摘要

コンピュータ囲碁は，コンピュータチェスに変わる人工知能・知識工学の研究対象のひとつとして注目を浴びている．その理由は，コンピュータ囲碁がチェスに比べて，可能手が膨大な数であることと，各局面の優劣を定義し評価関数で表現することが困難であるという問題を持つからである．本研究は，9路盤を対象としたコンピュータ囲碁システムを実際に作成し，システム全体から見たときのコンピュータ囲碁が抱える問題点を明らかにすることを目的としている．本論文では，実際に作成したシステムの全体構成，データ構造を説明し，その後，システムの各部について具体的な実装方法を述べている．システムは，対局に使用される状況判断部・候補手選出部・着手決定部と対局前に知識を獲得する役割を持つ事例獲得部の4つの主要部分からなる．最初の状況判断部は，局面データを受け取ると石のグループ化を行い，その後死活判定や各マス目がどちらの勢力に属しているかなどを判定し結果を候補手選出部に送る．候補手選出部は，膨大な数の可能手の中から数手の候補手を選択する役割を持つ．本システムでは，この候補手の選出に事例ベース推論の考え方を応用している．つまり，可能手の価値を求めるために，事例ベースの中からよく似た手を見つけだし，それを基に手の価値を決定している．事例ベースには，プロ・上級アマの実際の棋譜から獲得した事例が入っている．着手決定部では，選ばれた候補手を枝にして木探索を行い，最良の一手を選び出す部分である．事例獲得部は，候補手選出部で使用される事例を棋譜から自動的に獲得する役割を持つ．事例は，勢力パターン，評価改善量，及び1次検索用パターンからなる．勢力パターンは，事例に示される手が打たれたときの局面を石の影響力から計算される数値で表した5×5のパターンであり，可能手と事例の類似性を求めるのに使用される．評価改善量は，可能手の価値を求めるのに使用される数値である．1次検索用パターンは，事例の分類に使われる3×3のパターンである．本論文では，事例の形式以外にも，事例の獲得方法について述べ，さらに獲得された事例についての分析も行っており，棋譜に瀕出する形を示している．盤面の価値判断は評価関数で行っており，主に死活と地を基準に評価を決定している．本論文では作成したシステムを基に候補手選出に関して実験を行っている．実験は，実際にプロ・上級アマが打った手が候補に含まれるかを事例獲得に未使用の棋譜に対して行った．候補手の選出方法は，事例と形が完全に一致するものを候補とする，事例との類似性を表す事例有効率が高いもの上位5手を候補とする，事例を基に手の価値を求めその上位5手を候補とするの3つの方法で行った．結果は，形が完全に一致するものより，事例有効率を基に候補手を5手選んだものの方が性能が良く，序盤で53.5%，中盤で35.2%，終盤で47.2%の割合でプロ・上級アマの手を候補に含んでいた．手の価値を求める方法は事例有効率とほぼ同じ評価であり，期待される性能を発揮することはできなかった．これについては，手の価値計算に用いるパラメータが適切な値になっていないことが原因であった．各実験の分析の結果から，手の価値を求める方法には，性能が向上する余地があることが分かったので，今後の課題は，着手決定部の評価関数と合わせて，適切な価値算出方法を決定することである．

目 次

1 はじめに **1**

2 コンピュータ 9 路盤囲碁システムの全体構成 **4**

2.1 システム構成 4

2.2 データ構造 6

3 状況判断 **10**

3.1 石群の形成と評価 10

3.1.1 グループ化 10

3.1.2 勢力計算 12

3.2 死活判定 16

3.2.1 眼形候補による死活判定 16

3.2.2 木探索による死活判定 18

4 候補手選出および着手決定 **19**

4.1 候補手の選出に用いる事例 19

4.1.1 事例の形式 19

4.1.2 事例ベースの構成方法と分類 20

4.1.3 事例の獲得方法 22

4.1.4 獲得された事例の分析 24

4.2 事例検索による候補手選出方法 25

4.2.1 事例ベースにおける検索範囲の決定 25

4.2.2 手の価値の決定方法 30

4.3 評価関数による着手決定 31

4.3.1 評価関数 31

4.3.2 着手決定方法 34

5 実験および考察 **36**

5.1 実験 36

5.1.1 実験方法 36

5.1.2 実験結果 37

5.2 考察 43

6 おわりに **46**

図目次

1 囲碁システムの全体構成 4
2 囲碁システムの構成要素 5
3 連と群 8
4 群の大きさ 8
5 コスミ 10
6 コスミの形において切断となる特殊条件 11
7 一間トビ 11
8 一間トビの形に置いて切断となる例 11
9 ケイマと大ゲイマ 12
10 二間トビ 12
11 石の影響力 13
12 大ゲイマ間の影響力 13
13 盤の端を利用した味方領域の確保 14
14 盤の端からの影響力 14
15 局面 A 15
16 局面 A の勢力計算例 15
17 眼による死活判定 16
18 眼形候補による死活判定の全体の流れ 17
19 眼形候補決定の過程 17
20 事例の形式 19
21 事例の具体例とその獲得時の局面図 20
22 序盤の事例分類 21
23 中盤の事例分類 21
24 終盤の事例分類 21
25 1 次検索用パターンのコード変換方法 23
26 同一の 1 次検索用パターンの種類 24
27 序盤において出現頻度の高かった形 26
28 中盤において出現頻度の高かった形 27
29 終盤において出現頻度の高かった形 27
30 検索範囲の決定方法 28
31 回転および左右対象を含めた検索範囲の決定方法 29
32 ハッシュ表が二通り以上の変換を要求する例 29
33 手の価値を求める手順 30
34 手の価値算出の具体例 32
35 死活が不定の群の得点 33
36 着手決定の様子 34
37 木の各節点における評価 35

38 候補手数による平均正答率の変化 43

表目次

1 システム全体を通して使用されるデータ構造 7
2 事例ファイルに記録される項目 22
3 獲得された事例数 24
4 死活判定による得点 33
5 “地とアゲハマ” の計算に用いられる重み 34
6 形の完全一致による候補手選出結果 38
7 事例有効率に基づいて候補手を5つ上げた場合の実験結果 39
8 事例有効率に基づいて候補手を3つ上げた場合の実験結果 40
9 事例有効率に基づいて候補手を7つ上げた場合の実験結果 41
10 手の価値に基づいて候補手を5つ上げた場合の実験結果 42

1　はじめに

最近，日本国内において，コンピュータ囲碁・コンピュータ将棋などゲームプログラミングの研究が年々盛んになってきている．例えば，コンピュータ囲碁では，80年代頃から本格的な研究が始められ，世界大会でまずまずの成績を上げるプログラムが作られるに至っている．また，ゲームプログラミング専門のワークショップも開催されるようになり，システムの提案，詰碁の研究，および認知科学的側面からの分析など様々な研究が発表されている．

一方，外国ではもっと古くから研究が行われており，また非常に盛んである．コンピュータ囲碁を例にあげると，古くは60年代にRemusが囲碁の好手や悪手の学習に関する研究を行っている．そしてそれ以後，パターン認識や数学的な解析など様々な研究が行われ，80年代に入り，商用になったり，現在でも活躍しているコンピュータ囲碁プログラムが出現するに至っている[1]．また最近では，インターネット上にさまざまな囲碁関連のリソースが置かれ，研究の活性化に貢献している[2]．

このようにゲームプログラミングの研究が盛んになってきた理由は，これらが人工知能・知識工学の格好の研究材料であるからである．すなわち，ルールが非常に単純であるためコンピュータに乗せやすく，研究の達成度が具体的にわかるということが上げられる．実際，ゲームプログラミングの研究（特にチェス）を通して，ミニマックス法，α-β探索法，反復深化，B^*探索など，人工知能の様々な手法が提案されてきた[3]．

ここで，チェスに代わる次の研究対象として注目を浴びている囲碁について，ルールを簡単に説明しておく．囲碁は簡単にいうと陣取りゲームであり，ルールの概要は以下の通りである．

- 19×19，13×13，あるいは9×9のいずれかの大きさの碁盤を用いる．一方が黒石を持ち，他方が白石を用いる．
- 対局は，お互いが交互に盤面にひとつずつ石を置いていく．ただし，置いた石は動かせない．
- 相手の石を，自分の石と盤の端で囲めば，それを取ることができる．取った石は，その数だけ自分の得点になる．
- 置いた瞬間，相手に取られてしまうような手は禁止される．ただし，その手が相手の石を取る手の場合は許される．
- 石および盤の端で囲った領域（これを“地”と呼ぶ）は，その囲ったマスの大きさに応じてゲーム終了時に得点となる．
- 最終的に，お互いがこれ以上得点を稼げないような状態になるか，一方が投了したときゲームは終了となり，多くの得点を稼いだ方が勝ちとなる．

他にも細かなところでは，後手にハンディキャップを与えるコミ，千日手を避けるためのコウ，眼がなくても死なないセキなどのルールがあるが，囲碁のルールは以上のように簡単である．また，ルールではないがルールから導かれる重要な法則があるので説明しておく．それは，眼が二つ以上ある石は生きており，相手から取られることはないというものである．眼とは，味方石と盤の

端で囲われた空白のマス目からなる閉領域である．囲碁では，交互に石を置いていくので，二つの眼を同時に埋めることはできない．よって，眼が二つ以上あるときは，生きとなる．

コンピュータ囲碁が注目を浴びている理由は，これまでの手法が通用しないからである．これは，以下の二つが大きな原因である．

1. 各局面に対する可能な手が膨大である．
2. 各局面の優劣を定義し，評価関数で表現することが困難である．

前者は，ルールをみれば分かるとおり，囲碁には手に対する制約がほとんどない．いい替えれば，可能手は盤面の空白のマス目の数にほぼ等しい．そして，ゲームが進んでも，可能な候補手が急激に減少するということはない．9 路盤について，考えうるゲームの進行手順は単純に計算すると，$(9 \times 9)!$である．もちろん，これは回転などによって同一の盤面になる可能性や，戦略的に絶対にあり得ない展開が含まれるが，一つの目安として計算してみる．スターリングの公式を使って近似解を求めると，$(9 \times 9)! \approx 5.79 \times 10^{120}$ となる．19 路盤では，当然これより更に大きな数となるので，コンピュータ囲碁において可能手の大きさが如何に問題であるかが分かる．序盤は定石を使用できるとはいえ，可能な手がほとんど減少しないことを考えると，定石を外れた瞬間から膨大な探索が必要になってしまう．よって，いかに少数の有意義な候補手を選び出すかということになる．これについては，パターンマッチングが有効であると考えられる．囲碁では，ある形（石の配置）が現れると，その形に応じた特定の場所に打つべきであるということが多い．これは，定石だけでなく手筋と呼ばれるものが存在していることからもいえる．そこで，このような“形”と“それに対応してどこに打つべきか”をあらかじめデータベースに蓄えておき，実際の局面とマッチングさせることで候補手の選出に役立てる方法は有効であると考えられる．他にも，ルールを用いて人間の思考のように，有意義な候補手を選び出すという方法もある．

後者については，専門家の知識をいかにうまく獲得し表現するかという問題であり，時間をかけて改善していくしかない．

現状では，システムの個々の部分についての研究が多く行われている[4][5]．システム全体について評価を行った研究もあるが，非常に少ないという状態である[6][7]．そこで，本研究では，コンピュータ囲碁システムの抱える問題点を明らかにすることを目的とし，最初から最後までを通して打てるシステムを構築することにした．囲碁は，9 路盤であろうと 19 路盤であろうと同じ感覚で打て，考えなければならない概念は同じである．そこで，一局の対戦時間が短く，問題点の分析が行いやすい 9 路盤を最初の研究対象とすることにした．

本システムでは，ルール的に可能な手の中から少数の候補手を選び出すために事例ベース推論の考え方を応用している．事例ベースを用いた理由は，この手法は明確な解法がない分野においても正解に近い解を求めることが可能であり，決め手となる方法を持たない候補手選出部に適応可能であると考えたからである．また，形以外の情報を持たせた事例を使用することで不要な手を候補から外すことを考えたからである．つまり，これにより形は完全にマッチングしているが価値は低い手を候補から除外することができ，より強い囲碁システムが実現できると考えている．これは，囲碁に関する認知的研究において“碁打ちはサーチしない”という結果が得られていることに基づいている[8]．これは，碁の上手な人はサーチ（探索）をせず，先読み (探索と違い分岐

