

## 合議アルゴリズムプレイヤーの棋譜を用いた 将棋の評価関数の学習

中澤 隆久<sup>†1</sup>      林 伸也<sup>†1</sup>      鶴岡 慶雅<sup>†2</sup>  
田浦 健次朗<sup>†3</sup>      近山 隆<sup>†2</sup>

現在、コンピュータ将棋の評価関数は、機械学習による調整が主流となっており、通常、教師データにはプロ棋士の棋譜が使用される。学習データを増やす方法として、自己対戦による棋譜を用いる方法が考えられるが、これまでのところ、自己対戦棋譜による実用レベルでの学習の成功例は報告されていない。この原因は、自己対戦による棋譜の質が、プロ棋士の棋譜と比較して劣るからだと考えられる。本研究では、合議アルゴリズムを学習棋譜生成部分に利用して、棋譜の学習価値の向上と学習後の評価関数の改善を目指した。

### Learning shogi evaluation functions from records of players with a voting algorithm

TAKAHISA NAKAZAWA,<sup>†1</sup> SHINYA HAYASHI,<sup>†1</sup>  
YOSHIMASA TSURUOKA,<sup>†2</sup> KENJIRO TAURA<sup>†3</sup>  
and TAKASHI CHIKAYAMA<sup>†2</sup>

Today, evaluation functions in Shogi programs are usually tuned by machine learning with game records of professional Shogi players. One possible way to increase the amount of training data is to incorporate self-play game records. However, no successful attempt of the self-play approach has been reported so far, presumably because the quality of the game records created by self-play is far below than that of the game records of professional players. In this study, we apply a voting algorithm to the record generation process in order to generate game records of higher quality, which could lead to a better evaluation function.

### 1. はじめに

コンピュータゲームプレイヤーは人工知能の一つの分野として研究されており、中でもコンピュータ将棋プレイヤーは、近年、人間のプロ棋士に伍すほどの強さになって来たこともあって、研究が盛んである。

現在、コンピュータ将棋では、棋譜を読み込ませ、評価関数のパラメータの自動調整を行う機械学習が有効とされており、近年のコンピュータ将棋選手権では決勝出場のほとんどのプログラムがこの手法を採用している。これらのプログラムに読み込ませている棋譜は通常プロ棋士同士の対戦の棋譜であるが、プログラムに行かせた自己対戦の棋譜を学習することでも評価関数の改善を見込めることが分かっている<sup>1)</sup>。学習に用いる以上は、自己対戦で作成する棋譜もなるべく学習させる価値がある棋譜となっていることが望ましいが、単に自己対戦を行うだけでは質の悪い棋譜も生成してしまい、結果として評価関数の性能が低下する恐れがあるのが自己対戦の棋譜を学習に用いる際の欠点である。よって、自己対戦の棋譜を学習に用いて評価関数を改善するためには、何らかの調整を加えて、良い学習が出来る様に生成される棋譜の質を上げる必要があると言える。

本研究では近年提案された合議アルゴリズム<sup>3)</sup>をこの棋譜生成に応用することで、自己対戦によって生成された棋譜を学習させた時の評価関数を改善する。合議アルゴリズムでは、複数のプログラムを別々に動かし、それぞれのプログラムが選出した候補手群から、最終的に指す一手を決定するため、通常より多くの資源を必要としてしまうが、有意に棋力が向上することが知られている。自己対戦時に合議アルゴリズムを採用することによって、対戦で生成される棋譜の質、すなわち学習させる価値を上昇させることが可能と考えられる。

### 2. 関連研究

#### 2.1 指し手の比較に基づく学習

現在は主流となっている機械学習を用いた評価関数の学習であるが、将棋プログラム Bonanza<sup>8)</sup>の登場以前はバックギャモン等では成功例があったものの<sup>5)</sup>、将棋は機械学習を用いての学習は難しいとされていた。

<sup>†1</sup> 東京大学工学部電子情報工学科, Engineering Department, The University of Tokyo

<sup>†2</sup> 東京大学大学院工学系研究科, Graduate School of Engineering, The University of Tokyo

<sup>†3</sup> 東京大学大学院情報理工学系研究科, Graduate School of Information Science and Technology, The University of Tokyo

保木は、教師データである棋譜中の差し手と minimax 探索の結果の一致度を測る目的関数を設計し、棋譜との差し手一致の度合いを大きくするような重みベクトルを求めることで、機械学習を用いた評価関数の重みベクトルを自動調整を実現した<sup>2)</sup>。差し手の比較に基づいた学習は現在多くの将棋プログラムが採用しており、本研究で使用する将棋プログラム『激指』<sup>6)</sup>も例外ではない。

## 2.2 合議アルゴリズム

合議アルゴリズム<sup>3)</sup>は、図1のように複数の思考プログラムに処理を行わせ、その結果を集約して差し手を決定するもので、ある種の並列処理手法といえる。

