大規模な対局に基づいた教師データの重要度の学習

佐藤 佳州[1,2]　高橋 大介[3]

概要：近年，ゲームプログラミングの分野では機械学習が大きな注目を集めており，評価関数の学習，探索深さの制御など，多くの場面で成功を収めている．現在のゲームプログラミングにおける機械学習では，人間のエキスパートの棋譜を教師として，その指し手に近づけるようにパラメータの調整を行っている．しかし，将棋などのゲームでは，コンピュータは既に人間のトッププレイヤに迫る強さとなっており，単純に人間の指し手を再現することが必ずしも「強い」プレイヤの生成に結びつくとは限らない．本論文では，このような課題を解決するため，教師データに重要度（各教師データがどの程度「強い」パラメータの生成に寄与するか）を導入した学習手法を提案する．提案手法では，勝率を適応度とした進化的計算による重要度の学習と，重要度に従ったパラメータ学習を組み合わせた学習を行う．提案手法を将棋の評価関数，実現確率の学習へ適用した結果，従来手法との対局実験において有意に勝ち越すことに成功し，その有効性を示した．また，実験結果から局面の進行度や戦術等によって教師データの重要度に違いが生じることがわかり，教師データの効果的な利用により，より高度な知識の獲得が可能となることを示した．

Learning Weights of Training Data by Large Game Results

Yoshikuni Sato[1,2]　Daisuke Takahashi[3]

Abstract: Recently, machine learning is attracting much attention in the field of game programming, and it has succeeded in making evaluation functions, adjusting of search depth, etc. Existing machine learning methods in game programming learn parameters from records of human expert players. However, computer programs have almost the same strength as human professional players in some games such as shogi. Thus, learning by simply using human records is not necessarily good for generating strong computer players. In this paper, we propose a new machine learning method that estimates the importance of each training record by playing many games and learns parameters according to the importance. The experimental results show the effectiveness of our method for learning evaluation functions and realization probability search. Moreover, the results show that feature of training data such as progress of games or tactics affects its importance.

1. はじめに

近年，ゲームプログラミングの分野では，機械学習が大きな注目を集めている．現在のゲームプログラミングで用いられている学習手法の多くは，人間のエキスパートの指し手を教師とし，その指し手に近づけるように各特徴のパラメータを調整している．この方法は，評価関数の学習，探索深さの制御など，多くの場面で成功し，プログラムの性能向上に大きく貢献してきた．

一方で，このような人間の棋譜を教師とした学習によって獲得されるパラメータは，教師データの性質に大きく依存する．人間の棋譜は教師として完全に理想的なものとはいえず，悪手が含まれる場合が存在する，人間の手を真似ることが必ずしもコンピュータにとって最善とは限らない，といった問題点が存在する．従来は，将棋などの複雑なゲームでは，コンピュータの強さは人間のプレイヤに大きく劣っていたため，このような問題を含む棋譜も理想的な教師データとして近似することができた．しかし，現在

[1] 筑波大学大学院システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba
[2] パナソニック株式会社先端技術研究所
Advanced Technology Research Laboratories, Panasonic Corporation
[3] 筑波大学システム情報系
Faculty of Engineering, Information and Systems, University of Tsukuba

では，将棋などの複雑なゲームにおいても，コンピュータが人間のトッププレイヤに迫る強さとなっており，単純に人間の指し手を教師とする手法では，性能に限界が生じると考えられる．

このような問題は本来「強い」パラメータを学習したいのに対して，それを人間の指し手との一致率で近似しているために生じているといえる．人間の棋譜を教師としない学習も研究されているものの，現在までのところ将棋などの複雑なゲームでは人間の棋譜を教師とした学習を明らかに上回る手法は存在していない．

本論文ではこのような現在の機械学習の問題点を解決するため，教師データに重要度（その学習局面が「強い」プレイヤの生成にどの程度寄与するか）の概念を導入し，対局による重要度の学習と重要度に従った重み付き学習を組み合わせた学習手法を提案する．また，コンピュータ将棋を題材に評価関数の学習，実現確率の学習に提案手法を適用し，その有効性を検証する．

2. 関連研究

現在のゲームプログラミング，特に将棋を始めとする複雑なゲームでは，人間の棋譜を教師とした教師あり学習が成功を収めている．本章では，従来手法として，人間の棋譜を教師とした評価関数の学習，探索深さの制御（実現確率探索），および，ゲームプログラミングにおける提案手法と関連するその他の学習手法について述べる．

2.1 評価関数の学習

ゲームプログラミングにおいて，評価関数の機械学習は古くから研究されていた課題である．将棋において，初めて機械学習によってトップレベルの強さを獲得することに成功したのは，保木である [1]．文献 [1] の学習手法は Bonanza メソッドとも呼ばれ，PV（Principal Variation）の生成と評価関数のパラメータの学習を繰り返し行うことを特徴としている．文献 [1] の学習手法では，具体的には式 (1) の目的関数の最適化を行っている．

$$J(\boldsymbol{P},\boldsymbol{v}) = \sum_{p\in\boldsymbol{P}}\sum_{m=2}^{M_p} T_p[\xi(p_m,\boldsymbol{v})-\xi(p_1,\boldsymbol{v})] + \lambda g(\boldsymbol{v}) + L(\boldsymbol{v}) \quad (1)$$

