様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　元 年　 ６ 月 ２０ 日現在

機関番号：３３９０３
研究種目：基盤研究(C)（一般）
研究期間：2016～2018
課題番号：１６Ｋ００５１０

研究課題名（和文）モダンヒューリスティックスとプレイアウトに基づく囲碁アルゴリズムの構築

研究課題名（英文）Construction of Go Algorithm Based on Modern Heuristics and Playout

研究代表者
伊藤　雅（ITOH, Masaru）

愛知工業大学・情報科学部・教授

研究者番号：８０２２１０２６

交付決定額（研究期間全体）：（直接経費）　　2,600,000円

研究成果の概要（和文）：　囲碁アルゴリズムの主流はモンテカルロ木探索である。この探索手法は大量のプレイアウトを生成する。しかし、それらが再利用されることはない。そこで、過去のプレイアウト履歴からゲーム木を生成して最善手を導出する手法を提案した。
　もうひとつ、深層学習とプレイアウトに基づく囲碁アルゴリズムを提案した。深層学習は多層畳み込みニューラルネットワークで実現した。ニューラルネットワークはモダンヒューリスティックスの代表的手法のひとつである。アルファ碁が提唱するロールアウトによる勝敗は使わず、この部分をプレイアウトで代用した。木探索のノード評価にはUCB1値ではなく、アルファ碁が提唱するアクション値を採用した。

研究成果の学術的意義や社会的意義
　過去のプレイアウト履歴からゲーム木を作成して最善手を求める方法は詰碁のような狭い探索空間で有効に機能した。しかし、9路盤囲碁では完全に無効であった。アイデアは興味深いが、モンテカルロ木探索と同等以上の手法にはなり得ないことが判明した。
　もうひとつの深層学習とプレイアウトに基づく囲碁アルゴリズムは少資源環境下で動作させることに成功した。既存のオープンソース囲碁に統計的に有意に勝利できることを確認した。Value-MCTSで必要となるノード展開閾値やMixingパラメータといった各種パラメータ値を適切に設定する必要があり、それらの調整は容易でない。さらに、実行時間の短縮という大きな課題も残った。

研究成果の概要（英文）：The recently major algorithm in computer go is based on the Monte-Carlo tree search. The search method generates a large amount of playouts to obtain the best next move. However, those playouts are rarely reused until the end. So, I proposed a method to get the best next move by making game tree from past playout history. It was confirmed that the proposed method can be applied to small problems such as tsumego.
I proposed another algorithm based on deep learning and playouts. Deep learning was realized by multi-layer convolutional neural networks. Neural network is one of the representative methods of modern heuristics. In the Value-MCTS, the search method did not use a win/loss of rollout proposed by AlphaGo, and instead substituted this part with playout. For node evaluation, not the UCB1 value but the action value proposed by AlphaGo was adopted. It was shown statistically that the proposed method is superior to the existing Go software if the parameter values are set correctly.

研究分野：システム最適化

キーワード：囲碁アルゴリズム　プレイアウト　モダンヒューリスティックス　ニューラルネットワーク　深層学習
　　モンテカルロ木探索　アクション値

様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９、ＣＫ－１９（共通）

１．研究開始当初の背景

平成 28 年 3 月に Google 傘下の DeepMind 社が開発した囲碁 AI ソフトウェア“AlphaGo”（以下、アルファ碁という）がプロ棋士である韓国のイ・セドル九段に 4 勝 1 敗で勝利した。アルファ碁は深層学習、強化学習、モンテカルロ木探索の 3 つを組み合わせた思考ルーチンを搭載していた。本研究課題の研究計画調書を提出したのが平成 27 年 10 月であり、交付内定を得たのが平成 28 年 4 月である。このわずか半年の間に囲碁アルゴリズムの研究がモンテカルロ木探索の改良から人工知能（AI）の導入に移行してしまった。研究計画作成時には、AI の導入は完全に考慮外であった。再利用可能な過去のプレイアウトを駆使して最善手を導き出すことしか念頭になかった。アルファ碁の深層学習は畳み込みニューラルネットワークで実現されており、ニューラルネットワークは広義の意味でモダンヒューリスティックスの一手法と位置づけられている。研究計画は大きく修正せざるを得なかったが、研究課題名そのものを改める必要はないと判断し、研究を続行することにした。

