

CEDEC 2010 超速碁九路盤 AI対決 まとめ

朝倉 順子 (スクウェア・エニックス)

2010年 10月

1 はじめに

本稿では、CEDEC 2010 超速碁九路盤 AI対決に向けて行ってきた取り組みと大会における対局を通じて得られた知見について述べます。超速碁九路盤 AI対決は「CEDEC CHALLENGE」の企画の一つであり、コンピュータ碁を題材とした大会です [7]。大会のねらいは一手一秒という高速な碁 AIの開発を通じたビデオゲーム開発にかかわる技術力の向上および人材発掘の促進です。私たちは布石や定石などの碁特有の知識の利用を可能な限り回避することで、より一般的な AIプログラムを作成することを目指しました。また計算環境と制限時間を考慮して、高速な候補手探索の手法を採用しました。

2 先行研究

人工知能の分野では碁プログラムに対するさまざまな研究が行われてきました。近年、とくに注目されているのがモンテカルロ碁です。モンテカルロ碁以前は主流であった評価関数を用いた手法などとモンテカルロ木探索を組み合わせることで急激に棋力が向上し、CrazyStoneをはじめ、さまざまな強力なモンテカルロ碁が発表されました [3,4]。また2009年には、MoGo [5] が九路盤においてプロ棋士との対局に互先（ハンディキャップなし）で勝利しました。

しかしながら、モンテカルロ木探索はクラスタ等の大規模計算環境に適した手法です。さらに与えられる思考時間が短いほど、勝率が低下する傾向にあります [5]。したがって、私たちはCEDEC CHALLENGEのように通常の計算環境かつ超高速な着手時間での対局にはモンテカルロ木探索は不向きであると考えました。

高速な着手に注目した研究の一つにChenによる探索アルゴリズムがあります。これは詰碁を解くことを目的として提案された手法ですが、候補手生成の論理は実戦においても有効であると考えられますので、私たちはChenの手法を候補手の生成手法として採用することにしました。

3 実装した碁プログラムについて

本節では、実装したプログラムで採用した既存手法の influence の評価手法 [1] およびChenの手法 [2] について説明します。

3.1 Influence の評価手法

盤上の石の間には見えない「つながり」があると考えられます。この「つながり」を数値化したものが influence です。Bouzy は influence を morphology(数理形態学)の演算により計算し、計算結果を Zobrist モデル [6] と組み合わせることで territory (地) の評価を行いました。盤全体の influence を計算したものは Bouzy map として知られていて、現在あるプログラムの多くで囲碁プログラムで形勢判断などに利用されています。

3.1.1 Morphology の演算

Influence の計算に用いられる演算は dilation と erosion です。

- dilation : dilation は morphology における「足し算」です。与えられた集合に、集合の境界からある範囲の近傍を加える演算です。
- erosion : erosion は morphology における「引き算」です。与えられた集合から、集合の境界からある範囲の近傍を取り除く演算です。

Influence の計算では、dilation を複数回連続で行った後、dilation に応じて決定された回数だけ erosion を行います。[1] では、erosion の回数 e を $e = d(d - 1) + 1$ により決定しています。ここで d は dilation の回数とします。

3.1.2 Zobrist モデル

Zobrist モデルは influence の認識方法の一つです。はじめに黒石が置かれた点に+64、白石が置かれた点に-64、それ以外に 0 を割り当てます。それから、正の値を持つ点は周囲に+1 を伝播し、負の値を持つ点は周囲に-1 を伝播する操作を 4 回くり返します。

3.1.3 盤上の influence の計算

上の方法を組み合わせて盤上の各点における influence を計算する手順を以下に示します。

1. 盤上の各点に対して、黒石が置かれた点に+64、白石が置かれた点に-64、それ以外の点に 0 を割り当てる
2. 正数、負数の dilation をそれぞれ行う。すなわち、正数をもつ点の近傍には+1、負数をもつ点の近傍の点には-1 を伝播させる。ただし、各点に割り当てられる値はそれぞれ $[0, 64]$ 、 $[-64, 0]$ の範囲を越えないものとする。また、伝播先の influence が異符号であったら、伝播させない
3. 正数、負数の erosion をそれぞれ行う。すなわち、正数をもつ点の近傍には-1、負数をもつ点の近傍の点には+1 を伝播させる。ただし、各点に割り当てられる値はそれぞれ $[0, 64]$ 、 $[-64, 0]$ の範囲を越えないものとする。また、伝播先の influence が異符号であったら、伝播させない

