3P-07

局面タブーリストを内包したモンテカルロ木探索手法

太田　雄大　　伊藤　雅
愛知工業大学大学院　経営情報科学研究科

1　はじめに

モンテカルロ木探索におけるプレイアウトの強化改善に関する研究は数多く行われている．プレイアウトとは，終局までランダムにシミュレーションを行うことである．しかし，プレイアウトに関する研究は効率化や精度向上を目的としたものが多い．

プレイアウトの多様化に着目し，タブーサーチで用いられるタブーリストをモンテカルロ木探索に内包する手法 [1] が提案されている．詰碁と 9 路盤囲碁では有効であるが，19 路盤囲碁では有効性を示せていない．

そこで，本稿では局面タブーリストを内包したモンテカルロ木探索手法を新たに提案する．数値実験でオープンソースプログラムと 19 路盤で対局させ，プレイアウトの多様化と勝率の向上を検証する．

2　モンテカルロ木探索

モンテカルロ木探索は，有望なノードに対してプレイアウトを行い，徐々に木を成長させていく手法である．有望なノードを選択する手法の一つに UCB1（Upper Confidence Bound）アルゴリズムがある．図 1 中のノード u における UCB_u の定義を式（1）に示す。ここで，$\overline{X_u}$ はノード u の勝率，n_u はノード u 以降のプレイアウト数，n は u の親ノードのプレイアウト総数である．

$$UCB_u = \overline{X_u} + C\sqrt{\frac{2\ln n}{n_u}} \tag{1}$$

木探索に UCB1 アルゴリズムを組み込んだ手法は UCT（UCB applied to Tree）アルゴリズム [2] と呼ばれ，モンテカルロ木探索の一つに分類される．UCT でのプレイアウトは，式（1）が常に最大である葉ノードで行われる．図 1 中のノード t のような中間ノードではプレイアウトは行わない．プレイアウトの回数が閾値を超えた場合，t のように展開し，u のような葉ノードを適当数生成し木を成長させる．プレイアウトの結果は，$u \to t \to r$ のように親ノードを辿って根ノードまで順次伝播していく．このように UCT は最良優先探索をしながら木を成長させている．

A Monte-Carlo tree search algorithm including tabu lists of game states
Takehiro Ohta　Masaru Itoh
Graduate School of Business Administration and Computer Science, Aichi Institute of Technology

3　提案手法

タブーリストに一度探索した手を追加する従来法 [1] は，19 路盤囲碁では局所的であるためか有効に機能しない．そこで，手よりも全域的な局面を管理する局面タブーリストの導入を提案する．

3.1　局面タブーリストの構造

局面の保持にはゾブリストハッシュ[3] を用いる．ゾブリストハッシュとは，現局面のハッシュ値と手の乱数値との排他的論理和（*XOR*）をとることで一手打った後の局面を表す手法である．碁盤 x 軸を記号 'I' を除く A~T で，y 軸を 1~19 で表せば，手は xy となる．例えば，図 1 中の $D4$ や $J7$ のように表す．このとき，図 1 におけるノード s のハッシュ値 $hash(s)$ はノード r のハッシュ値 $hash(r)$ と手 $D4$ の乱数値 $rand(D4)$ の排他的論理和 $hash(r) \oplus rand(D4)$ で表すことになる．

プレイアウト相手初手から第 M 手目までにタブーリストを導入したのが図 1 右側である．このときノード s が管理するタブーリストは M 個あり，キュー構造（queue structure）を持たせる．

タブーリストにはプレイアウトで第 m 手目（$1 \le m \le M$）に出現した局面のハッシュ値 $hash(s_m)$ が順次キューに追加される．タブーリストへの追加は，一手着手した直後に行う．もしその局面がタブーリストに追加済みならば，代わりに NIL を追加する．と同時に局面を別途生成する．タブーリストに追加された局面は，タブーリスト長の L 回は探索できず，$L+1$ 回目で再度探索可能となる．

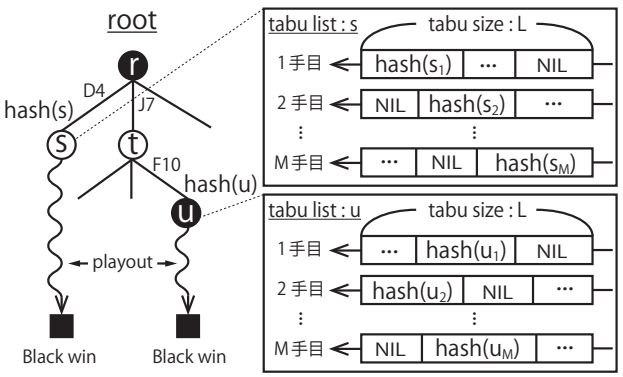

図 1: 局面タブーリストを内包したモンテカルロ木探索

3.2 勝敗に基づいたタブーリストへの追加

節 3.1 の手法では，プレイアウトで一手着手した直後に局面をタブーリストへ追加し，同時に更新も行った．それをプレイアウトの勝敗に基づいてタブーリストへ追加するか否かに変更する．全てのプレイアウトで一度探索した局面を追加するよりは，効率的にプレイアウトを多様化できると考えた．