２．研究の目的

現在の囲碁アルゴリズムの主流はモンテカルロ木探索である。ある局面から終局まで交互に適当に石を打つことをプレイアウトという。ある局面で次の一手を決定するのに 5 千回から 1 万回のプレイアウトを実行する。プレイアウトでは手番や手数、打った石の色と位置情報が記録可能である。しかし、通常のモンテカルロ木探索ではそれらを一切利用しない。モンテカルロ木探索をその手法の視点から研究目的を 2 つに分けて記述する。

(1) プレイアウト履歴を用いた囲碁アルゴリズムの構築

局面毎に大量に生成されるプレイアウトを再利用可能なプレイアウト履歴として構築し直し、木探索ではなく表探索で次の一手を決定する。アイデアの元となるのはタブー探索で使われるタブー表である。表探索から得られるゲーム木を使って単純なモンテカルロ木探索と同程度の棋力を有するアルゴリズムを構築する、これがひとつ目の研究目的である。

(2) 畳み込みニューラルネットワークとプレイアウトに基づく囲碁アルゴリズムの構築

モダンヒューリスティックスのひとつであるニューラルネットワークとプレイアウトを組み合わせて最善手を導出する、これが 2 つ目の研究目的である。ベースにあるのはモンテカルロ木探索（Monte-Carlo Tree Search : MCTS）である。アルファ碁は APV-MCTS（Asynchronous Policy and Value MCTS）というマルチスレッドに対応した非同期方策と Value Network を内包した MCTS で次の一手を求めている。一方の提案法は、Value-MCTS の部分こそ使うが、アルファ碁のような分散非同期型ではなく、マルチプロセスを用いた単体非同期型の囲碁アルゴリズムである。単純な乱数によるプレイアウトではなく、ヒューリスティックを取り入れたプレイアウトを活用する。因みに、アルファ碁はプレイアウトではなく、この部分にロールアウトを採用している。ロールアウトとは、活性化関数を softmax 関数にしたロジスティック回帰モデルであり、3 層ニューラルネットで実現される。

３．研究の方法

研究計画調書に記載した研究計画・概要の内容をかなり変更した。理由は、１．研究開始当初の背景で記述した通り、研究計画調書提出後かつ研究課題採択通知前にアルファ碁が登場し、その論文が発表されたからである。当初のアイデアである過去のプレイアウト履歴を利用した候補手生成とアルファ碁が採用した深層学習の融合は極めて難しいと判断し、研究の方法を次の 3 つに改めた。

(1) プレイアウト履歴から成るゲーム木を用いた詰碁の数値実験

プレイアウト履歴からゲーム木を作成し、その木を使って最善手を導出する囲碁アルゴリズムを考案した。過去のプレイアウト履歴からゲーム木を生成する過程でノード生成回数と勝利回数を計算し、これらが共に閾値を超える特徴点をすべて抽出する。その特徴点から最大勝率をもつゲーム木のノードを選択して、そこから根ノードに遡って、次の最善手を決定する。黒先白死の 5 つの詰碁に対して提案法と単純モンテカルロ木探索をそれぞれ 100 回試行させる。黒初手正答数によって提案法の性能を評価する。

(2) プレイアウト履歴から成るゲーム木を用いた 9 路盤囲碁の数値実験

詰碁の数値実験で用いた提案法を単純モンテカルロ木探索と 9 路盤囲碁で対局させる。先手後手を交互に入れ替えて互先で 100 回対戦させ、その勝敗数で優劣を決定する。

さらに、提案法を 2 つの観点から改良を施す。ひとつ目の改良は、プレイアウト履歴を作成する過程でランダムなプレイアウトではなく、パターンを導入したプレイアウトに改める。パターンには、蜂の巣型の 3×3 パターンを 61 個、5×5 パターンを 21 個、計 82 個のパターンを導入する。2 つ目の改良は、プレイアウト履歴からゲーム木を生成する際に合流を検知できるようアルゴリズムを改良する。これら 2 つの改良を施して再び 9 路盤囲碁で単純モンテカルロ木探索と 100 回対戦させてその勝敗数で改良提案法の棋力を測る。

