情報処理学会論文誌 Vol.55 No.11 2353–2361 (Nov. 2014)

難しさが手番で異なるゲームのモデル化とモンテカルロ木探索の性能の分析・改善

今川 孝久[1,a)] 金子 知適[1]

受付日 2014年2月20日, 採録日 2014年9月12日

概要：モンテカルロ木探索（MCTS）は囲碁をはじめとして広い応用を持つ探索手法である．しかし，チェスや将棋では性能が限定的であり，囲碁でも置き碁などが苦手であると経験的に知られているように，MCTS が有効に働くためのゲームの性質は明らかでない．これまでに仮想的なゲームを通じた分析が行われており，Finnsson らによって極端に単純化されたゲームであっても MCTS の性能が探索空間の大きさだけでは説明できないことが明らかになっている．本研究では「最善手を互いに選び続ければ引き分けになるが，次善手を選んだ場合の度合いが手番によって異なる」という状況を表すゲームを提案し分析を行った．実験において，提案ゲームは Finnsson のものと同程度の単純さを持つにもかかわらず，その設定は MCTS の性能に大きく影響することを示した．またこのゲームにおいては性能を改善する鍵が，終局時のスコアを利得に変換する過程にあることを示し，勝ち負けの評価の境界をプレイアウト終了時のスコアの頻度が最大の位置に移動する手法を用いると，既存の dynamic komi を用いる場合より効果的であることも示した．

キーワード：2 人ゲーム，難しさが手番で異なるゲーム，モンテカルロ木探索

Analyses of Monte Carlo Tree Search in Artificial Games Where Difficulty Differs by Turns

Takahisa Imagawa[1,a)] Tomoyuki Kaneko[1]

Received: February 20, 2014, Accepted: September 12, 2014

Abstract: Monte Carlo Tree Search (MCTS) is widely applied in many search problems, as well as Go. However, its effectiveness is limited in Chess and Shogi. As it is also empirically known that MCTS does not work effectively at Okigo in Go, features of games is not clear where MCTS works effectively. Analyses on virtual games have been done so far, Finnsson unveiled that MCTS's effectiveness cannot be explained only by size of search space even in extremely simple games. In our research, games in a situation were proposed and analyzed, where two players have same penalties if they keep choosing best move, and different penalties if they missed best one. In our examination, nevertheless the games is as simple as Finnsson's ones, it is shown that the settings greatly influences effectiveness of MCTS. Also it is shown that the problem will be mitigated by adjusting a function that converts game scores gained by playouts into rewards (win, draw and loss) in the games. In addition, it ! i s shown that MCTS with a method adjusting a threshold between win and loss to a score of the maximum frequency is more effective than plain MCTS and MCTS with an existing method, dynamic komi in the games.

Keywords: 2 player game, Games where difficulty differs by turns, Monte-Carlo tree search

1. はじめに

モンテカルロ木探索（MCTS）は，囲碁などの探索空間が大きなゲームでも，おおむね効果的に動作する優れた探索手法である [3], [7]．囲碁では，MCTS が旧来の Min-Max

[1] 東京大学
The University of Tokyo, Meguro, Tokyo 153–8902, Japan
[a)] imagawa@graco.c.u-tokyo.ac.jp

探索を基本とした手法よりはっきり勝るとされているが，他のゲームではまた異なる状況にある．チェスや将棋では，いくつかの研究はあるものの [9]，旧来の探索を MCTS が超えたという報告はまだない．Lines of action では評価関数を組み込んだ変更などを行って旧来手法に並んだという報告がある [10]．囲碁においても置碁のように，互先と比較して，MCTS が苦手とする状況があることが知られている．このように MCTS が有効に働く条件はまだ明らかになってはいない．本研究では，そのような条件の 1 つとして，勝ちやすさが手番で異なる状況について議論する．

探索の性能を論ずるうえで，仮想的なゲームを利用して実験・議論することは有力な手段である．仮想ゲームでは，実際のゲームでは一般に分からない，厳密な最善手などの情報を分析の際に利用でき，また，探索空間の広さなどに関するパラメータの調節が容易であるという利点がある．代表的な例として P-Game [5], [11] や，Ramanujan のもの [8]，Finnsson のもの [4] がある．その中で Finnsson のゲームは P-Game の枝の重みを特別な規則でつけた場合に相当し，このゲームにより MCTS の性能が探索木の深さと幅以外の要因に強い影響を受けることが示されている [4]．詳しくは 2.3 節で述べる．

本研究では，Finnsson の設定を一歩拡張し，「Min-Max 探索で得られるゲームの値は 0（引き分け）であっても，実際には片方のプレイヤが勝ちにくい状況」について議論する．いい換えると，お互い最善手を打てば互角だが，次善手以降の価値が手番によって異なる状況である．これはたとえば将棋で，先手は入玉して安全な状況だが，後手はまだ途上で捕まる危険があり，しかし最善手を指し続ければ持将棋で引き分けになるという状況と共通の性質を持つ．また片方のプレイヤが勝ちにくいという点では，置き碁とも関係が深い．ただし，置碁の上手は最善手を打ち続けても下手が間違えなければ負けると予想されるが，本稿では最善手を逃さないことに着目するために Min-Max 値が 0 の局面のみを扱う．

本研究が対象とする局面は既存の仮想ゲームの一部を変更することで容易に作成可能であり，これを用いて以下の結果を示した．実験から MCTS はこのような状況を苦手とするが，その原因はプレイアウトの勝率の期待値が引き分けから（Min-Max 値から）大きく離れていることにある．性能改善のためには，スコアを勝ち負けに変換する境界を定める閾値の調整が有効であり，ゲーム作成時のパラメータを知る神の視点では UCB1 を前提とした理想的な閾値を求められる．プレイヤの観測できる情報からは，プレイアウト結果の頻度が最大となるスコアを閾値とする手法が有力であり，実験からは dynamic komi などと比較しても有効に働く．

