コンピュータ囲碁における対局の進行度判定を用いたモンテカルロ木探索

田中 一樹[1,a)]　藤田　玄[1]

概要：近年，モンテカルロ木探索，主に UCT に局面評価関数を組み合わせた手法が広く知られている．しかし，いまだコンピュータ囲碁の局面評価関数の多くは，演算量に課題がある傾向にある．この問題に対し，前回の研究報告では，UCT の中で着手決定に用いられるプレイアウトの終局盤面に注目し，終局状態の局面の統計情報を用い，少ない演算コストで盤面評価を行う手法を提案したが，問題点として，特に終盤では，ダメなどの着手する意味がない箇所が優先されてしまうことがあげられた．そこで本研究では，モンテカルロオーナーに基づく局面評価関数に，対局の進行度判定を組み合わせることで木探索の効率化を図る．既存手法である Fuego に提案手法を適用して性能評価を行った結果，UCT 単独の Fuego に対して 6 割の勝率を示した．

Monte Carlo Tree Search Using Degree of Progress in Computer Go

Kazuki Tanaka[1,a)]　Gen Fujita[1]

Abstract: In recent years, a method to combine Monte Carlo tree search such as UCT and evaluation function have been widely known. On the other hand, most of the evaluation functions of the computer go still tends to be heavy computational cost. In the previous study, we have proposed a efficient evaluation function which focus on results of play out based on UCT.　 However, some pointless moves such as “ dame ” have been occurred in the middle and later. To address the issue, in this paper, degree of progress based on statistical information of play out has been proposed. Adaptive method using degree of progress seems to make efficient search possible.

1. はじめに

近年の研究で他のゲームでは，モンテカルロ木探索と局面評価関数を組み合わせた探索が有効であることが示されている [1][2]．2000 年代以降のコンピュータ囲碁では，局面評価関数に頼らない，モンテカルロ木探索 [3]，主に探索に期待値 UCB（Upper Confidence Bound）[4] を用いた UCT(UCB applied to Trees)[5] が主流になっている．

しかしコンピュータ囲碁では，途中局面の有効な局面評価を行う際に多くの時間を必要とするため [6]．モンテカルロ木探索と局面評価関数を組み合わせた研究は少ない．

局面評価関数を使わずにモンテカルロ木探索を効率化する手法も検討されている．例えば，UCT を採用している囲碁思考エンジン Fuego[7] では，探索する候補をパターンマッチングにより評価することで選出し探索の効率化を図っている．しかし少ない演算量で，精度の高い局面評価関数を実現できれば，さらなる探索の効率化と棋力の向上が期待できる．

そこで，前回の研究報告 [8] では，コンピュータ囲碁におけるモンテカルロ木探索に局面評価関数を組み合わせる手法を提案した，しかし [8] においては，中盤以降にダメなどの着手する意味がない箇所が優先されてしまうことが課題としてあげられた．モンテカルロオーナーの値を直接利用していることが問題点の一因として考えられる．

そこで本研究では，対局の進行度などプレイアウトの統計情報に基づいた値 PTS(Playout Territory Statistics) を

1　大阪電気通信大学

a)　mi14a005 @ oecu.jp

用いた効率的なモンテカルロ木探索手法を提案する．具体的にはモンテカルロオーナー，プレイアウトの結果の差，対局の進行度が PTS に含まれる．[8] の問題点を解決するため，局面評価関数の使用を進行度により制限する木探索を提案する．

さらに探索候補手選出を行う評価関数に既存のモンテカルロオーナーではなく，勝利したプレイアウトの統計情報を用いることで木探索の改善を図る．

2. UCT

UCT は，2006 年 9 月に Kocsis と Szepesvari によって発表された，モンテカルロ木探索の代表的なアルゴリズムである．UCT では，UCB 値を利用し，以下の 1 から 5 を時間内に繰り返すことで図 1 に示すような木を作成する [6]．

