

モンテカルロ木探索

—コンピュータ囲碁に革命を起こした新手法

美添一樹

独立行政法人 科学技術振興機構

囲碁は、主なボードゲームの中でコンピュータの挑戦を拒み続けてきた唯一のゲームである。囲碁の難しさは良い評価関数を作ることが困難であるということに起因していた。しかし 2006 年にコンピュータ囲碁の世界にまったく新しいアルゴリズムがもたらされた。評価関数が不要という画期的な探索アルゴリズム、通称、モンテカルロ木探索と呼ばれるものである。登場から 2 年あまりで 9 路盤ではプロ棋士を破るほどの強さを獲得した。そのアルゴリズムの性質や理論的背景について述べ、今後の展望を探る。

コンピュータ囲碁に起こった革命

2008 年 3 月末に、パリ囲碁トーナメント^{☆1}のエキシビジョンとして、ルーマニア出身で日本棋院中部総本部所属のプロ棋士 Catalin Taranu (タラヌ・カタリン) 五段と囲碁プログラム MoGo の対局が行われた。9 路盤では互角の条件で 3 局対戦し、対戦成績は MoGo の 1 勝 2 敗であった。19 路盤では MoGo が 9 子のハンデをもらって対戦したが、カタリン五段が勝利を収めた(目安として、9 子のハンデでプロに勝てればアマ有段者)。通常の 19 路盤より小さい 9 路盤での対局ではあるが、盤のサイズを問わず、公の場でコンピュータがプロ棋士から 1 勝を挙げたことは史上初の快挙である。

1997 年に IBM の Deep Blue が当時のチェスの世界チャンピオンであった Kasparov を破ったことが特に有名だが、それ以外のほとんどのボードゲームにおいても、コンピュータは人間のプレイヤーに迫る、あるいはそれ以上の強さを獲得している。しかし主なボードゲームの中で、囲碁だけがコンピュータの挑戦をはねのけてきた。

その状況を覆す可能性を持ったアルゴリズムが登場し、囲碁プログラムの棋力はこの 2 年余りで急激に向上した。原動力となったのは 2006 年に登場した画期的なアルゴリズムである。まだ日本語の呼称すら定着していないが、英語では Monte-Carlo Tree Search という通称で呼ばれているので、**モンテカルロ木探索**と呼ぶことにする。

囲碁の難しさはどこにあったのか、なぜ囲碁には新たなアルゴリズムが有効であるのか、理論的背景なども含めて説明していきたい。

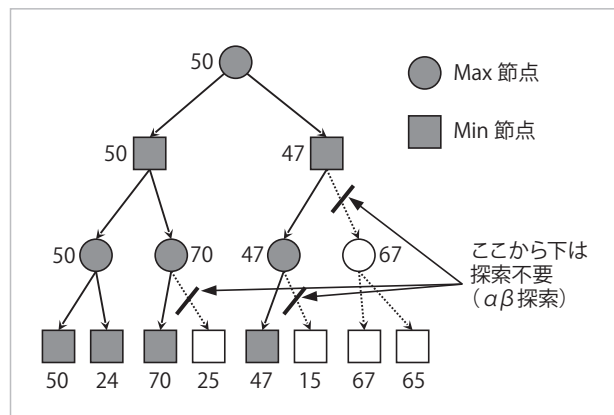


図-1 mini-max 探索と alpha-beta 探索

コンピュータプレイヤーの進歩

二人零和完全情報ゲームは mini-max 探索により最善手を求めることができる。

ゲームのスコアが数値で与えられるとする。先手番(のプレイヤー)が有利な場合は数値が大きく、後手番が有利な場合は数値が小さいとする。先手番は数値を可能な限り大きくしようとし(max プレイヤーと呼ぶ)、後手番は逆に小さくしようとする(min プレイヤーと呼ぶ)はずである。ゲームの探索は図-1のように、木で表すことができる。max プレイヤーの手番の節点を max 節点、min プレイヤーの手番の節点を min 節点と呼ぶ。端点の数値がゲームのスコアを示す。双方のプレイヤーが最善手を選択すると、このゲームは 50 点の節点に到達する。

mini-max 木を探索する際に、数値の大小関係に注意すると、探索しなくてもよい部分があることが分かる。必要最小限の枝だけを探索する手法が有名な alpha-

☆1 <http://paris2008.jeudego.org/>

チェッカー	1994年に世界チャンピオンに勝利 (2007年に初期配置の引分証明)
オセロ	1996年に人間の世界チャンピオンに完勝
チェス	1997年にIBMのDeep Blueが当時世界チャンピオンのKasparovを破る
将棋	現在アマ5段、奨励会初段程度の強さと言われている
囲碁	アマ初段をようやく超えた程度

