

6L-5

囲碁における終盤探索のための欠け眼判定アルゴリズムの開発

荒張 敬一郎、石原 孝一郎、原田 紀夫、近藤 秀文
 拓殖大学 大学院 工学研究科

1. はじめに

囲碁のヨセにおいては組み合わせゲーム理論を使うアプローチ[1]が研究されている。しかしこれには対象とする盤面の範囲が限られているなどの問題がある。

そのため本研究では組み合わせゲーム理論を使わず、縦型探索を行うアプローチでより実用的なヨセプログラムの開発を目指した。

縦型探索を行う方法には組み合わせ的爆発などの難しさがあるが、「欠け眼判定アルゴリズム」を独自に開発することで今までよりも現実的な終盤探索が行えるようになり、縦型探索のアプローチが有効である可能性を示すことができた。

2. 欠け眼判定の導入

2.1 欠け眼判定による探索効率の向上

欠け眼判定とは無駄な着手を見つけるための方法であり、この欠け眼判定によって得られた無駄な着手を候補から除外することで探索の効率を向上できる。

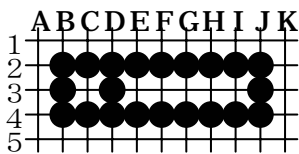


図 1 欠け眼判定の例

たとえば図 1 のように石が並んでいるとき、E3~I3 の 5 箇所について囲碁のルール上着手候補にしてしまうところを、欠け眼判定を適用すれば着手候補から外すことができるのである。もちろん、図 1 の場合だけでなく、もう少し複雑な形にも適用できる場合がある。

2.1.1 欠け眼判定の手順

① 生きていない石を取り除く

死活判定を行い、生きていない石を盤上から取り除いてしまう。このときの死活判定は、そのときに 2 眼を確保しているかどうかで判定し、将来にわたって生きるかどうかは考慮しない。

② 残った石が囲んでいる領域ごとに注目し、その領域に着手すべきかを考える

一方の色の石のみが囲んでいる領域を判定の対象とする。両方の生きている石に接している領域は対象としない。

③ 相手の石（囲んでいないほうの石）がその領域の中で生きられるかどうかを判断する

その領域の中で 2 眼を確保できるかどうかを判断するため、まず領域の中で、眼になり得ない場所を消去していく。眼になり得ない場所とは、上下左右のいずれかに相手の石があるか、斜めに 2 つ以上（盤面の隅では斜めに 1 つ以上）相手の石がある場所のことである。このような場所は周りを可能な限り囲んでも完全な眼にすることができない。

そして消去が終わったあとに残った目の数を数える。残った目が 3 目以上、または 2 目が離れているような場合には 2 眼を確保することが出来、生きられる可能性があることになる。2 目未満、または 2 目が隣り合う形でしか残らなかった場合は、その領域の中で 2 眼を作れないことになり、少なくとも生きられない。

④ 生きられない場合はその領域に着手しない

少なくとも生きられないという領域を見つけたら、その領域の中は全て着手候補から（そのときの盤面の手番や囲んでいる石の色に関係なく）除外する。

図 1 の例では、黒石の領域の中は全て眼になり得ない場所として消去され、着手候補から除外される。

消去された目を直ちに着手候補から除外するのではなく、消去した結果、2 眼を作れる可能性がない場合には、（消去されずに残った目も含めた）領域中のすべての目が候補から除外され、2 眼を作れる可能性がある場合には、（消去された目も含む）領域中のすべての目が着手候補になることに注意してほしい。

2.2 欠け眼判定アルゴリズムの拡張

2.2.1 ダメ判定

一方の色の石に囲まれていない領域（地になっていない領域）において、次の条件を両方も満たすときにダメと判定している。

- その場所が両プレイヤーにとって眼になり得ない
- 周り 8 箇所が両プレイヤーにとって眼になり得ない

欠け眼判定を利用し、このような条件にすることでより多くのダメを判定し、着手候補を減らすことができる。

2.2.2 守り手拡張（守り手による着手絞込み）

実際は無駄な着手であるにも関わらず、欠け眼判定では無駄な着手と判断されないケースが多く存在する。

そこで、探索をする前に 2 手の先読みを行い、その結果得られる欠け眼判定の結果をすべて着手から除外する守り手拡張を開発した。この方法を使えば無駄な着手をより多く見つけることができるようになる。

Development of KAKEME judgment search algorithm for the last stage in the game of Go