$\boldsymbol{P}$ は学習対象の局面集合，M_p は局面 p における合法手数，p_1 は記譜中で実際に指された手，p_m はそれ以外の手，$\xi(p_m,\boldsymbol{v})$ は p_m を選択した際の探索結果の評価値を表す．また，T_p は損失関数，$\lambda g(\boldsymbol{v})$ は拘束条件，$L(\boldsymbol{v})$ は正則化項を表している．

評価関数の学習は，コンピュータ将棋の強さを大きく向上させることに成功し，現在ではトップレベルのプログラムの多くが，Bonanza の手法をベースとした学習を用いている [2], [3].

2.2 実現確率探索

実現確率探索（実現確率による探索打ち切りアルゴリズム）[4] は鶴岡によって提案された手法である．実現確率探索は，プロの棋譜を基にした学習により，その手がどの程度の確率で指されそうか（遷移確率）を求め，その遷移確率によって探索の深さを制御するというものである．

実現確率の学習は，従来は単純に実際の棋譜における選択確率を用いていたが，近年はロジスティック回帰による予測が良い結果を得ており，主流となっている [5].

この手法では，n 個の特徴が存在し，i $(1 \le i \le n)$ 番目の特徴の値を x_i とすると，特徴の値 $(x_1, x_2, ..., x_n)$ を持つ指し手の遷移確率 p は式 (2) で表される．

$$p(x_1, x_2, ..., x_n) = \frac{1}{1+\exp(-\sum_{i=1}^{n} w_i x_i)} \quad (2)$$

ロジスティック回帰による実現確率探索では，プロの棋譜から各特徴が選択される割合を求める代わりに，式 (2) における各特徴の重み w_i を学習により求める．学習の際には，プロの棋譜において実際に指された手を正例，指されなかった手を負例とすることで，各特徴の重み w_i を推定する．本論文中では，この重み w_i の算出には，LIBLINEAR[6] を用いる．具体的には式 (3) の最適化を行うことによって w_i を算出している[*1].

$$\min_{w} \quad \|\boldsymbol{w}\|_1 + C\sum \log(1+e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}) \quad (3)$$

実現確率探索も将棋において成功を収めている探索手法であり，評価関数の学習と同様，多くのプログラムが採用している手法となっている．

2.3 その他の学習手法

ゲームプログラミングの分野では，前述の手法に限らず，従来から様々な学習手法が提案されている．例えば，文献 [7] では，強化学習の一種である TD 法を用いた駒の価値の学習が行われており，文献 [8] では進化的計算を用いた評価関数の学習が行われている．

また，複数の学習手法を組み合わせた手法も提案されている．文献 [9] では，局所探索を得意とする TD 学習と大域的探索を得意とする GA を組み合わせたハイブリッド GA により評価関数のパラメータ学習を行っている．この方法により，オセロの実験では従来手法（単一の学習手法を用いた場合）を上回る結果を得ている．

しかし，これらの強化学習や進化的計算によるパラメータ学習は，現在までのところ将棋のような複雑なゲームでは，人間の棋譜を教師とした学習を明確に上回る成果は得られていない．

[*1] LIBLINEAR のオプションは「-s 6」（L1 正則化ロジスティック回帰）

3. 提案手法

3.1 人間の棋譜を教師とした学習の問題点

本節では，現在のゲームプログラミングにおいて主流となっている人間の棋譜を教師とした学習の問題点について述べる．人間の棋譜を教師とした学習は，現在のところ多くの課題に対して良い結果を得ているものの，以下に示すような問題点が存在する．

第一に，人間の棋譜を絶対的に信頼しているため，教師とするのにふさわしくない局面を学習対象とすることがある．これまで，囲碁や将棋といった高度に複雑なゲームでは，コンピュータは人間にはるかに及ばなかったためこの点は問題とならなかった．しかし，現在では，コンピュータは人間のトッププレイヤに迫る強さとなっており単純に人間の強いプレイヤの棋譜を信頼した場合，性能に悪影響を及ぼす可能性があると考えられる．

第二に，従来手法では，すべての棋譜，局面を均等に扱っており，全体としてプロの指し手との一致率が上がるようにパラメータの調整を行っている．しかし，これは必ずしも強いプレイヤの実現に結びつくとは限らない．このような手法では，序盤の駒の位置関係が重視されがちであるが，中盤や終盤の，より戦術的な，勝敗に直結する局面において良い手が指せるかを重視したほうが強いプレイヤとなる可能性があると考えられる．

第三に，本来は学習手法によって有効な教師データには違いがあると考えられる．例えば複雑な局面における好手，妙手といった教師例は，駒の価値などの単純な特徴しか用いていない学習ではその意味を表現する事は難しく，意味のない教師データとなる可能性が高い．一方で，十分に表現力のある特徴を用いた場合には，そのような戦術的な局面は非常に有用な学習対象となると考えられる．