(3) 深層学習とプレイアウトに基づく囲碁アルゴリズムによる 19 路盤囲碁の数値実験

プレイアウトと深層学習を組み合わせた少資源環境下で動作する囲碁アルゴリズムを構築する。ここで、少資源環境とは、1 CPU & 1 GPU 程度で構成されるデスクトップ環境をいう。深層学習には 16 層からなる畳み込みニューラルネットワークを利用する。2016 年 1 月に発表されたアルファ碁の再現プロジェクトのひとつに RocAlphaGo がある。RocAlphaGo が提供する Python で開発された深層学習部分を利用する。教師付学習の SL Policy Network、強化学習の RL Policy Network、盤面評価関数として機能する Value Network の 3 つである。

提案法の特徴は、Value Network とモンテカルロ木探索を融合させることとアルファ碁が提唱する Tree Policy や Rollout Policy を使わずに囲碁 AI を動作させることである。アルファ碁は分散非同期型の囲碁 AI である。これに対し提案法は単体非同期型の囲碁 AI である。提案法では、Tree Policy の処理過程を省略し、葉ノード展開時に SL Policy Network と同期させて、着手確率が高い有望手の上位 20 手のみを探索木に追加する。さらにロールアウトによる勝敗は使わず、この部分を Ray のプレイアウトで代用する。Ray はプレイアウトに非決定論的なヒューリスティックを取り入れた思考ルーチンであり、2016 年に BSD ライセンスで公開された。C++言語で開発されているため、RocAlphaGo が提供する Python スクリプトとは相性が悪い。そこで Cython 言語を導入して共有ライブラリを構築する。木探索で使うノード評価値にはモンテカルロ木探索で常用される UCB1 値ではなく,この部分はアルファ碁が提唱するアクション値を採用する。RocAlphaGo が 19 路盤にしか対応していないため、提案法の棋力判定は互先の 19 路盤囲碁で検証する。

第 2 章 研究の目的(2)で述べたように提案法では Value-MCTS を使う。開発した囲碁 AI には様々な工夫を凝らしている。マルチプロセスで使う共有メモリ空間には、探索木と 2 つの実行タスクキューを置いている。SL Policy Network 用のタスクキューと Value Network 用のタスクキューの 2 つである。そして、GPU 実行用プロセスと探索プロセスの 2 種類を用意した。GPU 実行用プロセスは、先の 2 つの実行タスクキューから GPU で実行する 2 つの深層学習(SL Policy Network と Value Network) の片方のみを選択させ、タスクを順次 GPU 側に送る処理を担う。葉ノード展開時に使用する SL Policy Network の優先度は高い。展開と同時に子ノードを生成する必要があるからである。そのため割り込み処理ができるようにした。もう一方の探索プロセスには Virtual Loss という手法を併用した Tree 並列化を導入した。共有メモリ空間に置いたひとつの探索木を使って複数の探索プロセスを並列で実行させ、ノードの選択・評価・更新を行う。このとき Value Network は非同期に起動され、ノード諸元値を適宜更新する。葉ノードでプレイアウト回数が事前に定めておいたノード展開閾値 (node expansion threshold) を超えた場合、SL Policy Network が同期的に起動され、子ノードが葉に追加されて、探索木の形状に変化が生じる。この時点で探索プロセスを再び並列実行し直す。SL Policy Network は探索木の形状に関わるため同期を取りつつ起動され、Value Network は並列実行されている複数の探索プロセスから非同期に起動される。このように提案法は単体非同期型のアルゴリズムとなっている。ここまでは主としてマルチプロセスに関する工夫である。