2. 背景

この章では本研究で扱う MCTS の有力な一手法として UCT の概要と，形勢が偏った状況下での UCT の改善手法である dynamic komi，また仮想ゲームの性質と UCT の性能についての既存の研究を紹介する．

2.1 UCT

UCB1 [1] は多腕バンディット問題 [6] に対するアルゴリズムの 1 つである．この問題では，各選択肢ごとに確率分布が決められていて，ある選択肢を選ぶたびにその利得が確率的に得られる．プレイヤは，確率分布を知らない状況で決められた回数だけ選択肢を選び，利得の総和の最大化を目指す．戦略としては，利得を得るために期待値が高そうな選択肢を選ぶことと，より良い選択肢を探すために情報の少ないものを試すことの両方が重要である．UCB1 アルゴリズムでは，有望さとその評価の不確かさを組み合わせた評価基準である UCB 値を定義し，それが最大となる選択肢を選ぶ．つまり，取りうる選択肢の集合を A，選択肢 j を選んだ場合の平均利得を $\bar{r_j}$ その選択回数を n_j とすると

$$\arg\max_{j \in A} \bar{r_j} + \sqrt{\frac{2\log(s)}{n_j}} \tag{1}$$

という選択肢を毎回選択する．平均利得 $\bar{r_j}$ により，有望な選択肢が，それに加えられる補正項により，評価が相対的に不確かなものが選ばれる．

UCB1 では n 回の試行中，選択肢 i を選ぶ回数の期待値 $\mathbb{E}[T_i(n)]$ の上限が示されている．具体的には選択肢 i の利得の期待値を μ_i，利得の期待値が最大のものを $*$，その期待値を μ_* と表記すると $\Delta_i \equiv \mu_* - \mu_i$ として

$$\mathbb{E}[T_i(n)] \leq \frac{8\ln(n)}{\Delta_i^2} + 1 + \frac{\pi^2}{3} \tag{2}$$

という不等式が成り立つ．このような戦略を MCTS に応用した手法が UCT である．UCT [5] では探索木のどの葉からシミュレーションするかの決定を多腕バンディット問題の変種と見なして，UCB1 をその決定に用いる．UCT ではまず，根から UCB 値が最大の手を繰り返し葉に到達するまで選んでいき，葉に到達したら，互いにランダムに着手するシミュレーションを行い勝敗を求める（これをプレイアウトと呼ぶ）．なお，その節点を訪問した回数が閾値を超えていたら探索木を成長させる．次に，プレイアウト結果（利得）を親・先祖に伝えていく．そのことを繰り返し行い，評価の精度を高めていく．UCT でも，式 (2) に相当する評価として，様々な仮定を置くと，$\mathbb{E}[T_i(n)]$ に対して Δ_i の自乗に反比例するという上限が知られている．そして，探索終了後に最善手を選ぶ方法は「最も訪問した回数の多い節点を最善手として選ぶ」とすることが一般的

Algorithm 1 Score Situational (Simplified)

$$\text{BoardOccupiedRatio} \leftarrow \frac{\text{OccupiedIntersections}}{\text{Intersections}}$$
$$\text{GamePhase} \leftarrow \text{BoardOccupiedRatio} + s$$
$$\text{KomiRate} \leftarrow (1 + \exp(c \cdot \text{GamePhase}))^{-1}$$
$$\text{Komi} \leftarrow \text{Komi} + \text{KomiRate} \cdot \mathbb{E}[\text{score}]$$

Algorithm 2 Value Situational (Simplified)

```
if Value < red then
  if Komi > 0 then
    Rachet ← Komi
  end if
  Komi ← Komi - 1
else
  if Value > green ∧ Komi < Rachet then
    Komi ← Komi + 1
  end if
end if
```

図 1 2 種の dynamic komi（文献 [2] から必要部分を抜粋）
Fig. 1 Two methods for dynamic komi (extracted from literature [2]).

で，本研究でも採用した．

2.2 dynamic komi

置き碁のように一方のプレイヤが有利な状況では，有利な側は勝ちをより確実なものにする手を，不利な側は不利な状況を拮抗した状態に近づける手を指すことが求められる．しかし，UCT で思考を行うと有利な状態から互角に，不利な状態からより不利な状態になってしまうことがある．このことの主な原因は，有利な場合はどんな手でもプレイアウト結果はほぼ勝ちで，不利な場合はその逆となり，各手の評価に差が付かず，その結果，悪い手を選びやすくなってしまうことにあると考えられる．

文献 [2] の dynamic komi はそのような問題への対処として提案された手法である．dynamic komi では，仮想的にコミを調整することで勝ち負けの境界をずらし，勝ち負けの頻度を同程度にする．これは一般のゲームでは，ゲームの値（スコア）を調整することに相当する．スコアの調整方法は，Score Situational と Value Situational の 2 種類提案されている．

Algorithm 1 に示した Score Situational はスコアの大きさに応じて補正する方法である．GamePhase というゲームの進行状態をシグモイド関数で変換して KomiRate という補正速度を定める．そして，スコアの期待値（補正後）に KomiRate を掛けた量を Komi に足す．このようにして補正後にスコアの期待値を 0 に近づけることを目指す．

Algorithm 2 に示した，Value Situational は利得に応じて補正する方法である．利得平均が green という閾値を超えたら（勝ちが多いので）Komi を増やし，逆に red という閾値を下回ったら（負けが多いので）Komi を減らすことが主な動作である．それに加えて，Rachet という Komi の上限を設け，Komi のむやみな上げ下げを抑えている．

G	x	x	x	x	x	x	x	S
0	1	1	1	x	1	1	1	0
0	1	1	1	x	1	1	1	0
0	1	1	1	x	1	1	1	0
S	x	x	x	x	x	x	x	G

図 2 対称な仮想ゲームの例
Fig. 2 An example game with a symmetric configuration.