これらの問題は本質的には「強い」パラメータを得ることが目的であるのに対して，それを「人間の指し手との一致率」といった基準で近似していることに起因していると言える．強化学習など人間の棋譜によらない手法も提案されているが，現在までのところ，人間の棋譜を教師とした学習を明らかに上回る成果は得られていない．

3.2 重要度を導入した学習

前節で述べた課題の通り，教師あり学習によって生成されるパラメータは教師データの性質に大きく依存するが，従来手法では教師データとの一致率を向上させることのみを目的としており，どのような教師データが「強い」パラメータを生成するかといった点について考慮されていない．本提案手法では「強い」パラメータを得るため，教師データの各局面に重要度を導入し，強さに寄与する局面を重視した学習を行う．ここでの重要度は，ある教師局面が

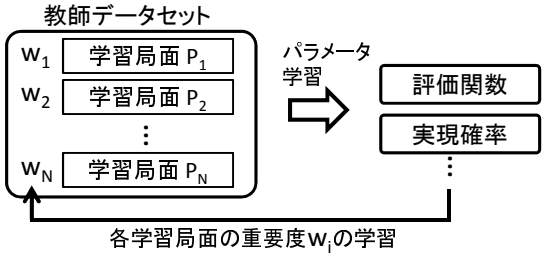

図 1 提案手法の概要

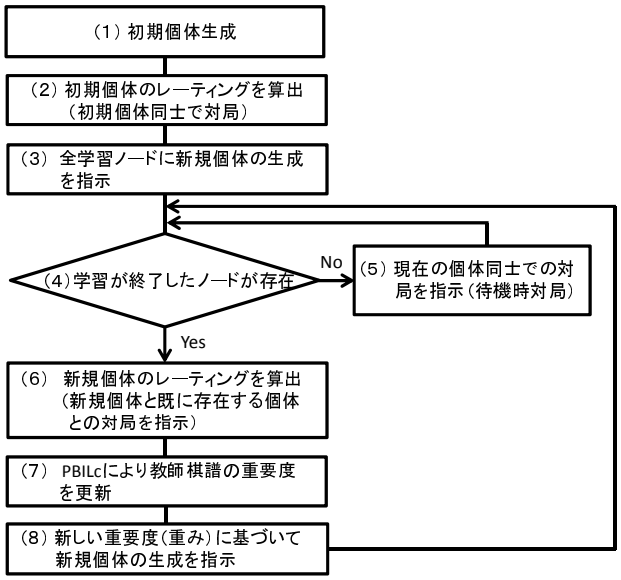

図 2 提案手法のメイン処理フロー

「強い」パラメータの生成にどの程度寄与するかを意味する．提案手法では，この重要度を対局の勝率を適応度とした進化的計算によって学習し，重要度の学習と，重要度に基づいたパラメータ学習を組み合わせることにより，「強い」パラメータを獲得する．

提案手法の概要を**図 1** に示す．提案手法では図 1 のように，教師あり学習による個体（評価関数，実現確率）の生成と，生成された個体同士の対局に基づいた進化的計算による教師データの重要度の学習を繰り返し行うことで，「強い」個体を生成する．

重要度は進化的計算の一種である PBILc（Continuous Population Based Incremental Learning）[10] を用いて学習する．学習の際には，大規模な計算環境を用いた対局を行い，勝率の高い個体を生成するように重要度を調整する．強化学習等の学習手法と進化的計算を組み合わせた手法は過去にも提案されている [9] が，前述のとおり，将棋等の複雑なゲームでは強化学習や進化的計算により人間の棋譜を教師とした学習を上回ることは現状では困難と考えられる．本手法のポイントは，進化的計算により評価関数や実現確率のパラメータを直接求めるのではなく，教師あり学習において「強い」パラメータを生成するための教師データの重要度を学習する点である．

図 2 に提案手法の具体的な処理フローを示す．ステップ (1) では，重要度を正規乱数 $\mathcal{N}(1.0, \sigma^2)$ に従って生成し，その重要度により教師データの各局面に重み付けしたパラメータ学習を行う．この操作を繰り返し，N 個の初期個体を生成する．本論文中では実験的に，$\sigma^2 = 0.1$，$N = 50$ とした．重要度は局面単位に持たせても良いし，棋譜単位など意味を持ったまとまり毎に持たせても良い．本論文では，進行度に応じた重要度を用いる手法，及び棋譜毎に応じた重要度を用いる手法の 2 種類を実験する．

ステップ (2) では，ステップ (1) で生成された各初期個体同士で対局を行い，その強さをレーティングとして算出する．なお，レーティングの算出には文献 [11] の手法を用いた．

ステップ (3)〜(8) では，対局結果に基づいた重要度の更新と，その重要度に従った新規個体の生成を繰り返す．その際，対局及び重要度の学習と新規個体の生成は，別のマシンを用いて並列に行う．重要度を学習するマシンでは，新規個体が生成されていた場合には，新規個体と既に存在する個体との対局を行うことによって新規個体のレーティングを算出し，その結果に基づいて重要度を更新する．その後，新規個体生成側のマシンに，新たな重要度に従った新規個体の生成を指示する．新規個体生成側のマシンは，受け取った重要度に従ったパラメータ学習を行い，新たな個体を生成する．

