

プレイアウト履歴から成るゲーム木を用いた囲碁アルゴリズム

川合 諒

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

コンピュータ囲碁における主流の1つにモンテカルロ木探索がある。この手法の登場により棋力が向上した。現在も多くのコンピュータ囲碁で採用されている。モンテカルロ木探索の問題点の1つとしてプレイアウトの結果から勝敗のみを利用している点が挙げられる。これではプレイアウトの実行時間に対して得られる情報量が少な過ぎる。

この問題点を改良するため、プレイアウトの結果として得られる内容を勝敗のみとせず、開始局面、手番を含めて記憶、蓄積させ利用することを提案する。また、記憶したプレイアウト結果を基にゲーム木の生成を行い、探索の評価に履歴を使用する。

2 モンテカルロ木探索

モンテカルロ木探索とは、Auer らによって提案された UCB1 アルゴリズム (Upper Confidence Bound)[1] に木探索を組み込んだ UCT(UCB applied to Trees) アルゴリズム [2] のことである。

ある局面が与えられると次の着手点を決めるため、モンテカルロ木の根から葉までのノードを UCB1 値の大きい方に辿って降りていく。葉まで到達すると、葉の局面でプレイアウトを実行する。プレイアウトとは現局面に対して合法手をランダムに生成し、終局までをシミュレートすることである。プレイアウトによって得られた勝敗結果を今度は葉から根まで伝播する。プレイアウト数が閾値を超えれば、その葉を展開する。

これらの操作を繰り返し、最終的に UCB1 値が最も大きくなった交点へ着手する手法がモンテカルロ木探索である。

モンテカルロ木探索の問題点の1つにプレイアウトの勝敗のみを取得している点がある。プレイアウトの終局面や手番の情報をすべて破棄している。これでは計算時間に対して得られる情報が少な過ぎる。

本稿ではこの問題点に着目し、プレイアウト結果を履歴として記録し、履歴をゲーム木作成や探索評価に利用することを提案する。

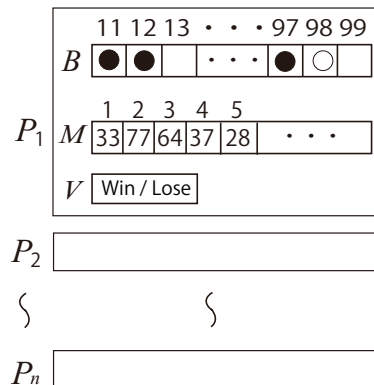


図 1: プレイアウト履歴の構造

3 提案手法

プレイアウト履歴とゲーム木の構造を説明する。

3.1 プレイアウト履歴の構造

提案手法ではプレイアウトから得られる勝敗の他に開始局面、着手順を使用する。プレイアウト履歴の構造を図 1 に示す。履歴構造は 1 回のプレイアウトの履歴 P を事前に設定した個数 n だけ集めて構成する。各履歴 P は開始局面 B 、着手順 M 、勝敗 V から成る。

次にプレイアウト履歴の記録方法について説明する。まず、プレイアウトの開始局面を B に記録する。プレイアウトを実行し、ランダムな着手を行う。この着手 1 つひとつを着手順 M へ記録する。これを着手可能点が無くなるまで実行する。最後に終局面の勝敗を計算し、勝敗 V にその結果を記録する。これを設定した回数だけプレイアウトを実行し、履歴を作成する。

現局面に着手が行われ、局面が進行すると履歴の中には現局面と矛盾した局面が生じる。矛盾した履歴が残らないようにプレイアウト履歴は現局面が進行した時点で逐次更新する。

3.2 プレイアウト履歴から成るゲーム木

モンテカルロ木探索では仮想局面での着手によってゲーム木を作成する。それに対し本提案ではプレイアウト履歴によってゲーム木を作成する。プレイアウト履歴の着手順 M を辿って現局面の根ノードから枝を作成していく。