文献 [2] 中の両アルゴリズムでは，それぞれ $s = 0.75$，$c = 20$，red $= 0.45$，green $= 0.50$ が用いられている．なお，前者では，s，c を十分大きく設定し，後者では，red を 0 に green を 1 に設定することで通常の UCT と同じ振舞いになる．また，両アルゴリズムでは，初期局面から 20 手未満の局面で探索を行うとき，例外として置き石に応じて線形にコミを加える．しかし，本稿で扱う状況では置き石に相当するものはないので，省いた．

2.3 仮想ゲーム

ゲームの性質が MCTS の性能にどのように影響するかを調べた研究として，文献 [4] がある．この研究では囲碁などのゲームと同様，2 人零和有限確定完全情報ゲームに属する仮想ゲームを提案している．また，その仮想ゲームでの探索空間の大きさと MCTS の性能の関係を示している．

仮想ゲームのルールを説明する．ゲームでは**図 2** のような盤を用いる．ゲームの初期局面では駒は S（Start）に置かれる．各手番では駒を 1 つ前に進め，その際，横に並ぶどの升を選んでも良い．各プレイヤは順番に手を指し，ともに G（Goal）に達したらゲームが終了する．各升には秘密のペナルティが割り当てられていて，指した手（選んだ升）に対応したペナルティが科せられる．ゲーム終了時，各自のふんだペナルティの和が少ない方を勝ちとする．なお，各プレイヤのペナルティはゲーム中は隠されており，ゲーム終了時に初めてスコアが明かされる．仮想ゲームでのスコアは初期局面から各プレイヤがふんできた升のペナルティの総和の差である．つまり，スコア=（相手のペナルティの総和）−（自分のペナルティの総和）となる．

本稿で扱う仮想ゲーム（図 2）は，各局面に最善手が 1 つだけあり，それ以外の手を選んだ場合にどのような不利益があるのかをモデル化したゲームである．“x” は壁を意味し，そこには移動・通過できないことを意味する．このゲームには，ペナルティが 0 である手がつねに存在し，最善の戦略はその手を打ち続けて引き分けとすることである．それ以外の手のペナルティは様々な設定が可能だが，図 2 ではすべて 1 である．

このようなゲームは分析の際，探索空間の大きさの変更が容易である．また，分析者は，プレイアウト終了時のス

コアとそれにともなう勝敗というプレイヤに利用可能な情報に加えて，途中の各局面でのペナルティを利用できる．対局により勝率を測定する場合も最善手をつねに選ぶプレイヤ（最適プレイヤと呼ぶ）を対戦相手とし，相手のミスにより偶然勝つようなことがなく，正しい手を選べたかどうかを測定できる．

Finnsson はこの研究で，探索空間の広さが同じでもペナルティの設定次第で難しさが異なることを実験的に示した．なお，プレイアウト数は，局面の訪問数の合計を一定（5,000 回）にすることで決めている．これは，実時間でプレイアウト数を決めるのと似た効果を得るための工夫と考えられる．1 つの節点を訪問するのにかかる時間が一定と仮定すれば，一定の時間でプレイアウト数を決めるのと等しくなるという性質がある．以後このゲームを対称ゲームと呼ぶことにする．盤の広さは，ペナルティが与えられている列の長さを盤の長さ，行の長さを盤の幅とそれぞれ表現する．たとえば，図 2 の盤は長さ 3 幅 4 である．なお，左の盤を先手が使い，右の盤を後手が使う．

3. 新しいモデルゲーム

この章では各探索手法の得手不得手を分析する対象として，手番によって難しさが異なる 2 種類の状況を提案し，具体的な仮想ゲームとして定義する．

3.1 非対称ゲーム（一律）

非対称ゲーム（一律）では後手のプレイヤの最善手以外のペナルティを大きく設定した．そのペナルティには，1 回でもその升を選ぶと負けが確定する値を用いる．具体的には後手番の最善手以外のペナルティは（盤の長さ）+1 とした．一方，先手のペナルティは図 2 の対称ゲームと同じままとする．長さ 3 幅 4 の盤の例を**図 3** に示す．ペナルティの差から後手にとっては勝ちにくいゲームであるが，最善手を選び続ければ影響がないため，初期局面の Min-Max 値は元の図 2 と同じく 0 である．

3.1.1 対称ゲームとの勝率の違い

この節では，対称なゲームと比べた難しさを具体的に議論するため，両ゲームで対戦実験を行い勝率の差を示す．プレイアウト数は Finnsson に倣い，節点の訪問数を 5,000 回として決めた．対戦相手は最適プレイヤとし，最適プレイヤは後手とした．実験は 1,000 試合行い，盤の幅 4 とした．利得は UCT の囲碁での一般的な使用法に倣い，スコアを勝ちを 1，引き分けを 0.5，負けを 0 に利得を変換して用いた．

長さを変えた場合の勝率の変化を**図 4** のグラフに掲載する．非対称ゲームでは対称ゲームの場合と比べて UCT の性能が明らかに低い．長さを 6 以上にすると，勝率がほとんど 0 となる．長さが短い場合には，読みきれることができるが，盤が長くなるにつれて，そのようなことも少なくなり勝率が下がると予想される．

G	x	x	x	x	x	x	x	S
0	1	1	1	x	4	4	4	0
0	1	1	1	x	4	4	4	0
0	1	1	1	x	4	4	4	0
S	x	x	x	x	x	x	x	G

図 3 非対称ゲーム（一律）の例

Fig. 3 An example game with an asymmetric configuration (uniform).

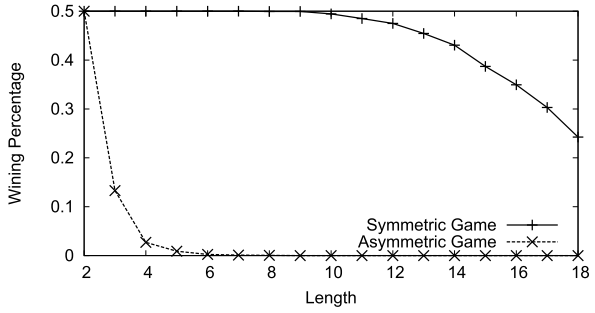


図 4 対称・非対称ゲーム（一律）で盤を長くした場合の UCT アルゴリズムの勝率の変化

Fig. 4 Relationship between winning percentage of UCT and board length, in asymmetric (uniform) and symmetric configurations.