重要度の学習では，各個体の正確な強さ（順位）を算出するために，非常に多くの対局を行う必要がある．ステップ (2) では各個体 4,000 局ずつ，ステップ (6) では 2,000 局[*2]の対局を行う．本論文では，クラスタ環境による大規模な計算資源を利用して，これらの対局と新規個体の生成を行う．

本手法では，重要度の更新には，進化的計算の一種である PBILc を用いる．PBILc は進化的計算の中でも，分布推定アルゴリズム（EDA: Estimation of Distribution Algorithm）と呼ばれる手法である．通常の遺伝的アルゴリズム（GA: Genetic Algorithm）が直接個体を進化させるのに対して，EDA では個体の生成確率を進化させる点が特徴である．

PBILc では，次世代の個体を $\mathcal{N}(X_i, \sigma_i^2)$ の正規乱数に従って生成する．X，σ は以下の式により更新する．

$$X_i^{t+1} = (1-\alpha)X_i^t + \alpha(X_i^{best,1} + X_i^{best,2} - X_i^{worst}) \quad (4)$$

$$\sigma_i^{t+1} = (1-\alpha)\sigma_i^t + \alpha\sqrt{\frac{\sum_{j=1}^{K}(X_i^j - \overline{X}_i)^2}{K}} \quad (5)$$

ここで $(t+1)$ 世代目の X^{t+1} は，前世代の最良個体 $X_{best,1}^t$ と 2 番目に良い個体 $X_{best,2}^t$，及び最も悪い個体 X_{worst}^t から求められる．また，σ の値は，上位 K 個の最良個体の標準偏差を基に更新する．このような式を用いることで，よさそうなパタメータの値の付近かつ不確定な（よさそうな個体群のなかでばらつきが大きい）部分を重点的に探索することができる手法となっている．本論文中では，PBILc のパラメータは $K = 10$，$\alpha = 0.01$ とした．

[*2] ステップ (6) では，対局数が 1,000 局以上で，レーティング算出対象の新規個体の順位が最下位の場合，その時点で対局を打ち切っている．

4. 実験

提案手法により，評価関数及び実現確率の学習を行った．評価関数のパラメータ学習は文献 [1] の手法を用い，学習時の探索は静止探索のみとした．実現確率のパラメータ学習には，LIBLINEAR[6] を各学習局面に重み付けして学習できるよう改変したものを用いた．重要度の学習，及び対局実験における探索ノード数は 1 手 10 万ノードとした．学習では，以下に示す将棋における一般的な特徴を用いた．

(1) 駒割り
(2) 自玉，敵玉との位置関係
(3) 利きが関連する駒同士の位置関係 [12]
(4) 王手
(5) 駒を取る手，リキャプチャ
(6) 成る手
(7) あたりをかける手
(8) 逃げる手
(9) 持ち駒を打つ手
(10) 玉の移動

このうち，(4)〜(10) は指し手の特徴となるため，実現確率の学習のみで用いている．本実験における特徴の総数（重みが 0 でないもの）は，約 400 万個となっている．学習にはプロやアマ高段者の棋譜[*3]を用いた．学習棋譜数は，評価関数は 10,000 局，実現確率は 5,000 局とした．

4.1 実験環境

実験環境を**表 1** に示す．重要度を学習するための個体同士の対局は 15 台のクラスタ環境で行い，新規個体の生成には 18 台の学習用マシンを用いた．学習は，提案手法における図 2 のステップ (4)〜(8) の処理を 400 回行った．学習に要した時間は，評価関数の学習では約 12 日，実現確率の学習では約 3 日となっている．

表 1 実験環境

用途	CPU	メモリ	台数
対局（重要度学習）	Core2 Quad Q9650	8GB	15
パラメータ学習	Core i7-3930K	16GB	7
	Core i7-990X	12GB	1
	Xeon E5506×2	24GB	6
	Opteron 6134×2 [*4]	16GB	4

[*3] 内訳はプロ 9,617 局，女流 278 局，アマチュア 83 局，奨励会 22 局．
[*4] 実現確率学習時は対局用として使用．

4.2 対局のマッチメイク方法

本提案手法では，複数の個体同士の対局結果を基に，重要度の学習を行う．なるべく少ない対局数で正確な個体の強さを算出するために，マッチメイクの方法は重要である．本実験では，囲碁や将棋のオンライン対局サーバ [13], [14] でよく用いられている手法を参考に，どのようなマッチメイク方法が適しているかを実験した．

図 3，**図 4** は，強さが既知の 50 プレイヤで繰り返し対局を行った時の，理想的な順位との誤差の平均を示したものである．本実験では，1,000 回のマッチメイクを行う実験を 10 回行い，各プレイヤの理想的な順位との誤差の平均値を求めている．なお，本実験では，1 回のマッチメイクにつき，同一局面から先後を入れ替えた 2 局をセットで行っている．

