乱数合議の有効性に関する一考察

小幡拓弥† 保木邦仁†† 伊藤毅志†

概要

本研究では、合議法の一種である乱数合議法の原理の解明を目指した実験を行った。その結果、乱数合議法はB* searchや共謀数探索の目指す"最も正解のありそうな方向に進む"という効果を手軽に得る、容易に実装可能な方法であるということが示唆された。本稿では、この実験結果とその考察について報告する。

A discussion on efficiency of consultation method with pseudo-random number

Takuya Obata†, Kunihito Hoki††, Takeshi Ito†

Abstract

In this research, the mechanism of the consultation method with pseudo-random number is investigated by means of several experiments. The results suggest that the method is a simplified implementation of B* or conspiracy number search to find the most probable pathway to the right answers. This paper presents the experimental results and corresponding discussion.

† 電気通信大学情報工学科
Department of Computer Science ,
University of Electro-Communications

†† 電気通信大学先端領域教育研究センター
University of Electro-Communications,
Center for Frontier Science and Engineering

1. はじめに

1965年にゴードン・ムーアによって提唱された、集積回路におけるトランジスタの集積密度が2年ごとに倍になるという経験則は(1、近年になって当てはまらなくなった。プロセッサ単体の性能は伸び悩み、マルチプロセッサ化が進んでおり、それにつれて、並列計算の重要性が高まっている。ゲームプログラミングにおいても並列計算は極めて重要なテーマである。

疎結合並列環境を利用した思考プログラムの強さ向上の手法として、合議法が報告されている(2,3。第19回世界コンピュータ将棋選手権3位の文殊、同第20回5位のBonanza Felizは、合議法の一種である乱数合議法を採用した将棋プログラムである。合議法は、複数の思考ゲームプログラム（あるいは思考ゲームプレイヤ）から合議システムを構築する手法

である。合議システムの各プレイヤはゲームの局面毎に、自身の最善と思う手といくらかの付加情報を提供する。合議システムは、これらを一定のアルゴリズムで処理してグループとしての指し手を決定する。

関連する研究として、将棋において疎結合並列環境を利用し、ゲーム木をさらに深く探索する試みがある。これに関しては、文献 4 による報告やそこに記述の関連研究を参照されたい。

2009 年に、我々は合議法が将棋プログラムを有意に強くすることを報告した。しかしながら、合議法が効果を発揮する原理については不明であった。

本研究では、合議法の中でも乱数合議法に着目し、その原理の解明を目指した実験を行った。その結果、乱数合議法が B* search(5, 6)や共謀数探索(7, 8)に近い性質の探索を、実用可能な実装方法で実現する手法であるということが示唆された。本稿では、この実験とその考察について報告する。

2. 乱数合議法

乱数を用いて 1 つの思考プログラムから複数のゲームプレイヤを生成し、合議システムを構成させる手法を乱数合議法と呼ぶ。この手法は、Min-Max 探索をベースとしたプログラムに対して適用することが可能である。乱数合議法では、プログラムの静的評価関数によって算出される評価値に、適当な大きさの乱数を加える。個々のプレイヤに異なる乱数系列を与えることによって、それぞれ異なる形勢判断を持つプレイヤを生み出すのである。この手法は非常に単純であり、このように生成されたプレイヤによる合議が元のプログラムより強くなることは既に報告している(2, 3)。

我々は、乱数合議法が効果を示す原理の説明として、「評価値に乱数を加えたプレイヤは、自身にとって合法手が多く、相手にとって少なくする戦略を取り、このことが棋力を向上させている」という仮説を立てた。Min-Max 探索において、ある Max ノードで子ノードの評価値が完全な乱数であるとすると、その Max ノードの評価値は子ノードの数が多いほど高くなりやすい。Max 値を取るため、乱数のサンプルが多いほど高い値を得るのは自明である。反対に Min ノードでは、子ノードの数が多いほど小さい値をとりやすくなる。乱数を使用するために、1 回 1 回の思考は不安定なものとなる。しかし、複数のプレイヤによる思考を集めて合議をすることで、この不安定性を解消しつつ上記の仮説のような手を選択しているのではないかと考えられる。将棋などのゲームでは一般に、合法手が多い方が少ないよりも有利であると言われている。このことが正しければ、上記のような指し手の選択は有効に働くと考えられる。