表-1 コンピュータの強さ（二人零和完全情報ゲーム）

ゲーム	局面数
チェッカー	10^{20}
オセロ	10^{28}
チェス	10^{50}
将棋	10^{71}
囲碁（9路盤）	10^{38}
囲碁（19路盤）	10^{171}

表-2 探索空間（局面数）の推定

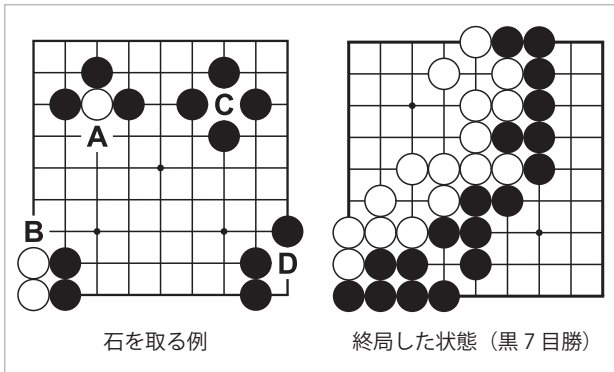


図-2 囲碁のルール

beta 探索である。mini-max 探索ですべての節点を探索する場合に N 個の節点を探索する必要があるとすると、alpha-beta 探索では最善の場合、ほぼ \sqrt{N} 個の節点を探索するだけで mini-max 探索で見つかった最善手と同一スコアに達する手を見出すことができる。

表-1 に示されるような、囲碁以外の多くの二人零和完全情報ゲームでのコンピュータの成功は、すべて alpha-beta 探索によるものと言ってよいであろう。

その中でコンピュータ囲碁の弱さが際立っている。囲碁のどのような性質が、これまでコンピュータによる挑戦を退けてきたのだろうか。今回はまずその点から説明を始めよう。

なぜ囲碁は難しいのか

● 囲碁のルール

囲碁は黒と白がお互いの石を交互に盤に置いていき、最後に占拠した領域の広さを競うゲームである。盤面には格子状に線が引いてあり、石は交点の上に置く。石を置くことを打つと言い、また領域のことを地と言う。

正式な競技は19路盤で行われるが、スペースの都合上、9路盤の図を用いて説明する。9路盤は、入門者の練習用や短時間の勝負を楽しみたい場合などに用いられる。

黒と白の2人のプレイヤーが盤面の交点の上に石を打っていく。基本的なルールは以下の3つである。

- 相手の石に隙間なく囲われた石は取られる。図-2の左側は石を取る例である。Aとマークされているところに黒が石を打つと、囲われた白石は取られて盤上から取り除かれる。同じ色の石同士は上下左右の4方向に連結する。連結した石の周囲が囲われるとひとまとめで取られる。よって、Bに黒石を打つと白石を2個まとめて取ることができる。
- 打った時点で隙間なく囲われているような手は禁止（着手禁止点）。図のCに白が石を打つことはできない。Dはまだ隙間があるので打っても良い。
- 同型反復は禁止。

図-2の右側は終局の例である。日本ルールでは同じ色の石によって囲われた領域を数える。中国ルールでは石の数と囲われた領域の合計を数える。単位は目（もく）という。また、囲碁はそのままでは先手番の黒が有利であるので、その差を是正するためのハンデを黒に負わせるのが普通である。それをコミと言い、日本ルールでは6.5目が用いられる^{☆2}。端数は引き分けをなくするための工夫である。

ルールの一部ではないが重要な概念に、石の死活というものがある。着手禁止点のことを眼と言う。2つ以上の眼を持つ石は自殺しない限り絶対に取られることがなくなるため、これを「生きている」と言う。逆に、眼を2つ持つ余地がなく、いつかは取られてしまう石のことを「死んでいる」という。

● 探索空間の大きさ

囲碁の難しさの中で分かりやすい要素としては、他のゲームと比較して、合法手が多いこと、終局までの手数が長いことが挙げられる。

囲碁においても5路盤では完全に最善の手順が解明されており、盤面が十分小さければコンピュータによるプレイも容易なはずである。では19路の囲碁の探索空間はどの程度か、表-2に他の主なゲームと比較した結果を示す。このように19路盤の囲碁は主なゲームの中で

^{☆2}中国ルールでは、ルールの制約により6.5目と7.5目がほぼ同等であるために7.5目が用いられている。

ポーン	ビショップ	ナイト	ルーク	クイーン	キング
1	3 + α	3	5	9	∞

表-3 チェスの駒の価値

も可能な局面の数が一番大きく、勝敗が決するまでの探索を行うことは到底不可能である。

ところであまり言及されることがないのだが、囲碁の9路盤の探索空間はチェスよりも小さい。にもかかわらず2005年以前の囲碁プログラムは9路盤でも19路盤でもそれほど強さが変わらず、どちらもアマ初段程度であった。このことは、囲碁の難しさには、探索空間の大きさ以外の要素もあることを示唆している。