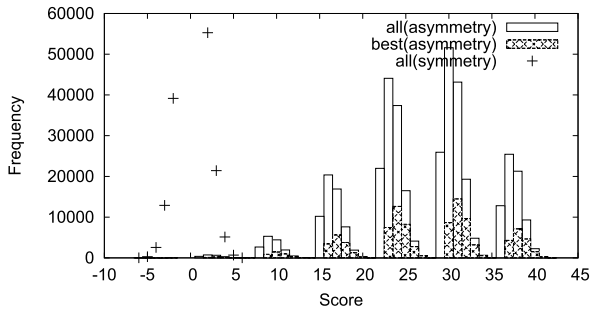


図 5 非対称ゲーム（一律）での最善手のみの，および全プレイアウト結果のスコアのヒストグラムならびに対称ゲームでの全プレイアウト結果のスコアのヒストグラム

Fig. 5 Histograms: the number of playouts for each score; the number of playouts played for all moves in an asymmetric configuration (uniform), that for the best move, and that for all moves in a symmetric configuration are measured.

3.1.2 スコアの分布

このゲームの性質（特に難しさ）について調査するため，プレイアウト結果（スコア）のヒストグラム（**図 5**）を作成した．この節ではそのことについて述べる．データは，長さ 6 幅 4 の盤の初期局面で，UCT で探索を行った際のものである．頻度を安定させるため，探索は 1,000 回行い，合計した．他の設定は 3.1.1 項での実験と同じである．ヒストグラムは，プレイアウト結果を根節点で最善手を選択した場合のみ抜き出したもの（best）と全プレイアウト結

G	x	x	x	x	x	x	x	S
0	1	1	1	x	1	1	1	0
0	1	1	1	x	1	1	1	0
0	1	1	1	x	11	11	11	0
S	x	x	x	x	x	x	x	G

図 6 非対称ゲーム（最終手）の例

Fig. 6 An example game with an asymmetric configuration (last move).

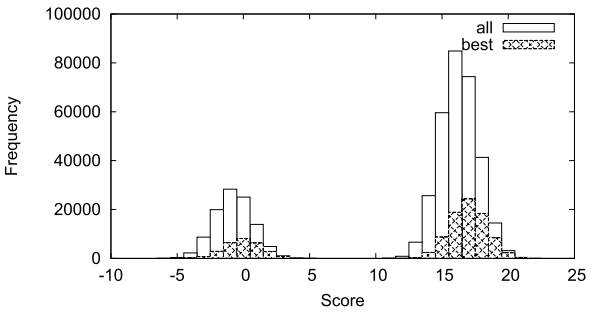


図 7 非対称ゲーム（最終手）での最善手のみの，および全プレイアウト結果のスコアのヒストグラム

Fig. 7 Histograms: the number of playouts for each score; the number of playouts played for all moves in an asymmetric configuration (last move) and that for the best move are measured.

果を含むもの（all）で作成した．また，比較のため，対称ゲームでの全プレイアウト結果を含む（all）ヒストグラムも作成した．

結果は図 5 のように多峰になる．スコアの対応を見ると分かるように，この山はそれぞれ，後手が最善手を逃した数に対応している．よって，山の数は盤の長さを増やすごとに多くなることが分かる．また，図 5 を見るとスコアが 0 以下になる頻度は，非常に低いことも分かる．つまり，このゲームではプレイアウト結果がほぼ勝ちで，各手の評価に差がほとんど付かない．この点が UCT で最善手を選ぶ難しさと考えられる．このゲームで先手の勝ち以外の結果が出るためには，後手が根からプレイアウトの終局までペナルティ 0 の手を毎回選ぶ必要があり，合法手を等確率で選ぶとするとなかなか起こらない．

3.2 非対称ゲーム（最終手）

非対称なペナルティを持つ別のゲームとして，後手の盤の一番最後の行に 1 つの升を除いて，非常に大きなペナルティを与え，それをふむかどうかでスコアが大きく変わるという設定を考える．具体的には 2·（盤の長さ）+5 とした．

ここでは省略するが，3.1.1 項と同様に勝率を計測した所，このゲームも UCT プレイヤの勝率は低下する（図 4 と後で紹介する実験の図 12 の Plain の差を参照）．また，3.1.2 項と同様に，スコアのヒストグラムを作成したところ**図 7** のようになり，最終手でのペナルティの有無によってプレイアウト結果のスコアは大きく変化し，ヒストグラムが二分されていることが分かる．スコアとの対応を見ると分かるとおり，この 2 つの山はそれぞれ，後手が最終手で最善手を打ったか否かに対応している．

このようなゲームでもスコア補正が有効な対策となりうる．なぜなら，補正なしの場合，最終手が最善手の場合に限り，プレイアウト結果が勝ちや負けに分かれる．つまり，最終手が最善手でない場合，プレイアウト結果は勝ちにしかなり得ず，（幅 −1）手分の情報を捨てているといえるからである．補正を行い，最終手が最善手でない場合で，プレイアウト結果が勝ちや負けに分かれるようにすれば，1 手分の情報を捨てるだけですむ．

4. 閾値補正方法の検討と提案

この章では新たな補正方法を 2 種類提案する．1 つは，プレイヤには入手できない情報を利用した方法である．これを比較のための目標値として用いる．もう 1 つは，プレイヤに入手可能な情報だけを利用した，現実的な利用を想定した手法である．