合議アルゴリズムの最たる利点はその実装の手軽さである。ソフトウェアの性能向上には効率の良い並列処理が一つの重要な課題であり、コンピュータ将棋においてもそれは例外ではない。しかし、将棋の思考アルゴリズムの主要部分である - 法に基づいたゲーム木探索の効率的な並列化は容易ではない。その点、複数プログラムを別々に動かして、結果のみを集約して決定すれば良い合議アルゴリズムでは、疎結合並列処理として非常に実装が簡単である。合議アルゴリズムでは、集約された結果から、評価値の比較や多数決など、一定のアルゴリズムを用いて差し手の選択がなされる。複数プログラムの合議では、それぞれのプログラム単独との対戦で有意に勝ち越すことが分かっており、性能の向上が見受けられる。また、複数のプログラムを用意せずとも、単一のプレイヤーに対し乱数を用いてゲーム木の末端評価に振れ幅を持たせることで擬似的に複数のゲームプレイヤーを生成し、合議システムを構築する手法があり、これを乱数合議法と呼ぶ。乱数合議法は一つのプレイヤーがあれば簡単に適用することが可能であり、元のプログラムよりも強くなることわがわが<sup>4)</sup>。

## 2.3 コンピュータ将棋の棋譜を用いた学習

通常、コンピュータ将棋の評価関数の学習にはプロ棋士の棋譜が用いられることは先に触れたが、コンピュータ将棋が作成した棋譜も十分学習する価値のある物が作成出来ることが報告されている<sup>7)</sup>。プロ棋士の棋譜、アマチュアの棋譜、コンピュータの棋譜を用意し、それぞれ学習を行い比較を行った結果、概ね評価関数の改善が確認され、特にプロ棋士の棋譜を学習に用いた物が強く、優位性が確認されると共に、コンピュータ将棋の棋譜が学習にも適用出来ること示されている。

## 3. 提案手法

本研究では将棋プログラムとして『激指』<sup>6)</sup>を用いる。激指も Bonanza 式の、兄弟の差し手の比較に基づく手法での学習が行えるので、棋譜生成における自己対戦部分に合議アル

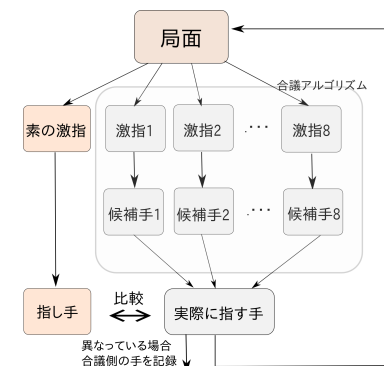


図1 提案手法

ゴリズムを組み込んでいくという手法をとる。合議アルゴリズムを用いた上での自己対戦によって生成した棋譜を激指に学習させることで評価関数を改善した激指を作成し、その性能を評価する。

### 3.1 合議アルゴリズムの組み込み

本研究では将棋プログラム『激指』のみを用いるので、合議には2.2節で述べた激指同士による乱数合議法を採用する。提案手法の概要を図1に示す。末端評価に一定の乱数の振れ幅を持たせた8台分の激指によって合議を行い、乱数を振らない激指と異なる差し手を記録しておく。この時の乱数は正規乱数を採用し、標準偏差を調整する。候補手の選択の仕方としては、単純に最も多くの激指が選択した手を採用する「多数決合議」を用いる。尚、複数の手が同数選択された場合は挙げられた同数の手の中で返された時の評価値の最も高い手を採用することとした。

### 3.2 合議激指の調整

乱数合議法に用いる乱数の標準偏差は重要なパラメータである。合議には大量の資源を消費する以上、最高の成果を出す調整が求められる所である。合議がなるべく力を発揮出来るような最適な値を探すにあたり、「盤面の進行度に応じて、乱数合議法に用いる乱数の標準偏差を変化させる」という手法を提案する。

将棋は進行度に応じて序盤・中盤・終盤に分類することが出来る。それぞれの場面において、優先されるべき事柄は異なるため盤面の進行具合を判断し、その情報から戦略を変えていくことの重要性はコンピュータ将棋でも提唱されていた<sup>9)</sup>。

ゲームが進行するにつれて、形勢は動き易くなる。即ち候補手間の評価値の差も大きくなると考えるのが自然である。

乱数合議法の末端評価に与える乱数の標準偏差の設定は絶対値である。序盤では最も適したと言える乱数幅が終盤でも最適だとは言い難い。そこで、盤面の進行度から振れ幅を変化させていくという手法が有効となると言える。

### 3.3 学習の取捨

『激指』の自己対戦により生成された棋譜を用いて学習することになるが、この際

- ・単純に生成した棋譜をすべて学習する
- ・手番ごとに合議が採択した手と非合議に採択させた手とに違いがあるかを保存しておく、両者が異なっていた局面とその指し手のみに絞って学習を行う