がほとんどない) を行っているということであり，この考えに基づくと，候補手はごく少数に絞り込まなければならず，手の価値を求めることは必須であると考えられる．

事例は，“勢力パターン”，“評価改善量”，“1 次検索用パターン” から成り立っている．“勢力パターン” は，形を勢力値に変換した 5×5 の大きさのパターンであり，局面と事例の類似度を求めるのに使用される．勢力値とは，碁盤の各マス目が持つ “黒（あるいは白）側にどれほど属しているか” を表す値であり，各石からの影響力を計算することで求められる．“評価改善量” は，この事例により最大でどの程度盤面の得点を改善できるかを調べるのに使用され，“1 次検索用パターン” は事例の分類と検索の省力化のための 3×3 の大きさのパターンである．また，事例は知識として数多く持つ必要があることから，プロ・上級アマの実際の棋譜を入力して自動的に獲得している．

本システムは，大きく分けて状況判断・候補手選出・着手決定・そして事例獲得部の 4 つから成り立っている．

まず状況判断部において，与えられた局面を分析し，石のグループ化を行う．同じグループに所属する石は，お互いにつながっている，あるいは将来的につなげられるという意味合いを持っており，これらは死活判定などの単位として用いられる．次に，これら石群について，取られる（死ぬ）か取られない（生きる）かを判定する（これを “死活判定” と呼ぶ）．死活判定が終ると，盤面の勢力値計算を行う．これらの結果は，次の候補手選出部に渡される．

候補手選出部では，可能手の中から，有意義と思われる手を選出する部分である．ここでは，先ほど説明した様に事例ベースを用いて候補手を選出している．事例ベースは，事例獲得部により対局に先だって構築される．

着手決定部では，選ばれた候補手の中から，最良の手を一つ選び出す部分である．これには，木探索と評価関数を用いている．評価関数は，状況判断ルーチンで計算される “勢力” や “死活” などの情報を基に盤面の価値を計算する．この評価値を一番高くする手が，木探索により最良手として選択される．

本論文では，第 2章において本システムの全体構成とデータ構造について述べ，第 3章においてどのように局面を分析し死活判定を行っているかを述べ，第 4章において候補手の選出方法及びその中から着手を決定する方法について述べ，第 5章において本システムを用いて行った実験の方法とその結果及び考察について述べ，そして最後に第 6章において本研究のまとめと今後の課題について述べている．

2　コンピュータ 9 路盤囲碁システムの全体構成

2.1　システム構成

本システムは，大きく分けて状況判断部，候補手選出部，着手決定部，そして事例獲得部からなる．この様子を図 1に示す．

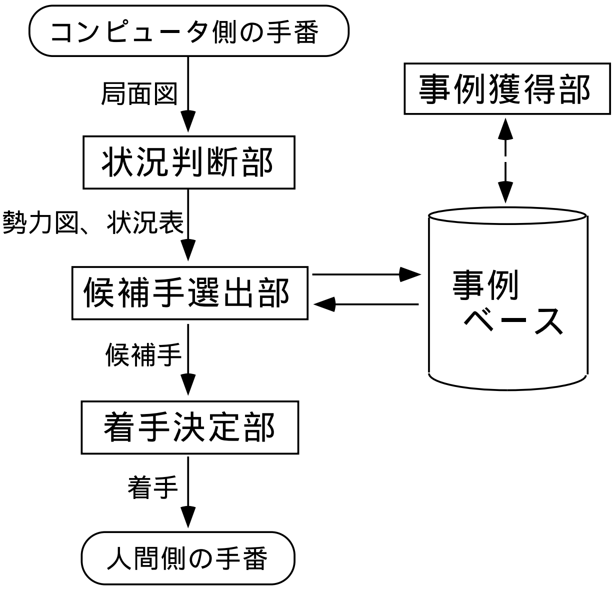


図 1: 囲碁システムの全体構成

状況判断部では，与えられた局面を分析し，石のグループ化を行う．同じグループに所属する石は，お互いにつながっている，あるいは将来的につなげられるという意味合いを持っており，これらは死活判定などの単位として用いられる．次に，これら石群について，取られる（死ぬ）か取られない（生きる）かを判定する（これを “死活判定” と呼ぶ）．死活判定が終ると，盤面の勢力値計算を行う．

候補手選出部は，状況判断部で得られた情報を基に，その局面で打つべき候補手を数手選び出す．本システムでは，事例ベース推論の考え方を応用した方法を採用している．すなわち，前もって “この場合にはこう打つ” といった事例を収集しておく．そして，実際の対局においてそれら事例に似た可能手が現れたならば，それがどの程度の価値を持つかを事例と比較することにより検証する．有効と判断された手は候補手のリストに入れられる．候補手選出部は，盤面のすべての可能手に対し価値を求めた後，リストを手の価値を基にソートし，上位数手を次の着手決定部に送る．

事例は，事例獲得部により前もって収集されている．すなわち，棋譜から事例を獲得し事例ベースを構築する部分が事例獲得部である．事例は知識として数多く持つ必要があることから，実際の棋譜を入力すれば自動的に事例が獲得されるようになっている．棋譜はプロ・上級アマのものを使用することで，質の高い事例を獲得している．事例は，“勢力パターン”，“評価改善量”，及

び“1 次検索用パターン”から成る．“勢力パターン”は，形を勢力値に変換した 5×5 の大きさの配列であり，局面と事例の類似度を求めるのに使用される．勢力値とは，碁盤の各マス目が持つ“黒（あるいは白）側にどれほど属しているか”を表す値であり，各石からの影響力を計算することで求められる．“評価改善量”は，この事例により最大でどの程度盤面の得点を改善できるかを調べるのに使用される．“1 次検索用パターン”は，事例が示す“手”の周囲 1 マスの形を表す 3×3 の大きさの配列であり，事例の分類と検索の省力化に用いられる．

着手決定部は，選ばれた候補手の中から実際に打つべき手を選択する．これは，木探索と評価関数を用いて実現されている．すなわち，候補手を枝にして木探索を行い，末端で評価関数を用いて盤面の価値を計算し，最も高い評価を得られる手を着手として採用する．評価関数は，状況判断ルーチンで判定される勢力や死活情報を用いて盤面の価値を計算する．

以上のシステムを実現するため，本システムは図 2に示す要素から構成されている．

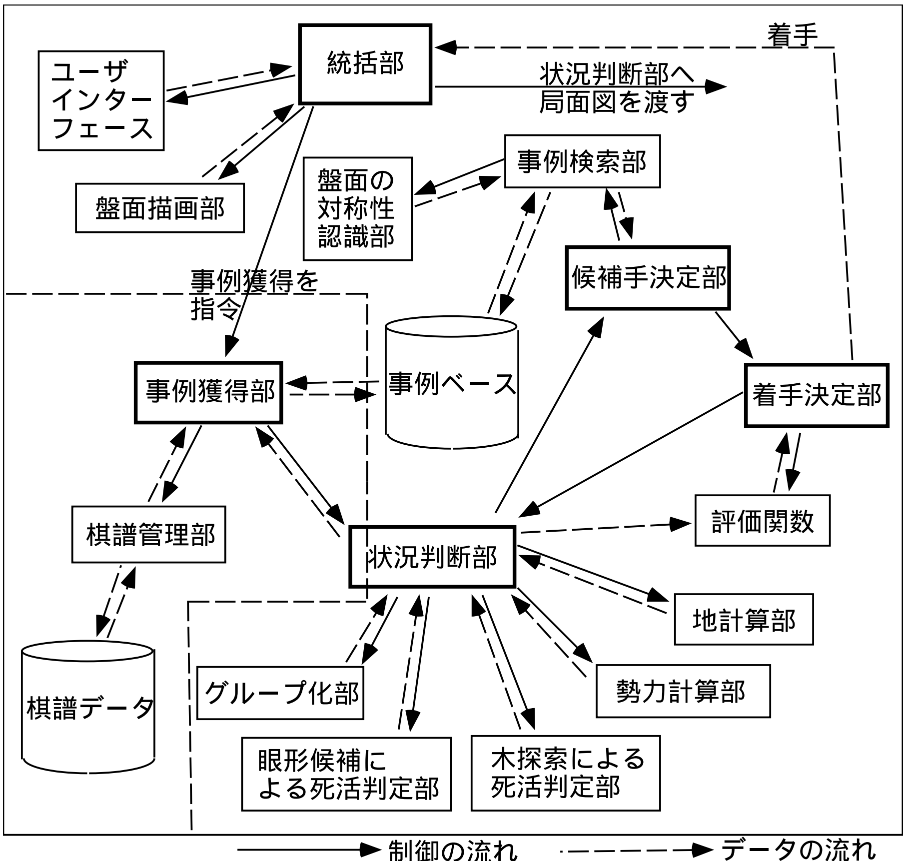


図 2: 囲碁システムの構成要素

本システムは，UNIX の X ウィンドウシステム上に C 言語を用いて実装されている．X ウィンドウへの碁盤および石の表示は，盤面描画部が担当している．入力とその他の出力はユーザインターフェースが受け持つ．統括部は，システム全体に動作の指示を与え，全体を統括する部分である．統括部の仕事は，大きく分けて二つある．一つは，対局時に“コンピュータ側の思考”を駆動することとユーザインターフェースを介して人間側の手を受けとることである．もう一つは，

非対局時の事例獲得作業を制御することである．

事例獲得部は，棋譜管理部を通して棋譜データを受け取り，局面を再現しながら分析を行う．獲得すべき手が見つかると，事例作成のため状況判断部を呼び出し，事例に必要なデータを計算させる．そして，それらデータから事例を作成し，新たに事例ベースに登録する．

棋譜管理部は，事例獲得部に棋譜データを渡す他に，棋譜に示される手を一手ずつ再現し，その局面を表示することで，ユーザに棋譜の鑑賞機能を提供する．この棋譜管理部と事例獲得部は対局中には全く駆動されない部分である．

コンピュータ側の思考の開始は統括部により指示され，状況判断部に現在の局面図（敵プレイヤーが打った直後の局面図）が送られる．

状況判断部は，システムにおける様々なデータを計算するため，必要に応じて各種のサブルーチンを呼び出す．石のグループ化を行うためには，グループ化部が呼び出される．死活判定には，まず眼形候補による死活判定部が呼び出され，その後さらに詳しい結果が要求されるときには，木探索による死活判定部が呼び出される．また，勢力計算や地計算も同様に各ルーチンが呼び出される．状況判断部は，勢力図と状況表を次の候補手決定部に送り，制御をそちらに移す．ただし，状況判断部は，先程述べた事例獲得部から送られた局面図を分析し結果を送り返す仕事を受け持ったときだけは，制御を候補手決定部に移さない．

候補手決定部は，可能手の価値を調べるため事例検索部を呼び出す．事例検索部では，その可能手に似た手が過去に打たれていないか事例ベースを検索する．似た事例が見つかると，それを基に今回の可能手の価値を計算し，結果を返す．この過程で，状況判断部からの分析結果が使用される．また，候補手決定部では，不要な検索を行わないために，盤面の対称性認識部を呼び出し，盤上の検索範囲（すなわち可能手）を限定している．盤面の対称性認識部とは，盤面全体が上下・左右に対称か，対角線をはさんで対称か，あるいは回転すると対称になるかを判定する部分であり，最低限必要な検索範囲を指定する．候補手決定部は，事例検索部と協力し，すべての可能手の価値を計算する．そして，その価値をキーにしてソーティングを行い，上位数手を候補手として次の着手決定部に送る．