(1) UCB1 値による局面選択
根節点から，UCB 値の高い子接点を選択しながら末端節点まで枝をたどる．

(2) 局面の展開
末端の節点のプレイアウト回数が閾値を越えていれば，末端節点の局面から合法手を選出し，末端の子節点として節点を展開する．

(3) プレイアウト
(2) で展開した節点の局面からプレイアウトを行う．

(4) UCB1 値の更新
プレイアウトの結果によって得られた評価値を根節点までたどって節点の値を更新する．

(5) 探索続行の判断制限時間や総プレイアウト数などの制限に達していなければ終了する．そうでなければ (1) に戻る．

囲碁に置き換えた場合，最後的に根節点の子節点から評価値が一番高い手を選択し着手を決定する．

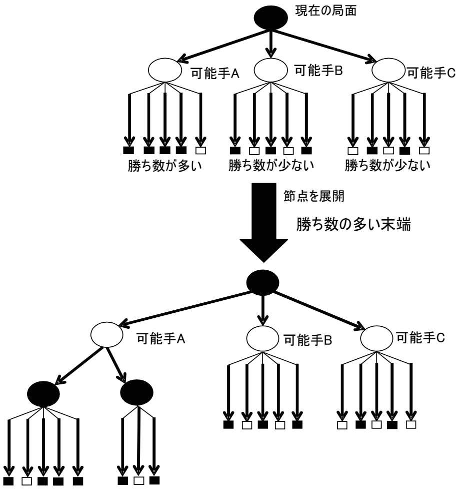

図 1 UCT のノードの展開

2.1 Fuego の探索候補手選出

コンピュータ囲碁は，持ち時間の制約によりすべての着手可能手を探索評価することができない．そのため Fuego では，前節の手順 (1) の段階で図 2 のように，盤面上の合法手に対し盤面の左上から列挙し，順番にパターンマッチングを用いて，UCB 値を求め先頭から閾値を超えたか否かで候補手選出を行い，プレイアウトをよりよい候補に集中させている．このとき UCB1 値が等しい場合は，候補手列の先頭にある合法手が優先的に探索される．しかし Fuego の合法手列挙は盤面の左上から順番に列挙されていくので，有効な手が探索されないことがある．

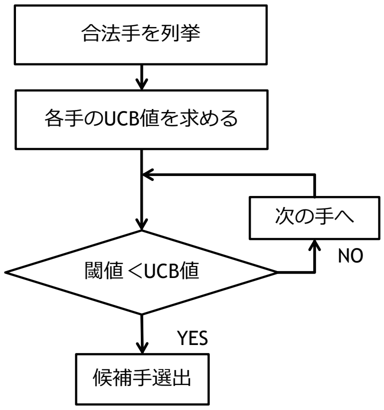

図 2 Fuego の探索候補手選出

3. モンテカルロオーナー

モンテカルロオーナー [9] はプレイアウトの最終局面の石の配置の統計を取ることで，盤面上の石の存在する確率を示す値である．モンテカルロオーナーにより，現局面まだ着手されていない未確定な座標点が，どちらの色に属しているかを計ることができる．またプレイアウトの情報を再利用することでモンテカルロオーナーを求めることによる計算量の増加はわずかである．

評価値が取る範囲は+1 から-1 の範囲であり，評価値が-1 に近いほど白に属した地，+1 に近いほど黒に属した地である．0 に近いほどどちらにも属していない．座標 p のモンテカルロオーナーの値 $MO(p)$ は式 (1), (2) で求められる．

$$P(n_i, p) = \begin{cases} 1(state(p) = \text{黒のとき}) \\ 0(state(p) = \text{空点のとき}) \\ -1(state(p) = \text{白のとき}) \end{cases} \quad (1)$$

$$MO(p) = \Sigma_{i=1}^{n} \frac{P(n_i, p)}{n} \quad (2)$$

n：プレイアウトの総数
n_i：i 番目のプレイアウト
$state(p)$:座標 p の状態
$P(n_i, p)$:i 番目のプレイアウト時の点 p の状態

4. 従来手法

[8] では，図 3 のように，Fuego の探索候補手選出に加え，評価関数を用いた探索候補手列のソートを行うことで探索の効率化を図っている．プレイアウトの統計情報に基づいたモンテカルロオーナーを評価関数に用いている．問題点として，この手法を適用した場合，中盤以降にダメなどの着手する意味がない箇所が優先されてしまうことがあげられる．

