

コンピュータ将棋における直近の決定履歴を用いた合議アルゴリズム Consultation Algorithm using Recent Decision History for Computer Shogi

日高 信[†] 布目 淳[‡] 平田 博章[‡] 柴山 潔[‡]
Makoto Hidaka Atsushi Nunome Hiroaki Hirata Kiyoshi Shibayama

1. はじめに

コンピュータ将棋において、複数の異なる将棋プログラムを搭載したコンピュータ（以下、プレーヤとする）を用いて採用手を決定する合議アルゴリズムが研究されている[1]。また、各プレーヤに適切な重み付けの値（以下、ポータリングウェイトとする）を与えることにより、単純な多数決によって採用手が一手に定まらない場合でも、より最善に近い候補手を選択するアルゴリズム[2]が提案されている。本稿では、直近の合議決定履歴を考慮してポータリングウェイトを設定する方式を提案する。これにより、試合展開に柔軟に対応した合議を実現する。

2. 関連研究

當間らによる採用実績を用いた合議アルゴリズム[2]では、単純多数決の結果、対局中により多く候補手が採用されたプレーヤほど、より正確に局面を評価できていると考え、ポータリングウェイトの値を大きく設定する。

将棋では、序盤、中盤および終盤の各段階では指し手の戦略が異なる。例えば、序盤や中盤では相手の強い駒を取ることは有効であるが、終盤では、それよりも相手の王を寄せる手の方が有効である場合が多い。當間らのアルゴリズムは、各プレーヤが提示した候補手の採用実績を対局開始時からすべて記録し、採用手の決定に利用する。しかし、この手法では、終盤になっても、序盤のときの採用実績がポータリングウェイトに影響を与えてしまう。そのため、仮に終盤の評価に特化しているプレーヤが存在したとしても、序盤での採用実績の影響で、終盤にそのプレーヤの候補手を軽視してしまう可能性がある。そこで、提案方式では、直近の採用実績のみを用いて採用手を決定する。

3. 直近の決定履歴を用いた合議アルゴリズム

3.1 概要

提案方式では、一定以上過去の合議における決定履歴が悪影響を与えないよう、各プレーヤの候補手が採用された回数を一定の期間のみ記録する。ここで、記録する直近の決定履歴の数を履歴参照回数と呼ぶことにする。履歴参照回数を大きくするほど、多くの履歴を用いて採用手を決定できる反面、中盤以降で戦略決定を誤る可能性が高くなる。また、履歴参照回数を小さくすれば、ポータリングウェイトを設定するための情報が不足し、適切な候補手を選択できなくなる。

3.2 採用手の決定方法

3種類の合議方式を組み合わせることで採用手を決定する。

合議方式 A (単純多数決方式)

最も多くのプレーヤに支持された候補手を採用手とする。単純多数決によって、対局中により多く候補手が採用されたプレーヤほど局面を正確に評価していると考えられる。この方式で採用手が決定した場合、各プレーヤのポータリングウェイトを更新する。

合議方式 B (ポータリングウェイトによる決定方式)

それぞれの候補手を支持したプレーヤのポータリングウェイトの合計が、最大となる候補手を採用手とする。

合議方式 C (ランダム決定方式)

単純多数決によって採用手が決定できず、かつポータリングウェイトを用いても採用手が一手に絞れない場合は、各候補手の間には優劣の差が無い。この場合、プレーヤをランダムに選択し、そのプレーヤが支持した候補手を採用手とする。

合議方式 B と合議方式 C においては、多数決によって採用手が決定されたわけではない。よって、その局面のみでは各プレーヤの間に優劣をつけることができないので、ポータリングウェイトは更新しない。