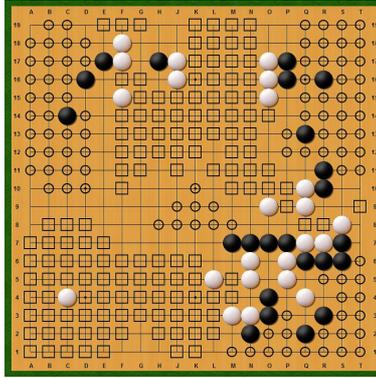


図 1: Bouzy Map

この手順により正数および負数が割り当てられた点をそれぞれ黒および白の territory とみなします。 $d = 5$ 、 $e = 21$ の Bouzy Map による territory の認識結果を図 1 に示します。 Bouzy Map により白および黒の territory と判断された空点をそれぞれ四角および丸で示しました。

3.2 Chen の手法

Chen の手法は詰碁 (つめご) を解くことを目的として提案された手法です。同色の石が形成するかたまりの周囲の liberty (ダメ) の概念を 3 次までに拡張し着手の候補点の評価に利用したり、周囲の liberty を多く持たない相手の連の数で調節を行ったりするなど、ヒューリスティックな探索を行う点が特徴的な手法です。

3.2.1 Chain の定義

Chen の手法で必要となる要素に "chain" があります。chain とは、同色の block により形成された、相手によって分断することのできない集合のことです。

Definition 3.1 二つの *block* が $C1$ から $C3$ のいずれかを満たすとき同一の *chain* に属すると定義する。

$C1$ 二つ以上の共通の *liberty* をもつ

$C2$ 一つの共通の *liberty* をもち、それが "protected" である、ここで空点が "protected" であるとはその点が相手にとって着手禁止点であるか相手が着手した後に取り返すことができる点である (例えば相手はその空点にコウやウツェガエシで再度打つことができないなど)

$C3$ 共通の相手の死んだ *block* に隣接している

chain の例を図 2 に示します。図 2 において盤上の白石および黒石はそれぞれ一つの chain を形成しています。

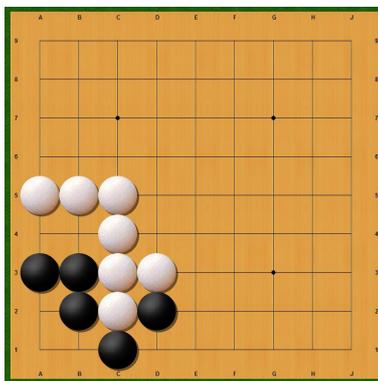


図 2: Chains

3.2.2 Position Evaluation

各点の価値は、通常の liberty である 1 次 liberty に加えて、2 次および 3 次 liberty を定義して、それらの総数から計算される基本評価値を基に計算されます。

Definition 3.2 *block* に水平または垂直に隣接する空点を 1 次 liberty と定義する。*block* に仮想的に一つ石を加えてできる *block* の liberty を 2 次 liberty と定義する。加えた石により他の連に接続することも許す。同様に、仮想的に二つ石を加えてできる *block* の liberty を 3 次 liberty と定義する。

n_1 、 n_2 および n_3 をそれぞれ 1 次、2 次および 3 次 liberty の総数とすると、基本評価 V は

$$V = n_1 \times 16 + n_2 \times 8 + n_3 \times 4$$

で与えられます。この基本評価 V に周囲の liberty を多く持たない相手の連の数に応じた調整やシチョウの追い・逃げを考慮した評価を加えて、最終的な評価値を得ることができます。