という2種類の手法が考えられる。前者と比較したとき後者の方が学習に利用可能な局面は少なくなるが、利点としては、既に評価関数が正しく評価出来ているような局面は省くことで、合議アルゴリズムが導出した手を特に学習出来る、といった事が挙げられる。本研究では両者についてそれぞれ学習を行って評価した。

## 4. 評価

### 4.1 乱数合議における適切な振れ幅の選定

提案手法を取り入れた激指の作成のために、乱数合議法を採用した自己対戦で学習用の棋譜を生成する必要がある。3.1節で述べたように乱数合議法を用いる場合、適切な振れ幅にする必要がある。そこで、最適な振れ幅の調査とその調整を行った。

乱数の標準偏差を一定間隔で変更しながら、『激指』の片方の手番に乱数合議法を用いた自己対戦を2,000局行い、有意な勝率を得られる振れ幅を調査した。深さによっての効果も調べるため、深さ6, 8, 10で実験を行った。その結果を図2の左図に示す。

対戦数は2,000局のため、53%弱勝ち越せば有意水準5%の二項検定で有意に強くなっていると考えられる。深さ6において乱数合議法は十分その強さを高めると言える一方で、深さ8, 10を見ると、有意に勝ち越しているとは言い難い。深さを増やし、指し手がより正確になるとそれに伴って合議法自体の効果が薄くなってしまふためと考えられる。

そこで、更なる結果を導くため提案手法で触れた進行度による調整を行う。激指には盤面の駒の配置と種類に応じてゲームの進行度を0から127の値で示す関数が実装されているため、この進行度に対して乱数の標準偏差の倍率を線形に上げる手法を取った。式1に示す(激指では1歩の価値がほぼ100点)。

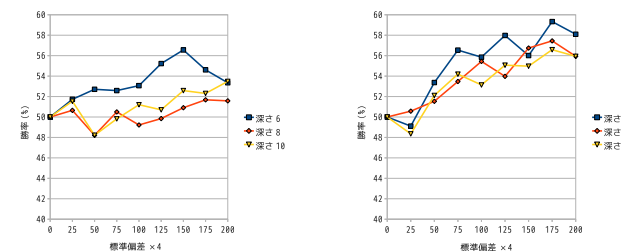


図2 乱数振れ幅による合議を用いた激指の勝率の変化  
左図が通常の合議を、右図が調整を行った合議の結果である。

$$\text{正規乱数の標準偏差} = (\text{初期値}) \times \left( 1 + \frac{(\text{最終的な倍率} - 1) \times (\text{進行度} + 1)}{128} \right) \quad (1)$$

深さ6, 8, 10で最終的な倍率は4倍とし、初期値を変更しながら調査を行った。結果を図2の右図に示す。深さ8, 10でも大きく勝ち越せるようになり、最大で58%近くの高い勝率を出すことが出来ている。

### 4.2 合議棋譜を学習した激指の評価

適切な調整を加えることによって、乱数合議法を用いた激指は元の1台の激指と比較して十分な強さを出すことが出来た。この合議激指を用いて、提案手法の実装を行う。以下、合議激指はその深さに応じて最も高い勝率を出せる調整にしたものを指すこととする。

まずプロ棋士の棋譜を30,000用い、これを用いて学習を行った激指を用意する。これをオリジナルの激指とする。次にオリジナルの激指に乱数合議法を用いた8台の合議激指で自己対戦を行い、棋譜を50,000生成した。これらの合議棋譜をn棋譜加えた30,000+n棋譜で学習を行った激指の強さを一手3秒の2,000局対戦で評価する。

学習の際は、提案手法でも挙げたように棋譜生成の時点で合議激指とオリジナルの激指の候補手の相違を一手ごとに記録しておき、合議激指がオリジナルの激指と異なる手を指した物のみの学習(合議のみ)と、相違に関わらず生成されたすべての棋譜の学習(全部)を学習させる棋譜の数を変更しながら行った。

本研究では評価関数の学習時の探索深さを6としている。したがって、教師データとしては深さ6よりも深い探索を行うのが望ましい。そこで深さ10で棋譜を生成し、実験を行った。また、比較対象として先に挙げた2手法に加え、合議を行わずに、オリジナルの激指を