図1に本方式における合議アルゴリズムの流れを示す。図中の VW はポータリングウェイトの略である。

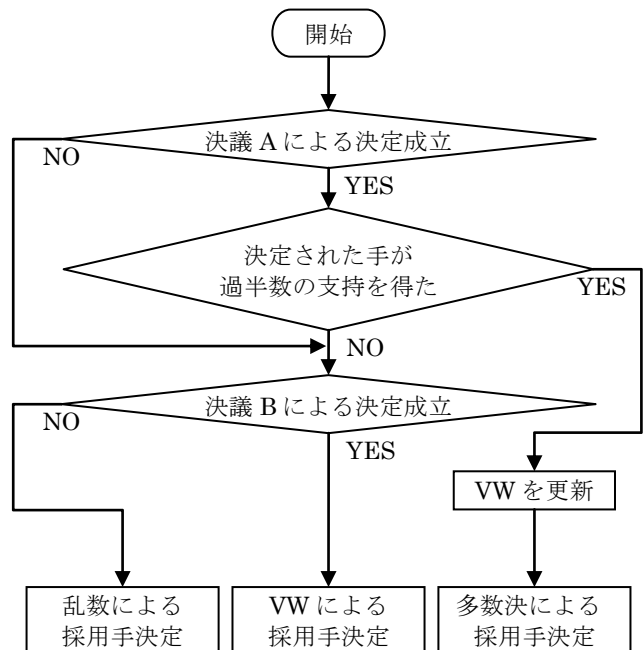


図1: 提案方式の合議フロー

[†] 京都工芸繊維大学大学院工芸科学研究科情報工学専攻
[‡] 京都工芸繊維大学大学院工芸科学研究科情報工学部門
Dept. of Information Science, Kyoto Institute of Technology

本方式では、まず決議方式 A による合議を行う。ただし、単純多数決で採用手を決定できても、全プレーヤの過半数の支持がない場合は、決議方式 B によって採用手を決定する。これは、決議方式 A によって決定した手が過半数の票を集められなかった場合、他の候補手との差がほとんど無いので、過去の決定履歴を参照するためにボーティングウェイトを使うためである。

3.3 ボーティングウェイトの更新方法

履歴参照回数を n 回とすると、直近 n 回の合議における決定履歴を、プレーヤごとに合議サーバ上のリングバッファに記録する。決議方式 A において採用手が決定したとき、候補手が採用されたプレーヤのリングバッファには 1 を、採用されなかったプレーヤには 0 を記録する。決議方式 B および決議方式 C ではリングバッファの値を更新しない。ある局面でのプレーヤのボーティングウェイトの値は、リングバッファ内の 1 の個数の和である。

4. 評価

4.1 評価条件

将棋プログラム Bonanza[3]による乱数合議[4]を用いる。Bonanza の評価関数に乱数を作用させたクローンプログラムを生成し、擬似的に複数の将棋プログラムとして動作させる。図 2 に示すように、5 台の Bonanza クローンプレーヤ 1, 2, 3, 4, 5 を合議サーバに接続し、オリジナルの Bonanza を搭載したノードと対局させる。

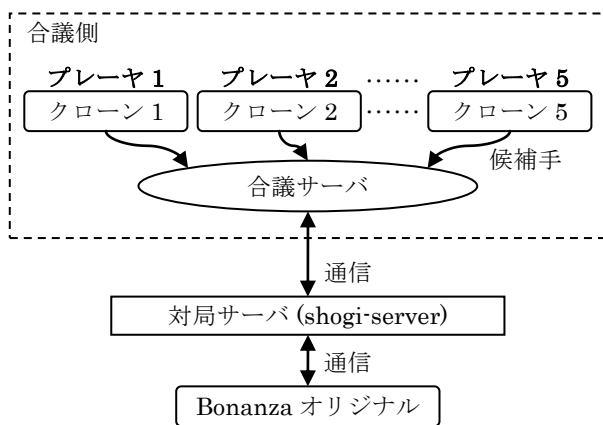


図 2: 評価システムの構成

実験では、ハードウェア構成がすべて同じコンピュータを用意した。将棋プログラム同士の対局には、ネットワーク対局用サーバである shogi-server[5]を使用した。乱数合議を行う上で、Bonanza クローンに与える乱数の標準偏差は 15 とした。

