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

五十嵐治一[†1]　　森岡祐一　山本一将[†2]

本論文では強化学習の一手法である方策勾配法をコンピュータ将棋に適用する際に，全 leaf 局面の静的局面評価値をその局面への遷移確率値で重み付けた期待値を用いた指し手評価方式を提案する．探索木の各ノードにおける指し手の選択として Boltzmann 分布に基づく確率的戦略を採用すると静的局面評価関数に含まれるパラメータの学習則が再帰的に計算できる．しかしながら，処理対象とする leaf 局面数が大幅に増加するのでいくつかの近似解法も考案した．

Learning Static Evaluation Functions Based on Policy Gradient Reinforcement Learning

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

This paper applies policy gradient reinforcement learning to shogi. We propose a move's evaluation function, which is defined by the expectation of the values of all leaf nodes produced by the move in a search tree, that is weighted by the transition probabilities to the leaf nodes from the root node produced by the move. Boltzmann distribution function gives the probabilities of taking branches in a search tree instead of the minimax strategy. The learning rules of the parameters in the static evaluation function of the states can be calculated recursively. Since the number of leaf nodes for evaluation increases substantially, we also consider approximation methods to reduce the computation time.

1. はじめに

近年，コンピュータ将棋の実力はプロ棋士に迫るものがある [1]．この一因となっているのが，将棋ソフト Bonanza で提案された評価関数の自動学習である [2]．一方で，序盤定跡やプロ棋士の棋譜データベースを全く用いないで，将棋のルールと勝敗信号とだけを用いてコンピュータの棋力をプロ棋士レベルまで向上させることが可能であるかという問題が存在する．この問題に対する解決策の一つとして，教師付き学習ではなく強化学習により評価関数を学習する方法が考えられる．この代表的な強化学習法が TD(λ)法と TDLeaf(λ)法である．TD(λ)法はバックギャモンでは大成功を収めており [3]，TDLeaf(λ)法はチェスにおいて有効性が確認されている [16]．しかし，将棋では良い結果が報告されるまでには至っていない．

そこで本報告では“方策勾配法”と呼ばれる別の強化学習法の適用についての考察を行った．方策勾配法は報酬を自由に設定することが可能なので，棋力向上だけでなく棋風の学習など様々な学習目的に対して幅広く適用できる．

2. 方策勾配法による学習

2.1 方策勾配法とは

強化学習では，Q 学習や TD 法のように価値ベースの強化学習法がよく知られている [3]．一方，方策中にパラメータを入れておき，パラメータ空間内での期待報酬関数の最急勾配を計算することにより，方策を直接学習する強化学習法がある．Williams の REINFORCE アルゴリズム [4]や木村らの確率的傾斜法 [5]などである．また，MDP（マルコフ決定過程）を仮定して Q 値により上記の勾配関数を表現する方式 [6-8]や，自然勾配の利用 [9]も考案されている．これら一連の強化学習法は，“方策勾配法”と呼ばれ，例えば，Peters らの文献[10]中に簡潔にまとめられている．

本研究では，五十嵐らが提案している方策勾配法 [11]を用いる．この方式は，Williams[4]のエピソード単位の学習方式（episodic REINFORCE algorithm）に近いが，環境モデル（状態遷移確率と報酬）と方策に関する単純マルコフ性を必要としない．また，可変長のエピソードを取り扱うことができ，報酬もエピソード全体の状態・行動列を評価して計算する非マルコフ的な関数として与えることができる．さらに，一般的な方策勾配法では単位時間あたりの報酬を極大化することを目的とすることが多いが，本方式はエピソードあたりの報酬を極大化することを特徴としている．なお，これまでに追跡問題や粒子群を用いた最適化手法である PSO (Particle Swarm Optimization)等へ適用され，その有効性が確認されている [12),13)]．