● 局面の評価関数

理論的には、二人零和完全情報ゲームは mini-max 探索により最善手を求めることができる。ところが、厳密に最善手を求めるためには、図-1の末端の数値はゲームの勝敗を実際に示すスコアでなくてはいけない。三目並べのような単純なゲームならば勝敗が決まるまででのゲーム木をすべて探索し、最善手を求めることができる。しかしより複雑なゲームでは、探索範囲が大きすぎるために実際に勝敗がつく深さまで探索を行うことは現在の技術では不可能である^{☆3}。

そのため、実際には探索を途中の適当な深さで打ち切り、その時点で分かる範囲で一番良さそうな手を求めることになる。適当な深さで打ち切った場合にはどのように探索を行うかということ、ゲームのスコアを近似的に数値化する評価関数を用いる。

人間のプレイヤー向けのチェス入門書では、最初の方に必ず駒の価値が示されている(表-3)。駒の価値を合計すれば、ごく簡単な評価関数を作ることができる。

チェスや将棋などのゲームでは、実際にコンピュータが用いる評価関数も、駒の価値に局面の状況を加味して作ることが多い。将棋では、駒の価値、玉の安全度、駒が動きやすいか、などが評価関数の主な要素である。また、オセロでは隅や辺などの重要な範囲のパターンを学習して評価関数が作成されている。

評価関数に望まれる性質としては、十分正確であることはもちろんだが、さらに十分に高速であることが挙げられる。探索の末端では毎回評価関数を計算する必要があるため、最低でも1秒間に数万回程度は計算可能でないと探索が非常に遅いものになってしまう。

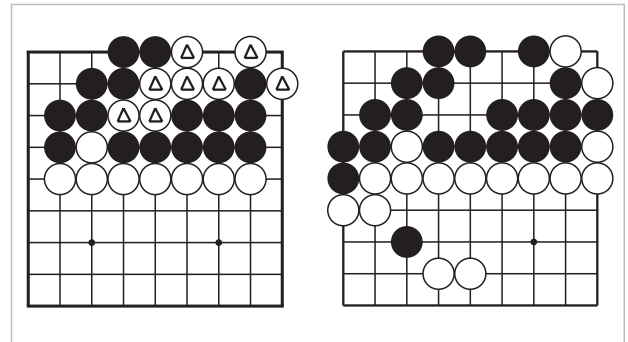


図-3 捨石が成功した実戦例
白は△のついた石8個を捨て、代償に下辺をすべて確保した

● 囲碁の評価関数の難しさ

チェスなどと異なり、囲碁では各石の価値は平等である。占拠した領域の広さを競うゲームであるからには領域の広さを評価基準に用いることができればよいのだが、実際に互いの領域が確定するのはゲームの最終盤になってからであるために、それ以前には領域の広さを求めることは難しい。また、特に19路盤は盤面が広いために、オセロのように明らかに特徴的な個所が少ない。

局所的な最善手が全局的な最善手になりにくいという性質もある。たとえば、相手の石を取ることは局所的には得な手であるが、意図的に自分の石を捨てて全局的に利益を得ることは囲碁では基本的なテクニックである。ある局面では非常に重要だったはずの石が、状況の変化によって価値が下がって、むしろ見捨てた方が有利になるなどということも頻繁に起こる。図-3は捨石が見事に成功した実戦例である。

序盤には定石もあるが、あくまで1つの隅での手順が定石化されているものであり、他の隅の状況によっては、定石通りに打ったのに不利になることも多い。

中盤の評価はさらに難しい。人間のプレイヤーは「厚い／薄い」「形が良い／悪い」「味が良い／悪い」「石が軽い／重い」などの囲碁用語を用いて局面を説明するが、これらの用語は有段者でなければ意味を理解することが難しいであろう。

つまり、囲碁には、チェスや将棋の駒得のような明らかな評価基準がなく、何かの要素の足し算だけで局面の評価を行うことは非常に困難である。過去に多数の研究者、プログラマが取り組んだが、精度が高くかつ高速な評価関数はいまだに開発されていない。コンピュータ囲碁の研究者の間では、囲碁の評価関数を作成することは非常に困難であるという知見が共有されている。