将棋において、1 局の平均終了手数は 110 手から 120 手であるとされる。すると、合議側に手番が回ってくるのは、1 局につき平均 50 回から 60 回程度となる。このうち、対局開始から数十回程度までは、定跡データベースを用いて採用手を決定することが多いため、ほとんどゲーム木探索はしない。 $n = 100$ としたとき、終局時でもリングバッファには常に空きが存在していたので、これを履歴参照回数を無制限としたときのデータとして取り扱う。

4.2 実験結果と考察

実験では、履歴参照回数 n の値を変化させながら、100 局ずつ対局を行った。このときの合議側の勝率を図 3 に示す。横軸が履歴参照回数 n 、縦軸が $n = 100$ のときの勝率を 1 としたときの勝率の比である。

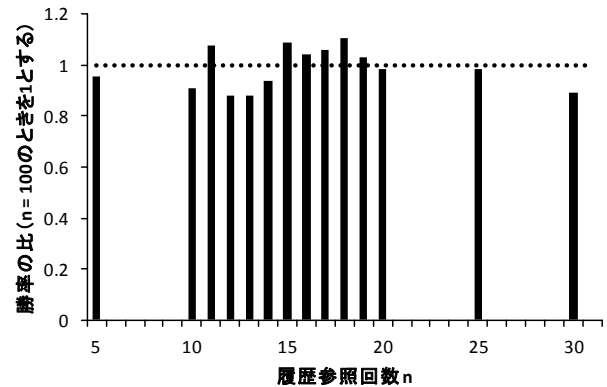


図 3: 履歴参照回数を変化させた場合の勝率の比

図 3 より、 n の値が 15 から 20 までの区間で、履歴参照回数を無制限にしたときの勝率と比べて最大で約 10.6% 高くなった。 n の値が 15 より小さいときの勝率は $n = 100$ のときよりも下回った ($n = 11$ のときを除く)。これは、履歴が不足することにより、適切なボーティングウェイトを設定するための情報が不足したことが原因であると考えられる。また、 n の値を 20 よりも大きくすると、実験結果より、 $n = 100$ のときにほぼ近い値を取っていく。 n の値を適切に設定することで、勝率の向上が期待できることが分かった。

5. むすび

本稿では、過去の決定履歴を用いた合議アルゴリズムにおいて、試合展開に柔軟に対応するために、直近の決定履歴のみを参照する方式を提案した。また、実験により、参照する決定履歴の最適な数を求めた。この結果、対局開始時からの決定履歴をすべて用いるよりも高い勝率を得ることができた。

今後、対局数を増やして実験データの信頼性をさらに高めていくことが課題である。

参考文献

- [1] 小幡 拓弥, 埴 雅織, 伊藤 毅志: “思考ゲームによる合議アルゴリズム～単純多数決の有効性について～”, 情報処理学会研究報告, 2009-GI-22, Vol.2009, No.2, pp.1-5 (2009).
- [2] 當間 啓介, 遠藤 聡志, 當間 愛晃, 赤嶺 有平, 山田 孝治: “合議アルゴリズムにおけるリーダー決定法の検討”, The 26th Annual Conference of the Japanese Society for Artificial Intelligence, 301-R-13-5 (2012).
- [3] “Bonanza – The Computer Shogi Program”, http://www.geocities.jp/bonanza_shogi/.
- [4] 杉山 卓弥, 小幡 拓弥, 斎藤 博昭, 保木 邦仁, 伊藤 毅志: “将棋における合議アルゴリズム～局面評価値に基づいた指し手の選択”, 情報処理学会論文誌, vol.51, no.11, pp.2048 - 2054 (2010).
- [5] “shogi-server”, <http://shogi-server.sourceforge.jp/>.