2.2 方策勾配法による学習

t 回目(t=1,2,…,L_a)の手番局面 u_t において学習エージェント A が指し手 a_t を選択する確率(方策)を

$$\pi_a\left(a_t\middle|u_t;\omega\right)=\exp\left(-E_a\left(a_t,u_t;\omega\right)\middle/T_a\right)\middle/Z_a \qquad (1)$$

$$Z_a\equiv\sum_{a'}\exp\left(-E_a\left(a',u_t;\omega\right)\middle/T_a\right) \qquad (2)$$

とする．ただし，ωは評価関数中の学習パラメータ，T_aは温度パラメータである．(1)の右辺は Boltzmann 分布と呼ばれる確率分布関数であり，$E_a\left(a_t,u_t;\omega\right)$は手番局面 u_tにおける指し手 a_tの評価を表す指標であり”目的関数”と呼ぶ．一方，対戦エージェント B の方策は$\pi_b\left(b_t\middle|v_t\right)$と与えられて既知であるとする．ただし，$v_t$は対戦エージェントの t 回目(t=1,2,…,L_b)の手番局面であり，b_tはそのときの指し手

†1 芝浦工業大学工学部情報工学科
Shibaura Institute of Technology
†2 (株) コスモ・ウェブ
Cosmoweb Co., Ltd.

を表している．

一局の指し手と出現局面との時系列データ（棋譜）を”エピソード”と定義する．エピソード終了後，学習エージェントに報酬 r を与える．一般に，両対局者の指し手の決定は確率的方策によるものとする．したがって，学習エージェント A の指し手数（≡エピソード長 L_a）や報酬 r の観測値もエピソードごとに変動する．

ここでは文献[11]の方策勾配法を適用して，一局当たりの期待報酬値 $E[r]$を極大化するように学習パラメータ ω を学習する．それによれば，$E[r]$の勾配ベクトルが

$$\partial E[r]/\partial\omega = E\left[r\sum_{t=1}^{L_a} e_\omega(t) \right] \tag{3}$$

$$e_\omega(t) \equiv \partial \ln \pi_a\left(a_t|u_t;\omega\right)/\partial\omega \tag{4}$$

と表されることから，学習則として

$$\Delta\omega = \varepsilon \cdot r\sum_{t=1}^{L_a} e_\omega(t) \tag{5}$$

を用いる．ただし，ε は学習係数で小さな正数にとる．今，方策が(1)である場合，(4)の”特徴的適正度”$e_\omega(t)$は

$$e_\omega(t) = -(1/T_a)\Big[\partial E_a(a_t,u_t;\omega)/\partial\omega - \sum_{a'}\pi_a(a'|u_t;\omega)\partial E_a(a',u_t;\omega)/\partial\omega\Big] \tag{6}$$

と表される．

3. 探索と静的局面評価関数による指し手評価

指し手の評価は，読み（探索木の展開）を伴う方が精度が高いと考えられる．そこで，(1)の目的関数 $E_a(a_t,u_t;\omega)$ を，着手後の局面 $v=v(a_t,u_t)$ではなく，探索木 $G_D(a_t,u_t)$の末端（以下，leaf 局面)の評価値を用いた関数とする．ここで，$G_D(a_t,u_t)$は局面 $v(a_t,u_t)$を root とする深さ D の探索木で，学習エージェント A の手番から次の手番までを深さの 1 単位とし，出現局面 u_t を深さ 0 の，$G_D(a_t,u_t)$の leaf 局面を深さ D の A の手番局面とする．

本論文では，以下の“指し手評価の期待値”

$$E_a^*(a_t,u_t;\omega) \equiv \sum_{u\in U_D(a_t,u_t)} P(u|a_t,u_t;\omega)E_a^s(u;\omega) \tag{7}$$

を(1)の目的関数 $E_a(a_t,u_t;\omega)$として用いることを提案する．ただし，$U_D(a_t,u_t)$は探索木 $G_D(a_t,u_t)$の全 leaf 局面の集合を，$E^s_a(u;\omega)$は leaf 局面 u での静的局面評価関数である（図 1）．

ここで，全幅探索はもちろん，通常用いられる min-max 探索（またはα β 探索）やヒューリスティクスによる枝刈りを行う選択探索の他，モンテカルロ探索も(7)の右辺の期待値を厳密あるいは近似的に計算していると解釈できる．