3.2.3 Candidate Move Generation

上で述べた評価に基づき、chain の 1 次および 2 次 liberty から 4 手、隣接する相手の block の liberty から 2、3 手、周囲の chain に含まれない自分の block の liberty から 2 手、候補手として選出します。さらにその他に相手の石を捕獲できる手などを加えた候補手を優先度に基づき並べ替え、候補手リストを作成します。

3.3 アルゴリズム概要

以上の手法を組み合わせて実装した囲碁プログラム”lchan”のアルゴリズムを以下に示します。

プログラムコード名	勝敗	順位
nomitan	22 - 2	1
Tombo	21 - 3	2
Kasumi	18 - 6	3
Hope	18 - 6	3
Impromptu	16 - 8	5
fudogo	15 - 9	6
Rock	14 - 10	7
eTalk	12 - 12	8
kiwosakada	8 - 16	9
kisaragi	7 - 17	10
lchan	5 - 19	11
mnagaku	2 - 22	12
PB002	0 - 24	13

表 1: 予選結果

Algorithm of lchan

1. Make a legal moves list
2. Check atari stones
3. Check stair capturing stones
4. Calculate influence on the current board by using Bouzy map
5. Select a move by the above diagnosis
6. Form crucial chains
7. Generate candidate moves by Chen's method

4 大会の結果

4.1 対戦結果

予選および決勝の対戦結果を表 1 に示します。私たちのプログラムコード名は”lchan”です。表 1 に示す通り、私たちのプログラムは 5 勝をあげました。直前の不具合の為に、実装したアルゴリズムを一部を削除した状態での対局としては健闘したのではないかと思います。興味深い結果は対 Tombo 戦で、パスをせずに相手の地に石を打ち続ける lchan に対して Tombo がパスの対応を間違えたことで得られた一勝です。これはヒトの常識を持たない囲碁 AI 同士の対局ならではの結果であるといえます。

	黒番		白番	
準決勝 第一回戦	Hope	×	—	nomitan
準決勝 第二回戦	Tombo		—	Kasumi ×
決勝	nomitan		—	Tombo ×

表 2: 決勝トーナメント結果

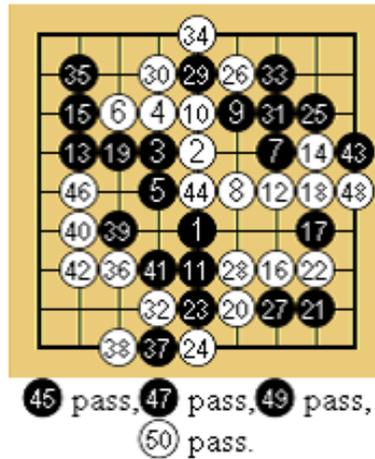


図 3: エキシビションの結果 (黒番：nomitan, 白番：万波プロ)

4.2 決勝トーナメント・エキシビション

決勝戦は予選上位の4チームによりトーナメント方式で行われました。対戦結果を表2に示します。表2に示すとおり、予選を一位通過したnomitanが優勝しました。

決勝戦終了後に、エキシビションマッチとして優勝プログラムのnomitanと万波佳奈プロ四段による対局が行われました。黒番がnomitan、白番が万波プロです。この対局の棋譜を図3に示します。エキシビションでは一手一秒のルールは廃止されましたが、万波プロは手早くnomitanを全滅させてしまいました。万波プロ、王プロ共に序盤の数手で既に終局を見通しており、ヒトとAIとの大きな差を実感させられたエキシビションでした。

5 おわりに

本稿では、CEDEC 2010 超速碁九路盤 AI 対決に向けて行ってきた取り組みと大会における対局を通じて得られた知見について述べました。一手一秒という高速なルールにおいても、モンテカルロ囲碁の強さは十分に他を圧倒していました。今後もモンテカルロ木探索を基本として、与えられた計算環境下でいかに探索を最適化できるかが勝敗を分けるのではないかという印象を持ちました。