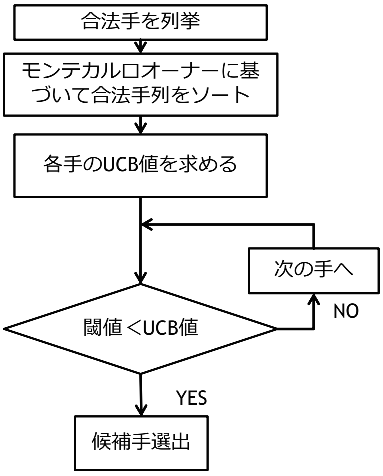

図 3 従来探索候補手選出

5. 提案手法

本研究では，Fuego の探索候補手選出の問題点に着目し，局面評価関数を用いた候補手選出を行うことで探索の効率化を図る．対局の進行度などプレイアウトの統計情報に基づいた値 PTS を用いた効率的なモンテカルロ木探索手法を提案する．具体的には，対局の進行度による候補手生成処理の切り替えを用いた候補手生成手法，プレイアウト差を用いた候補手生成手法である．

5.1 PTS(Playout Territory Statistics)

PTS は，プレイアウトの統計群であり PTS に基づき，局面を評価する．具体的には以下の統計情報が含まれる．

(1) モンテカルロオーナー　$MO(p)$　p：盤面の座標
(2) プレイアウト差 $WD(p)$
(3) 進行度 Pr

5.1.1 プレイアウト差

プレイアウト時に，結果の勝ち負け別に統計値 E_{win}，E_{lose} を求める．E_{win}，E_{lose} の値はそれぞれ以下の式により求める．N_w は結果が勝ちのプレイアウトの集合，N_l は結果が負けのプレイアウトの集合を意味している．S は自分の手番が黒なら 1，白なら −1 が与えられる．また n_w と n_l の和はプレイアウトの総数になる．

$$E_{win}(p) = \Sigma_{i \in N_w} \frac{S \times P(i,p)}{n_w} \tag{3}$$

$$E_{lose}(p) = \Sigma_{i \in N_l} \frac{S \times P(i,p)}{n_l} \tag{4}$$

$$WD(p) = E_{win} - E_{lose} \tag{5}$$

5.1.2 モンテカルロオーナーを使った対局の進行度判定

各座標点のモンテカルロオーナーの絶対値を求め，平均を取ることで現局面が序盤, 中盤, 終盤のどれであるかの指標として用いることができる．図 4，は対局の進行によって $|MO(p)|$ の度数分布が遷移していく様を示している．この図から Pr の値が終盤になるにつれて 1 に近づいていくことがわかる．

$$Pr = \Sigma_{i=1}^{n} \frac{|MO(p)|}{n} \tag{6}$$

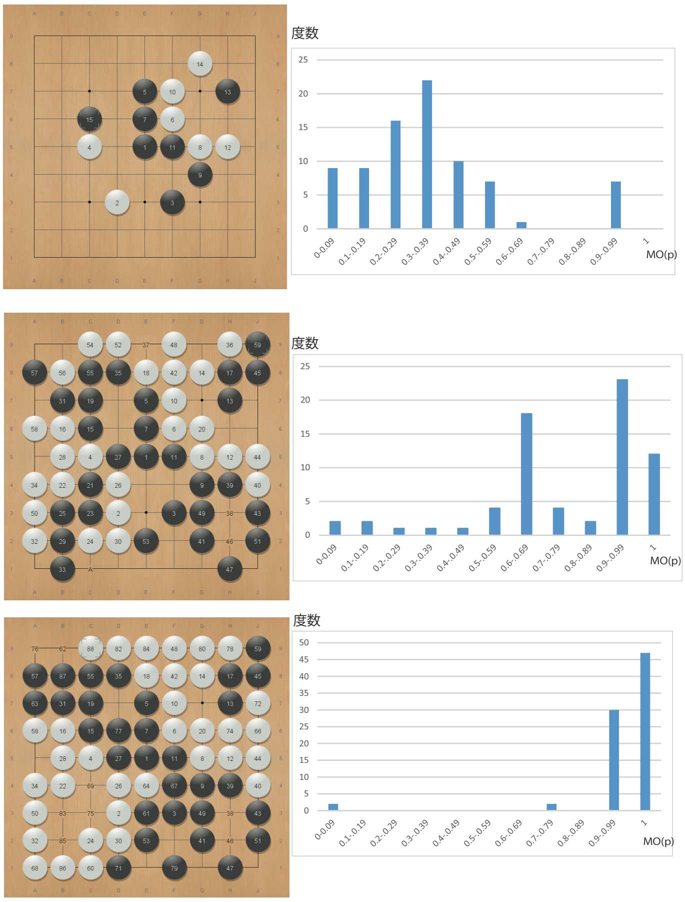

図 4 対局の進行によって遷移する $MO(p)$ 度数分布

5.2 進行度による候補手生成処理の切り替え

[8] では，問題点として，終盤以降にこの手法を適用した場合，ダメなどの着手する意味がない箇所が優先されてしまうことがあげられたが，本研究では，Pr を用いることで局面を判定し終盤に局面評価関数による優先処理を止めることで提案手法の問題点の解決を図っている．式を満た