通常，ゲーム木探索における指し手評価では，探索木 $G_D(a_t,u_t)$に対して min-max 探索法あるいはその高速計算版であるα β 探索法を適用して得られた leaf 局面での静的局面評価値を指し手 a_tの評価とするのが一般的である．これは，(7)の右辺の計算において期待値計算を厳密に行わないで，探索木の最善応手手順(principal variation)の leaf 局面 $u^*_D(a_t,u_t)$(principal leaf)の静的局面評価値 $E_a^s\left(u_D^*(a_t,u_t);\omega\right)$で代表するという一種の近似計算に相当する．すなわち，(7)の右辺で $P(u^*_D(a_t,u_t)|a_t,u_t)=1$ とし，他の遷移確率 $P(u|a_t,u_t)$は 0 と置いたことに相当する．

また，ヒューリスティクスを用いた探索木の枝刈りも，(7)の右辺の探索過程において途中で leaf 局面への遷移確率をゼロとおくことに相当する．例えば，激指チームの“実現確率”（＝“親の実現確率”×“指し手の遷移確率”）による枝刈り [1]も同様な操作と考えられる．

さらに，近年囲碁などのゲーム探索において盛んに利用されているモンテカルロ探索 [14]は，局面評価のために多数回のプレイアウトを行う．これは，(7)の右辺の期待値操作を，あるシミュレーション方策（例，ランダム方策）により生成した leaf 局面の評価値の単純平均操作で置き換えた近似計算と見なすことができる．

本論文で指し手の評価として(7)のような期待値を提案した理由は次の２つである．まず，上で述べたように従来の様々な指し手探索法を導くことが可能で理論的な見通しが良いことである．次に，min-max 戦略のように最善応手手順や principal leaf だけを利用すると，読みの深さや評価関数の精度に限界がある場合には，指し手評価に大きなリスクが伴う可能性があると考えられるからである．つまり，うまい手順が見つかってそれに飛び付いてしまうというリスクを避けて，それ以外の変化手順をも十分考慮して指し手の評価を行う方が，読みの深さと評価値の限界に起因する探索の揺らぎに対して頑健な評価法を与えてくれるのではないかと考えたからである．

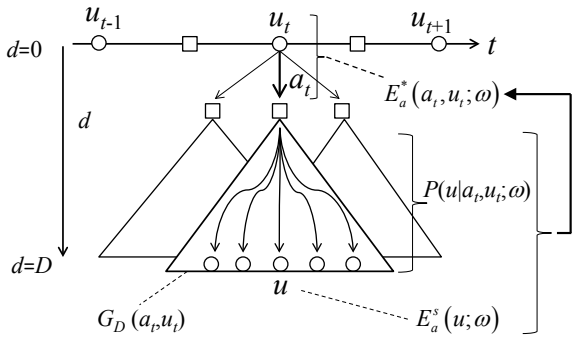

図 1　指し手評価の期待値 $E_a^*(a_t,u_t;\omega)$ と leaf 局面での静的局面評価値 $E_a^s(u;\omega)$, 遷移確率 $P(u|a_t, u_t;\omega)$の関係を表す．t は時間順序，d は読みの深さ，$G_D(a_t,u_t)$は深さ D の部分探索木を表している．また，○印のノードは学習エージェント A の手番局面, □印のノードは対戦相手 B の手番局面を表している．

Figure 1 Expected value of move a_t, $E_a^*(a_t,u_t;\omega)$, static evaluation function of state u, $E_a^s(u;\omega)$, and transition probability from u_t to u, $P(u|a_t,u_t;\omega)$.