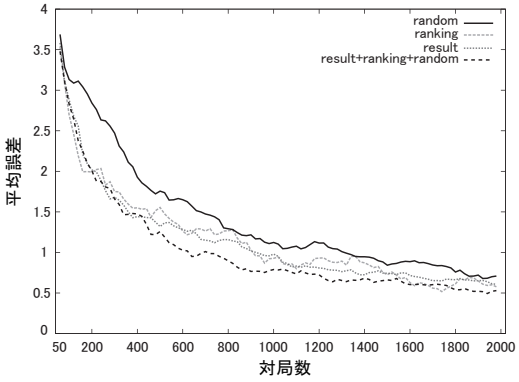

図 3 各マッチメイク手法における理想的な順位との誤差（同一プログラムで探索ノード数を変更したプレイヤ間での対局）

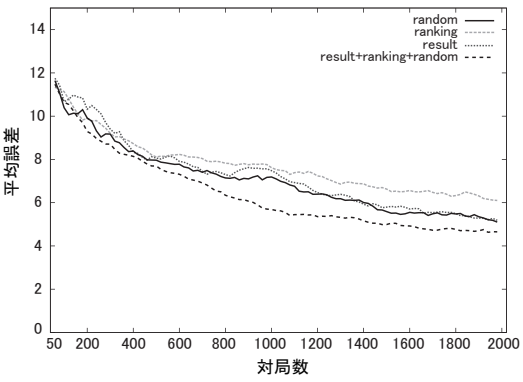

図 4 各マッチメイク手法における理想的な順位との誤差（異なるパラメータの評価関数を持つプレイヤ間での対局）

図 3 は，同一プログラムでノード数を変更[*5]したプレイヤ間での実験，図 4 は，評価関数学習時におけるすべての初期個体同士で対局を行った実験の結果である．図 4 の実験では，すべての個体同士で事前に 1,000 局ずつ総当りの対局を行い，その結果を理想的な順位としている．総当りの対局は非常に時間がかかるため，本実験では探索ノード数は 10,000 とした．つまり，図 3 は相性が存在しない理想的な順位を持つプログラム間で対局を行う環境での実験，図 4 は相性や同程度の強さのプログラムが複数存在する，より実際の状況に近い環境での実験となっている．

実験では，次の 4 通りのマッチメイクの方法を比較した．図 3，図 4 中の random は完全にランダムなマッチメイク，ranking は正規乱数により順位の近いプレイヤ同士を優先的に対局させる方策[*6]，result は勝ったプレイヤ同士，負けたプレイヤ同士を対局させる方策である．result+ranking+random は，勝ったプレイヤ同士，負けたプレイヤ同士で順位の近いものを優先して対局させ，さらに 10 回に 1 回完全にランダムなマッチメイクを混ぜた方策である．この意図は同じプレイヤ同士の対局のループを防ぐためである．

実験の結果，図 3 の理想的なプレイヤ同士の対局では，ranking，result，result+ranking+random のいずれの方策もランダムなマッチメイクと比較して良い結果を得た．一方で，図 4 の実験では，result は random とほぼ同等，ranking は random に劣る結果となった．これは，マッチメイクの工夫がプレイヤ間の相性に影響されやすいことを示していると言える．特に ranking によるマッチメイクでは，順位の近いプレイヤとの対局回数が多くなるため，局所的には正確な順位を求められるが，相性が存在する場合には全体の中での順位に不適当な偏りが生じることが多くなったと考えられる．

result+ranking+random によるマッチメイクは，図 3，図 4 の実験ともに最も良い結果を得た．これは，ranking によって順位の近いプレイヤとの優劣を正確に求めつつ，result, random を組み合わせることによって，局所解に陥らないようにすることができているためであると考えられる．以降の実験では，個体のレーティング算出 (図 2 のステップ (2)，(5)) のマッチメイクには，result+ranking+random を用いた．なお，図 2 のステップ (6) の対局では，対局プレイヤの一方は新規個体で固定のため，ランダムなマッチメイク（random）を用いた．

4.3 対局実験による提案手法の評価

提案手法を将棋の評価関数の学習，及び実現確率の学習に適用した際の有効性を，対局実験により評価した．比較対象のプレイヤ（従来手法）は，教師データに重要度による重み付けをせず学習したものである．評価関数の学習では，予備実験においてそれ以上反復回数を増やしても性能向上が認められなくなるまで十分な学習を行ったものを比較対象とした．対局は定跡で 16 手進めた局面から先後を入れ替えて 1,000 セット，2,000 局を行った．