すとき [8] の手法 2 を用い，満たさないとき既存の Fuego の候補手生成を用いる．

$$0 < Pr < Th_1 \quad (7)$$

5.2.1 E_{win} 値に閾値を設けた候補手列生成

閾値 Th により以下の式 (8) を満たす探索候補点を優先する手法．すでに，自分の手番が有利であることが確定している点ではなく，まだ勢力争いにある点を優先して着手することを意図している．

$$0 < E_{win} < Th_2 \quad (8)$$

6. 性能評価

進行度による候補手生成処理の切り替えの有用性を示すため対戦による性能評価を行う．さらに，プレイアウト差を用いた候補手生成に関する性能評価も行う．

表 1 に環境を示す．

表 1 評価条件

使用プログラム	Fuego Version1.0
プロセッサ	Intel(R)i7-2600 CPU3.4GHz
メモリ	16GB
OS	Windows7 64bit
一手あたりの持ち時間	1 秒

6.1 進行度判定評価

進行度判定の性能評価を行ったいくつかの終局までの着手数が違う盤面を用意しそれぞれの進行度を計算した．図 5 に進行度判定結果を示す．

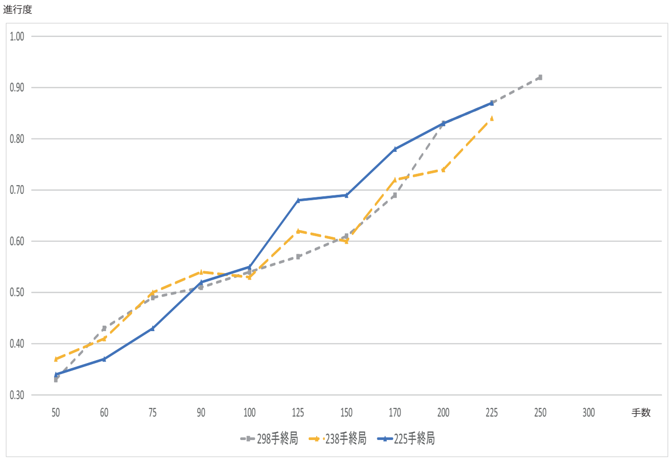

図 5 進行度判定結果

図 5 から進行度判定がほぼ単調に増加しておりある程度の有効に機能していることが分かる．また進行度が大きく変化している 225 手終局の 100 手目と 125 手目，進行度が後退している 238 手終局の 125 手目と 150 手目の実際の盤面を図 6 に示す．238 手終局の場合は，盤面左上の死活がはっきりとしない部分を検出したことで進行度が後退していると考えられる．

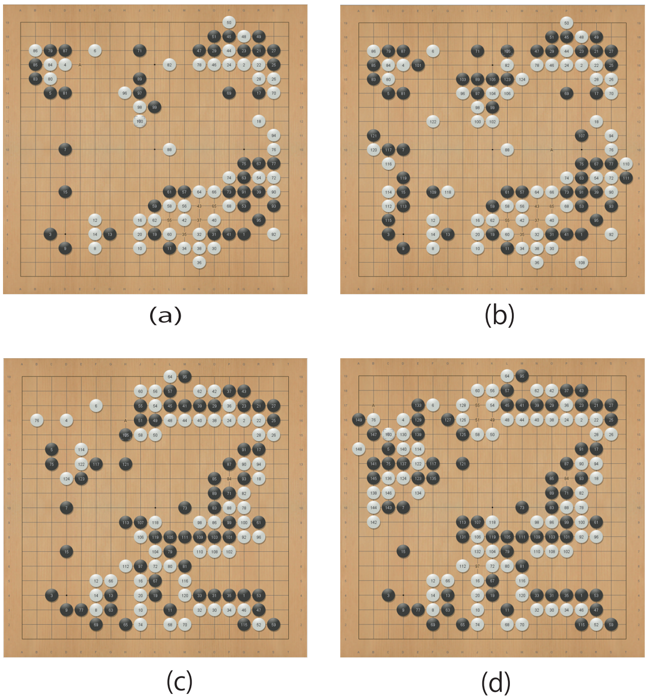

図 6 (a) 225 手終局の 100 手目 (b) 225 手終局の 125 手目 (c) 238 手終局の 125 手目 (d) 238 手終局の 150 手目