4.1 UCB1 での最適補正量の検討

この節では UCT でなく，1 手読みの UCB1 を用いた場合の補正量について考察する．最善手 $*$ の期待値 μ_* が他の手の期待値より大きい場合，最善手を見つけることが簡単であると期待される．理論的にも式 (2) のように，最善手に計算資源を投入する強い保証を持つ．この点をふまえて，$\min_{i \in A \setminus \{*\}} \Delta_i \equiv \Delta$ を最大化する補正を提案する．Δ を最大化する補正量は，プレイアウトを開始する節点ごとのスコアの確率分布を用いて求められる．つまり，本研究で扱うゲームであればプレイヤに分からない知識（手のペナルティの大きさはいくつか，またそのペナルティを持つ手は何個あるか）を使って求められる．

実際に非対称ゲーム（一律）で盤の長さ 6，幅 4 の初手の探索時に，各補正量に対応する Δ の値を 0.01 間隔でプロットした結果を**図 8** に示す．この例では多峰になること，Δ が 0 になる補正量があること，そして Δ は補正量 a に対して $30 < a < 31$ で最大になることが見て取れる．

このような関数の Δ 最大となる補正量 a を求めるためには，ペナルティと盤の長さに比例する個数の候補を調べれば十分であることを以下で示す．具体的には，次善手のペナルティを p とし，最善手の節点からプレイアウトした場合のスコアの最小値を MinScore，最大値を MaxScore とすると，MaxScore $-p-$ MinScore $+2$ 個の非整数値について Δ の値を調べれば十分である．なお，MinScore $<$ MaxScore $- p$ かつ $p > 0$ であること，また，ペナルティとスコアは整数値しか取らないことを仮定する．

本研究で扱っているゲームではプレイアウト中にペナルティを課される確率はそれ以前にどの升をふんだかとは独

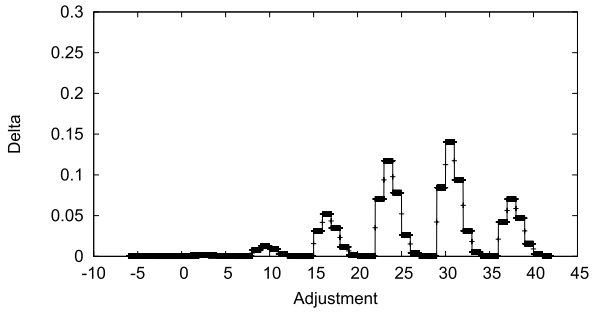


図 8 非対称ゲーム（一律）で，盤の長さ 6 幅 4 とした場合の各補正量での最善手と次善手の期待値の差（Δ）の値

Fig. 8 Difference in the expected rewards for the best move and that for second best move (Δ) in various values for adjustment (board length = 6, width = 4, asymmetric, uniform).

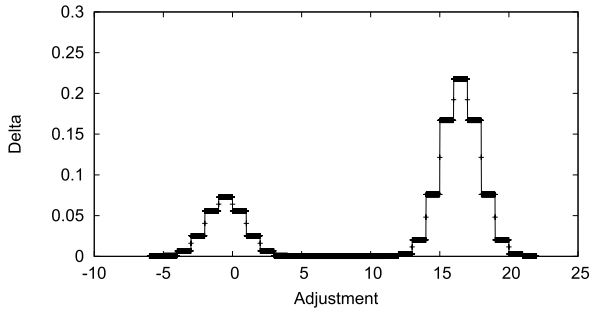


図 9 非対称ゲーム（最終手）で，盤の長さ 6 幅 4 とした場合の各補正量での最善手と次善手の期待値の差（Δ）の値

Fig. 9 Difference in the expected rewards for the best move and that for second best move (Δ) in various values for adjustment (board length = 6, width = 4, asymmetric, last move).

立という性質がある．そのため，最善手の節点からプレイアウトを始めた場合のスコア s となる確率を $\mathrm{P}[s]$ とし，次善手のペナルティを p とすると，次善手の節点からプレイアウトした場合のスコア s となる確率は $\mathrm{P}[s+p]$ となる．

関数 $\mathrm{P}(x)$ を x が整数のときに $\mathrm{P}[x]$，非整数のときに 0 をとるものと定める．補正量 a での Δ の値を $\Delta(a)$ と表記すると $\Delta(a)=\mu_*-\mu=(\sum_{s>a}\mathrm{P}[s]+0.5\mathrm{P}(a))-(\sum_{s>a}\mathrm{P}[s+p]+0.5\mathrm{P}(a+p))$ と表せる．等式 $\sum_{a<s\le a+p}\mathrm{P}[s]=\sum_{a<s<a+p}\mathrm{P}[s]+\mathrm{P}(a+p)$（$\sum$ の添字に注意）を用いて，式変形をすると，

$$\Delta(a)=\sum_{a<s<a+p}\mathrm{P}[s]+0.5\cdot\mathrm{P}(a)+0.5\cdot\mathrm{P}(a+p) \quad (3)$$

となる．

また，スコアは整数なので，非整数値での微小な補正では，利得が変化することはない．つまり，n を整数とし，任意の $a,b\in(n,n+1)$ に対し $\Delta(a)=\Delta(b)$ が成り立つ．また，$0<\delta<1$ とすると，$\Delta(n+\delta)$ と $\Delta(n-\delta)$ が最大値であるときのみ $\Delta(n)$ が最大値であることも成り立つ．つまり，補正量は隣り合う整数の間で 1 つだけ調べれば十分であることが分かる．以下，非整数値 a のみについて考える．すると，引き分けを考えなくてすむことから

$$\Delta(a+1)=\Delta(a)+\mathrm{P}[\lceil a+p\rceil]-\mathrm{P}[\lceil a\rceil] \quad (4)$$

となる．式 (4) は，$a<\mathrm{MinScore}-1$ のとき，第 3 項が 0 であり，全体は広義の単調増加となり，同様に，$\mathrm{MaxScore}-p<a$ のとき，第 2 項が 0 となり，広義の単調減少となる．加えて，式 (3) と $\mathrm{P}[s]=0$（$s<\mathrm{MinScore}$ または $\mathrm{MaxScore}<s$）より $a<\mathrm{MinScore}-p$ または $\mathrm{MaxScore}<a$ のとき，$\Delta(a)=0$ である．したがって $\Delta(a)$ が最大となりえる範囲は $\mathrm{MinScore}-1<a<\mathrm{MaxScore}-p+1$ である．