[*5] プレイヤ $n(0 \leq n \leq 49)$ の探索ノード数を $1,000 + 200 \times n$ とした．

[*6] 正規乱数のパラメータは予備実験により決定．

4.3.1 評価関数の学習における提案手法の評価

表 2，**表 3** に評価関数学習時における従来手法，および最良の初期個体との対局結果を示す（表中の太字の数値は有意水準 5%の二項検定で有意な結果）．

表 2 従来手法に対する勝率（評価関数）

重要度の学習	提案手法	初期個体（best）
進行度単位	**0.561**	**0.523**
棋譜単位	**0.581**	**0.552**

表 3 最良の初期個体に対する勝率（評価関数）

重要度の学習	勝率（提案手法）
進行度単位	**0.544**
棋譜単位	**0.531**

実験結果から，提案手法が従来手法と比較し，有意に勝ち越していることがわかる．最良の初期個体の重要度を用いて学習を行った場合にも従来手法を上回っているが，提案手法で学習された個体は，より高い勝率を得ることができていることがわかる．

4.3.2 実現確率の学習における提案手法の評価

表 4，**表 5** に実現確率学習時における従来手法，および最良の初期個体との対局結果を示す（太字は有意水準 5%の二項検定で有意な結果）．

表 4 従来手法に対する勝率（実現確率）

重要度の学習	提案手法	初期個体（best）
進行度単位	**0.527**	0.506
棋譜単位	**0.536**	0.486

表 5 最良の初期個体に対する勝率（実現確率）

重要度の学習	勝率（提案手法）
進行度単位	0.514
棋譜単位	**0.542**

実験結果から，実現確率探索においても提案手法が従来手法を有意に上回る結果を得た．ただし，評価関数の学習と比較すると，その効果はやや低い結果となった．これは，評価関数の場合には評価値にわずかでも差があれば選択される指し手に直接影響を及ぼすのに対して，実現確率探索の場合には，予測確率を探索深さに変換して利用することになるが，その際に予測確率の差が小さい指し手の間では探索深さとしては差がつきにくいこと等が原因として考えられる．

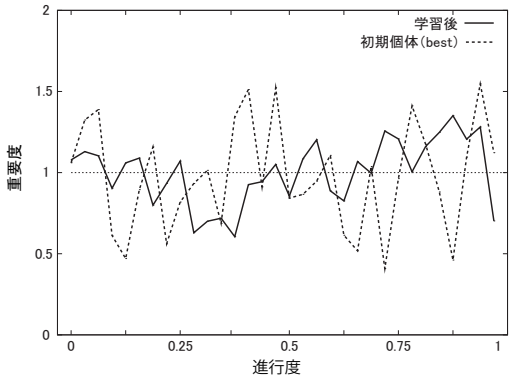

図 5 進行度に応じた教師データの重要度

4.4 進行度単位の重要度の分析

図 5 に提案手法を用いて学習した進行度別の重要度（評価関数学習時）を示す．本論文における進行度は，ある時点の手数を終局までの手数で割った値としている．図 5 では，初期個体のうち最もレーティングの高かった個体，及び提案手法を用いて学習した重要度を示している．なお，重要度の平均は常に 1.0 になるように正規化している．

図 5 から，提案手法を用いて学習した重要度は，序盤から中盤にかけてやや下がり，終盤で最も高くなっていることがわかる．終盤で重要度が最も高くなった理由としては，終盤では駒の位置関係が直接勝敗に影響しやすく，良い局面，悪い局面がはっきりと分かれるためであると考えられる．文献 [3] では，進行度に応じて終盤ほど正解手とその他の合法手の評価値のマージンが大きくなるように調整しており，終盤の重要度が高くなっている点では，今回の実験結果は一致した傾向となっている．

また，今回の実験では，序盤よりも中盤の前半において重要度が低くなる結果となった．これは，序盤は駒組みが明確であり，評価関数による局面評価がしやすいのに対して，中盤は駒組み（囲い）が崩れることも多く，明確な目指すべき形が定義しにくいことが原因と考えられる．序盤，終盤はある程度有力な手が限られているのに対して，中盤は自由度が高く局面の評価が難しいと言える．

さらに，最終盤（終局直前の 5〜10 手程度）では，重要度が急激に低くなっていることがわかる．これは最終盤では既に勝敗が決しており教師として不適切な局面が存在することや，今回の実験で評価関数の学習に用いた探索が静止探索という非常に浅い探索であったことに起因すると考えられる．文献 [1] の手法では，探索結果の末端局面を学習に用いるため，学習時の探索深さと局面を正確に評価するために必要な探索深さが大きく異なる局面は本質的に学習対象として向いていないといえる．最終盤は直接詰みが絡むことが多いため，特に今回用いた浅い探索では適切な探索結果を得ることができず，重要度が低くなったと考えられる．これらの結果から，提案手法では学習の性質に応じた教師データの重要度を算出できていると考えられる．

表 6 戦術による重要度の違い

順位	戦型	戦術		先手側勝率（棋譜数）	重要度の平均
		先手戦術	後手戦術		
1	四間飛車	四間飛車穴熊	銀冠	0.566 (30)	1.121
2	四間飛車	居飛穴模様	藤井システム	0.500 (20)	1.100
3	矢倉	森下システム	△９五歩・８四歩型	0.500 (96)	1.088
4	矢倉	▲４七銀３七桂	金矢倉	0.500 (18)	1.079
5	相掛かり	▲２六飛	△５四歩型	0.542 (24)	1.061
…	…	…	…	…	…
96	三間/四間飛車	居飛車穴熊	本美濃	0.700 (30)	0.951
97	矢倉	引き角	△６二飛戦法	0.474 (19)	0.922
98	三間/四間飛車	左美濃	高美濃	0.611 (18)	0.914
99	中飛車	角交換型	ゴキゲン中飛車	0.519 (27)	0.901
100	角換わり	相腰掛け銀	△６五歩型	0.538 (26)	0.899