3. 乱数のみの評価関数による合議

仮説を検証するために、ゲーム木探索をできる限り簡素なものに変更した Bonanza 4.1.2 を用いて実験を行った。主な変更点は、Late Move Reduction 等の枝刈りや局面の優劣関係を判断する処理、静止探索を取り除いたことである。さらに、局面評価の知識を全て削除し、評価関数が正規分布 $N(0, 1000^2)$ に従う乱数を返すように変更した。

このプログラムを用いて、静的評価関数が単なる乱数値を返すプレイヤと、そのプレイヤ複数により構成される合議プログラムの対戦の勝敗を調べた。合議方法は、多数決合議と楽観的合議の 2 通りとした。正規乱数の標準偏差 1,000 という値は理論上意味を持たないが、実装上指し手の生成順の影響を受けないために十分大きな値を与えた。合議プログラムのプレイヤ数は、多数決合議で 4, 8, 16、楽観的合議で 2, 4, 8, 16 とした。対局は両者一手 10 万ノード（1 秒に相当）で、各 1,000 局とした。表 1 に対戦成績を示す。

表 1 合議プログラム対単独プレイヤの対戦結果。各プレイヤの評価関数は分布 $N(0, 1000^2)$の正規乱数を返す。勝率は合議プログラム側からみたものである。

プレイヤ数	2	4	8	16
多数決	―	56.5%	67.9%	71.2%
楽観的	54.1%	62.4%	63.6%	68.9%

合議プログラム側が大きく勝ち越す結果となった。両者とも形勢判断の知識を一切持たず、単に乱数を返すのみであるにも関わらず、両者の間には強さの違いがある。多数決について考えると、グループとして良い手を選択するには、個々のプレイヤが一定以上の確率で「良い手」を選ぶ必要がある。つまり、単に乱数を返すだけのプレイヤも、ある程度知的な選択をしているといえる。一方、最も評価値の高い手を選択する楽観的合議でも勝率が向上している。評価関数が乱数のみを返す場合、評価値のより高い手とはつまり、仮説のような手である。そのような手を選ぶことが、棋力向上に繋がっているといえる。

実験の棋譜の定性的な分析からは、いくつかの興味深い傾向が見られた。各プレイヤは局面評価の知識を、駒の価値を含め一切持たないにも関わらず、駒の損得に対してある程度の反応を見せた。必ずではないものの、取れる駒を積極的に取りに行く傾向が見られた。これは、自分の合法手を増やし、相手の合法手を減らすことに繋がっている。また、王手を積極的にかける様子も見られた。王手は相手の合法手を減らす典型的な手である。

図 1 は、初期局面から深さ 3 までの 1,000 回の Min-Max 探索における、各ノードの評価値の分布を表したものである。評価関数は正規分布 N (0, $1{,}000^2$)に従う乱数値を返す。3 手目の先手の合法手が多くなる例として、初手に▲7 六歩を指した場合と、少なくなる例として、初手に▲3 八金を指した場合を示した。各ノードで評価値は、正規分布に近い形で分布した。