一方で、モンテカルロ囲碁であってもプロ棋士との実力差は明らかでした。現在のモンテカルロ囲碁は純粋な試行反復に、探索の効率化、囲碁特有の知識の活用などが加えられています。それらの要素をより発展するべきか、それとも全く新しい概念を取り込むべきかなど、依然として多くの興味深い課題が残されています。

参考文献

- [1] B. Bouzy, Mathematical morphology applied to computer go, IJPRAI **17**(2), 2007, pp. 257-268.
- [2] K. Chen and P. Zhang, A New Heuristic Search Algorithm for Capturing Problems in GO, CG, 2006, pp. 26–36.
- [3] R. Coulom, Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search, CG, 2006, pp. 72–83.
- [4] M. Enzenberger and M. Müller, A lock-free multithreaded Monte-Carlo tree search algorithm, Advances in Computer Games **12**, 2009.
- [5] S. Gally, Y. Wang, R. Munos and O. Taytaud, Modification of UCT with Patterns in Monte-Carlo Go, Technical Report **6062**, INRIA, 2006.
- [6] A. Zobrist, A model of visual organization for the game of Go, AFIPS Joint Computer Conferences, 1969, pp. 103-112.
- [7] <http://cedec.cesa.or.jp/2010/event/challenge/ai.html>.

付録 A Pseudocode of lchan

1. Make a legal moves list
FOR $p \in \{\text{board positions}\}$ and $\text{color} \in \{\text{BLACK, WHITE}\}$
is_legal_move()
Input p, color
IF
($p \neq \text{NO_STONE}$)
&& (!is_superko(p, color))
&& (!is_suicide(p, color))
THEN
RETURN TRUE
ENDFOR
2. Check atari stones
FOR $p \in \{\text{legal moves of board positions}\}$ and $\text{color} \in \{\text{BLACK, WHITE}\}$
is_atari_point()
Input p, color
IF
num_liberties(p, color) == 1
THEN
RETURN TRUE
ENDFOR
3. Check stair capturing stones
FOR $p \in \{\text{atari stones' positions}\}$ and $\text{color} \in \{\text{BLACK, WHITE}\}$
is_stair_point()
Input p, color
IF
stair_captured(p, color) == TRUE
THEN
RETURN TRUE
ENDFOR
4. Calculate influence on the current board by using Bouzy map
Influence()
Input the board state
FOR $i = 1$ to d DO
dilate()
ENDFOR
FOR $i = 1$ to e DO
erase()
ENDFOR
5. Select a move by the above diagnosis
select_init_candidate()

```

Input: state  $\in$  {diagnosis} and color  $\in$  {BLACK, WHITE}
select a move s.t.
avoid (normal or stair) capturing
maximize color's influence
RETURN move

6. Form crucial chains
FOR color  $\in$  {BLACK, WHITE} and b1  $\in$  {blocks of color}
FOR b2  $\in$  {blocks of color} \ b1
IF cond_C1(b1, b2) || cond_C2(b1, b2)
THEN
add b2 to the chain which includes b1
ENDFOR

7. Generate candidate moves by Chen's method
generate_candidates()
Input information about crucial chains
FOR p  $\in$  {1st and 2nd liberties of a crucial chain} V = evaluation_value(p)
ENDFOR
add 4ps which have large V to the candidates list CAND_LIST
add 2 or 3 liberties of relevant hunter blocks
add 2 liberties of relevant prey blocks
RETURN CAND_LIST

```

探索部分は大会に提出したプログラムには未実装ですが、一手一秒という制限時間の短さを考慮して、はじめにシチョウやアタリ、そして influence の総和から選出された候補手を確保した後、制限時間いっぱい探索を行う計画でした。

chain の形成では死石の判定を避けるため、chain の定義の C1 と C2 のみを用いて block が属する chain を判断しました。また C2 において相手が空点を取り返す可能性の確認は省きました。

探索の対象とする chain は相手の直前手が属する chain のみとしました。

また、探索打ち切り時間、influence の morphology 演算の実行回数および Chen の手法内部のパラメータは実験的に決定しました。