4. 探索と方策勾配法による評価関数の学習

4.1 学習則

3.では(1)の目的関数として出現局面 u_tにおける指し手 a_tの評価値 $E_a(a_t,u_t;\omega)$ ではなく，(7)に示したように，a_t 以下の全 leaf 局面の静的評価値 $E_a^s(u;\omega)$ [$u\in U_D(a_t,u_t)$]と leaf 局面への遷移確率 $P(u|a_t,u_t;\omega)$ とを用いて計算することを提案した．よって，学習エージェント A の方策(1),(2)は，

$$\pi_a(a_t|u_t;\omega) = \exp\left(-E_a^*(a_t,u_t;\omega)/T_a\right)/Z_a \tag{8}$$

$$Z_a \equiv \sum_{a'} \exp\left(-E_a^*(a', u_t; \omega)/T_a\right) \quad (9)$$

と表される．このときの学習則は，(5),(6)より，

$$\Delta\omega = \varepsilon \cdot r \sum_{t=1}^{L_a} e_\omega(t) \quad (10)$$

$$e_\omega(t) = -(1/T_a)\Big[\partial E_a^*(a_t, u_t; \omega)/\partial\omega - \sum_{a'} \pi_a(a'|u_t; \omega)\partial E_a^*(a', u_t; \omega)/\partial\omega\Big] \quad (11)$$

となる．

(8)～(11)は，$E_a^*(a_t, u_t; \omega)$ と $\partial E_a^*(a_t, u_t; \omega)/\partial\omega$ の値が局面 u_t における合法な指し手 a についてすべて分かれば計算できる．ただし，これらの値は局面 u_t において指し手 a を指した局面以下の部分木 $G_D(a,u_t)$の全 leaf 局面 $u \in U_D(a,u_t)$ に依存する．したがって，2.2 で述べた通常の方策勾配法の適用では，出現局面 u_t 以下の深さ 1 の局面に含まれる特徴量のパラメータのみが更新対象となるが，本方式では全 leaf 局面に含まれる特徴量のパラメータすべてが更新対象となり，対局あたりの学習の効率化が期待できる．

4.2 指し手評価の期待値とその勾配の再帰計算

(8)～(11)の $E_a^*(a_t, u_t; \omega)$ と $\partial E_a^*(a_t, u_t; \omega)/\partial\omega$ は再帰的に計算できることを示す．まず，深さ $d(0 \leqq d \leqq D\text{-}1)$における学習エージェント A の手番局面 $u^d{}_t$において，指し手 $a^d{}_t$ により生成される相手の手番局面を $v^d{}_t = v(a^d{}_t, u^d{}_t)$, その局面から相手が指し手 $b^d{}_t$ を指して得られた学習エージェントの手番局面を $u^{d+1}{}_t = u(b^d{}_t, v^d{}_t)$とする（図 2）．ただし，対戦相手のエージェント B の方策 π_b は既知とする．

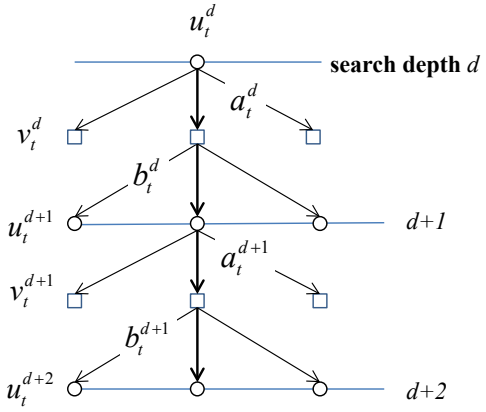

図 2　探索の深さ，手番局面，指し手の関係．

Figure 2　Search depth *d*, states *v* and *u*, and moves *a* and *b*.

この時，探索の深さ d における $E_a^*(a_t^d, u_t^d; \omega)$ と $\partial E_a^*(a_t^d, u_t^d; \omega)/\partial\omega$ は次のように再帰的に書ける．

$$E_a^*(a_t^d, u_t^d; \omega) = \sum_{b_t^d} \pi_b(b_t^d|v_t^d) \cdot \sum_{a_t^{d+1}} \pi_a(a_t^{d+1}|u_t^{d+1}; \omega) \cdot E_a^*(a_t^{d+1}, u_t^{d+1}; \omega) \quad (12)$$