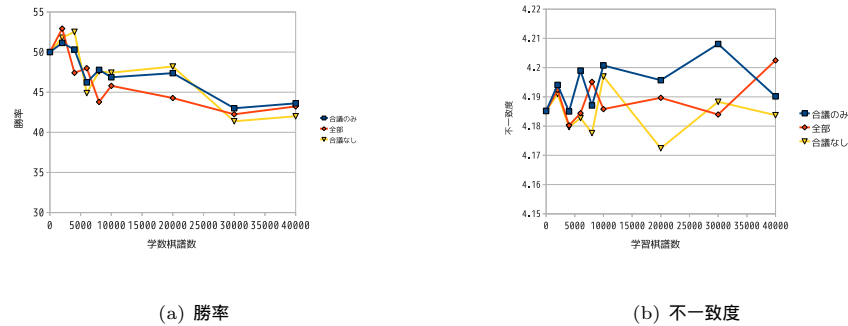


図 3 深さ 10 の合議棋譜を学習させた激指

用いて深さ 10 で自己対戦を行い対戦棋譜を生成した (合議なし)。

合議を行わないで作成した棋譜を学習に用いた激指を作ること、乱数合議法を棋譜生成に用いたことの効果の検証をより正確に行う事が可能である。更に、学習棋譜数 10,000 以下で更に細かく学習を行い、学習曲線を詳しく調べた。その結果を図 3(a) に示す。x 軸が学習に用いた自己対戦棋譜数である。

3 種類とも同じような曲線を描いており、学習棋譜数が増加していくと勝率は緩やかに低下する。2,000 棋譜、4,000 棋譜の学習を行ったときやや勝率が上がっており、高い点では有意といえる 53%を上回るが、学習自体に揺らぎがあることを考えると強くなると言えるかは疑問が残る。勝率が低下する理由としては、学習に用いた自己対戦棋譜がプロ棋士の棋譜と比較すると、十分なレベルに達していないというのが一番の要因と考えられる。このため、自己対戦棋譜を加えていく学習することで、棋士の棋譜の割合が相対的に減ってしまい、悪影響を与える結果となった。学習に用いる棋譜としては深さ 10 でも浅すぎたと考えられる。一方で、合議の有無による差異はあまり無かった。しかし、合議激指が違った手を指した局面のみを選んだ場合、学習に用いる局面数は棋譜全体の 2 割ほどとなるため、学習局面数で考えると他の 2 つに比べて劣っている。

次に不一致度を図 3(b) に示す。不一致度は低いほど良いとされる基準である。10,000 までの結果から、ある程度の揺らぎが生じる事がわかる。その上で図 3(b) を見ると、合議の指した局面のみを選んだ手法は他の 2 つの手法に比べるとやや不一致度が高めに出ている。学習局面数が少ない事を考慮するとより劣っているため、局面選別の手法としては良くない

かったと考えられる。一方で他の 2 つの手法では不一致度は大差が無く、合議棋譜が不一致度に与える影響は合議棋譜の物とほぼ変わらないと考えられる。

## 5. おわりに

本研究では合議激指の強化と学習棋譜生成への応用に取り組み、進行度に対しての乱数幅の変化を導入することで合議激指は勝率を大きく上昇させることが出来た。合議を用いた質のいい棋譜生成は、深さ 10 とした場合では良くないという結果となった。合議により強さは確実に上がっていても、同じ深さの非合議棋譜と学習した時の評価関数があまり変化しないという事実は、良い学習に必要な要素が単純な強さでは無いことを考えさせられ、興味深い。今後の課題としては、まずプロ棋士の棋譜に劣らない程度の深さでの棋譜生成が挙げられる。他には初期のプロ棋士の棋譜数が 30,000 と多かったため、プロ棋士の棋譜数が少ない場合での効果の検証を行う、合議棋譜からの学習させる局面の選択を合議の票の割れ方を考慮して行うなどを考えている。

## 参考文献

- 1) 柿木義一. 5 五将棋における評価関数の自動学習,2008.
- 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. The 11th Game Programming Workshop. pp78-83, 2006.
- 3) 小幡拓弥, 杉山卓弥, 保木邦仁, 伊藤毅志. 将棋における合議アルゴリズム:既存プログラムを組み合わせる強いプレイヤーを作れるか?. The 14th GameProgramming Workshop pp51-58, 2009.
- 4) 小幡拓弥, 杉山卓弥, 伊藤毅志. 乱数合議の有効性に関する一考察. The 15th Game-Programming Workshop. pp9-14, 2010.
- 5) G. Tesauro. Programming backgammon using self-teaching neural nets. Artificial intelligence. pp181-199, 2002.
- 6) 将棋プログラム「激指」. <http://www.logos.t.u-tokyo.ac.jp/~gekisashi/index.html>.
- 7) 金子知適. コンピュータ将棋の評価関数と棋譜を教師とした機械学習. 人工知能学会誌 Vol.27, 2012.
- 8) Bonanza - The Computer Shogi Program [http://www.geocities.jp/bonanza\\_shogi/](http://www.geocities.jp/bonanza_shogi/).
- 9) Tomohito Esaki.A Classification of the Phases in Shogi GameUsing SOM and FCM.CyberGames, 2006.