探索木のノード諸元値について補足する。最も重要なのがアクション値である。これは従来のモンテカルロ木探索で使われる UCB1 値に相当する。アクション値は加重平均値とバイアス項の和で計算される。バイアス項は誤差項と考えればよい。加重平均値は Value Network からの平均勝率とプレイアウトからの平均勝率の加重平均で算出される。[0, 1]定数λを使って 2 つの平均勝率をそれぞれ (1－λ) 倍、λ倍する。プレイアウトの優位性を定数λで表現していると捉えると分かり易い。定数λのことを Mixing パラメータという。

４．研究成果

研究の方法で書いた詰碁と 9 路盤囲碁、そして 19 路盤囲碁の 3 つの数値実験から得られた各研究成果について次の項(1)～(3)で記述する。

(1) プレイアウト履歴から成るゲーム木を用いた詰碁の成果

第 3 章 研究の方法(1)に基づいて開発した提案手法の実験結果を示す。プレイアウトの履歴サイズを 40,000、一手の生成で発生させるプレイアウト数を 8,000 とした。5 つの詰碁を A, B, C, D, E と表記する。初期乱数値を適当に変更しながら提案法と単純モンテカルロ木探索でそれぞれ 100 回試行して各詰碁の黒初手を導出する。100 回中の正着数を「詰碁 (提案法の正着数, 単純モンテカルロ木探索の正着数)」の形式で表現する。結果は次のようになった。

A(94, 98), B(31, 1), C(37, 22), D(27, 16), E(72, 58)。この結果から提案法は単純モンテカルロ木探索とほぼ同等の正答を導けることが分かった。詰碁 A では提案法が劣り、詰碁 B では提案法が優っていた。黒先白死の詰碁のみで、しかも黒初手正着でしか評価していない。詰碁の一本道を辿れるか、統計的に提案法の優位性を言えるか、といった踏み込んだ議論の言及には至っていない。言及を避けた理由は、次に述べる 9 路盤囲碁の対局結果が予想以上に悪く、詰碁の性能に言及する必要はない、と判断したからである。

(2) プレイアウト履歴から成るゲーム木を用いた 9 路盤囲碁の成果

詰碁の数値実験で用いた提案法を単純モンテカルロ木探索と 9 路盤囲碁で 100 回対戦させた。結果は提案法の 2 勝 98 敗で大敗を喫した。

プレイアウト履歴からゲーム木を生成するので、質の良いプレイアウトを生成できるようパターン 82 個を導入した。さらに、ゲーム木生成時に合流を検知できるように提案法を改良した。改良提案法を再び単純モンテカルロ木探索と先手後手を交互に入れ替えて互先で 100 回対戦させて改良提案法の棋力を測定した。結果は 1 勝増えただけの 3 勝 97 敗であった。提案法の惨敗であった。

プレイアウト生成時のパターン一致率は約 15%と低く、ゲーム木生成時に発生する合流も一局当り 15%程しかなかった。追加で改良した 1) プレイアウトへのパターンの導入、2) ゲーム木生成時の合流検知、これらは提案法の棋力向上への貢献がほとんどないことが判明した。

結局、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は、詰碁のような狭い探索空間では有効に機能するが、少し広い 9 路盤囲碁では単純モンテカルロ木探索に棋力で遠く及ばないことが明らかとなった。プレイアウト履歴からゲーム木を作成し、そのゲーム木から特徴点を抽出して次の一手を導出する。その挙動が大量のプレイアウトを利用する原始モンテカルロ囲碁の挙動と似ているのが、棋力向上を達成できなかった理由のひとつかもしれない。

(3) 深層学習とプレイアウトに基づく囲碁アルゴリズムによる 19 路盤囲碁の成果

事前の予備実験で 4 つのオープンソース囲碁エンジンの棋力を測定した。Ray, Fuego, Pachi, GNU Go の 4 つである。結果は Ray の圧勝であった。よって、提案法の棋力判定の対戦相手を Ray のみに限定した。提案法の囲碁エンジンは Python/Cython 言語で開発されているため、Cygo と称すことにする。

