

逆算法による詰碁問題の生成方式

上西 知陽[†] 中村 克彦[‡]

[†] 東京電機大学大学院理工学研究科

[‡] 東京電機大学理工学部

1 まえがき

コンピュータ将棋やコンピュータチェスでは着手決定を行う際、駒の配置や役得などから局面に数値的な評価を与える評価関数と Mini-Max 探索を組み合わせることで局面の評価を行っている。囲碁は探索空間が大きいので、Mini-Max 探索のような動的解析を行うことが非常に難しい。最近ではモンテカルロ法を用いた UCT (Upper bound Confidence for applied to Trees) を用いることによりこの問題の解決がはかられ格段に棋力が上がり、現在ではアマチュア最高レベルに達したと言われている。

われわれはコンピュータによる詰碁の解法とこれを応用した詰碁問題の作成に取り組んでいる。詰碁は、一般に正しい手はひとつだけであるため、モンテカルロ法は必ずしも適していないと考えられる。詰碁問題では標準的な棋力をもつ人が読みをせずに判定できる死活パターンを基本として「手詰め」の問題などと呼ばれる。このため人と同程度の静的解析による死活判定を実現することが第 1 の目標である。

コンピュータ将棋では詰め将棋の問題を作成するプログラムなどが作成されているが、コンピュータ囲碁では詰碁プログラムはつくられているが、詰碁を作成するプログラムはあまり見られない。この報告書では静的に死活を判定するための、平面グラフに対するオイラーの公式を応用した目の推定や、数値的パラメータを用いた死活判定などの方法と逆算法による詰碁問題作成方法について述べる。

2 ブロックとグループ

ブロックとは、図 1 のような独立した石、もしくは同色の石が隣接した極大集合のことである。囲碁では打ち上げの際や、石の結合などをすべてブロック単位で処理を行っているのでブロック管理が必要となる。

グループとは、相手の石によって切断されること

ないブロックの極大集合のことである。(例：図 2 の黒石) ブロックが相互にコスミの関係によって接続されていて静的に容易に判別できる場合もあるが、一般には静的な判定は困難であり、厳密な判定は動的な判定を必要とする。

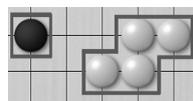


図 1: ブロック

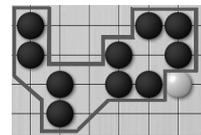


図 2: グループ

3 オイラーの公式に基づく目の数の推定

オイラーの公式によれば、連結平面グラフの k 個の辺の数、 n 個の頂点の数に対して、辺で囲まれた領域の数は $N = k - n - 1$ で求められる。これを、次のように応用し囲碁における閉領域の個数を求めることができる。

$$N = E - V - C + 1$$

ここで N は領域数、 E は辺の数、 V は石の数、 C は三角団子の数である。三角団子とは、図 3 の形のように 3 つの石が連結したものである。

盤面の隅で死活判定を行う際、図 4 のように辺に接して空点ができる場合がある。その場合、盤外に earth という仮定の石をあると仮定し、石が earth と接続されているとして、リンクや石の数を数えて計算する。図 4 の例では、石の数が 10 個、辺が 16 個、三角団子数が 5 個であることから、 $N = 16 - 10 - 5 + 1$ となり、このグループの閉領域は 2 個であることがわかる。

4 数値的パラメータを用いた死活判定

人間が視覚的に判定できるような閉領域を含むグループの死活は、先読み探索でも判定できるが、一般に多くの探索時間を要する。

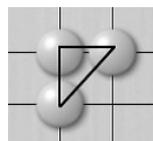


図 3: 三角団子

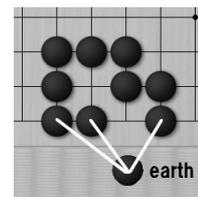


図 4: earth を用いる例

Generation of Life and Death Problem by Retrograde Method

[†]Graduate School of Science and Engineering, Tokyo Denki University

[‡]School of Science and Engineering, Tokyo Denki University

[†]11rmj10@ms.dendai.ac.jp, [‡]nakamura@rd.dendai.ac.jp

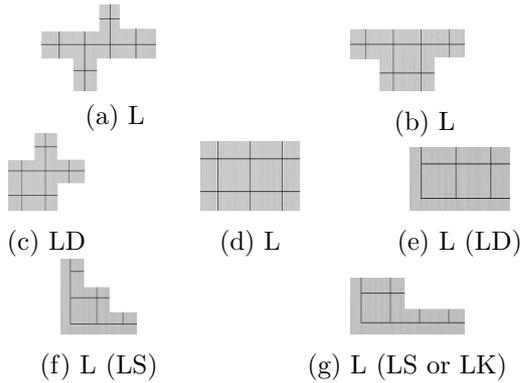


図 5: 大きさ 6 の主な閉領域とその死活

この死活判定においては，閉領域の形状とその中にある相手の石の配置が問題になる．閉領域の形状を識別するためには，領域中の各点間の隣接数，一点についての最大隣接数などのパラメータを用いる．これらによって曲り四目，自動車 (5 目)，花六目などと呼ばれる形状を含む 7 目以下の形状を識別できる．