4.5 棋譜単位の重要度の分析

表 6 に棋譜単位の重要度を学習した時の，戦術による重要度の違いを示す．戦術の判別には，文献 [15] のソフトを用い，出現数が上位 100 個の戦術について分析を行った．

表 6 から，穴熊や銀冠，矢倉を始めとする，囲いを発展させる戦術の重要度が高くなっていることがわかる．一般的に，穴熊等はコンピュータ将棋でも勝ちやすい戦術と言われており妥当な結果と考えられる．また，今回の実験結果では，戦術で分類した時に先手と後手の勝率が互角に近い棋譜の重要度が高い傾向があった．これは，本実験では棋譜単位で重要度を付与しており，先手，後手の戦術の重要度をそれぞれ表現する事はできないことが原因だと考えられる．棋譜単位の重要度では，一方が戦術的に有利である棋譜に対して，先後両方の戦術の重要度を同等として学習してしまうため，先後で互角の棋譜の価値が高くなったと考えられる．今回は棋譜単位で重要度を割り当てたが，同じ棋譜でも戦術や対局者は先手と後手で異なるため，先後で異なる重要度を用いることも有効と考えられる．

表 7 に棋戦による重要度の違いを示す．棋戦は，棋譜数が 10 局以上あるものを分析対象とした．表 7 から，棋戦単位で分析した場合，アマや女流の棋戦の重要度はやや低くなる傾向があったが，プロ棋士の棋戦の間ではほぼ差は見られなかった．実験結果から，少なくとも今回の実験で用いた評価関数の特徴，および学習方法では，女流やアマ等明らかに実力の差がある棋譜は学習に悪影響をおよぼす可能性があるものの，一定以上の強さが保証されている場合には，その中での棋譜の質の差は重要度に大きな影響を与えてはいないと考えられる．

その他，対局者，対局年，持ち時間，戦型（戦術で分類しない），終局までの手数での分析も行ったが，いずれも明確な傾向は見られなかった．以上の結果から，今回の実験で学習された重要度は，(1) コンピュータが勝ちやすいと言われている戦術が重視されている，(2) 教師としての質（強さ）に大きく差があるものは除外する方向に調整されている，といった傾向が読み取れる結果となった．

表 7 棋戦による重要度の違い

順位	棋戦	棋譜数	重要度の平均
1	名将戦	41	1.049
2	近将カップ	64	1.043
3	天王戦	85	1.025
4	十段戦	118	1.020
5	達人戦	27	1.020
…	…	…	…
31	全日プロ	400	0.989
32	朝日アマ名人戦	20	0.974
33	早指し戦	299	0.972
34	女流名人戦	62	0.946
35	全日アマ名人戦	10	0.917

表 8 実際の対局における戦術選択の違い

戦術	各戦術の選択回数	
	従来手法	提案手法
本美濃	131	77
高美濃	45	54
銀冠	61	175
流れ矢倉	28	15
金矢倉	33	45
居飛車穴熊	210	215
四間飛車穴熊	38	40
三間飛車穴熊	65	39

4.6 実際の対局時における戦術選択の違い

提案手法により学習された重要度が，実際の対局においてどのように反映されているかを検証した．

表 8 に，4.3.1 節の 2,000 局の対局実験において，提案手法，従来手法のプログラムが選択した戦術の違いを示す．表 8 から，今回の実験結果では，実際の対局時に選択された戦術として，美濃囲い（本美濃）と銀冠で提案手法と従来手法に特に大きな差が表れていることがわかる．提案手法では，美濃囲いのまま戦うよりも，高美濃，銀冠といった囲いに発展させている傾向が顕著に表れている．その他にも提案手法では，流れ矢倉よりも，より固さを重視した金矢倉が選択される割合が高くなるといった傾向が表れており，学習時の重要度の違いが，実際の対局時の戦術にも影響を与えていることが確認できた．

なお，穴熊については，実際の対局時には，提案手法で選択した割合が従来手法と比較して必ずしも高くなっているわけではないことがわかる．これは，従来手法においても，穴熊は十分に高い価値を獲得しており，提案手法と差が生じにくかったことなどが理由として考えられる．

5. 今後の課題

本論文における実験では，教師データに重要度を導入し，対局による重要度の学習と重要度に基づく重み付き学習を繰り返すことによって，性能向上が実現できることを示した．本提案手法では，重要度の学習には分布推定アルゴリズムの一種である PBILc を用いた．PBILc は分布推定アルゴリズムの中でも基礎的な手法となっており，その他の学習手法を用いることによって，より精度の高い重要度が獲得できる可能性がある．また，本論文では，実験の都合上，初期個体 50，学習ステップ 400 という条件で学習を行ったが，これらの条件を増やすことによる性能向上も期待できる．その他，今回の実験では，進行度単位，棋譜単位に重要度を割り当てたが，局面単位，戦術単位，先手と後手で異なる重要度を用いるなど，重要度を割り当てる単位にも改善の余地があると考えられる．