まず、教師付学習の SL Policy Network で使用したデータ数やパラメータは、棋譜数 59,976、盤面数 94,731,144、エポック数 7、ミニバッチ 16 である。教師付学習での最終精度は 52.3%であった。次の強化学習 RL Policy Network で使用したパラメータは、save-every = 10, game-batch = 10, iterations = 6000, record-every=1 である。強化学習後の重みで Ray と 500 局対戦させたが 142 勝 358 敗で Ray に負け越していた。最後に盤面評価関数として機能する Value Network を CNN (Convolutional Neural Network：畳み込みニューラルネットワーク) で学習させた。CNN に与えるデータセットの作成には学習済み SL/RL Policy Network を使用した。作成したデータセット数は 194,122 個、盤面数で 1,552,976 個である。CNN のパラメータはバッチサイズ 16、ミニバッチ 16、エポック数 100 などである。学習推移を図 1 に示す。最終精度 51.4%を達成した。

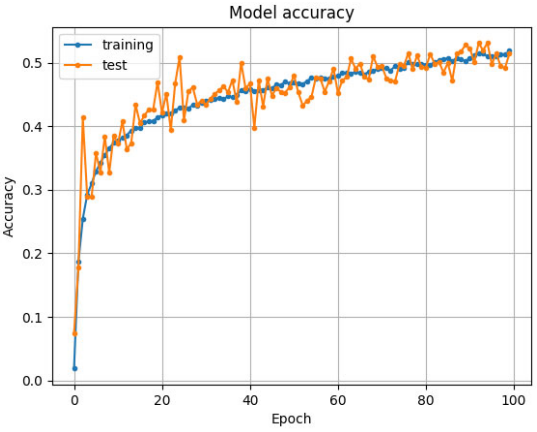

図 1 Value Network の学習推移

得られた重みで Cygo と Ray を先手 250 局後手 250 局の計 500 局対戦させた。コミを 6 目半、プレイアウト数を 6,000 で統一した。Cygo は少資源環境下で動作するよう開発した。CPU プロセス数は動作の安定性を最優先させて 2 とした。

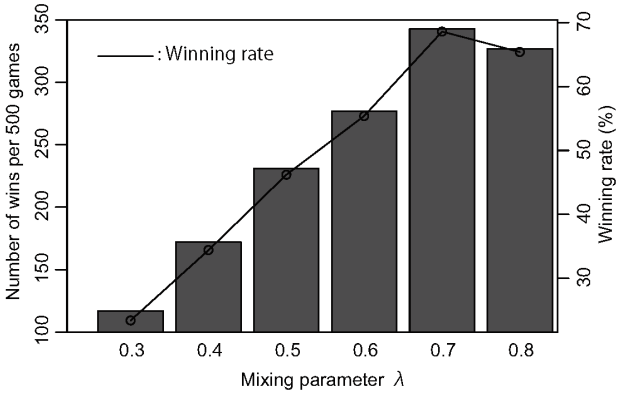

図 2 ノード展開閾値 20 での Cygo の勝数と勝率

数値実験で求めるべきパラメータは大別すれば 2 つある。ひとつは Cygo で使う最適な Mixing パラメータλの同定である。もうひとつはノード展開閾値の最適値の推定である。ノード展開閾値を 20 に固定してλの値を横軸に 0.3～0.8 の範囲で 0.1 ごとに変化させた場合の Cygo の勝数(棒グラフ：左軸)と勝率(折れ線グラフ：右軸)をプロットしたのが図 2 である。λが 0.7 のとき、Cygo の勝数が 343 で最大となり、そのときの勝率は 68.6%であった。有意水準 5%の二項検定によって Cygo の統計的有意性を確認した。λが 0.5 以下になると Cygo の棋力の統計的有意性は消失した。