プレイアウトで一手着手した直後にはタブーリストの更新のみ行う．このとき，リストへの追加は行わない．プレイアウトの勝敗が自分の手番で負けた場合，M 手目までの局面をタブーリストへ追加する．逆に自分の手番で勝った場合は，タブーリストへの追加は行わない．つまり，図 1 のノード s ではタブーリストへの追加を行うが，u では追加は行わない

4 数値実験

オープンソースの GNU Go 3.8*（思考ルーチン名: gnugo）に UCT と提案手法を組み込む．使用するプログラムは，gnugo（UCT）と文献 [1] の手法 gTabu18，節 3.1 の提案手法 gTabu18-hash，節 3.2 の提案手法 gTabu18-hash-win である．本実験では，タブーサイズ L を 18，タブーとする手数 M を 5，プレイアウト数を 8000 とした．また，GNU Go 3.8 の UCB1 値に関する式（1）相当の定数 C は $C = 1.5$ で実験した．

4.1 多様性の検証

提案手法が従来法 [1] と同程度にプレイアウトの多様性を確保できているかを検証した．初期局面に対して次の一手の探索を行い，プレイアウト初手から 5 手目までに探索した局面重複数で評価した．重複した上位 50 局面の結果を図 2 に示す．提案手法の局面重複数は gTabu18 の重複数とほとんど差がない．つまり，提案手法は gTabu18 と同程度の多様性が確保できていることが判る．

4.2 19 路盤囲碁での対局

gnugo と提案手法を先手後手入れ替え，各 500 局の計 1000 局対局させた．コミは 6 目半とした．結果を表 1 に示す．局面タブーリストを内包した提案手法 gTabu18-hash は，gnugo（UCT）や gTabu18 よりも勝率を改善していることが判る．さらに，勝敗に基づいて局面をタブーリストに追加する手法 gTabu18-hash-win はどの手法よりも勝率で改善している．有意水準 5%で二項検定を行うと p 値は 0.012 となり，有意な棋力向上がみられた．

*GNU Go http://www.gnu.org/software/gnugo/

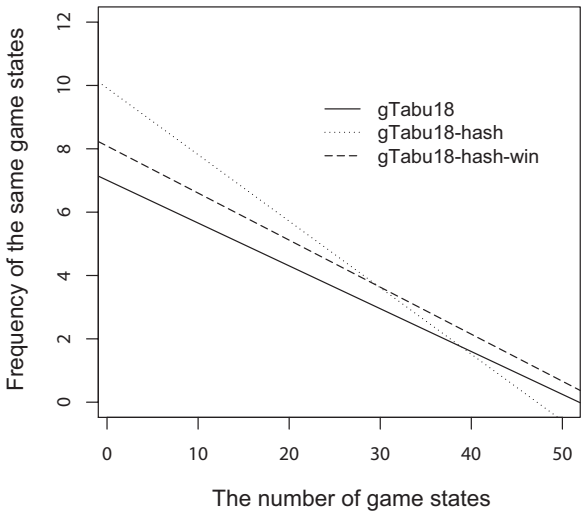

図 2: Top 50 frequencies of the same game states

表 1: Results of the games playing in 19×19 go

Method	Black		White		Total	
	wins	win. rate	wins	win. rate	wins	win. rate
gnugo	503	—	497	—	503	50.3%
gnugo（UCT）	266	53.2%	253	50.6%	519	51.9%
gTabu18	239	47.8%	252	50.4%	491	49.1%
gTabu18-hash	264	52.8%	259	51.8%	523	52.3%
gTabu18-hash-win	277	55.4%	263	52.6%	540	54.0%

5 おわりに

本稿では，19 路盤囲碁に対応した局面タブーリストを内包したモンテカルロ木探索手法を提案した．提案手法をオープンソースである GNU Go 3.8 に組み込むことで，従来法と同程度の多様性が確保できることを確認した．また，対局では勝敗に基づいた局面タブーリストを用いた提案手法が最も勝率が高く，二項検定によって棋力向上の有意性も確認できた．

参考文献

[1] 太田雄大, 伊藤雅: “タブーリストを内包したモンテカルロ木探索の詰碁と 9 路盤囲碁への応用”, 電気学会論文誌 C, Vol. 135, No. 3, March 2015 (掲載予定)

[2] L. Kocsis and C. Szepesvári: “Bandit based Monte-Carlo Planning”, Proc. of the 17th European Conference on Machine Learning (ECML 2006), pp. 282–293, Berlin, Germany, September 2006

[3] A. Zobrist: “A New Hashing Method with Applications for Game Playing”, ICGA Journal, Vol.13, No.2, pp. 69–73, 1990