着手決定部は，候補手選出部から受けとった候補手を枝として木探索を行うために，それぞれの手を進めたときの局面図を状況判断部に送る．送られたデータは，状況判断部，候補手決定部を通り，その局面における候補手となって帰ってくるので，これを再び枝にして同様の探索を続ける．そして，この探索が一定の深さに達したとき，盤面評価部を呼び出す．盤面評価部は，さきほど説明した評価関数を計算する部分であり，結果すなわち盤面の価値が着手決定部に返される．着手決定部は，最終的に一番高い評価を得られる手を選び出し，着手としてこれを統括部に送る．

統括部は，コンピュータ側の着手を受け取ると盤面描画部を呼び出して盤面を更新し，次にユーザインターフェースを通じて人間側に次の手を入力するように促す．

2.2 データ構造

本システムを通して用いられるデータの構造を表 1に示す．

盤上の各石は，それぞれ“石番号”と“所属する連”の情報を持つ．“石番号”は，盤上のそれぞれの石に対し，独立に割り当てられた整数であり，各石を区別するためのものである．この石番

表 1: システム全体を通して使用されるデータ構造

データ名	判定される項目	意味とその取りうる値
石	石番号	座標に対応する 0 以上の整数
	所属する連	その石の所属する連の連番号
連	連番号	それぞれの連に独立に付けられる 0 以上の整数
	構成石	{ 連を構成する石（石番号）の集合 }
	所属する群	その連の所属する群の群番号
	呼吸点	0 以上の整数
群	群番号	それぞれの群に独立に付けられる 0 以上の整数
	構成連	{ 群を構成する連（連番号）の集合 }
	群の大きさ	0 以上の整数
	死活	“生き” or “死に” or “不定” or “不明”
マス目	勢力値	整数
プレイヤー	地	0 以上の整数
	アゲハマ	0 以上の整数
	全勢力範囲数	0 以上の整数

号は，座標から計算して求めている．“所属する連” は，どの “連” に所属しているかを連番号で表している．ここで “連” とは，“縦横に隣接した同色からなる石のグループ” を意味する．図 3(a) に，連の例を示す．連は，盤上における最も基本の単位であり，同じ連番号を持つ石同士は必ず生死を共にする．連番号は，盤上に現れたそれぞれの連に対し，独立に割り当てられる．

盤上の各連は，それぞれ “連番号”，“構成石”，“所属する群”，そして “呼吸点” の情報を持つ．“構成石” は，その連を構成する石の集合で表される．“所属する群” は，どの “群” に所属するかを群番号で表している．“群”（あるいは “石群”）とは，“連絡可能と考えられる連の集合” を表す．いい替えれば，敵プレイヤーの妨害に関わらず，将来的に連としてつなげられるものの集合である．図 3(b) に，群の例を示す．群は，死活判定における基本の単位であり，それぞれに独立な群番号が与えられる．“呼吸点” は，連に対して縦横に隣接する空白マス目の数である．呼吸点が 0 になるということは，周り全てを敵石あるいは盤の端に囲まれたということであり，その連の死を意味する．

群について説明を行う前に，先に “勢力値” について述べておく．盤上の各マス目は，“勢力値” を持つ．“勢力値” は，盤上の各マス目が “どのくらい黒（あるいは白）側に属しているか” を表す．正の数は黒側に属していることを表し，負の数は白側に属していることを表す．勢力値は，盤上の各石がそのマスへ及ぼす影響力を合計することに求められる．この値は，黒石が近くにあればあるほど，そして多くあればあるほど，より高い値を持ち，逆に白石の場合は，より低い値を持つ．また，この勢力値がある一定値以上のとき，そのマス目は黒の勢力下にあり，逆に一定値以下のとき，白の勢力下にあるとしている．

盤上の各群は，それぞれ “群番号”，“構成連”，“群の大きさ”，そして “死活” の情報を持つ．

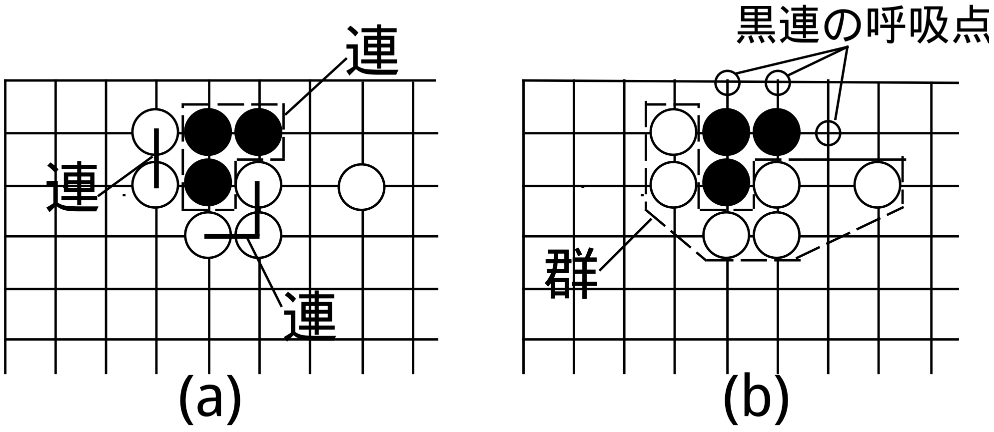


図3: 連と群

“構成連” は，その群を構成する連の集合で表される．“群の大きさ” とは，群の安定度を表す数値であり，その群がどれだけ多くのマス目を勢力下においているかを示している．周囲のマス目を勢力下においているかは，さきほどの勢力値により判定される．この様子を図4に示す．図には4つの群があり，‘B’ はその空点が黒の勢力下にあることを示し，‘W’ はその空点が白の勢力下にあることを示している．群Aは大きさ11，群Bは大きさ10であり，群Cと群Dは共に大きさ1である．“死活” は，石群に対する生死の状態を表し，“生き”，“死に”，“不定”，“不明” のどれかの値を取る．“生き” は，その群が眼を二つ持つあるいは眼を二つ以上作ることができることを表す．“死に” は，逆にその群が眼を二つ作れないことを表す．“不定” は，生きとも死にともいえない状態を指している．例えば，序盤の石群の多くがこれに属する．“不明” の石群は，探索を行わないと生死の判断がつかない状態を指している．この状態に分類された石群は，後にシステムの木探索ルーチンにより個別に死活判定が行われ，最終的に上の3つの状態のどれかの値を取る．

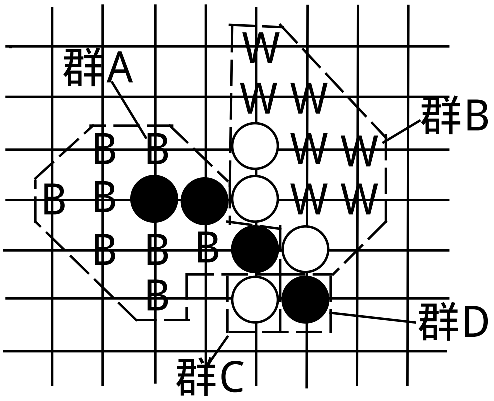


図4: 群の大きさ

黒側および白側のプレイヤーは，それぞれ “地”，“アゲハマ”，そして “全勢力範囲数” の情報

を持つ．“地” は，文字通り地をいくつ稼いだかを表す．これはプレイヤーを対象としているので，そのプレイヤーの各群が持つ地の数を合計したものである．囲碁では，(コミを加えた後に) 地とアゲハマが多い方が勝利者となるので，評価関数において特に中・終盤以降から重要視される項目である．“アゲハマ” は，敵の石を幾つ取ったかを表す．この項目を改善するということは，敵の石を多く取り，そして味方の石を取られないようにするということである．すなわち，重要視されるべきは死活であり，アゲハマの主な目的は得点計算への使用である．“全勢力範囲数” は，各プレイヤーがどれだけの空白地を自身の影響下においているかを表す．これは，各プレイヤーについて自身の勢力下にある空白地を数える事で計算される．ただし，“群の大きさ” における勢力下の判定より，より厳しい “しきい値” を用いて判定を行う．これは，この項目が評価関数の序盤・中盤において地の予想として用いられるため，確実に勢力下になければならないからである．

3　状況判断

3.1　石群の形成と評価

3.1.1　グループ化

石のグループ化は，連の形成，そして群の形成の二段階に分けて行われる．連は，縦・横に隣接した同色の石からなるグループであるので，簡単に求めることができる．連を求めると，次に群を求めるわけであるが，群は連絡可能と考えられる連の集合であり，現在実際につながっているわけではないので，連より判定が困難である．一番，正確な判定方法は，お互いに石を置いていった場合に石をつなげられるかを調べることであるが，時間がかかり過ぎることから現実的でない．そこで本システムでは，囲碁の対局においてよく現れるパターンを検出することで石のつながりを認識する方法を採用することにした [11]．

囲碁においてよく現れる石のつながり方には，“並び”，“コスミ”，“一間トビ”，“ケイマ”，“二間トビ”，“大ゲイマ” などがある．この他にも，幾つか代表的な石のつながり方があるが，それらはこれらの組合せである．そこで各々について以下の様な方法で，つながりを判定する．

1. コスミ

   コスミの形を，図 5に示す．

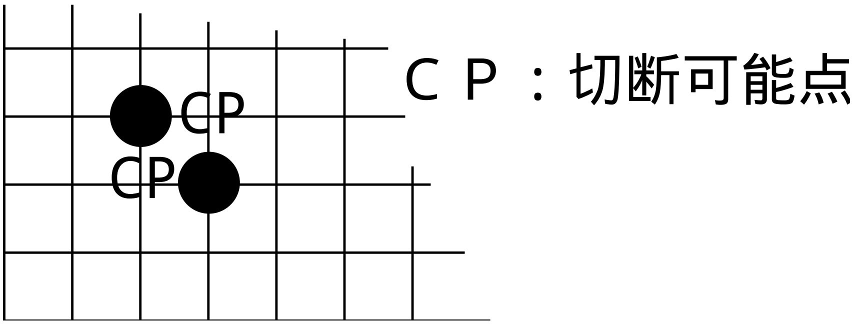
   

   図 5: コスミ

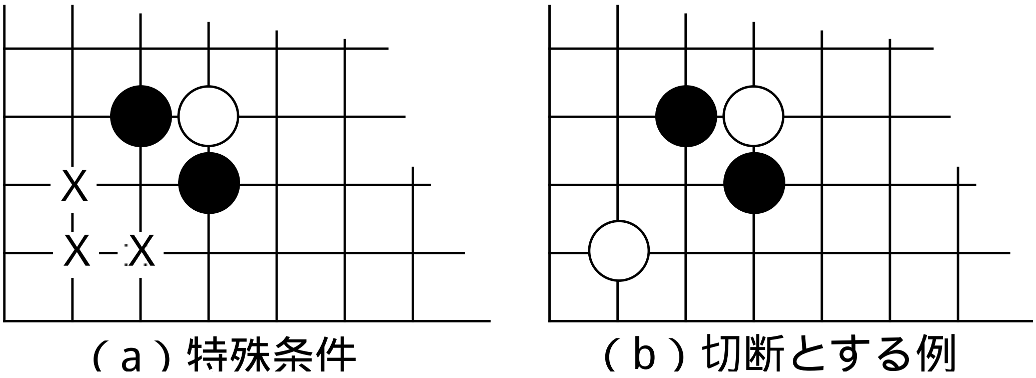
   図の ‘切断可能点 (CP)’ と示されたマス目は，この二つの黒石の連絡を断つ（切断するという）ために重要な場所を示している．コスミでは，CP のどちらかが空いていれば，連絡と判定する．ただし例外として，図 6の ‘X’ に敵石（この場合，白石）がある場合は切断とする．