以下簡単のため非整数値補正量 a を $\lceil a\rceil-0.5$ で代表させることにすると $a'=\mathrm{MinScore}-0.5$，$n=\mathrm{MaxScore}-\mathrm{MinScore}-p+1$ である非整数 a' と整数 n について，

Algorithm 3 MAX FREQUENCY

Adjustment $\leftarrow \arg\max_{s\in\mathrm{Score}}\mathrm{Histogram}[s]$

$\Delta(a'),\Delta(a'+1),\cdots,\Delta(a'+n)$ のいずれかが最大値をとる．このようにして，本稿で扱うゲームでは任意の局面での最適補正量を求めることができる．

非対称ゲーム（一律）では，図 8 から，盤の長さ 6 のとき，補正量 30.5 で Δ 最大となることが分かる．また，非対称ゲーム（最終手）でも，同様にプロットした（**図 9**）．補正量 $4<a<12$ で Δ はほぼ 0 であることや，$16<a<17$ で Δ は最大になることが見て取れる．

4.2 最大頻度法の提案

このような状況で有効と思われる手法の 1 つとして新たな手法を提案する．提案手法（Algorithm 3）は，プレイアウト結果のスコアのヒストグラムを作成し，ヒストグラムの最大値のスコアを補正値とする調整法である．Score は探索局面から取りうるスコアの集合とする．Histogram は探索中の全プレイアウト結果の補正前のスコアの頻度を格納する配列とする．なお，刻み幅は 1 としている．Adjustment は dynamic komi のコミに相当する補正量であり，囲碁に限定していないので，Adjustment とした．

この手法の目的は，大きな Δ を持つ補正量を選ぶことである．直感的には，図 5 および図 7 に描いた頻度のヒストグラムと，図 8 および図 9 に描いた Δ が同じ形で変化していれば，最大頻度の補正量が最適に近いと期待できる．両者がどの程度一致するかは条件に依存するが，Δ の定義である式 (3) から明らかなように，少なくともスコア s の頻度が 0 でない限り $\Delta(s)$ は 0 にはならない．すなわち提案する最大頻度法は，選ばれた補正量で Δ が 0 になることはないという保証を持つ．一方，dynamic komi を用いた場合はその保証はない．dynamic komi では期待値付近を補正量として選ぶため，スコアのヒストグラムが多峰の場合に補正量は谷に設定されうる．そのようにして $\Delta=0$

となった場合は，1 手読みの UCB1 で探索した場合に，プレイアウト数をいくら増やしても最善手とその他を区別できないことを意味する．図 7 では正に期待値が谷に来ているので，dynamic komi が有効ではないと予想される．

最大頻度法のもう 1 つの利点として，図 5 のようにスコアのヒストグラムが複数の山に分かれている場合に，最善手が識別しやすいことがあげられる．図 5 ではそれぞれの山の中で最善手とそれ以外の手が少しずれて分布している．一番頻度が多い山に境界を移動することにより，それらを効率的に見分けられると期待できる．

5. 実験

この章では，提案手法と既存手法を用いた場合の勝率を 2 種類の非対称ゲームで対戦実験により比べる．提案手法の比較対称として，それぞれ，4.2 節で定義した最適補正量，dynamic komi の Score Situational（以下 Score と表記）と，Value Situational（Value）に基づく補正とさらに補正しない通常の UCT の計 5 つの手法を用いて図 4 と同様に盤の長さを変えた場合の勝率の変化を測定する．特に非対称ゲーム（一律）での盤の長さは，3.1.2 項で述べたように，図 5 のヒストグラムの山の数に対応する．したがって，この実験は，各手法の山の数に対する性能を調査することにも関係している．さらに，勝率を Δ の大小の観点から論じる．なお，dynamic komi のパラメータは Score で $s = 0$，$c = 5$，Value で red $= 0.3$，green $= 0.85$ とした．これは，非対称ゲーム（一律）を使った予備実験で勝率が最大となったときのパラメータである．その他の設定は 3.1.1 項と同じとした．

5.1 非対称ゲーム（一律）

まず，非対称ゲーム（一律）の結果を示す．**図 10** から読み取れるように，勝率の高い順に最適補正（Theoretical），提案手法（MaxFrequency），Value，Score，通常の UCT（Plain）となった．提案手法は最適補正には及ばなかったが，既存の手法を大きく上回る勝率となった．

次に，この結果について Δ の大きさの観点から，ある 1 試合での長さ 6 の盤の初手の探索時のデータを用いて分析する．長さ 6 の場合を取り上げた理由は，勝率が一番低いものでも勝率の下限 0 より高く，また一番高いものでも勝率の上限 0.5 よりは低いためである．プレイアウト回数を増やした場合の各アルゴリズムでの補正量の変化を**図 11** に示す．図 8 に示した補正量と Δ の対応は，補正量 0 の場合 Δ の値は $1.2 \cdot 10^{-7}$，最適補正量 30.5 の場合 0.14 となっている．各アルゴリズムの補正量は，dynamic komi の Score の場合は，補正量が 10 から 40 前後を揺れ動いて，平均するとあまり高くない．ただし，補正量 0 のときよりは改善している．その次に dynamic komi の Value の場合は，補正量が 1 ずつ増え 27 で安定している．補正量 27 で

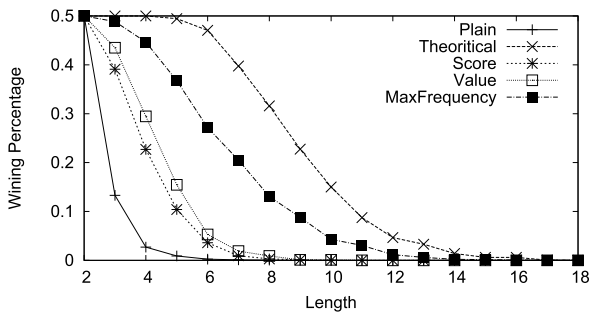


図 10 非対称ゲーム（一律）で盤を長くした場合の各アルゴリズムの勝率の変化

Fig. 10 Relationship between winning percentage of the algorithms and board length in an asymmetric configuration (uniform).