Keiichiro Arahari, Koichiro Ishihara, Norio Harada, Hidefumi Kondo

Graduate school of Engineering, Takushoku University

3. 実行性能の比較

欠け眼判定と守り手拡張について比較した結果を次に示す。実行環境はいずれも Windows XP Pentium4 2.4GHz VMオプション: -Xmx256m -Xms256m

また、いずれもハッシュテーブル（一度探索した盤面の結果を記憶して再利用する）を使用している。

3.1 欠け眼判定とその拡張の効果

3.1.1 実験の設定

欠け眼判定と守り手拡張の効果を調べるために、終盤探索用の簡単な問題を独自に用意して実験した。結果を表 1 に示す。いずれも探索法は NegaMax である。

K は欠け眼判定、KD は欠け眼判定と守り手拡張を使用していることを表す。また、欠け眼判定の中にはダメ判定を含んでいる。ただし、欠け眼判定を使用しなくても、簡単なダメは着手しないようにしている。

問題の並びは、欠け眼判定を使用しなくても解くことができた問題を先にしている以外に意味はない。

表 1 欠け眼判定と守り手拡張の効果 単位: 秒

問題	ナシ	K	KD	問題	ナシ	K	KD
kake01	6	5	1	kake07	272	74	0
kake02	37	5	0	kake08	-	57	0
kake03	109	25	0	kake09	-	-	3
kake04	121	62	2	kake11	-	-	85
kake05	55	51	63	kake12	-	-	52
kake06	66	2	0				

3.1.2 実行結果の検討

多少の違いはあるものの、欠け眼判定は確かに効果が出ている。また守り手拡張については多少遅くなる場合もあるものの、非常に大きな効果があるといえる。特に、守り手拡張を使わなければ解けない問題がある。

3.2 NegaMax 以外の探索法での効果

表 2 NegaMax 以外の探索法での効果 単位: 秒

問題	NegaScout		MTD(0)		MTD(bi)	
	ナシ	K	ナシ	K	ナシ	K
kake01	5	4	4	4	5	5
kake02	27	4	26	3	33	4
kake03	70	20	55	17	81	21
kake04	113	62	86	50	100	54
kake05	53	51	33	33	39	41
kake06	58	2	56	2	60	2
kake07	243	57	157	41	161	43

3.2.1 実験の設定

3.1 の問題のうち、欠け眼判定を使用しなくても解けた問題 7 問を選び、NegaScout、MTD(f)、MTD(bi) の探索方法で欠け眼判定の効果を試してみた。MTD(f) の初期値は 0、MTD(bi) は解の範囲を -64~64 で初期値を 0 にしている。実験の結果を表 2 に示す。

3.2.2 実行結果の検討

kake01 と kake05 であまり効果が現れなかったのは 3.1 と同様であるが、3 つの探索方法それぞれで確かに効果があることがわかる。

3.3 ヨセプログラムとしての実力

3.3.1 実験の設定

欠け眼判定と守り手拡張を使用したプログラムに、広い領域を着手候補から除外する機能を組み合わせたプログラムで、文献[1]の巻末付録に掲載されている問題集（全 22 問）を解かせた。そのうち、解くことができた問題についての結果を表 3 に示す。

表 3 問題集の問題 単位: 秒

問題	時間	問題	時間
MathGo_01	7	MathGo_10	3925
MathGo_02	137	MathGo_14	190
MathGo_06	448	MathGo_20	15639
MathGo_08	250		

3.3.2 実行結果の検討

22 問中 7 問を解くことができた。いずれも 9 路盤である。また、他の問題が解けない理由としては

- 盤面が広すぎるため探索空間が広がりすぎる
- 探索空間が大きすぎてハッシュテーブルに記憶する量が実メモリの容量よりも大きくなる
- 石を捕獲することで探索空間が収束しにくくなるなどが挙げられる。

4. おわりに

欠け眼判定と守り手拡張アルゴリズムは大きな効果を示すことが確認できた。特に守り手拡張については 2 手の先読みしか行っていないので、引き続き検討していくことでさらに大きな効果が期待される。

また、本研究では探索の際に候補手の並び替えを行っていないので、候補手の並び替えをすることでより良い結果が期待できる。

参考文献

- [1] エルウィン・バーリカンプ/デビッド・ウルフ 著、吉川竹四郎/小林美治/石原孝一郎 訳：囲碁の算法—ヨセの研究—、株式会社トッパン(1994)