2. 一間トビ

   一間トビの形を図 7に示す．

   一間トビの形が連絡と判断されるための条件は，3 箇所の CP の二つ以上が空白となっていることである．例えば，図 8の形が現れたときは，切断となる．

3. ケイマ・大ゲイマ



（ａ）特殊条件

（ｂ）切断とする例

図 6: コスミの形において切断となる特殊条件

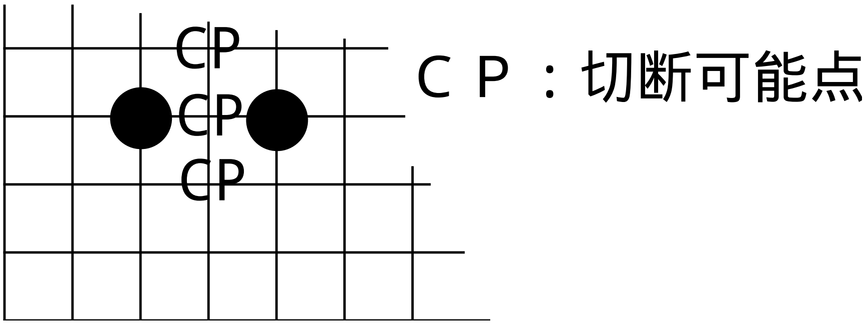


図 7: 一間トビ

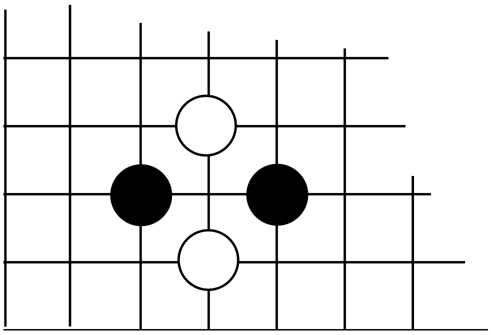


図 8: 一間トビの形に置いて切断となる例

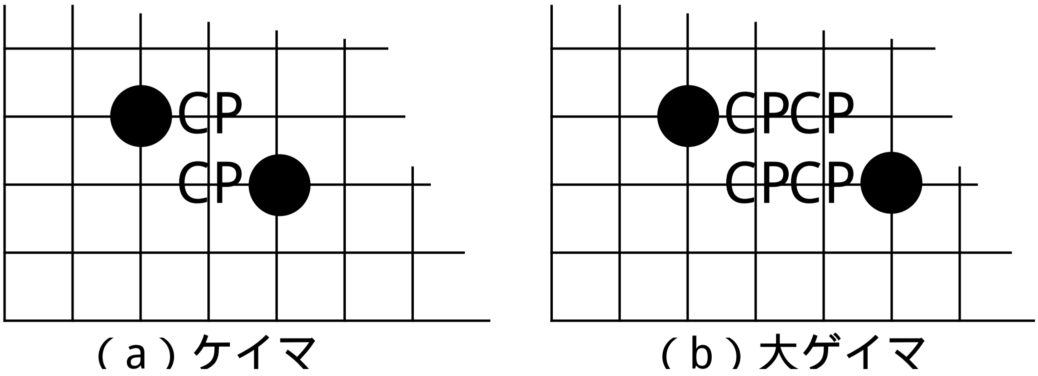


図 9: ケイマと大ゲイマ

ケイマの形を図 9 (左) に，大ゲイマの形を図 9 (右) に示す．

これらが連絡と判断されるための条件は，CP の少なくとも一つが敵石ではないことである．

4. 二間トビ

二間トビの形を図 10に示す．

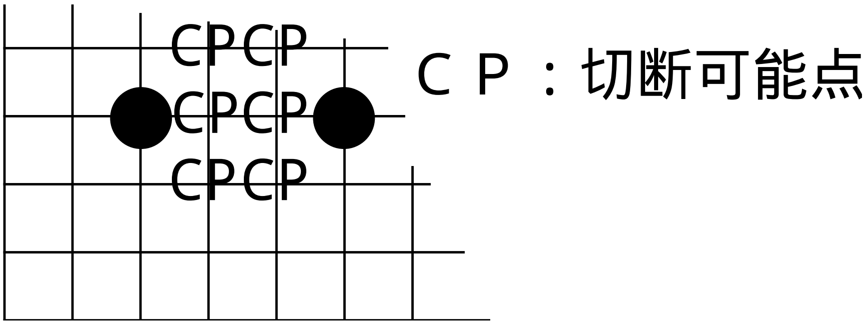


図 10: 二間トビ

二間トビの連絡のための条件は，CP の全てが敵石でないことである．

3.1.2 勢力計算

勢力値は，盤上の各石からの影響力を用いて計算される．本システムで現在用いている石の影響力の値を図 11に示す．

石の影響力は，システム作成者の経験に基づいて設定されている．石の影響力の大きさは，大きく細かく取ればより正確になるが，計算の労力が増大する．そこで，一定の正確さを保ちつつも計算の労力を最小にできる大きさが求められる．本システムでは，一定の正確さを保つための最低限の目標を，以下の二つとした．

- 囲碁の基本的な石の配置であるケイマの距離に勢力を及ぼせること．

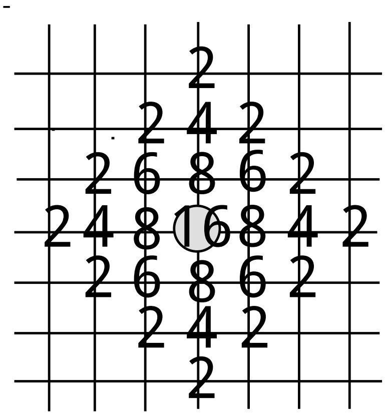


図 11: 石の影響力

- 大ゲイマの位置にある石同士の間が，自勢力として十分な勢力値を持つこと．

大ゲイマとは，図 12の様な位置関係にある石を指す．この図 12は，大ゲイマについて勢力値を計算した結果を示しており，大ゲイマ間が両方の石の影響を受け自勢力として強固になっていることが分かる．

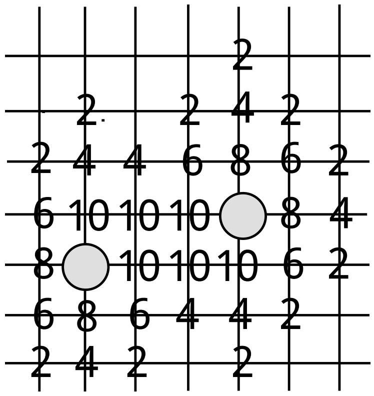


図 12: 大ゲイマ間の影響力

石の影響力は，黒側に正，白側に負の値を与えている．石の影響力は，石自身の占める位置を除けば縦横に隣接するマスに一番影響を及ぼすようになっており，斜めにはそれより低い値となっている．これは，囲碁のルールが縦横には隣接しているが，斜めは隣接していないとすることに基づいている．そして，距離が離れるにつれて減少するようにしている．

他のゲーム同様，囲碁においても盤の端は重要である．しかし，囲碁では敵石を囲んで殺せるというルールがあるため，盤の端の性質が他とは少し異なる．将棋やチェスなどのゲームでは，盤の端は駒の移動を制限する効果を持っている．ところが，囲碁では敵石を囲むのに味方の石だけでなく盤の端を利用してよいので，味方の石（あるいは逆に敵方の石）という効果も持っている．

また囲碁では，空白地を囲って味方の領域とするためにも盤の端を利用する．この様子を図 13に示す．黒側は，黒石の間をその気になればつなぐことができ，同様に盤の端と黒石をつなぐこともでき内部を味方の領域であると主張することができる．従って，この場合においても，盤の端は味方石と同様の効果を持っているといえる．そこで，本システムでは盤の端についても勢力値を計算し，一定値以上を持つものについては，石と同じように周りに影響力を及ぼすこととした．この様子を図 14に示す．盤の端からの影響力は，石の影響力より低くしている．これは，ゲームの進行につれて味方ではなく敵側に有利に働くかもしれないからである．

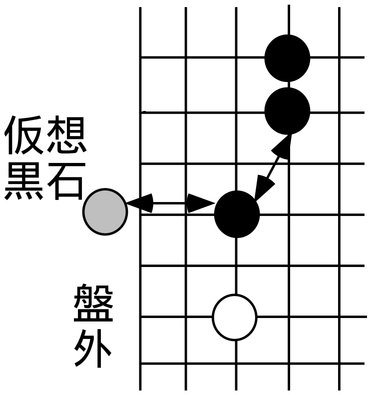


図 13: 盤の端を利用した味方領域の確保

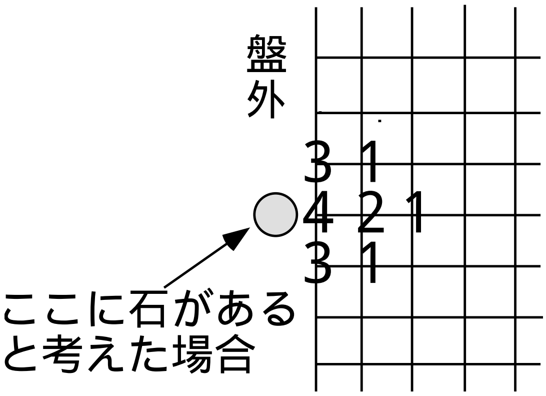


図 14: 盤の端からの影響力

勢力値の計算は，そのマス目への各石および盤の端からの影響力を合計することで求められる．ただし，石自身の影響力 （図 11の石の上の値）を越えたときは，“石自身の影響力-1” の値となるようにしている．評価関数では全勢力範囲数といったような一定値以上のマス目の数を評価対象にしているため，この計算でも実用上十分である．また，事例検索において類似度を求めるときに勢力値を用いており，この方法であれば類似度計算を簡単に行えるという利点がある．加算のみの方法としたのは，本システムのような対局を行うシステムでは，割算を含むような時間の

かかる処理は極力避ける必要があるからである．以上の影響力を使って，図 15の局面を勢力値計算した例を図 16に示す．

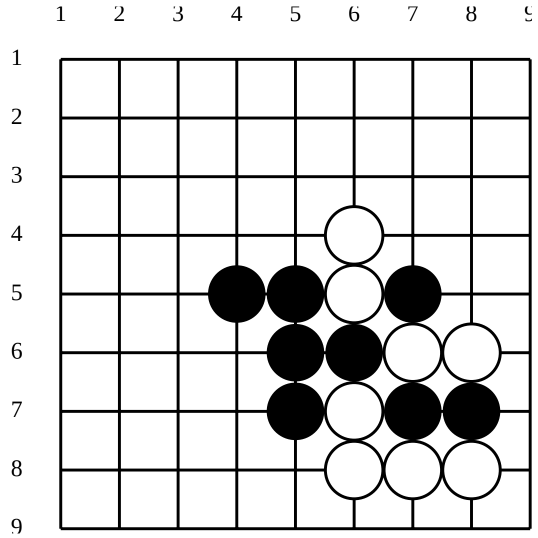


図 15: 局面 A

0	0	0	0	0	-2	0	0	0
0	0	0	2	0	-6	0	0	0
0	0	2	4	0	-6	-6	-2	0
0	2	6	10	10	白	-6	-4	-2
2	6	12	黒	黒	白	黒	-5	-7
0	4	15	15	黒	黒	白	白	-6
0	2	6	15	黒	白	黒	黒	-12
0	0	0	2	-3	白	白	白	-13
0	0	0	0	-7	-15	-15	-15	-12

図 16: 局面 A の勢力計算例

盤面の勢力値が実際に計算されるのは，新たな局面が生成され状況判断が行われるときだけである．後に，事例の検索時などにおいて必要となれば，計算済みの結果から必要部分だけを切り出し送っている．

3.2 死活判定

3.2.1 眼形候補による死活判定

死活判定において眼の概念は重要であり，“2 眼以上ある石は必ず生きである” という法則がある．図 17にこの様子を示す．

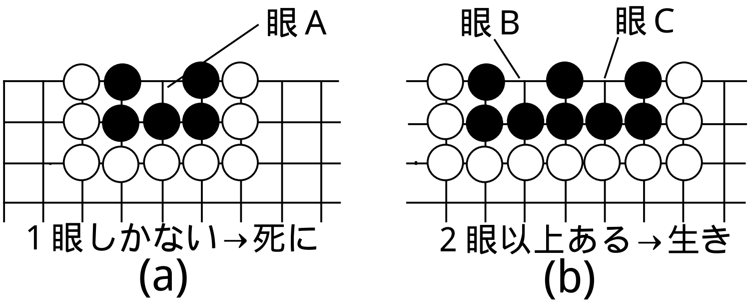


図 17: 眼による死活判定

眼とは，味方石と盤の端により囲まれた閉鎖領域のことであり，眼が一つしかない石群は，そこを敵石に埋められてしまうと死んでしまう．図 17の (a) では，‘A’ の地点に白石が置かれると黒石は全て取られる．一方，眼が二つあると，その二箇所を敵に埋められてしまわない限り，取られることはない．しかし囲碁では，一度に置ける石は一つずつであり，同時に二箇所を埋めることはできない．図 17の (b) では，‘B’ と ‘C’ の地点を埋めるため，先にどちらかに白石を置こうとすると，それは自殺手 (禁じ手) になってしまい，どうやっても黒石を取ることはできない．