表 2 に対戦結果を示す．

表 2 19 路盤持ち時間一秒評価結果

Th_1	勝率	勝ち数 (黒勝ち，白勝ち)	負け数
0.10	57 %	226(121,105)	174
0.15	59 %	235(118,117)	165
0.20	58 %	230(114,116)	170
0.25	52 %	206(96,110)	194
0.30	64 %	256(132,124)	144
0.35	52 %	206(96,110)	194
0.40	56 %	222(114,108)	178
0.45	56 %	225(113,112)	175
0.50	57 %	228(120,108)	172
0.55	57 %	228(121,107)	172
0.65	57 %	228(117,117)	166
0.70	59 %	234(98,118)	184
0.75	54 %	216(114,121)	165
0.80	59 %	235(115,132)	153
0.85	62 %	247(110,124)	116
0.90	59 %	234(109,111)	180
0.95	55 %	220(111,116)	173
1.00	57 %	230(121,109)	170

表 2 から，モンテカルロオーナーの値に閾値を設けた候補手列生成は序盤までに適用することが有効なのがわかる．

6.2 プレイアウト差を用いた候補手選出

プレイアウト差を用いた候補手選出手法と既存の Fuego との対局による評価実験を 9 路盤で行いプレイアウト差を用いた候補手選出の評価を行う．E_{win} 値に閾値を設けた候補手列生成手法の有効な閾値を探すため複数の閾値でそれぞれ 9 路盤で従来手法との対局による性能評価を行った．結果を表 3 に示す．

表 3 9 路盤閾値変更

Th_2	勝率	勝ち数 (黒勝ち , 白勝ち)	負け数
0.40	65 %	650(273,377)	350
0.60	67 %	670(296,374)	330
0.65	65 %	653(283,371)	347
0.70	66 %	675(299,376)	325
0.75	65 %	649(279,370)	351
0.80	65 %	648(279,369)	352

表 3 から，この手法は有効だと考えられる．

6.3 おわりに

本研究では，プレイアウトの統計情報から局面の進行度を求め，進行度に応じて処理を切り替えるモンテカルロ木探索を提案した．

今後の課題として，対局の進行度を算出する際にモンテカルロオーナーの平均を利用したが，ヒストグラムの値をより有効に活用することが考えられる．

参考文献

[1] 橋本剛，前原彰太，川島哲哉，小林康幸: 局面評価関数を使う新たな UCT 探索法の提案とオセロによる評価, 情報処理学会論文誌, Vol.54, pp.1930-1936 (2013).

[2] 松本渉, 小林康幸, UCT 探索における局面評価関数の使用方法と性能評価, The 18th Game Programming Workshop 2013

[3] Coulom, R.: Efficient selectivity and backup operators in montecarlo tree search, Proceedings of the 5th International Conference on Computers and Games, Turin, Italy (2006).

[4] Auer, P., Cesa-Bianchi, N. and Fischer, P.: Finite time analysis of the multi-armed banditProblem, machine learning, Vol. 47, pp. 235-256 (2002).

[5] Kocsis, L. and Szepesvari, C.: Bandied based montecarlo planning, Proceedings of the 15th European Conference on Machine Learning, pp.282-293 (2006).

[6] 美添 一樹 , 山下 宏 , コンピュータ囲碁 ―モンテカルロ法の理論と実践―, 共同出版社,(2012).

[7] Fuego, ”http://fuego.sourceforge.net/”

[8] 田中 一樹，藤田 玄，コンピュータ囲碁における PTS を用いたモンテカルロ木探索，研究報告ゲーム情報学（GI）,2014-GI-32(3),1-4 (2014-06-28)

[9] R. Coulom.: Computing Elo ratings of move patterns in the game of Go, In Computer Game Workshop,Amsterdam, The Netherlands, 2007.