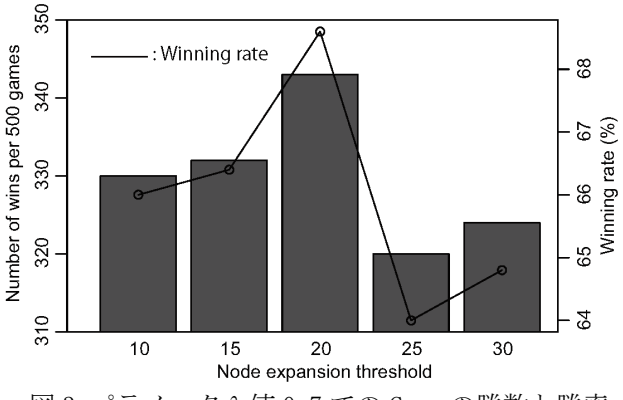

図 3 パラメータλ値 0.7 での Cygo の勝数と勝率

次に、Mixing パラメータ λ の値を 0.7 に固定して、ノード展開閾値を横軸 10～30 の範囲で 5 刻みに変化させた場合の Cygo の勝数をプロットすると図 3 のようになった。ノード展開閾値に限れば、10～30 の間であれば Cygo の統計的有意性が保たれることを二項検定により確認した。結局、Cygo は 343 勝 157 敗の勝率 68.8%で Ray に大きく勝ち越した。

提案法の Cygo には大きな問題がある。実行に要する計算時間である。Ray との対局実験で 1 局当りの平均で Cygo は約 1 時間 7 分 18 秒を消費していた。一方、Ray は約 4 分 28 秒の計算時間しか消費していなかった。CPU プロセス数をデフォルトの 2 から 8 まで増やしても、約 45 分 19 秒までしか短縮できなかった。このことから 1 手の導出に要する計算時間の短縮こそが Cygo の改良における今後の最大の課題である。

＜引用文献＞

① 川合諒, 伊藤雅, 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, 情報処理学会第 79 回全国大会講演論文集 第 2 分冊, pp.463-464, 2017.
② 伊藤雅, 伊藤有人, 深層学習とプレイアウトに基づく囲碁アルゴリズム, 愛知工業大学研究報告, Vol.54, pp.110-117, 2019.

５．主な発表論文等

〔雑誌論文〕（計 1 件）

① 伊藤雅, 伊藤有人, 深層学習とプレイアウトに基づく囲碁アルゴリズム, 査読なし, 愛知工業大学研究報告, Vol.54, pp.110-117, 2019.
http://repository.aitech.ac.jp/dspace/handle/11133/3491

〔学会発表〕（計 4 件）

① 伊藤雅, 伊藤有人, プレイアウトと深層学習を組み合わせた囲碁アルゴリズム, 平成 31 年電気学会全国大会, 北海道科学大（北海道・札幌）2019-3-13.
② 伊藤有人, 伊藤雅, 少資源環境下における RocAlphaGo の改良とその棋力検証, 情報処理学会第 80 回全国大会, 早稲田大・西早稲田キャンパス（東京・新宿）2018-3-13.
③ 川合諒, 伊藤雅, 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, 情報処理学会第 79 回全国大会, 名古屋大・東山キャンパス（愛知・名古屋）2017-3-18.
④ 山川雄史, 伊藤雅, RocAlphaGo に基づく囲碁アルゴリズム, 日本 OR 学会中部支部第 44 回研究発表会, 愛知県立大・サテライトキャンパス（愛知・名古屋）2017-3-4.

〔図書〕（計 0 件）

〔産業財産権〕
○出願状況（計 0 件）
○取得状況（計 0 件）

〔その他〕
招待講演（計 1 件）
① 伊藤雅, コンピュータ囲碁の最前線, 愛知工業大学総合技術研究所主催第 9 回 AIT テクノサロン, 愛知工大・八草キャンパス（愛知・豊田）2017-7-21.
ホームページ等
② 平成 30・29・28 年度 修士論文・卒業研究紹介
https://aitech.ac.jp/~milabo/member/(k24/k23/k22/)
本研究課題に関する研究テーマを掲載している
③ RocAlphaGo に基づく囲碁アルゴリズム
https://aitech.ac.jp/~milabo/member/k22/OR_chubu_44.pdf
平成 28 年度卒業研究の 1 件を日本 OR 学会中部支部第 44 回研究発表会で発表した予稿

６．研究組織

(1) 研究分担者　なし

(2) 研究協力者　3 名
研究協力者氏名：伊藤　有人, 川合　諒,　山川　雄史
ローマ字氏名：（ITOH, arito, KAWAI, makoto, YAMAKAWA, yushi）