眼を基に死活判定を行うことは有効であるが，対局が進まないと眼は現れてこない．そこで，将来，眼になりそうな領域 (以後，眼形候補と呼ぶ) を検出し，死活判定を行う方法を採用している [11]．この方法は，大サイズの死活問題を探索で解こうとすると一般に時間が掛かりすぎるので，問題をいくつかの比較的小さな眼形領域に分割し，個々の眼形領域で眼数を求め，それを合計することにより全体の眼数を調べようという考え方に基づいている [9]．

眼形候補による死活判定の全体の流れを図 18に示す．

最初に眼形候補 (領域) の検出を行い，その領域数 (サイズ) と境界の状態が検出される．眼形候補の検出は，眼形を構成する可能性のある空点 (以下，内点と呼ぶ) を検出することで行われる．その方法を図 19を用いて説明する．

まず最初に，石群の外の境界と内部の仕切りの予想を行うため，仮想結線の処理が行われる (図 b)．そして，その内部の空点を内点とするが，例外として仮想結線の隣にある空点は除かれる (図 C)．これは，仮想結線に敵石が侵入したとき，仮想結線の隣に石を置き侵入をくいとめることになるため，内部の空点とはならないからである．内点が決定すると内点のグループ化を行い，グループ化されたそれぞれの領域がひとつの眼形候補となる (図 D)．次に，眼形候補の境界が完全であるのか，それとも不完全であるのかを調べる．これは，眼形候補を形成する内点の 4 方向の状態を調べることで判定される．具体的には，以下のような内点により形成される境界は完全であり，それ以外は不完全となる．

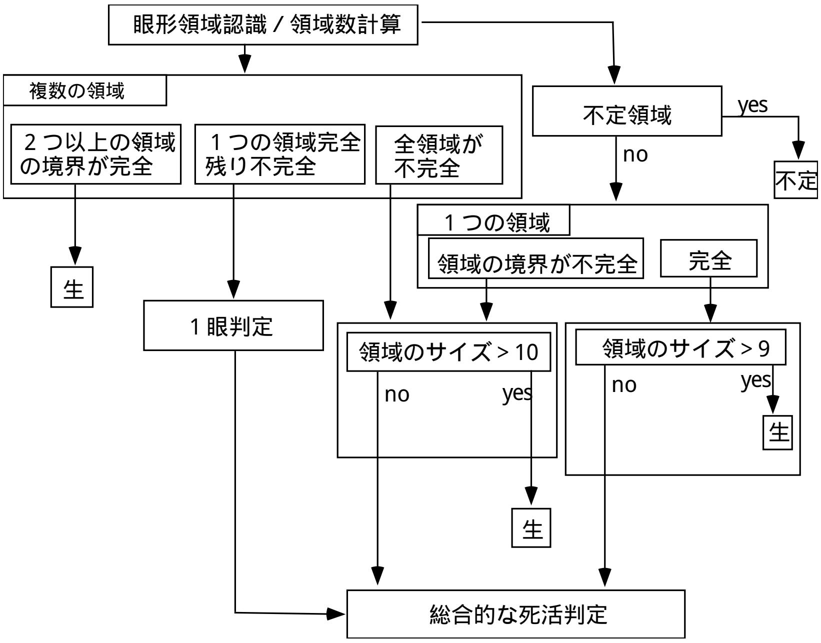
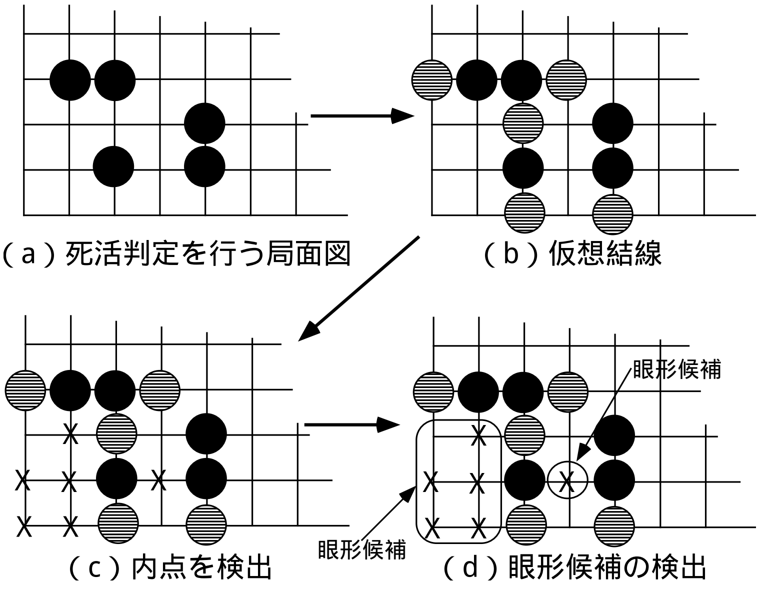


図 18: 眼形候補による死活判定の全体の流れ



（ a ）死活判定を行う局面図　（ b ）仮想結線

（ c ）内点を検出　（ d ）眼形候補の検出

図 19: 眼形候補決定の過程

- 3 方向以上において，味方石，仮想石にぶつかる．
- 味方石，仮想石にぶつかるのは 2 方向であるが，残りの 2 方向が盤の端に到達する．

眼形候補（及びその領域数とその境界の状態）が求まると，先ほどの図 18 に結果を当てはめることで，死活が決定される．

3.2.2 木探索による死活判定

木探索による死活判定部では，眼形候補による死活判定で “不明” と判断されたものについて死活を求める．現在のところ，この部分は，“生死を調べる石群に隣接する空白のマス目” への着手を枝とする木探索を行っている．そして，以下の条件により結果を判定している．

- 呼吸点が無くなると “死に”
- 呼吸点が一定値以上に増えると “生き”
- 一定の深さまで読んでも死なないときは “不定”

この方法だけでは，時間もかかり正確性にも欠ける．しかし，他の研究で行われているような探索する枝に順序づけを行う，中間結果をハッシュ表に蓄えて利用するなどの工夫を行ったり [10]，攻め合いなどを判定し攻めの手を候補に入れるなどを実装することにより，これらの問題はある程度まで解決が見込める．

膨大な時間のかかる木探索の問題を根本的に解決するためには，眼形候補などを用いることで可能な限り木探索を避ける方法を検討することがより重要であると考えられる．従って，本研究では現在のところ，眼形候補による死活判定の方に力を入れている．もちろん，死活判定における木探索の役割は重要であるので，眼形候補による方法と合わせて改善を加えていく必要がある．

4　候補手選出および着手決定

4.1　候補手の選出に用いる事例

4.1.1　事例の形式

本システムで用いる事例の形式を図20に示す．

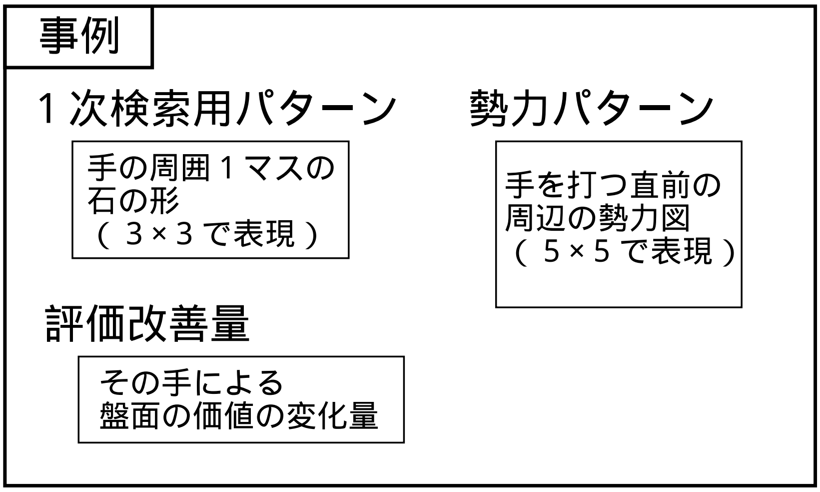


図 20: 事例の形式

事例は，“勢力パターン”，“評価改善量”，及び“1次検索用パターン”からなる．

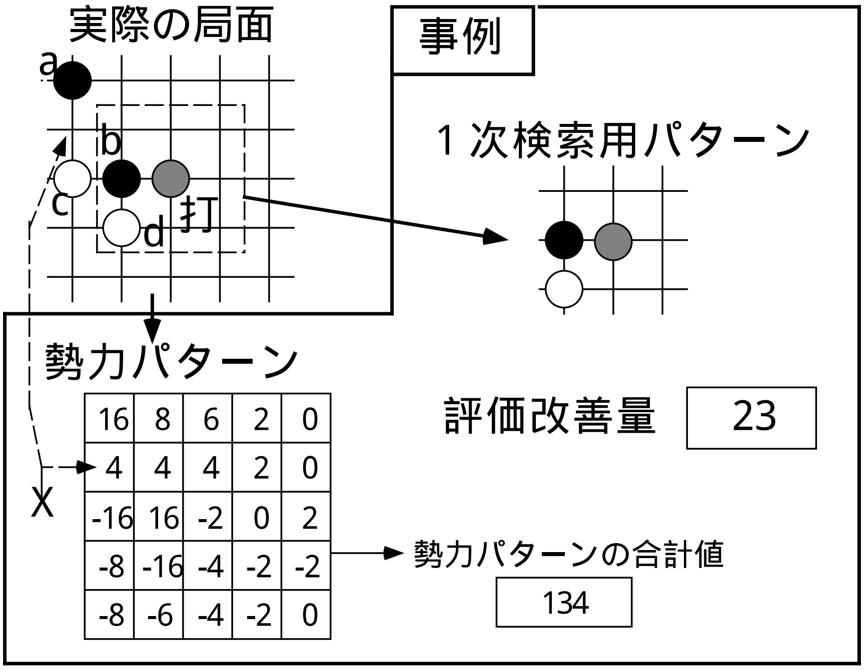
“勢力パターン”は，事例に示される手（以後，これを単に“手”と記す）が打たれたときの石の形を勢力値を用いて表した5×5の大きさのパターンである．勢力値を用いているので，石の形だけでなく石の力関係も表現されている．このパターンは，“手”を中心として表現されている．すなわち，この5×5のパターンに示される力関係が盤上に現れたとき，その中心に手を打てばよいことを示している．勢力パターンは，“事例に示される手”と可能手との類似性を調べるために用いられる．

“評価改善量”とは，その手を打つ前と後で盤面の価値がどれだけ変化したかを表す数値である．数値は，正のとき盤面が改善されることを表し，値が大きければ大きいほど，より大きく盤面の価値が変化することを表す．盤面の価値は，着手決定部で使用される評価関数により求められる．評価改善量は，可能手に対して“最大でどの程度の効果を期待できるか”を調べるために使用される．

“1次検索用パターン”とは，“手”が打たれる直前の縦横・斜め1マスの石の配置（形）を表す3×3のパターンである．“手”が打たれる直前のパターンであるので，中心は必ず空白である．このパターンは事例の分類に使用され，検索の省力化に使用される．

次に，事例の具体例とそれが獲得されたときの局面を図21に示す．

局面部分の中央に薄く描かれた黒石がこの事例の示す次の一手であり，この手が事例化されている．事例では，実際には黒石・白石という区別ではなく，手番側の石・非手番側の石と分けている．しかし，ここでは分かりやすいように手番側を黒石・非手番側を白石で表現している．勢