$$\partial E_a^*(a_t^d, u_t^d; \omega)/\partial\omega = (\partial/\partial\omega)\sum_{b_t^d} \pi_b(b_t^d|v_t^d) \cdot \sum_{a_t^{d+1}} \pi_a(a_t^{d+1}|u_t^{d+1}; \omega) \cdot E_a^*(a_t^{d+1}, u_t^{d+1}; \omega) \quad (13)$$

$$= \sum_{b_t^d} \pi_b(b_t^d|v_t^d) \sum_{a_t^{d+1}} \partial\pi_a(a_t^{d+1}|u_t^{d+1}; \omega)/\partial\omega \cdot E_a^*(a_t^{d+1}, u_t^{d+1}; \omega) + \sum_{b_t^d} \pi_b(b_t^d|v_t^d) \sum_{a_t^{d+1}} \pi_a(a_t^{d+1}|u_t^{d+1}; \omega) \cdot \partial E_a^*(a_t^{d+1}, u_t^{d+1}; \omega)/\partial\omega \quad (14)$$

$$= \sum_{b_t^d} \pi_b(b_t^d|v_t^d) \sum_{a_t^{d+1}} \pi_a(a_t^{d+1}|u_t^{d+1}; \omega) \cdot \Big[e_\omega(t, d+1) E_a^*(a_t^{d+1}, u_t^{d+1}; \omega) + \partial E_a^*(a_t^{d+1}, u_t^{d+1}; \omega)/\partial\omega\Big] \quad (15)$$

ただし，

$$e_\omega(t, d+1) \equiv \partial \ln \pi_a(a_t^{d+1}|u_t^{d+1}; \omega)/\partial\omega \quad (16)$$

$$= -(1/T_a)\Big[\partial E_a^*(a_t^{d+1}, u_t^{d+1}; \omega)/\partial\omega - \sum_{a'} \pi_a(a'|u_t^{d+1}; \omega)\partial E_a^*(a', u_t^{d+1}; \omega)/\partial\omega\Big]. \quad (17)$$

また，(12),(13)における再帰の終端は，もし，$u^{d+1}{}_t$ が leaf 局面，すなわち，d=D-1 ならば，

$$E_a^*(a_t^d, u_t^d; \omega) = \sum_{b_t^{D-1}} \pi_b(b_t^{D-1}|v_t^{D-1}) E_a^s(u_t^D; \omega) \quad (18)$$

$$\partial E_a^*(a_t^d, u_t^d; \omega)/\partial\omega = \sum_{b_t^{D-1}} \pi_b(b_t^{D-1}|v_t^{D-1}) \cdot \partial E_a^s(u_t^D; \omega)/\partial\omega \quad (19)$$

と書ける．図 3 に上記の依存関係を表した模式図を示す．

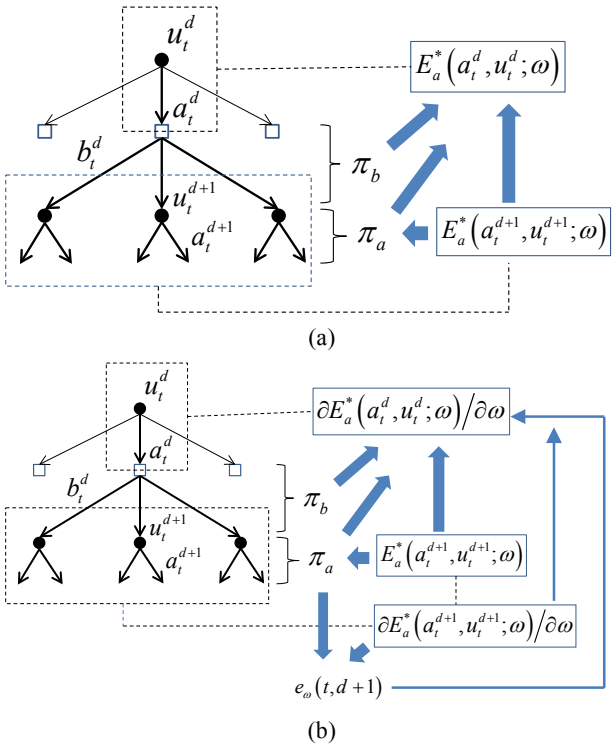

図 3　PG 行動期待値法の再帰計算における依存関係：(a) 指し手評価の期待値 $E_a^*(a_t^d, u_t^d; \omega)$，(b) 1 階微係数の値 $\partial E_a^*(a_t^d, u_t^d; \omega)/\partial\omega$.