☆3 チェッカーでは、双方が最善手をプレイすると引分になることが2007年に証明された。現在の技術での限界はチェッカーであると言える。



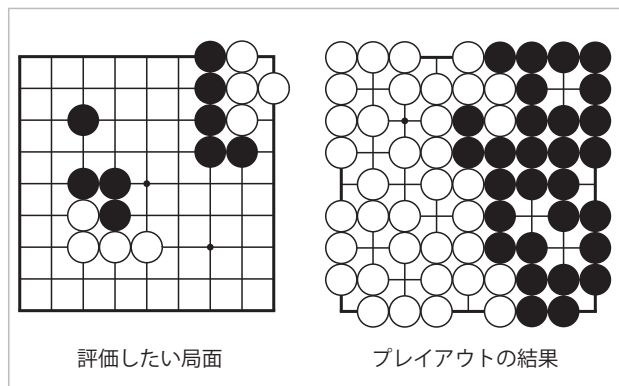


図-4 乱数を用いたプレイによる局面評価 (ゲームプログラミングワークショップ2007でのRémi Coulomの講演より引用)

棋力の向上が見られていない。

モンテカルロ木探索によるプログラム

すでに述べたように、囲碁においては局面を評価することが困難であった。しかし囲碁においても終局した時点であれば、どちらが勝利したのか簡単に計算可能である。Rémi Coulomによって開発されたCrazyStone、およびその後登場したSylvain GellyらによるMoGoはこの性質をたくみに利用したプログラムである。

● 原始モンテカルロ囲碁

一部の研究者によって、モンテカルロ法による囲碁プログラムが研究されていた。1993年のBrügmannによる研究に始まり、Bouzy, Cazenaveらによって受け継がれた一連のプログラムは、乱数を用いて適当に終局までプレイした結果によって局面を評価して着手を選択するというものであった。

囲碁ではゲームが終盤に近づくにつれて合法手の数が減少していくために、合法手の中からランダムに選んで打つだけのプレイヤーでも終局を迎えることはできる。ただし、以下の制約が必要となる。

先述のように、囲碁のルールから自然と導かれる性質として、眼と呼ばれる場所を2つ以上持つ石のグループは「生きて」おり、自殺しない限り相手に取られることはないという事実がある。つまり、自分で自分の眼に打たないようにするだけで、適当にランダムに打つプレイヤー同士の対局でも自然と終局する^{☆4}。このように、乱数を用いたプレイヤー同士で終局までプレイすることをプレイアウトと呼ぶ。

原始モンテカルロ囲碁の原理はきわめて単純である。ある局面からプレイアウトを行い、終局図を得る。図-4に、盤面とプレイアウトによるプレイの結果を示す。左側の盤面から単純なプレイアウトを行った結果、右側のようになった。黒34目対白47目なので白勝ちである。

この結果は人間の眼から見ると非常に不正確であるが、それでも回数を積み重ねることによってそれなりに良い評価値を得ることができる。当然ながら、プレイアウトの回数を増やすほど棋力も上がる傾向にある。

図-5のように各候補手に対して多数のプレイアウトを行い、その期待値によって局面の良し悪しを判断する。

しかしこの方式には本質的な弱点がある。双方のプレイヤーが乱数を用いてプレイすることを仮定しているた

従来の囲碁プログラムGNU Goの例

囲碁がコンピュータにとって難しいと言われながらも、既存のプログラムの棋力もアマ初段程度には到達していた。それらはどのように囲碁をプレイしていたのだろうか。多くのプログラムは商用であるために詳細な仕組みは知ることができない。そのため、ここではGNU Goを例に挙げて説明することにする。

GNU Goはオープンソースの囲碁プログラムであり、最強の商用プログラムよりも少し棋力が低い。GNU Goは多数の複雑な評価関数を用いて着手を選択している。大まかな手順を説明すると、まずパターンデータベースにマッチする手を発見し、それらにパターンに応じた評価値を割り当てる。また、着手の目的別に候補手をいくつか生成して、それらにも状況に応じた評価値を割り当てる。着手の目的とは、自分の石を守る、相手の石を攻める、自分の領域を広げる、などである。最後に複数の評価値の依存関係などをチェックして、一番評価の高い手を決定し、その手をプレイする。これは人間のプレイヤーの思考方法と類似した方法であると言える。

しかし上述の評価値はどれも意味が異なるものである。それらの数値を合計して局面評価を行うには職人芸的な調整が必要である。GNU Goのソースコードは80,000行以上にも及ぶ。またソースコードとは別に、GNU Goはテキストファイルで52,000行を超える充実したパターンデータベースを持っている。これらのパターンは単純な石の配置のパターンだけではなく、一部の石が取れるかどうかの探索結果によってパターンにマッチするかどうか場合分けをするものも多く含む。パターンデータベースはプログラムの棋力に大きく貢献している。GNU Goは多数の開発者による長年の労力の成果であると言える。