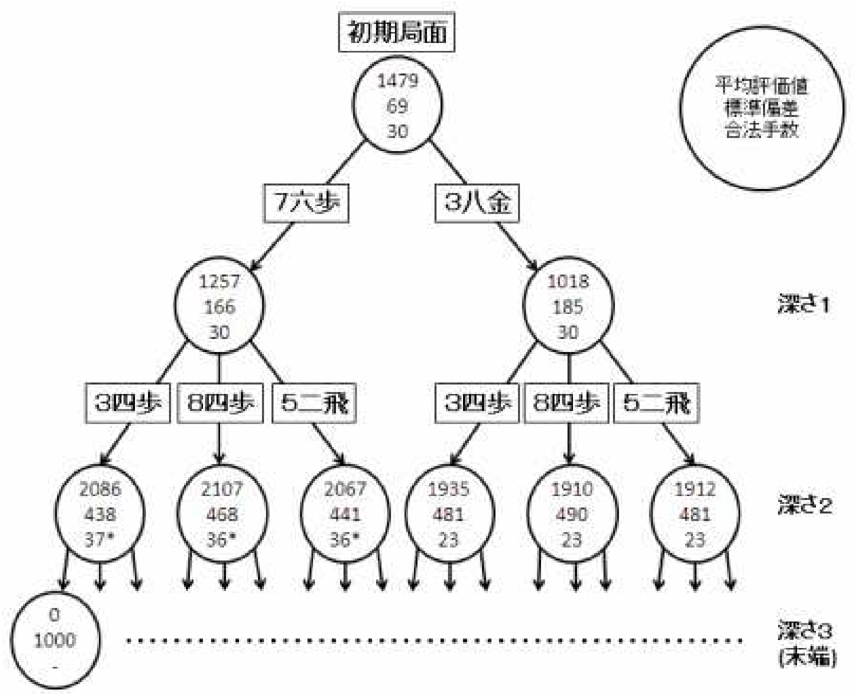

図 1 初期局面の深さ 3 の Min-Max 探索の評価値の分布。実際の探索では全ての合法手を展開しているが、図には一部を抜粋した。各ノード内の数値は、上から平均評価値、標準偏差、合法手数である。

表 2 乱数プレイヤの指し手の選択確率

指し手	1 プレイヤ	16 合議	256 合議
7 六歩	12.60	30.30	88.23
9 六歩	8.60	13.94	7.25
1 六歩	7.90	11.65	3.6
3 六歩	6.50	7.74	0.60
2 六歩	6.00	6.51	0.28
8 六歩	4.30	3.20	0.06
7 八飛	4.20	3.03	0.05
6 六歩	4.20	3.07	0.05
5 六歩	4.00	2.75	0.02
4 六歩	3.80	2.48	0.01
6 八王	3.70	2.33	0.003
9 八香	3.70	2.34	0.004
3 八飛	3.50	2.06	0.003
1 八飛	3.50	2.09	0.005
7 八銀	2.70	1.21	0.00
7 八金	2.10	0.69	0.00
5 八飛	2.10	0.70	0.00
1 八香	2.10	0.71	0.00
4 八飛	1.80	0.51	0.00
6 八銀	1.80	0.50	0.00
6 八金	1.80	0.50	0.00
5 八王	1.70	0.45	0.00
6 八飛	1.60	0.39	0.00
4 八王	1.40	0.31	0.00
4 八金	1.00	0.15	0.00
5 八金左	0.90	0.12	0.00
5 八金右	0.80	0.10	0.00
4 八銀	0.60	0.05	0.00
3 八金	0.60	0.05	0.00
3 八銀	0.50	0.04	0.00

(%)

図 1 に示される通り、▲7 六歩は▲3 八金よりも平均的に高い評価値を得ており、より選択されやすい手であるといえる。実際、1,000 回の探索で、▲7 六歩が選択された回数は最も多い 126 回であった。次点は 85 回の▲9 六歩で、これも先手 3 手目の合法手を増やす手である。一方、▲3 八金が選択されたのは 6 回で、▲3 八銀などと並んで最も少なかった。図の結果は数値的に求めたが、正規乱数の代わりに一様乱数を用いた場合は、このような結果を解析的にも導くことが可能である。

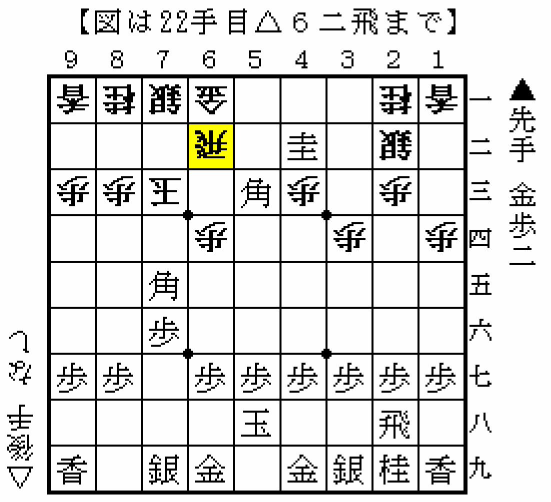

図 2 実際に現れたある局面

表 2 に、上記 1,000 回の思考で合法手 30 手のそれぞれが選択された割合と、16、256 プレイヤで多数決した場合に各手が選択された割合を示す。なお、多数決合議の指し手の選択確率は 100 万回のシミュレーションで計算し、同数の票で分かれた意見はランダムに選択されることとした。自明なことであるが、1 プレイヤで選択される確率の高い手が、多数決によってさらに選択されやすくなった。表で確率が高い手は先手 3 手目の合法手が多くなる手であり、確率が低い手は合法手の少なくなる手である。