また，本論文では評価関数，実現確率の学習において提案手法の有効性を示したが，今後はゲームプログラミングにおけるその他の学習への提案手法の適用も検討したいと考えている．特にモンテカルロ木探索におけるシミュレーション中の指し手の選択は大きな課題の一つである．モンテカルロ木探索中の指し手の選択は現在まで様々な手法が提案されている [11], [16], [17] が，強い方策が必ずしも強いプレイヤに結びつくわけではないという事がわかっており，どのような方策が強いプレイヤを実現するかは明らかになっていない．提案手法は，勝率をベースとして「強い」プレイヤの実現を目指す手法となっているため，このような問題に対して有効となる可能性がある．

6. おわりに

本論文では，教師データに重要度を導入し，対局による重要度の学習と重要度に基づいたパラメータの重み付き学習を組み合わせた学習手法を提案した．重要度は，大規模な計算環境を用いた対局を行い，その勝率を適応度とした進化的計算により学習した．提案手法をコンピュータ将棋における評価関数，及び実現確率の学習に適用した結果，従来手法を有意に上回ることに成功し，その有効性を示した．また，実験により学習された重要度は，終盤の重要度が高くなる，穴熊など一般的にコンピュータが勝ちやすいと言われている戦術の重要度が高い傾向となる，といった結果が得られ，提案手法を用いることによりプログラムの性質にあった教師データの重要度が算出されていることが確認できた．

現在，ゲームプログラミングでは，評価関数の学習，探索深さの制御，モンテカルロ木探索の指し手の選択等，多くの課題において機械学習が取り入れられており，今後も一層重要となると考えられる．一方で，コンピュータの強さは多くのゲームにおいて確実に人間の強さに迫るものとなっており，今後は単純に人間の指し手を教師とした学習から，一歩進んだ学習手法が必要になると考えられる．本論文の提案手法は，教師データの理解，効率的な利用によって，より高度な知識の獲得を可能とするものであり，今後このような学習手法はさらに重要となると考えている．

参考文献

[1] 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).

[2] 金子知適，山口和紀：将棋の棋譜を利用した大規模な評価関数の学習，情報処理学会論文誌，Vol. 51, No. 12, pp. 2141–2148 (2010).

[3] 鶴岡慶雅：「激指」の最近の改良について —コンピュータ将棋と機械学習—，コンピュータ将棋の進歩 6，pp. 71–83 (2012).

[4] Tsuruoka, Y., Yokoyama, D. and Chikayama, T.: Game-Tree Search Algorithm Based On Realization Probability, *ICGA Journal*, Vol. 25, No. 3, pp. 145–152 (2002).

[5] 鶴岡慶雅：最近のコンピュータ将棋の技術背景と激指，情報処理， Vol. 49, No. 8, pp. 982–986 (2008).

[6] Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R. and Lin, C.-J.: LIBLINEAR: A Library for Large Linear Classification, *Journal of Machine Learning Research*, Vol. 9, pp. 1871–1874 (2008).

[7] Beal, D. F. and Smith, M. C.: First Results from Using Temporal Difference Learning in Shogi, *Proceedings of the First International Conference on Computers and Games*, pp. 113–125 (1999).

[8] 鈴木彰，柴原一友，但馬康宏，小谷善行：条件付き確率 PIPE による将棋の評価関数の生成，第 10 回ゲーム・プログラミングワークショップ，pp. 56–62 (2005).

[9] 矢野友貴，柴田剛志，横山大作，田浦健次朗，近山隆：GA と TD(λ) 学習の組み合わせによるゲーム局面評価パラメータの調整，情報処理学会研究報告. GI, [ゲーム情報学]， Vol. 2009, No. 27, pp. 63–70 (2009).

[10] Sebag, M. and Ducoulombier, A.: Extending Population-Based Incremental Learning to Continuous Search Spaces, *Proceedings of the 5th International Conference on Parallel Problem Solving from Nature*, pp. 418–427 (1998).

[11] Coulom, R.: Computing Elo Ratings of Move Patterns in the Game of Go, *ICGA Journal*, Vol. 30, No. 4, pp. 198–208 (2007).

[12] 佐藤佳州，高橋大介：特徴の生成を組み合わせた機械学習，第 16 回ゲーム・プログラミングワークショップ，pp. 135–142 (2011).

[13] http://cgos.boardspace.net/

[14] http://wdoor.c.u-tokyo.ac.jp/shogi/floodgate.html

[15] http://www.geocities.jp/saltedeggplant/

[16] Gelly, S., Wang, Y., Munos, R. and Teytaud, O.: Modification of UCT with Patterns in Monte-Carlo Go, Technical Report 6062, INRIA, France (2006).

[17] Silver, D. and Tesauro, G.: Monte-Carlo simulation balancing, *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 945–952 (2009).