GNU Goは19路盤ではアマ初段より少し弱く、9路盤でもほぼそれと同等の棋力を持つが、最近はそのほど

^{☆4} 日本ルールだと問題があるが、中国式的数え方なら自分の地を減らすような手を打っても問題ない。

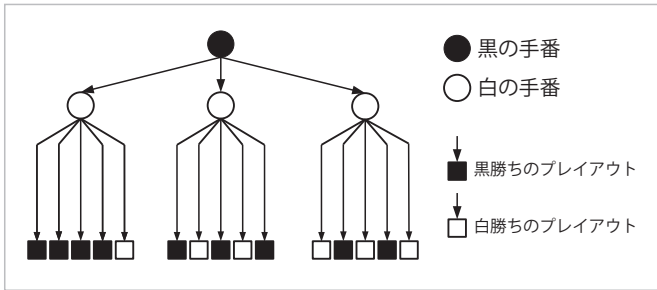


図-5 原始モンテカルロ囲碁

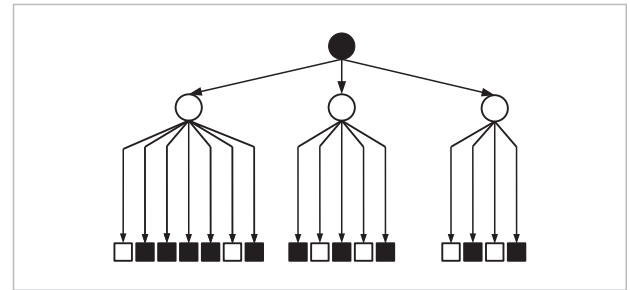


図-6 1段読みモンテカルロ木探索／有望な手に多くのプレイアウト

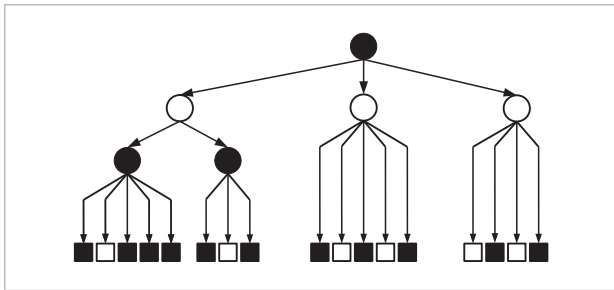


図-7 モンテカルロ木探索／木が成長する

め、相手のミスに期待した手を選択することがあるのだ。たとえば、相手が正しく応ずれば大損害をこうむるが、相手がミスをすれば大きな利益を得られるという手があるとす。正しい応手が1通りしかないような局面では、プレイアウト中に正しい手を選択される可能性は低いため、原始モンテカルロ囲碁によるプログラムは相手のミスに期待してその手を選択してしまう。そのために、深さ2以上の木に対しては、いくらプレイアウトの回数を増やしても最善でない手を選択する可能性がある。単純なプレイアウトを用いた場合に、プレイアウトの回数を増加させていくとどの程度で棋力が頭打ちになるのかを調べた論文も発表されている⁶⁾。

なお、後述する**モンテカルロ木探索**を用いたプログラムと区別をするためにこの方式を**原始モンテカルロ囲碁**と呼んでいるが、これはこの解説の便宜上付けた呼び名であり、一般的な呼称ではない。

● モンテカルロ木探索を用いたプログラム

2006年8月にトリノで開催された第11回 Computer Olympiad の囲碁9路盤部門で優勝を飾った CrazyStone およびその後登場した MoGo は、2006年から現在に至るまで、コンピュータ囲碁界の2強である。これらのプログラムが用いたアルゴリズムが、原始モンテカルロ囲碁と木探索を巧妙に組み合わせたアルゴリズム、**モンテカルロ木探索**である。

モンテカルロ木探索もまず最初は原始モンテカルロ囲碁と同様に各候補手でプレイアウトを行い、その結果得られたスコアを集計する。第1のポイントは、図-6に

示したように有望な手に多くのプレイアウトを割り当てる点である。

モンテカルロ木探索の第2のポイントは、各節点でプレイアウトが行われた回数を記録しておき、回数がある閾値を超えた場合、その手を展開し、プレイアウトを開始する節点を1つ深くするという点にある。このため、プレイアウトを繰り返すごとに図-7のように、有望と判断された部分だけ探索木が成長することになる。

CrazyStoneの開発当初は強さもそれほどではなかったそうだが、あるとき、たった1行プログラムを改良したことにより、対 GNU Go の勝率が3割台から6割以上に跳ね上がったそうである。開発者の Rémi Coulom によれば、当初利用していたスコアは、何目勝ったかのスコアそのものであった。そこを勝ち／負けの1か0かに変更したのである。結果として CrazyStone は、スコア差を最大化するのではなく勝率を最大化することを目指した打ち方をするようになった。