表 3 に、指し手選択確率の探索量依存性を示す。探索ノード数が 1,000、10,000、100,000 の 3 通りを比較した。探索が深くなるほど、指し手選択確率が偏る傾向が見てとれる。深くなるほど Max, Min ノードと交互に手を選択する回数が増えて、より評価値に差がはっきりと表れるからと考えられる。

図 2 は、評価値 0＋ 乱数のプログラム同士の対局で、実際に現れた局面である。相手の合法手を減らす代表的な手である、王手をかける手が存在するこ

の局面で、同じく評価値の分布を観察すると図 3 のようになった。

相手に王手をかける▲６二角成は次の相手の合法手が5手、全く関係のない手▲３八金は30手である。また、▲６二角成は駒を取る手でもあることから、その後の自分の合法手も前者の方が多い。図から見てとれるように、▲６二角成の手は▲３八金と比べて平均的にかなり高い評価値を得ている。

表 3 探索量別の各手の選択確率(%)

指し手	1,000N	10,000N	100,000N
７六歩	19.7	47.6	59.8
９六歩	4.4	4.8	4.2
１六歩	3.7	2.0	2.9
３六歩	2.9	2.6	1.1
２六歩	3.2	1.7	1.7
８六歩	3.7	1.9	1.8
７八飛	2.8	1.7	1.3
６六歩	3.0	1.4	0.4
５六歩	3.7	2.1	1.4
４六歩	4.7	2.1	1.7
６八王	2.5	1.9	0.9
９八香	2.8	2.0	1.4
３八飛	3.1	2.2	1.6
１八飛	3.7	1.4	1.2
７八銀	3.3	1.8	0.6
７八金	2.7	2.1	1.2
５八飛	2.1	1.8	1.1
１八香	2.4	1.8	1.5
４八飛	2.9	1.3	0.5
６八銀	2.1	2.0	1.1
６八金	1.8	1.3	2.2
５八王	2.3	2.0	1.4
６八飛	2.9	2.4	1.6
４八王	2.3	1.4	1.3
４八金	2.4	0.7	1.2
５八金左	1.7	1.4	1.1
５八金右	1.8	0.8	1.1
４八銀	2.4	1.8	0.9
３八金	1.5	1.0	0.7
３八銀	1.5	1.0	1.1

(%)

表 4 は、上の局面での各手の選択確率である。初期局面と比べて各手の合法手の数の差が大きいため、選択確率にも極端な差がついている。上位 4 つの手はいずれも王手をかける手であるが、その次の自分の手番で合法手が多くなる手、つまり相手の駒を取る手や成る手が特に高い確率となっている。

4.. おわりに

一般に、将棋などの思考ゲームではさせる手の多い方が有利であることが知られている。この手の広さが、末端評価関数に乱数を用いた深さ一定の Min-Max 探索にて統計的に考慮されることが示された。また、この傾向は探索量が多いほど強く表れた。さらに、当然ながら、合議法によりこの統計的な傾向は加速した。

通常、将棋プログラムの静的評価関数では、飛び駒が移動可能な升目の数等、合法手の数を考慮する。一方、乱数合議法では、Min-Max 探索を行う過程で、動的に合法手数を考慮するという特徴を持っている。

局面評価関数を使用していない本研究では、合議側の勝率は 70%を超えた。通常の静的評価関数値に乱数を足した前回の報告では、合議側の勝率は 60%に満たないことから(2, 3)、静的評価関数の精度が高い

表 4 各指し手の選択確率

指し手	選択確率
６二角成	52.6
６四角引成	22.5
６四角上	16.2
８四角	7.5
７四金	0.5
８四金	0.4
６三金	0.2
７二金	0.1
その他	0

(%)