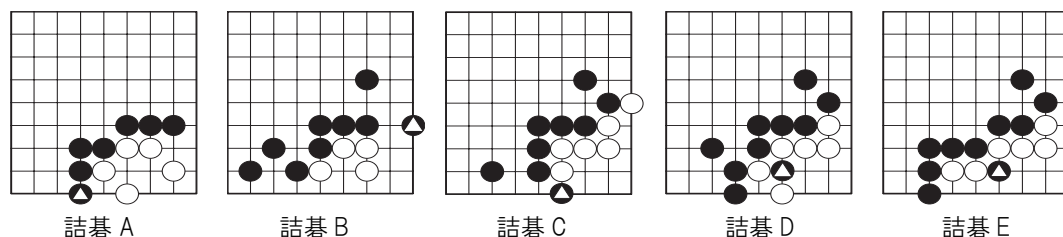


図 2: 詰碁問題

プレイアウト履歴の 1 回分の内容が 1 本の枝となる。プレイアウトの着手 1 手ごとにノードは生成され、各ノードごとに 1) ノード到達時に着手した交点, 2) 直前のノード, 3) このノードから派生するノード群, 4) ノードにおけるプレイアウトの生成回数と勝率, 以上 4 つが記録される。このプレイアウト履歴によって作成された枝を束ねることでゲーム木を構成する。また、同一の着手順で進んだノードは、ノードの生成回数と勝率のみが更新される。

現局面と矛盾したプレイアウト履歴を削除する際には、同時にゲーム木の枝刈りを行う。着手順 M を基にゲーム木の根ノードから枝を降りていき、ノードの生成回数と勝率を更新することで枝刈りを行う。

3.3 ゲーム木の利用

現局面への着手を選択するために、ゲーム木のノード生成回数と勝率に対して、それぞれ閾値 $MAKE$ と $RATE$ を設定する。この閾値と比較して、ノードの生成回数と勝利回数が閾値よりも高い記録を持つノードを抽出する。このようなノードを特徴点と呼ぶ。特徴点が一定数以上の場合は閾値 $RATE$ を再設定し、絞り込みを行う。

特徴点を抽出すると、次にそのノードから根まで枝を辿る。根に繋がっているノードの着手は現局面への着手候補点となる。この処理を各特徴点ごとに行い、着手候補点を複数個選択する。最後に着手候補点の中から最も勝率の高い手を現局面への着手して採用する。

4 数値実験

性能検証のために詰碁の正答検証を行った。文献 [3] で紹介されている単純なモンテカルロ木探索と提案法を導入した囲碁アルゴリズムとの正答数で比較する。

図 2 に 5 つの詰碁問題を示す。△の付いている石が正着である。これらの詰碁の 1 手目の正答数を各アルゴリズムごとに 100 回試行した。各々のプレイアウト数は 8000 とした。結果を表 1 に示す。

表 1: 提案法とモンテカルロ木探索の正答数

詰碁問題	A	B	C	D	E
提案法	94	31	37	27	72
モンテカルロ木探索	98	1	22	16	58

この結果から提案法はモンテカルロ木探索と同等の正答を導けることが分かる。また、詰碁 B のように、モンテカルロ木探索ではほぼ正答を導けなかった問題に対しても一定数の正答を導ける問題も存在した。

5 おわりに

本稿では、プレイアウト履歴を用いてゲーム木を作成し、それを探索に利用する囲碁アルゴリズムを提案した。また、提案法の性能検証のために詰碁による比較を行い、単純なモンテカルロ木探索と同程度の結果を得られることを確認した。

謝 辞

本研究の一部は文部科学省 科研費 基盤研究 (C) No. 16K00510 の助成を得た。ここに謝意を表する。

参考文献

- [1] P. Auer, N. Cesa-Bianchi, and P. Fischer, “Finite-time Analysis of the Multi-armed Bandit Problem”, *Machine Learning*, Vol. 47, No. 2, pp. 235–256, 2002.
- [2] R. Coulom, “Computing Elo Ratings of Move Patterns in the Game of Go”, *Computer Games Workshop 2007 (CGW 2007)*, Amsterdam, The Netherlands, 2007.
- [3] 美添一樹, 山下宏, 松原仁 (編), 「コンピュータ囲碁—モンテカルロ法の理論と実践」, 共立出版, 2012.