現在のモンテカルロ木探索による囲碁プログラムに共通する特徴として、リードしているときはひたすら安全に勝つ手を選び、逆に不利なときには無理な手を打ってくるということがあるが、その特徴は勝率を最大化するという手法により自然にもたらされているものである。

モンテカルロ木探索の理論的背景

ここではモンテカルロ木探索の代表的なアルゴリズムである、UCT の理論的背景を述べる。

● Multi-Armed Bandit

統計学や機械学習の分野で研究されてきた問題に、Multi-Armed Bandit 問題がある。Multi-Armed Bandit とは想像上の存在であり、腕が複数あるスロットマシンを意味する^{☆5}。

この問題は次のように定式化される。図-8のように複数のスロットマシンがあり、各スロットマシンは未知

☆5 One-Armed Bandit とはスロットマシンの俗称である。

モンテカルロ木探索

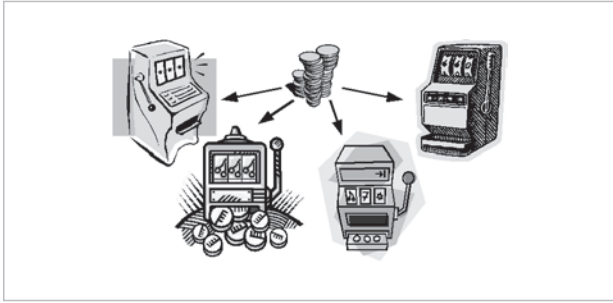


図-8 Multi-Armed Bandit 問題
どれにコインを投入すべきか

の確率分布で報酬を返すとする。目的は、与えられた枚数のコインを用いて、最大の報酬を得ることである。コインを投入するマシンを選択するための戦略を考えるのが Multi-Armed Bandit 問題である。

Multi-Armed Bandit の各マシンを各候補手とみなし、1枚のコインが1回のプレイアウトに対応するとすると、単純にすべてのマシンに同じ枚数のコインを投入して最善のマシンを発見しようとする戦略が、原始モンテカルロ囲碁の方法であると言える。

Multi-Armed Bandit 問題に対する最善の戦略は知られている⁵⁾が、計算量、消費メモリともに大きくなるため、現実の問題に用いる場合には適さないことが多い。

2002年に Auer らによって発表された UCB1 (Upper Confidence Bound) という戦略¹⁾は計算量を小さく保ちつつ、高い報酬を得ることができるものである。UCB1を用いることで、常に最善のマシンを選択した場合の報酬と実際に得た報酬の差(の期待値)が、ある値以下に抑えられることが証明されている。

この戦略は、各マシンについて UCB1 値という値を計算し、それが最大になるマシンにコインを投入するというものである。

$$\bar{X}_j + c \sqrt{\frac{2 \log n}{n_j}} \tag{1}$$

式1に示したものが UCB1 値である。 \bar{X}_j は j 番目のマシンの報酬のその時点での期待値、 n_j は j 番目のマシンにそれまでに投入したコインの数、 n はそれまでに全マシンに投入したコイン数の合計を示す。 c は何らかの係数だが、期待値の値域が $[0,1]$ の場合は $c=1$ で良い。

この式は(期待値) + (バイアス) という構成になっている^{☆6}。バイアスの部分は、そのマシンに投入されたコインの数が少ないほど大きくなる。UCB の考え方は、

- 期待値の高い所により多くのコインを投入する
- コインが少ない場合は、単に運悪く実際より期待値が低い可能性があるなのでその分を考慮して優遇するということである。

各候補手について UCB1 値を計算し、最大の値を返す候補手を選択するのが、図-6に示したような、有望な手に多くのプレイアウトを割り当てる方法に相当する。

● UCT (UCB applied to Trees)

CrazyStone の成功に刺激を受けた Kocsis と Szepesvári は、2006年に UCB1 を用いた木探索アルゴリズムを発表した。それが急速にコンピュータ囲碁界に普及した UCT (UCB applied to Trees) である⁴⁾。

UCT では根節点から順に UCB1 値の高い子節点を辿っていき、末端の節点まで到達したらそこからプレイアウトを行う。さらに、末端節点でのプレイアウトの回数が閾値を超えるとその節点を展開する(図-7参照)。

UCT を用いた場合、確率分布の時間変化がある仮定を満たす場合には、UCB1 値のバイアスにあたる部分が $O(\log n / n)$ となることが証明されている。つまり、木探索の場合でもプレイアウトの回数 n が十分多くなれば、単に期待値を元に手を選択すれば最善手に近づくことになる。

CrazyStone が最初に用いた方式では、この探索アルゴリズムがなげうまく動作するのかという理論的裏づけがなかったが、UCT は最善手にいくらでも近づけるとい意味で探索の正しさを保証するものである。