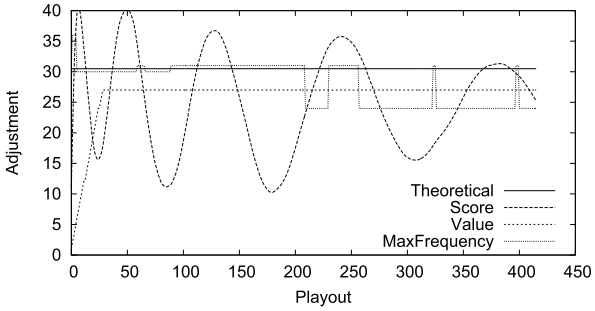


図 11 非対称ゲーム（一律）で各アルゴリズムで毎プレイアウトでの補正量の変化

Fig. 11 Adjustment computed by the algorithms (asymmetric, uniform).

は Δ は 0.0023 であり，Score での平均した値よりも低い値である．最後に提案手法（MaxFrequency）の場合は，補正量は最適補正量 ±0.5 を往復後に最後は 24 で安定している．補正量が最適補正量 ±0.5 のときは Δ は 0.11，0.12 であり，24 のときも 0.098 で他の手法と比べて高い値となった．このように Δ を高くできたために，他と比べて高い勝率を達成できたのだと予想される．

続いて，10 試合での補正量の変動を調べ，全体的な傾向を述べる．結果は 1 試合でのものと同じ傾向であり，まず Score では，各試合ごとに異なる周期の波が見られた．また，Value では，プレイアウト数が 20 から 30 程まで単調に増加し，その後一定値となった．最後に提案手法では，30 と 31，23 と 24 を往復するが，その移り変わりのタイミングは各試合で異なっていた．さらに，1,000 試合中で，提案手法での最終的な補正量の頻度分布をとると，23 と 24 を合わせて 638 回，30 と 31 を合わせて 362 回となって Δ 最大の山の頂点（30.5）付近よりも 2 番目の山の頂点（23.5）付近の頻度の方が高かった．この理由は不明である．

5.2 非対称ゲーム（最終手）

続いて，非対称ゲーム（最終手）での実験結果を示す．図 10 の実験と同様に長さを変えた場合の勝率を測定した．

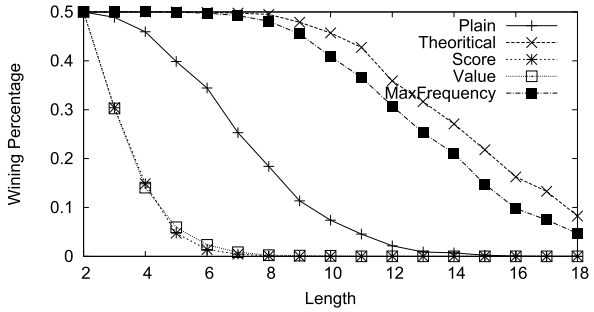


図 12 非対称ゲーム（最終手）で盤を長くした場合の各アルゴリズムの勝率の変化

Fig. 12 Relationship between winning percentage of the algorithms and board length in an asymmetric configuration (last move).

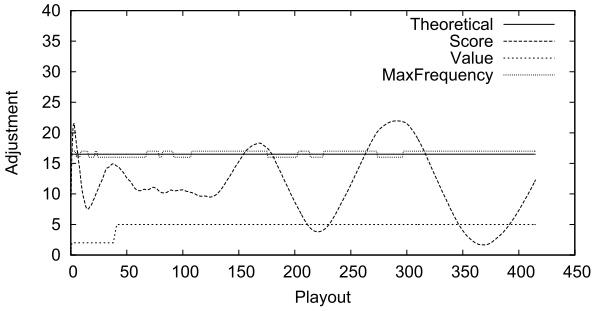


図 13 非対称ゲーム（最終手）で各アルゴリズムで毎プレイアウトでの補正量の変化

Fig. 13 Adjustment computed by the algorithms (asymmetric, last move).

その結果，**図 12** に示したように勝率の高い順から最適補正（Theoretical），提案手法（MaxFrequency），通常の UCT（Plain），dynamic komi となった．dynamic komi の 2 種はほぼ同じ勝率だった．重要な点は通常の UCT よりも dynamic komi は勝率が低くなったことと，一方で提案手法はこの仮想ゲームでも高い勝率を達成できたことである．

前節と同様に長さ 6 の盤で，ある 1 試合での補正量の変化を**図 13** に示す．図 9 と見比べると，提案手法（MaxFrequency）の補正量は最適補正量 ±0.5 で安定している．補正量 16，16.5（最適補正量），17 での Δ の値は，それぞれ 0.19，0.21，0.19 であり，提案手法での Δ の値は最適補正量での値に近いことが分かる．一方で，dynamic komi の 2 種類は，ともに Δ が 0 の所に境界を移動させるような補正量であることが多い．Score の場合，Δ がほぼ 0 となる範囲 4 から 12 に補正量が設定されて，2/3 近くのプレイアウトが行われている．また，Value の場合補正量 5 で安定しているが，このとき Δ は $2.9 \cdot 10^{-5}$ でほぼ 0 である．このゲームでは，dynamic komi を使うと Δ の値が補正量が 0 時の値 0.064 よりも小さい値となっていて，dynamic komi による補正が悪影響を及ぼしているといえる．

今度は，10 試合での補正量の変動を調査した．このゲームでも同様に 1 試合での結果と同じ傾向が見られた．具体的には，Score では，各試合ごとに異なる周期の波が見られた．また，Value では，プレイアウト数が 50 になるまでに安定して一定値となったが，その値は最小値 1 から最大値 17 まで各試合でばらつきが見られた．さらに，提案手法では，16 と 17 を往復するが，その移り変わりのタイミングは各試合で異なっていた．また，1,000 試合中の提案手法での最終的な補正量の頻度分布を調べると 15 が 5 回，16 が 635 回，17 が 359 回，18 が 1 回となって，補正量が理論値（16.5）付近になることが多かった．

