致率を 53.8%， Bonanza 6.0 のパラメータを用いたプログラムに対する勝率を 47.6%まで高めることができた．

今後の研究として学習時に深さ 2，3 程度の Minimax 探索を実装することや, 勝敗結果の利用などを予定している．さらに，強化学習をも組み合わせた学習実験を行うことも考えている．

参考文献

[1] 保木邦仁.「Bonanza 4.1.3」ソースコード. コンピュータ将棋の進歩⑥（松原仁 編著），共立出版, 2012, p. 1-23.

[2] Beal. D. F. and Smith. M. C.: Temporal difference learning applied to game playing and the results of application to shogi. Theoretical Computer Science. 2001. vol. 252. p. 105-119.

[3] 薄井克俊他.“TD 法を用いた評価関数の学習”，第 4 回 GPW, 1999, p. 31-38.

[4] 森岡祐一他. 方策勾配法と α β 探索を組み合わせた強化学習アルゴリズムの提案, GPW-12, 2012, p. 122-125.

[5] 五十嵐治一他. 方策勾配法による静的局面評価関数の強化学習についての一考察, GPW-12, 2012, p. 118-121.

[6] Baird, L. and Moore, A.. Gradient Descent for General Reinforcement Learning. Advances in Neural Information Processing Systems 11, 1999, p. 968-974.

[7] 五十嵐治一他. プロ棋士の棋譜データベースを用いない局面評価関数の学習法についての考察, 情報処理学会研究報告, 2015, vol.2015-GI-34, no.4, p. 1-8.

[8] 大串明他. コンピュータ将棋における方策勾配を用いた局面評価関数の教師付学習, GPW-15, 2015, p. 84-87.

[9] 海野裕也他. Chapter4 発展，オンライン機械学習，講談社,

g) 著者の知る限り，強化学習において生成項のエピソード誤差を考慮した研究例はこれまでにはなかったと思われる.

h) この手法については文献[7]に詳しく解説されている.

2015, p. 63-100.
[10] J.Duchi et al.. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. Journal of Machine Learning Research, 2011, vol.12, p. 2121-2159.
[11]“Bonanza - The Computer Shogi Program”. http://www.geocities.jp/bonanza_shogi/, (参照 2017-10-09).
[12] “コンピュータ将棋協会”. http://www2.computer-shogi.org/, (参照 2017-10-09).
[13] “棋譜データベース作成プログラムの使い方”. http://www.geocities.jp/shogi_depot/2chkifu.htm, (参照 2017-10-09).
[14]Williams, R. J.. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. Machine Learning, 1992, vol.8, p. 229-256.
[15] Baxter, J., Tridgell, A., and Weaver, L.. KnightCap: A chess program that learns by combining TD(λ) with game-tree search. Proceedings of the Fifteenth International Conference (ICML '98), 1998, p. 28-36.