原始モンテカルロ囲碁では1手の深さしか読まないために、探索時間をいくら増やしても最善手を知ることにはできない。それに対して UCT は木が成長していくために、探索時間を十分にかけさえすれば最善手を得ることができる。MoGo 以降に登場した囲碁プログラムは UCT か、または同様に木が成長するモンテカルロ木探索を用いている。

モンテカルロ木探索の改良

CrazyStone が登場した当初は、9路盤でこそアマ3級程度の強さを持っていたが19路盤では非常に弱かった。それが2年余りの間に9路盤ではアマ高段者でも苦戦するほどになり、現在では19路盤でもアマ有段者並みの棋力を持つと推測される^{☆7}。探索部分の改

☆6 通常の統計用語のバイアスとはやや意味が異なる。

☆7 2008年3月末の時点で、KGS というネット碁サイトでは CrazyStone は1級にランクされているが、これは日本の一般的な碁会所の基準では二段程度の棋力に相当すると推測される。

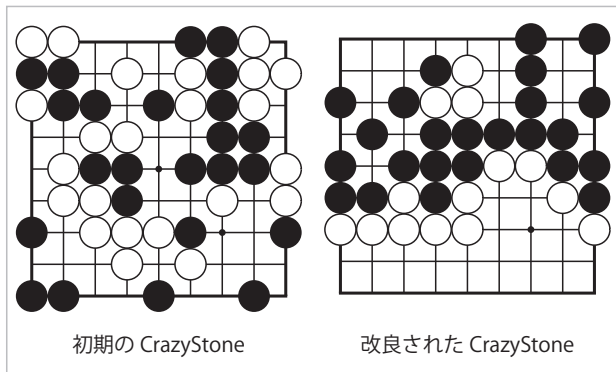


図-9 プレイアウトの比較

良とプレイアウト部分の改良の2つが組み合わせられた結果である。

● 探索部分の改良

現在の CrazyStone は探索部分に囲碁の知識を利用している。囲碁の知識を用いて良さそうな手から順に候補手をソートしておき、徐々に木に加えていく方法である。このテクニックは Progressive Widening と呼ばれている。これにより、まず良さそうな手から探索木が深くなることになり、それがダメな場合に他の手も探索されるということになる。また、最善手と思われる手はオリジナルの UCT の場合よりも優遇されるようになっている。MoGo も最新版では最善手を UCT の場合よりも優遇するようにしているそうである。

どのような改良が棋力向上につながるのか、現在まだ研究途上でありはっきりとした結論は出ていない。しかしいくつかの研究結果から言えることは、探索部分の改良の中には、次に述べるような改良されたプレイアウトと組み合わせられてはじめて効果を発揮する手法があるということである。

● プレイアウトの改良

CrazyStone の最初のバージョンで用いられたプレイアウトには、自分の眼を潰さない、あと1手で取られそうな石は逃げる、等の簡単な制約があるほかは、すべての合法手をほぼ平等な確率で選択していた。先に挙げた終局図(図-4右)はそのようなほぼランダムなプレイヤー同士によるものである。19路盤ではそのままでは非常に弱かった。

しかし現在ではプレイアウトにさまざまな改良が加えられている。人間のプレイヤーがパターンを用意したり人間の棋譜からパターンを学習するなどの手法によって、プレイアウト自体が、乱数を用いたプレイでありながらそれなりに囲碁らしい手を打つようになった。

図-9の左側は、図-4左の局面から単純なプレイアウト

トを行っている途中の図である。対して右側は、パターンの知識を使ってプレイアウトを改良した場合の途中経過を示す。明らかに右側の方が人間の評価にも近く、囲碁らしい手が多い。プレイアウトの際に用いられるパターンは、人間が手で用意した3×3のパターン(初期の MoGo³⁾)や、棋譜から自動的な学習で得られたもの(CrazyStone²⁾)等であり、GNU Go の場合のように開発者の大きな労力が必要なものではない。

パターンの制約を入れることにより、プレイアウトの速度は数分の1に落ちたそうである(もちろんプログラムによって異なるが、典型的には最近のCPUでは9路盤の場合、パターンを入れない場合で40,000回/秒程度のプレイアウトが可能であり、パターンの制約があると10,000回/秒程度と思われる)。しかし全体としての棋力は大きく向上し、19路盤でもアマ有段者と言える強さを獲得するに至った^{☆8}。

探索部分の工夫と、プレイアウトの工夫が車の両輪となってプログラムの棋力を急速に向上させつつある。

考察

● モンテカルロ木探索がなぜ囲碁に有効か

囲碁では乱数を用いたプレイヤー同士でもルールの終局と言えるところまで簡単に到達するという性質があった。そのためにこの手法が有効であったと言える。チェスなどでは今のところ、モンテカルロ木探索を用いた強いプログラムは出現していない。これは、単純なプレイアウトでは通常の終局に至るのが難しいからであると推測される。プレイアウトと探索を組み合わせるなどの手法が研究されているが、まだ大きな成果は出ていない。対して、たとえばオセロは単純なプレイアウトでも簡単に終局に到達可能なため、モンテカルロ木探索は有効だと推測される。他のゲーム、他の分野への応用はまだ研究途上であり、今後の研究が待たれる。