16	8	6	2	0
4	4	4	2	0
-16	16	-2	0	2
-8	-16	-4	-2	-2
-8	-6	-4	-2	0

図 21: 事例の具体例とその獲得時の局面図

力パターンは，事例の獲得が行われた局面を勢力値で表している．勢力パターンの中心が，次に打たれる手の位置に対応している．勢力値が ‘16’ の部分は黒石であり，‘-16’ は白石に対応している．他の部分は，勢力値計算の方法により求められる．例えば，図の ‘X’ の部分は，以下のようにして決定されている．‘a’ の黒石から見て’X’ は縦へ 1 マスの位置であり，先ほど図 11に示したように石は縦・横には 8 の影響力を持つので，‘a’ は 8 の影響力を及ぼしている．同様にして，‘b’ の黒石は 6 の影響力を及ぼしている．‘c’ の白石は，敵側であるので -8 の影響力を及ぼし，‘d’ は -2 の影響力を及ぼしている．よって，‘X’ の値は，$8+6+(-8)+(-2)=4$ となる．勢力パターンの横の “勢力パターンの合計値” とは，勢力パターン内の各マスの絶対値を合計したもので，事例の類似性を求めるときに用いられる値である．評価改善量は，この手が打たれたことにより，盤面の価値が 23 増加したことを示している．1 次検索用パターンを見ると，手の周囲 1 マスに対応していることが分かる．

4.1.2 事例ベースの構成方法と分類

本システムの事例ベースは，序盤（1 ~ 20 手目），中盤（21 手目 ~ 40 手目），そして終盤（41 手目以降）の 3 つのサブ事例ベースから成り立っている．これは，ゲームの進行に応じて，使用される事例の傾向が変化するからである．すなわち，序盤において有効な手が，必ずしも終盤では有効でないので，手数に応じて，アクセス先を切替えている．区切りとして 20 手目と 40 手目を選んだのは，実際の棋譜を分析した結果である．

本システムでは，可能手の価値を求めるとき，少なくとも斜めを含む周囲 1 マスが完全一致する事例を用いなければならないという条件をつけている．事例が可能手の周囲 1 マスと全く同じ

パターンを持つかは，1 次検索用パターンを調べればすぐに分かる．よって，あらかじめ 1 次検索用パターンに基づいて分類を行っておけば，検索対象を限定でき，検索の労力を減らすことができる．一次検索用パターンにより事例がどのように分類されているかを，序盤，中盤，そして終盤について，それぞれ図 22，図 23，図 24に示す．

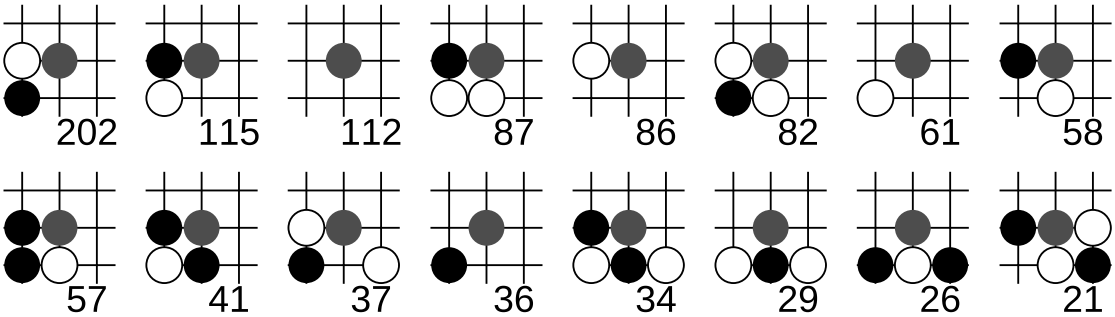


図 22: 序盤の事例分類

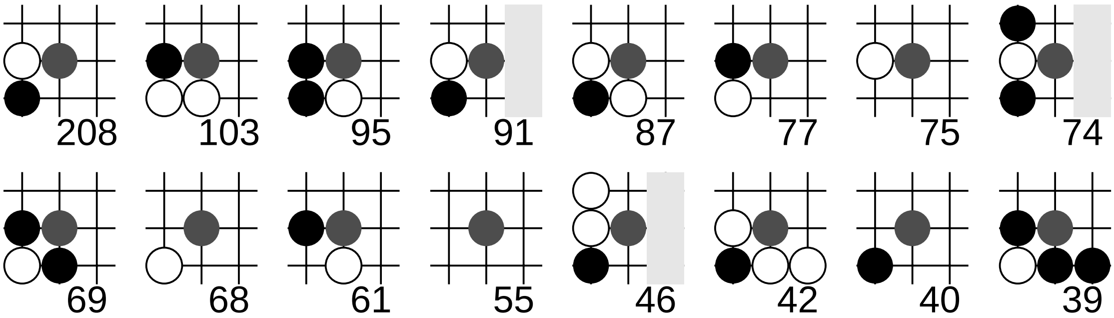


図 23: 中盤の事例分類

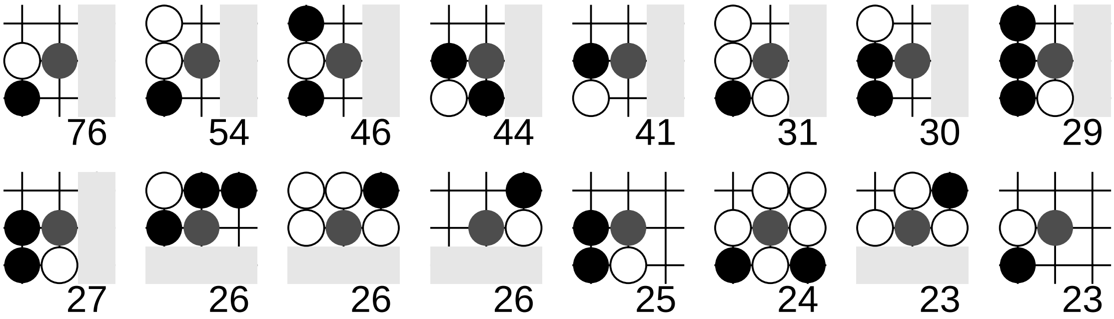


図 24: 終盤の事例分類

各図は，どのような 1 次検索用パターンの元に幾つの事例が格納されているかを表しており，

格納数の多い順に並べたときの上位16個を示している．図のパターンの下の数値が，格納数である．パターン内の薄く塗りつぶされた部分は，そこが盤外であることを示しており，中央の薄い黒石が事例に示される手に対応している．各局面において順番こそ違っているが，同じようなパターンが現れている．また，中盤・終盤と進むにつれて，盤の端での攻防が多くなることを反映して，盤端の事例を格納するパターンが多く現れている．事例の分類は，検索速度の関係から，できる限り均等に分散している方がよいが，序盤・中盤において最初の10個前後にかなり多くの数が集まっているので，今後，速度に問題が生じるようならば，さらなる分類を考える必要がある．

事例は，対局前にファイルからメモリ上に読み込まれる．ファイルには，表2の様な対局には使用しないデータも含まれている．これらは，事例の管理などに使用されるが，資源の節約のため対局時にはメモリ上に読み込まれない．

表2: 事例ファイルに記録される項目

項目	内容
獲得ファイル名	この事例を獲得した棋譜ファイル名
手数	上記棋譜において何手目の手か
形	手の周囲2マスの形
場所	事例が獲得された局面の場所

本システムでは，事例を序盤・中盤・終盤に分けているので，すべての事例を一度にメモリ上に読み込む必要はない．すなわち，序盤には序盤と中盤の事例のみを読み込んでおき，必要に応じて終盤の事例を読めばよい．序盤に中盤の事例を読んでおくのは，着手決定部において序盤と中盤をまたいだ木探索が行えるようにするためである．また，終盤の事例を読み込むときには，序盤の事例は不要であるので，これと入れ換える．本システムでは，このように事例を必要に応じて読み込み，不要な事例と入れ換える機能があるので，事例数の増加にも対応することができる．

4.1.3 事例の獲得方法

事例の獲得は，棋譜ファイルを入力することで自動的に行われる．事例獲得を行う対象は，棋譜ファイルのすべての手としている．ただし，事例ベースに獲得済みの手は，重複して獲得することはない．すべての手を対象としているのは，本システムが大量の事例を必要とするため，人間の手により作成や選択する作業は向いていないからである．この場合，好ましくない手が獲得される可能性があるが，プロや上級アマの棋譜を使用することで，質の低い手が獲得される割合を下げている．また，棋譜に含まれている悪手といっても，それは現在の囲碁システムの棋力からいって，それほどの悪手ではないともいえる．仮に，実験の結果，問題が生じたときは，事例に瀕度情報を持たせ，あまり使われない事例を削除するなどの対策をとることも可能である．

事例はコンピュータと人間の対局から獲得することも考えられる．これにより，コンピュータに不足した手を捕捉できると考えられるが，プロ・上級アマの棋譜から獲得した事例と同じ価値で扱うかなどの問題もあり，今回のシステムでは実装を見送っている．

獲得した事例を事例ベースに格納するにあたり，その格納先を序盤（1～20手目），中盤（21手目～40手目），及び終盤（41手目以降）で分けている．これは，ゲームの進行に応じて，使用される事例の傾向が変化するからである．すなわち，序盤において有効な手が，必ずしも終盤では有効でないので，手数に応じて事例ベースを切替えている．序盤を1～20手目，中盤を21手目～40手目，そして終盤を41手目以降としたのは，棋譜を眺めたとき，平均的にこのように分けられると判断したからである．もちろん，中盤がもっと早く来たり，中盤が非常に短いといった場合も存在し，中盤の手が序盤の事例ベースに学習される可能性はあり，本来計算機に自動的に序盤から中盤，そして終盤へ変化したことを認識させられればよいのだが，現在のコンピュータ囲碁システムではその認識が不可能である．しかし，序盤・中盤・終盤というのは厳密に定義できるものではなく，人間の感覚的なものである．よって，序盤と中盤の手が交じることにより，大きな問題が生じるとは考えていない．

事例の獲得は以下のようにして行われる．まず，棋譜に示される手を最初から再現し，局面を生成していく．その局面に対する棋譜の次の手が，事例ベースになければ，その手を獲得する．事例ベースにあるかどうかは，同じ形（5x5のパターン）があるかにより判断される．もし事例にない場合，その手を事例にするため必要な情報を計算する．勢力パターンは，局面を勢力値に変換し，その中から手を中心とする5×5の大きさのパターンを切り出すことで得られる．評価改善量は，その手が打たれる前と後のそれぞれの局面において評価関数を呼びだし，その差を求めることで得られる．1次検索用パターンは，局面から手を中心とした3×3の大きさを切り出すことで得られる．1次検索用パターンは内部では，2進数16ビットの値を持っている．変換の仕組みを図25に示す．

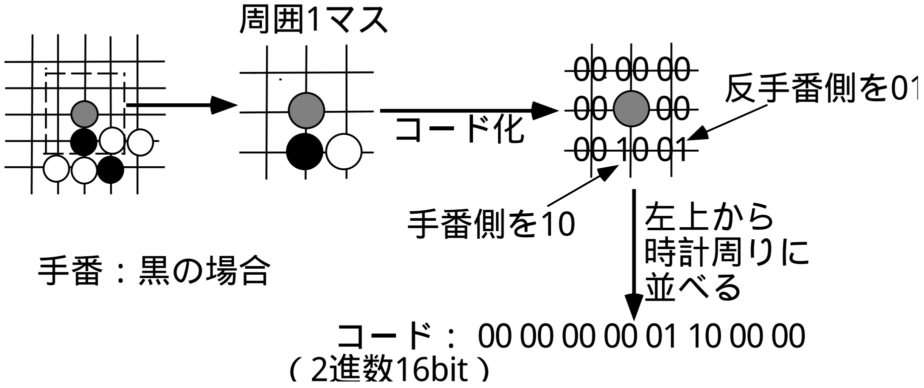


図25: 1次検索用パターンのコード変換方法

図の様に，まず切り出した手の周囲1マスのそれぞれのマス目を2桁の2進数で表す．手番側の石は“10”に変換され，非手番側の石は“01”，空白地は“00”，そして番外は“11”に変換される．変換されたコードは，左上から時計周りに並べられ，並べられた2進数16ビットが1次検索用パターンのコードとなる．