Figure 3　Recursive relations in “PG expectation method” for (a)the expectation values of moves, and (b) the first derivatives.

なお，本論文では 2.2 で述べた出現局面における指し手評価値を用いた方策勾配法を“PG 法”または単に方策勾配法，4.1 と 4.2 で提案した全 leaf 局面に基づく指し手評価

の期待値を用いた方策勾配法を“PG 行動期待値法”(Policy gradient expectation method)と呼んで区別することにする．

5. 学習に対する近似手法のアイデア

5.1 学習則の計算量

本論文で提案している 3.の指し手評価には(12)の再帰を用いる．この計算には，ある深さ D における全 leaf 局面の静的局面評価値を知る必要がある．さらに，4.で提案した PG 行動期待値法による学習では，その全 leaf 局面での勾配値も計算する必要がある．したがって，探索時の深さ D が大きくなるにつれて指し手決定と学習にかかる計算時間は膨大なものとなることが容易に予想される．そこで，学習時の計算量を削減するための近似手法に関するアイデアを本章では述べる．

5.2 min-max 探索またはαβ探索の適用：PGLeaf 法

学習エージェント A の方策として max 探索（(8)における $T_a \to 0$ に対応）を行う．すなわち，min-max 探索，あるいは$\alpha\beta$探索を行い，最善応手手順だけを考える．これは(7)の遷移確率において，

$$P\left(u \middle| a_t,u_t;\omega\right)=\begin{cases}1 & \text{if} \quad u=u_D^*\left(a_t,u_t;\omega\right)\\ 0 & \text{otherwise}\end{cases} \qquad (20)$$

と置いたと解釈できる．このように指し手評価の期待値 $E^*_a(a_t,u_t;w)$を principal leaf $u^*_D(a_t,u_t;\omega)$の静的局面評価値 $E^s_a(u^*_D(a_t,u_t;w);w)$で置き換えた指し手決定法と学習法を“PGLeaf 法”と呼ぶことにする．PGLeaf 法では学習時に(12)~(17)のような再帰計算は必要でなく，ゲームプログラミングでよく用いられている$\alpha\beta$探索アルゴリズムとそのプログラムをそのまま利用することができる．

5.3 反復深化法の適用

探索時に探索の深さDを段階的に増やす反復深化法を適用する方法が考えられる．ある深さDを設定し，leaf局面u^l_Dの集合とそれらの静的局面評価値$E^s_a(u^l_D;\omega)$を用いてleaf局面までの遷移確率$P(u^l_D|\, a_t,u_t;\omega)$と指し手評価の期待値$E^*_a(a_t,u_t;\omega)$を計算する．ただし，遷移確率$P(u^l_D|\, a_t,u_t;\omega)$が閾値以下であればそれ以下の部分木はカットする．次にDを1だけ増やしてこの操作を繰り返す．

指し手評価の期待値の計算やパラメータの学習時には，カットされないで残った枝のleaf局面だけを用いる．この場合，残った枝のleaf局面に含まれる特徴量に関するパラメータすべてが更新される．

5.4 異なる評価関数による期待値操作の適用

異なる評価関数を持った探索アルゴリズムk (k=1,2,…N)によりmin-max探索を行い，principal leaf $u^*_{D,k}$を求める．次に，それぞれのprincipal leafにおける静的局面評価値$E^s_a(u^*_{D,k})$を計算し，信頼度α_k ($\Sigma_k \alpha_k$=1, 0≦α_k≦1)を用いて，指し手評価の期待値を

$$E_a^*\left(a_t,u_t;\omega\right)\approx\sum_{k=1}^{N}\alpha_k E_a^s\left(u_{D,k}^*\left(a_t,u_t;\omega_k\right);\omega_k\right) \qquad (21)$$