また、囲碁においては多くの局面で、最善手と次善手の価値の差が小さい。さらに、手順に関係なくある位置を占めておけば後々有利になるという点が多く、この性質がモンテカルロ木探索の有効性を高めているものと思われる。もちろん、囲碁においても長手順の読みが必要になる局面も多く、現在のモンテカルロ木探索ベースのプログラムの1つの弱点はそのような局面である。

● 今後の展望

最近のコンピュータ囲碁の大会では、モンテカルロ

☆8 Rémi Coulom 氏には、CrazyStone の棋力向上の過程について説明していただいた。ここに感謝の意を表する。

モンテカルロ木探索



木探索を利用したプログラムが上位を独占する状況になっている。多数の強力なプログラムが登場しており、CrazyStone、Mogoの座も安泰ではない。このアルゴリズムが有望であるということに加えて、開発にかかる労力が以前の方法と比較して大幅に小さいことも大きな理由であると思われる。探索部分とプレイアウト部分を実装すればそれなりの棋力に到達できる上に、プレイアウトの改良に必要なパターンを機械学習で得る手法が有効であることも、開発者に高い棋力を要求しないという意味で重要である。CrazyStoneの開発者のRémi Coulom氏は自分のプログラムよりも囲碁が弱いのだが、開発チームの中に、完成したプログラムよりも囲碁が強い人間が1人も居ないプログラムはCrazyStoneが初めてだろうと思われる。2006年以降、一挙に多数の研究者がコンピュータ囲碁に参入し、たった2年余りの間の進歩は目を見張るものがある。

冒頭で述べたプロ棋士との対戦では、MoGoは合計コア数256からなるクラスタシステムを用いていた。しかしモンテカルロ木探索には自明な並列性はなく、まだ決定版と言える並列化手法は確立されていない。今まさに多数の研究者がモンテカルロ木探索の並列化に取り組んでいる。また、プレイアウト部分のさらなる強化も研究されている。特に、細い一本道の手順に弱いという弱点があるが、過去の研究成果とモンテカルロ木探索をうまく組み合わせる方法が開発されればその弱点を補えるかもしれない。

モンテカルロ木探索ベースのプログラムの棋力はまだまだ向上を続けており、今のところこの進歩がどこで止まるのか分からない状況である。今後の発展を非常に楽しみにしている。

謝辞 この解説の執筆のきっかけを与えていただいた、松原仁先生に深く感謝いたします。また、はこだて未来大学の岸本章宏氏、編集を担当していただいた田中哲朗先生には多くの有益なアドバイスをいただきました。お二人の助力なくしては本稿は完成しませんでした。

参考文献

- 1) Auer, P., Cesa-Bianchi, N. and Fischer, P.: Finite-time Analysis of the Multi-armed Bandit problem, *Machine Learning*, Vol.47, pp.235-256 (2002).
- 2) Coulom, R.: Computing Elo Ratings of Move Patterns in the Game of Go, *Computer Games Workshop 2007* (2007).
- 3) Gelly, S., Wang, Y., Munos, R. and Teytaud, O.: Modification of UCT with Patterns in Monte-Carlo Go, Technical Report 6062, INRIA (2006).
- 4) Kocsis, L. and Szepesvári, C.: Bandit Based Monte-Carlo Planning, *17th European Conference on Machine Learning (ECML 2006)*, LNCS, Vol.4212, pp.282-293 (2006).
- 5) Lai, T. L. and Robbins, H.: Asymptotically Efficient Adaptive Allocation Rules, *Advances in Applied Mathematics*, Vol.6, pp.4-22 (1985).
- 6) Yoshimoto, H., Yoshizoe, K., Kaneko, T., Kishimoto, A. and Taura, K.: Monte Carlo Go Has a Way to Go, *Twenty-First National Conference on Artificial Intelligence (AAAI-06)*, pp.1070-1075 (2006).
(平成20年4月15日受付)

美添一樹(正会員)
yoshizoe@acm.org

東京大学理学部情報科学科卒業。同大学院理学系研究科情報科学専攻修士課程修了。(株)富士通研究所、中央大学研究開発機構などを経て、2008年より科学技術振興機構研究員。探索アルゴリズム、情報セキュリティ等に興味を持つ。囲碁は自称アマ三段。電子情報通信学会、人工知能学会、ACM各会員。