グループの死活を，L：無条件生き，D：無条件死，S：セキ (生き)，K：コウなどの記号で表し，これらを組み合わせて，LD：先手無条件生き・後手死，などを表す．他の記号の組み合わせ LS，SD，LK，KD，SK，KS も同様に定義される．

図 5 は大きさ 6 の代表的な閉領域の形を囲む石を省略して示している．カッコ内は外ダメが 0 のときの死活である．これらは，閉領域の隣接数と最大隣接数に加えて，領域中の端点の個数および領域の外縁の個数から識別される．隅の領域 (e)，(f)，(g) は端の点を 4 または 5 個含むことで識別され，また (f) は外縁の個数が 4 であることから識別される．これらの形は，これを含むグループの外ダメの数が 0 のときには生きとはならないので区別が必要である．

閉領域が敵石を含む場合の死活は，敵石の数，その最大ダメ数などの数値的特徴によって判定できる．さらにセキやコウが関係しているときにはグループの外ダメも必要となる．これらの数値的特徴によって，大きさが 6~7 目の，敵石を含む閉領域 (辺と隅を含む) のセキ及びコウを含む多くの死活パターンが識別できる．

5 逆算法を用いた詰碁問題の作成

逆算法とは，詰め将棋の問題を作成する際にも用いられる手法の一つで，既存の問題や，ある場面から状況を逆算していき，途中で石を取り除くなどして問題を作成する手法である．詰碁問題では，正解が必ず存在，初手は一通り，正解の手は一通りという条件が必要である．この条件を元に逆算法に基づき詰碁問題を作成する．詰碁問題を作成するにあたっての手順の例

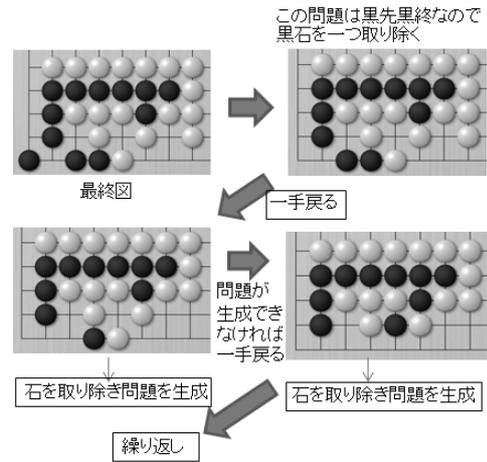


図 6: 逆算法を用いた詰碁作成の手順

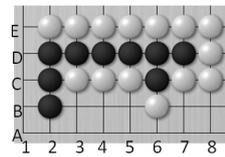


図 7: 詰碁問題

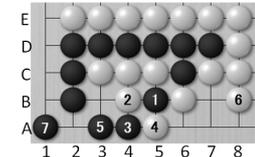


図 8: 詰碁問題の解答

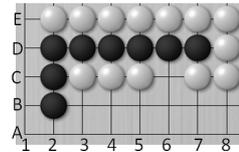


図 9: 逆算法を用いて作成した詰碁問題

を図 6 に示す．この手順を図 8 が解答である図 7 の問題に対して用いると，図 9 のような問題が生成できる．図 9 は解答である図 9 の黒番 1 手目までさかのぼり，そこから，黒石は C6，白石は B6 の 2 石を取り除いた形になる．ノゾキに対するツギ，オサエなどの絶対手の部分，連結している石，グループを接続している重要な石を取り除き，問題を作成する手法をとっている．

6 むすび

静的解析をより活用した詰め碁プログラムを作成した．また，それを用いて，逆算法に基づき既存の詰碁問題から新たな詰碁問題を生成する方式について述べた．

参考文献

- [1] K.Nakamura, Static analysis based on formal models and incremental computation in Go programming, 2005
- [2] 中村 克彦, 木戸間 周平: 数値的な特徴に基づく囲碁局面パターンの解析, 情報処理学会論文誌, Vol.43, pp.3-4, 2002