以上の項目が自動的に計算され，事例が作成される．作成された事例は，事例ベースに格納されるが，その前に1次検索用パターンのコードが最小となる向きに変換される．これは，1つの

パターンには回転・左右対象に形を考えることにより，最高8通りの同一のパターンが考えられるからである．この様子を図26に示す．図のように，同じ形であるにも関わらず，コード化の方

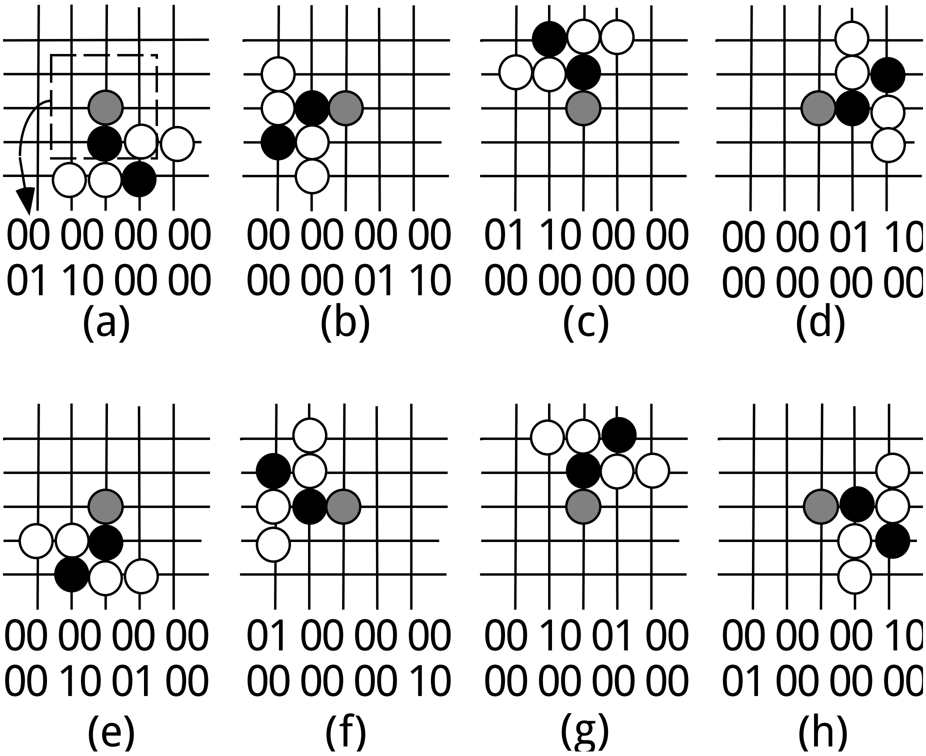


図26: 同一の1次検索用パターンの種類

向により異なった値を持ってしまう．しかし，回転・左右対象の関係にある1次検索用パターン同士が，同じ1次検索用コードを持つ方が，分類を簡単に行うことができる．そこで，コードを2進数と見たとき最小の値を持つ方向で事例化を行うことで方向をそろえている．この例では，事例は図の(b)の方向に変換されてから，事例に格納される．

4.1.4 獲得された事例の分析

事例の獲得は，プロ・上級アマの9路盤での棋譜200局を対象に行った．棋譜は，知り合いのプロ棋士の方から頂いた．獲得された事例数を表3に示す．中盤が一番多く，そして序盤，終盤の

表3: 獲得された事例数

序盤（1～20手目）	中盤（21手目～40手目）	終盤（41手目以降）
1808	3095	1561

順になっている．これは，中盤より序盤の方が，石の数が少なく，また密度も低いため，出現するパターンが決まったものになり，結果として事例数が少なくなったと考えられる．同様に考え

ると，中盤より終盤の方が事例数が多くなるべきであるが，そうはなっていない．これは，学習に使用した事例が上級者のものであったため，途中で投了することが多かったこと，また時間制限の条件下で対局したものがあり，それが時間切れとなる場合があったことなどが理由である．

棋譜の中で現れた瀕度の高かった形を，序盤，中盤，そして終盤ごとにそれぞれ図 27，図 28，図 29に示す．

パターン内の中央の薄い黒石は，そのパターンが現れた時に中央に黒石が打たれたことを表している．パターンはすべて黒側からみた形で表現している．パターンの下の数値は，200 個の棋譜における出現数であり，パターン内の薄く塗りつぶされた部分は，盤外を表している．

序盤において，最も出現数が多い形は，周りすべてが空白の形であったが，これは各棋譜の第一手目が必ずこの形になるので，当然といえる．その次に多いのは，一間トビの位置からの敵石へのカカリである．カカリとは，敵石に自由に地を作らせないなどを目的として敵石の近くに打つ手をいう．第三番目のパターンもケイマの位置からのカカリである．四番目以降は，カカリに対抗してよく打たれる手が現れている．序盤は，囲碁の本に記されているような形の出現瀕度が高いといえる．

中盤においては，序盤よりパターンに現れる石の数が多くなり，そのためパターンごとの出現瀕度が少なくなっている．中盤においては，盤端での攻防が多くなっている．これは 9 路盤であるため，中央の攻防が比較的すぐに盤端での攻防に移っていくためと思われる．中盤全体を通してよく現れているのは，第二番目の手のように盤端で地を稼ぐため敵の領域に一歩踏み込む手や，第三手目のように相手側がこちらの石を “アタリ” にしたため，それに対抗してあたりにされた石を囲っている敵石を取るという手などがある．“アタリ” というのは，次の一手で石がとられる状態になることをいう．

終盤においては，中盤よりさらに石の数が多くなり，全く同じ形が現れにくくなっている．終盤においては，第一番目のパターンのような盤の角でのコウに関する争いや，第四番目のように地を確定させる手が目だっている．コウとは，第一番目のパターンように，黒石を置くと黒石の上の白石を取れるが，次に白側が取られた白石の位置に白石を打てば，黒側が置いたばかりの黒石を取れるというような状態を指す．囲碁では，このような場合，すぐ次の手番でその相手石を取り返してはならないというルールがある．そこで，相手側が絶対に応じなければならない場所に石を打ち，相手がそこに応じた後，コウを取り返すという攻防が繰り返される．よって，このコウを表すパターンは，同じ棋譜に置いて複数回出現したものと判断できる．よって終盤においては，違う棋譜同士では，全く同じ形は非常に現れにくいといえる．よって，候補手の選出にあたっては，類似判定をうまく行う必要がある．

4.2 事例検索による候補手選出方法

4.2.1 事例ベースにおける検索範囲の決定

候補手は，可能手すべての “手の価値” を計算し，そのうちの上位 5 つを取り出すことで求められる．手の価値は，事例ベースからよく似た事例を探しだし，その事例に示されている “評価改善量” と “勢力パターン” を用いて求められる．しかし，すべての事例と似ているかを判定することは，時間の無駄である．そこで，本システムでは可能手と縦横・斜めの周囲 1 マスが同じ形で