6. おわりに

ゲーム木探索の研究は，チェスや囲碁などの実際のゲームでの強さを目指すとともに，要素技術を仮想ゲームなどを用いて分析し理解しながら進歩してきた．本研究では UCT が性能を発揮しにくい探索問題の性質の 1 つとして，Min-Max 値は 0 であるが，手番の勝ちやすさが異なるという状況に着目した．そのような状況が，既存の仮想ゲームに小さな変更を加えるだけで作成できることを示し，非対称ゲーム（一律）と非対称ゲーム（最終手）として実際に作成した．

もとの対称ゲームと作成した非対称ゲームでは，様々な探索空間の広さにおいて，UCT で最善手を選ぶ確率にはっきりとした差が現れる．たとえば長さ 5 から 10 では，対称ゲームではほぼ確実に最善手が選ばれたが，非対称ゲームではほぼ選ばれなかった．このような非対称ゲームでは，探索した際のプレイアウト結果の頻度が勝ちやすい側の勝ちに偏るが，一方 UCB1 を前提とすると最善手と次善手の利得の期待値の差（Δ）が大きいことが望ましい．そのため，プレイアウト結果を勝敗に変換する閾値を，Δ が大きくなるように調整することが有効であると考えられる．ゲームの内情を知っている神の視点ではそのような最適な閾値が求められ，プレイヤに入手可能な情報のみ用いる場合は頻度最大のスコアに閾値を合わせることが有力である．それらの調整は以下の実験結果により効果的であると示された．非対称ゲーム（一律）において，様々な探索空間の広さでの実験では，神の視点で求めた閾値が最も勝率が良く，続いて頻度最大の閾値，dynamic komi，何もしない場合の順となる．一例として，長さ 6 の場合の勝率はそれぞれ約 0.47，0.27，0.053，0.0025 であった（最善引き分けなので勝率は 0.5 が最大）．非対称ゲーム（最終手）についてはゲームの性質から，dynamic komi による調整が不利に働き，何もしない場合との勝率が逆転する．一例として長さ 6 の場合の勝率は，補正なしの約 0.034 に対して，dynamic komi では最大 0.023 である．また，これらの実験で，勝率と Δ の大小が実際に深い関係にあることが分かった．

囲碁を含めて人間にとって興味深いゲームは，複雑で

様々な性質を持つと予想される．ゲームの持ちうる性質を 1 つずつ仮想ゲームに反映させて分析することにより，MCTS が有効に動作する条件は何かを明らかにするという大きな研究目標に近づけると著者らは期待している．

参考文献

[1] Auer, P., Cesa-Bianchi, N. and Fischer, P.: Finite-time Analysis of the Multiarmed Bandit Problem, *Machine Learning*, Vol.47, No.2-3, pp.235–256 (online), DOI: 10.1023/A:1013689704352 (2002).

[2] Baudiš, P.: Balancing MCTS by Dynamically Adjusting the Komi Value, *ICGA Journal-International Computer Games Association*, Vol.34, No.3, p.131 (2011).

[3] Enzenberger, M., Muller, M., Arneson, B. and Segal, R.: Fuego—An open-source framework for board games and go engine based on monte carlo tree search, *IEEE Trans. Computational Intelligence and AI in Games*, Vol.2, No.4, pp.259–270 (2010).

[4] Finnsson, H. and Björnsson, Y.: Game-Tree Properties and MCTS Performance, *Proc. 2nd International General Game Playing Workshop* (*GIGA2011*), pp.23–30 (2011).

[5] Kocsis, L. and Szepesvári, C.: Bandit based monte-carlo planning, *Machine Learning: ECML 2006*, pp.282–293, Springer (2006).

[6] Lai, T.L. and Robbins, H.: Asymptotically Efficient Adaptive Allocation Rules, *Advances in Applied Mathematics*, Vol.6, No.1, pp.4–22 (1985).

[7] Lee, C.-S., Wang, M.-H., Chaslot, G., Hoock, J.-B., Rimmel, A., Teytaud, F., Tsai, S.-R., Hsu, S.-C. and Hong, T.-P.: The computational intelligence of MoGo revealed in Taiwan's computer Go tournaments, *IEEE Trans. Computational Intelligence and AI in Games*, Vol.1, No.1, pp.73–89 (2009).

[8] Ramanujan, R., Sabharwal, A. and Selman, B.: On the Behavior of UCT in Synthetic Search Spaces, *Proc. 21st Int. Conf. Automat. Plan. Sched., Freiburg, Germany* (2011).

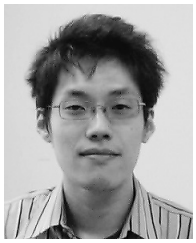
[9] Sato, Y., Takahashi, D. and Grimbergen, R.: A shogi program based on Monte-Carlo tree search, *Icga Journal*, Vol.33, No.2, pp.80–92 (2010).

[10] Winands, M.H., Bjornsson, Y. and Saito, J.-T.: Monte carlo tree search in lines of action, *IEEE Trans. Computational Intelligence and AI in Games*, Vol.2, No.4, pp.239–250 (2010).

[11] Yoshizoe, K., Kishimoto, A., Kaneko, T., Yoshimoto, H. and Ishikawa, Y.: Scalable distributed monte-carlo tree search, *4th Annual Symposium on Combinatorial Search* (2011).

今川 孝久 （学生会員）

2013 年東京大学教養学部広域科学科卒業．2013 年より同大学大学院修士課程在籍．



金子 知適 （正会員）

1997 年東京大学教養学部卒業．2002 年東京大学院総合文化研究科博士課程修了．博士（学術）．2002 年東京大学院総合文化研究科助手．2007 年助教を経て 2012 年より准教授．