と近似する．学習時には(21)を(11)へ代入して得られる特徴的適正度を用いる．探索アルゴリズムkは自分が探索したprincipal leaf $u^*_{D,k}(a_t,u_t;\omega_k)$に含まれている特徴量のパラメータを更新する．これは，複数の探索アルゴリズム（知識源の異なるエージェント）による一種の“合議”[15]による指し手決定と，探索アルゴリズムごとの評価関数の学習方法を与えており，並列処理向きの探索／学習アルゴリズムと言える．この際，異なる探索アルゴリズムの生成法として，評価関数にランダムノイズを付加する方法も考えられる．

5.5 その他の工夫

その他の計算量の削減方法として，対戦相手 B の方策π_bとして min 探索を用いることや，(15)の$\partial E^*_a\left(a,u_t;\omega\right)/\partial\omega$を再帰的に計算する際に，(16)の特徴的適正度 $e_\omega(t,d+1)$の項をすべて省略してしまうなどの近似方法も考えられる．

6. まとめ

本論文では強化学習の一手法である方策勾配法をコンピュータ将棋に適用する際に，全 leaf 局面の静的局面評価値をその局面への遷移確率値で重み付けた期待値を用いた指し手評価方式を提案し，評価関数の学習則を導出した．

参考文献

1) 松原仁 編著：コンピュータ将棋の進歩⑥プロ棋士に並ぶ，共立出版(2012).

2) 保木邦仁：局面評価の学習を目指した探索結果の最適制御，第 11 回ゲームプログラミングワークショップ，pp.78-83(2006).

3) Sutton, R. S. and Barto A. G. : Reinforcement Learning, The MIT Press, Massachusetts (1998).

4) Williams, R. J. : Simple Statistical Gradient- Following Algorithms for Connectionist Reinforcement Learning, Machine Learning, Vol.8, pp.229-256 (1992).

5) 木村元，山村雅幸，小林重信：部分観測マルコフ決定過程下での強化学習-確率的傾斜法による接近，人工知能学会誌，Vol.11, No.5，pp761-768 (1996).

6) Sutton, R.S., McAllester, D., Singh, S. and Mansour, Y. : Policy Gradient Methods for Reinforcement Learning with Function Approximation, NIPS’99, pp.1057- 1063 (2000).

7) Konda, V. R. and Tsitsiklis, J. N.: Actor-Critic Algorithms, NIPS’99, pp. 1008-1014 (2000).

8) 阿部健一：強化学習―価値関数推定と政策探索”，計測と制御，第 41 巻，第 9 号，pp.680-685 (2002).

9) Kakade, S.: A natural policy gradient, NIPS’01, pp.1531- 1538 (2002).

10) Peters, J., and Schaal, S.: Policy Gradient Methods for Robotics, IROS 2006, pp.2219-2225(2006).

11) 五十嵐治一，石原聖司，木村昌臣：非マルコフ決定過程における強化学習―特徴的適正度の統計的性質―，電子情報通信学会論文誌 D, Vol.J90-D, No.9, pp.2271-2280 (2007).

12) 石原聖司，五十嵐治一：マルチエージェント系における行動学習への方策こう配法の適用-追跡問題-，電子情報通信学会論文誌 D-I, Vol.J87-D1, No.3, pp.390-397 (2004).

13) 五十嵐 治一，半田 雅人，石原 聖司，篠埜 功：マルチエージェントシステムにおける行動制御―PSO における重み係数の強化学習―，電子情報通信学会論文誌 D, Vol. J94-D, No. 10, pp. 1612-1621 (2011).

14) 美添一樹：モンテカルロ木探索-コンピュータ囲碁に革命を起こした新手法-，情報処理，Vol.49, No.6, pp.686-693 (2008).

15) 伊藤毅志：コンピュータ将棋における合議アルゴリズム，人工知能学会誌，Vol.26，No.5，pp.525-539 (2011).

16) Baxter, J., Tridgell, A., and Weaver, L., : KnightCap: A chess program that learns by combining TD(λ) with game-tree search, ICML '98, pp.28-36 (1998).