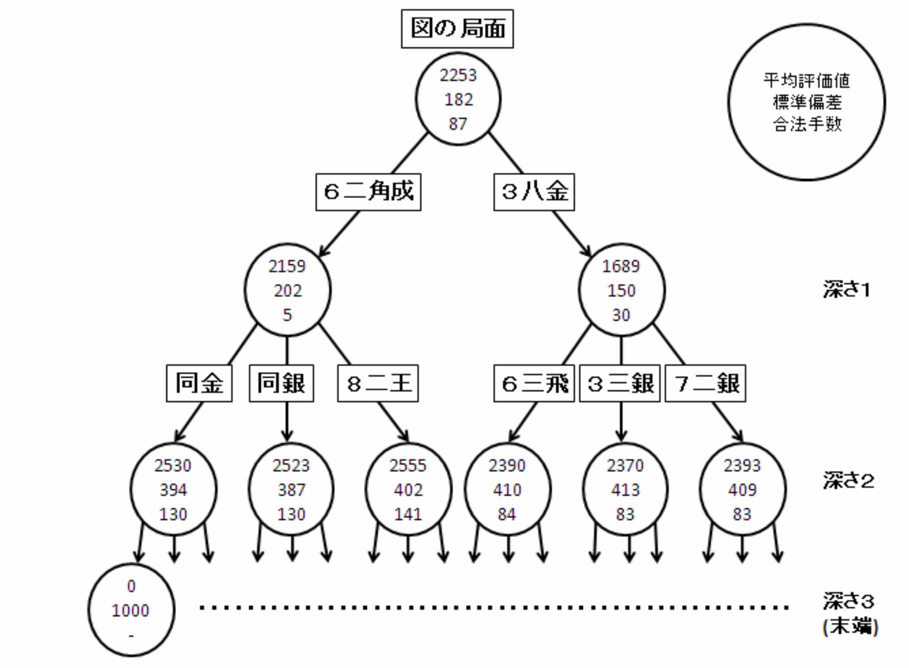

図 3 ある局面の深さ 3 の Min-Max 探索の評価値の分布。実際の探索では全ての合法手を展開しているが、図には一部を抜粋した。*Bonanza は角の不成を生成しない

ほど、乱数合議の効果は減少すると考えられる。従って、将来の展望として、局面ごとの評価関数の信頼度に基づき、正規乱数の分散を可変にする手法が考えられる。

乱数合議法は、評価関数の値に幅を持たせるという点において、B* search と類似している。どちらの手法も、有望な子ノードが多い手を発見するという効果を期待したものである。しかし、具体的な動作の内容は異なり、B* search では探索値の幅を用いて、有望な子ノードがより多くある局面の探索の延長を行う。一方、乱数合議法では単に有望な子ノードの多い手を指す。また、B* search では評価関数の幅を直接考慮して Min-Max 探索を行うのに対し、乱数合議法では乱数を加えた探索を多数行うことにより探索値の幅を評価する。

B* search は実装が難しく、チェスや将棋においてこの手法を用いた強いプログラムは現在のところ無い。乱数合議法は、実用可能な実装方法で B* search と近い効果を発揮する手法であると考えられる。

5. 参考文献

1) ”Cramming more components onto integrated circuits”, Electronics Magazine 19 April, 1965

2) 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て強いプレイヤを作れるか？. The 14th Game Programming Workshop pp. 51-58, 2009

3) 杉山卓弥, 小幡拓弥, 保木邦仁, 伊藤毅志. 将棋における合議アルゴリズム – 評価値を用いる効果について－. The 14th Game Programming Workshop, pp. 59-65, 2009

4) 田中哲郎, 金子知適. 大規模クラスタシステムでの実行－GPS 将棋の試み－. 情報処理学会誌, VOL.51, pp. 1008-1015, 2010

5) Berliner, H.J. The B* tree search algorithm: A best-first proof procedure. Artificial Intelligence 12, pp. 23-40, 1979.

6) Palay, A.J.. The B* tree search algorithm – New results. Artificial Intelligence 19, pp.145-163, 1982

7) D. A. McAllester. "Conspiracy Numbers for Min-Max Search". Artificial Intelligence 35, pp. 287-310, 1988.

8) J. Schaeffe. "Conspiracy Numbers". Artificial Intelligence 43, 67-84, 1990.