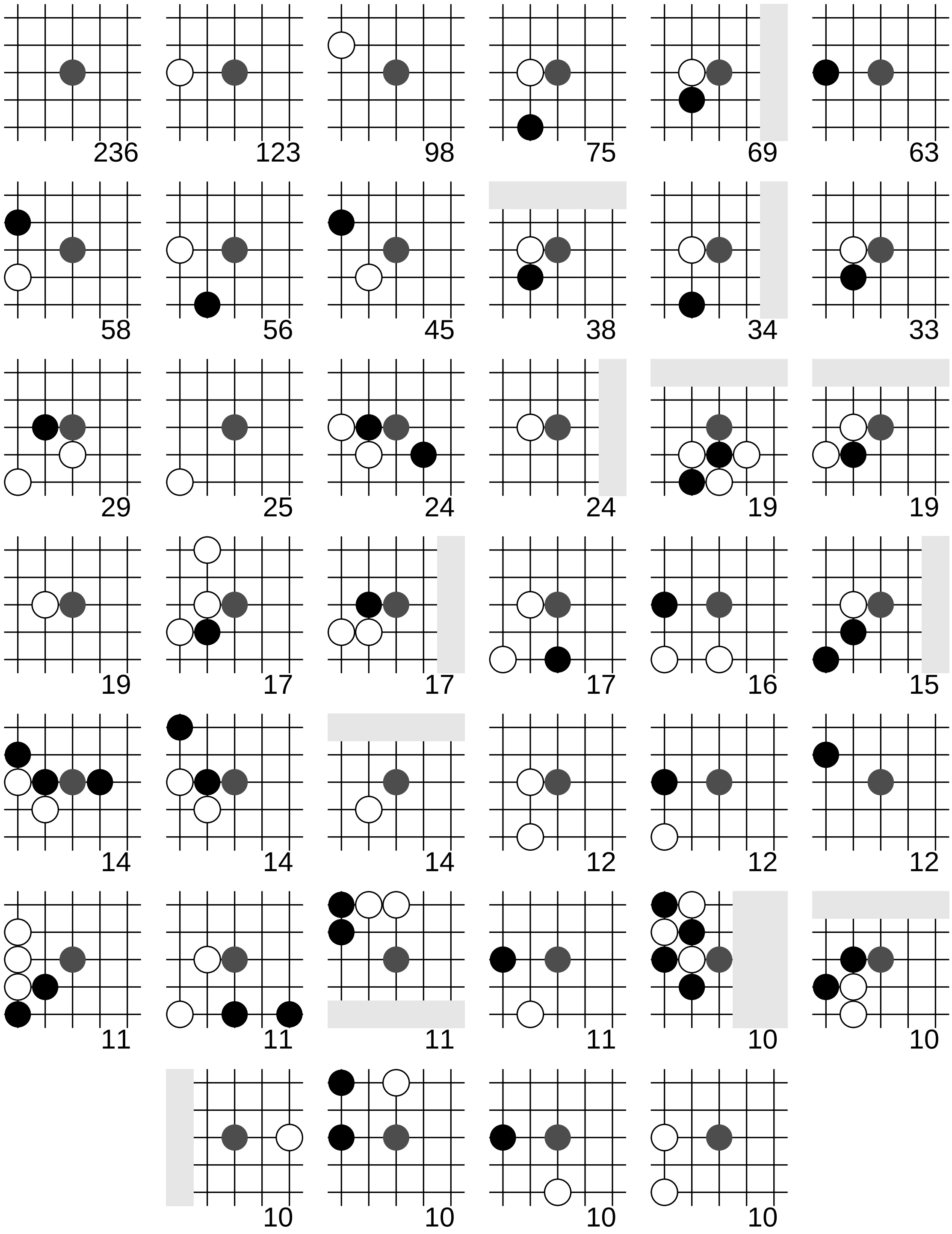


図 27: 序盤において出現頻度の高かった形

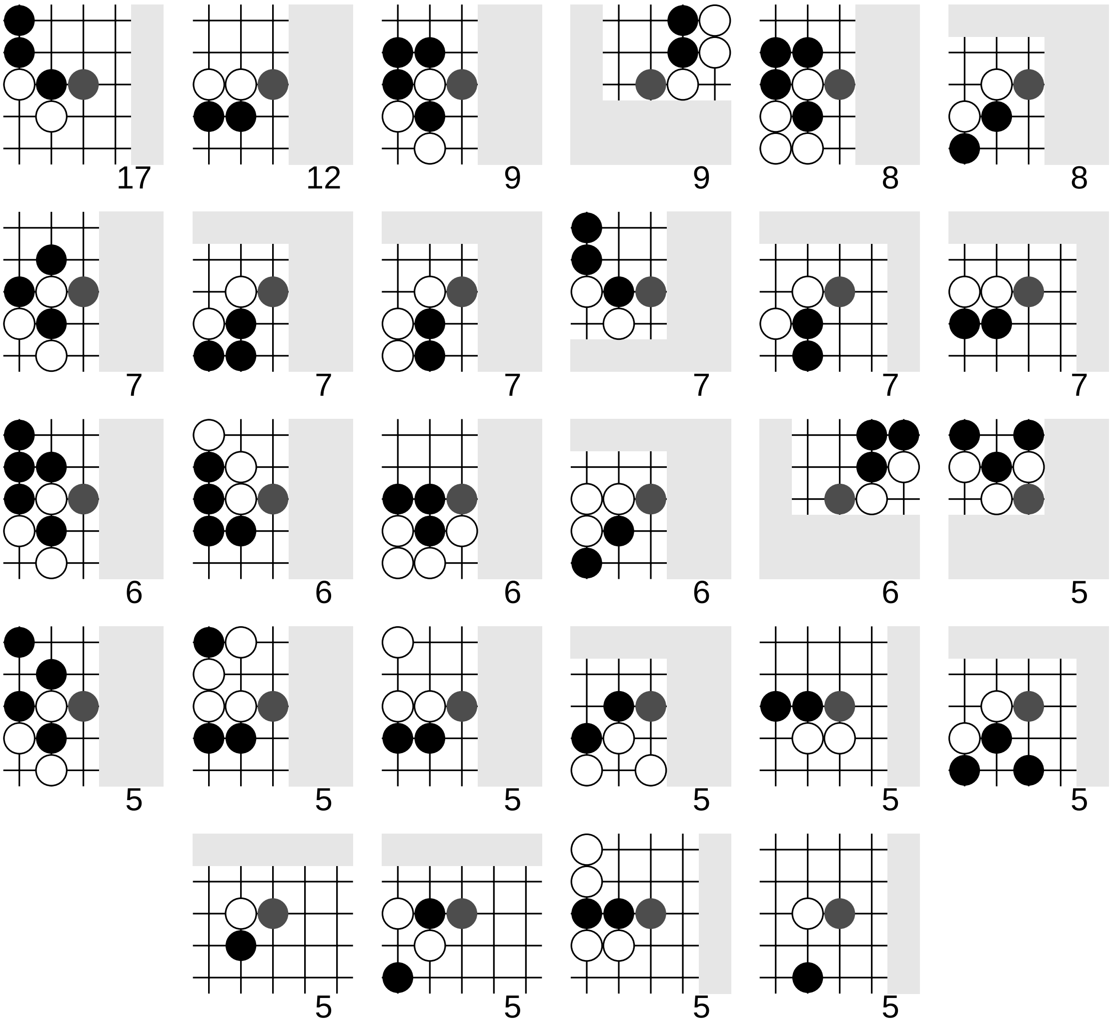


図 28: 中盤において出現頻度の高かった形

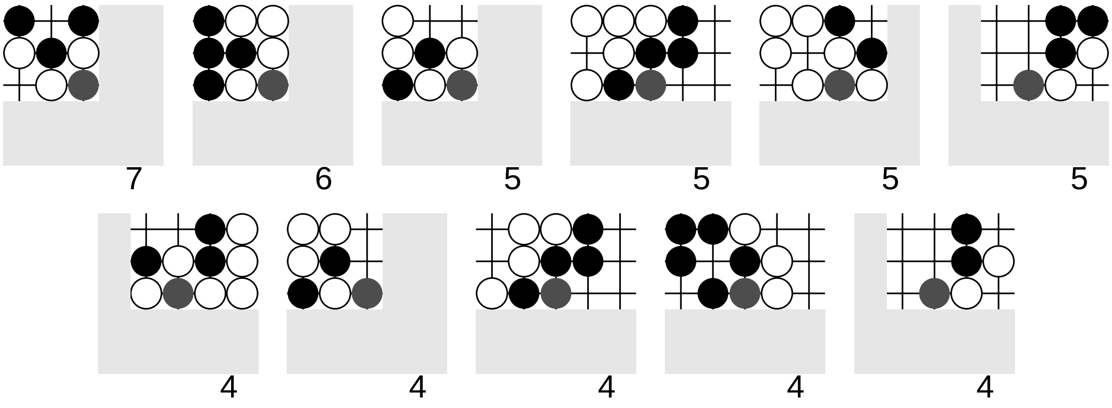


図 29: 終盤において出現頻度の高かった形

ある事例のみを検索対象としている．すなわち，可能手の周囲1マスのパターンと同じ1次検索用パターンを持つ事例のみを検索対象とする．この様子を図30に示す．

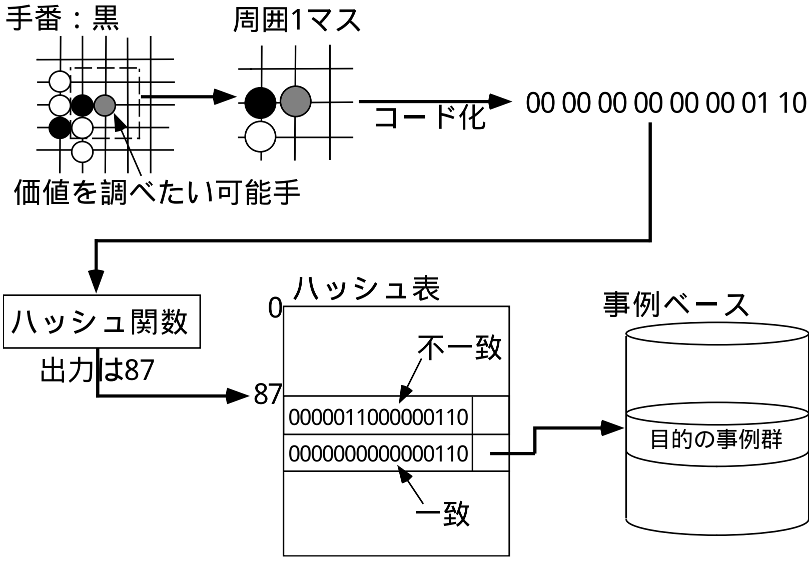


図 30: 検索範囲の決定方法

まず，最初に価値を求める可能手の周り1マスのパターンを切り出し，1次検索用パターンと同じ方法でコード化する．次に，このコードをハッシュ関数に入力し，結果をハッシュ表とつき合わせることで，目的の事例群の場所を知ることができる．

先に図26で示したように，1つのパターンに対し，他に7つの同一の形を表すパターンがある．よって，可能手と事例との比較を行うときには，事例をすべての方向に変換して類似性を調べる必要がある．しかし，それでは検索に時間がかかるので，本システムでは，可能手の方を類似事例が存在する方向に回転させてから類似判定を行っている．すなわち，上記8パターンのどれを入力しても，目的の事例群が見つけられるように，それぞれについて変換方法の情報を保持している．この様子を図31に示す．

図の3つのパターンのうち中央のものは，最小の1次検索用パターンを持つ方向であり，このままの向きで同じ1次検索用パターンを持つ事例と類似判定を行えばよい．しかし，一番上のものは最小の1次検索用パターンを持つ方向ではないので，このまま類似判定を行おうとしても同じ1次検索用パターンを持つ事例群が存在せず，判定を行えない．そこで，ハッシュ表の指示にしたがってパターンの向きを変換する．図では，ハッシュ表に“時計周りに90度回転させる”とあるので，この指示に従うと，中央のパターンと同じ形になり，類似判定を行う事例群を見つけることができる．

しかし，図32の様に，1次検索用パターンを最小値にする変換方法が二つ以上存在する場合がある．図では，‘a’のパターンを1次検索用パターンが最小となる方向に向けるには，“そのまま何もしない”か左右を入れ換える（‘b’のパターンになる）の二通りが考えられる．このような場合，ハッシュ表はこの二通りの方法による類似判定を指示する．これにより，パターン‘a’が入力

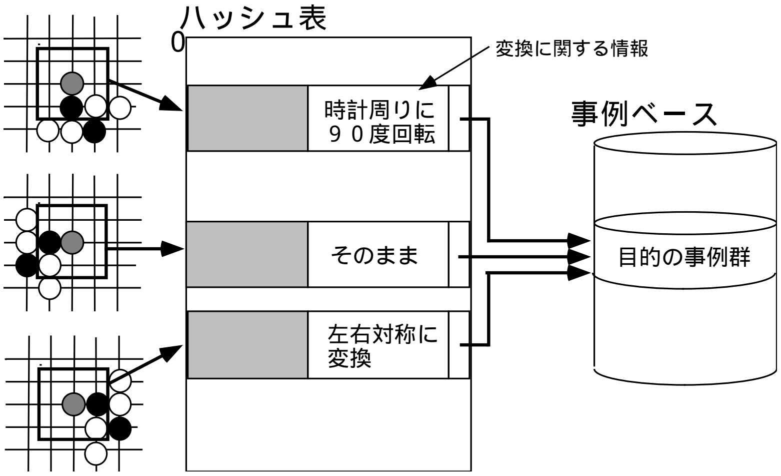


図 31: 回転および左右対象を含めた検索範囲の決定方法

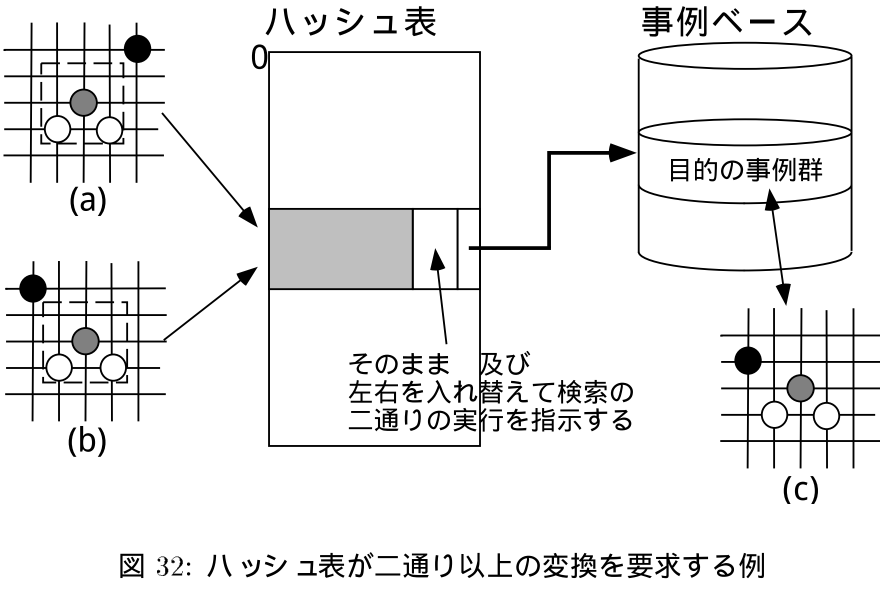


図 32: ハッシュ表が二通り以上の変換を要求する例

されても，事例‘c’との類似性が正しく判断される．

4.2.2　手の価値の決定方法

検索対象の事例を限定すると，手の価値を求める作業に入る．手の価値は，評価改善量と事例有効率から求められる．

評価改善量とは，その手により盤面の評価をどれだけ改善できるかを表す見込みの値であり，事例に記されている．可能手の局面と事例に示される局面が，形的にも意味的にも同じとき，事例に示されるのと同じ効果が発揮される．しかし，実際には全く同じではないので，それほどの効果は発揮されないと考えられる．すなわち，評価改善量とは，事例の手と同じ効果を発揮した場合に，盤面の評価が最大でどれだけ改善されるかを表す値である．

事例有効率とは，事例と同じ効果を発揮できる可能性を表す．可能手の局面と事例との局面が全く同じならば，可能手には事例と同じ効果があると考えられるが，実際には完全に一致することは少ない．しかし，可能手と事例の局面が類似しているほど，同じ効果を発揮する可能性が高い．そこで，事例と可能手の局面の類似性を基に事例有効率を求め，これを補正に使うことで，より実際に近い手の価値を求めることを目指している．

手の価値が求める手順を，図 33に示す．最初に状況判断部で計算された勢力図から，可能手の周囲の勢力パターンを取り出す．次に，取り出したパターンと事例の勢力パターンを比較する．比較は，パターン同士の各マスごとの差の絶対値を求めることにより行われる．求められた絶対値を足し合わせたものが類似度であり，低い値を持つほどより類似していることを表す．次に，類

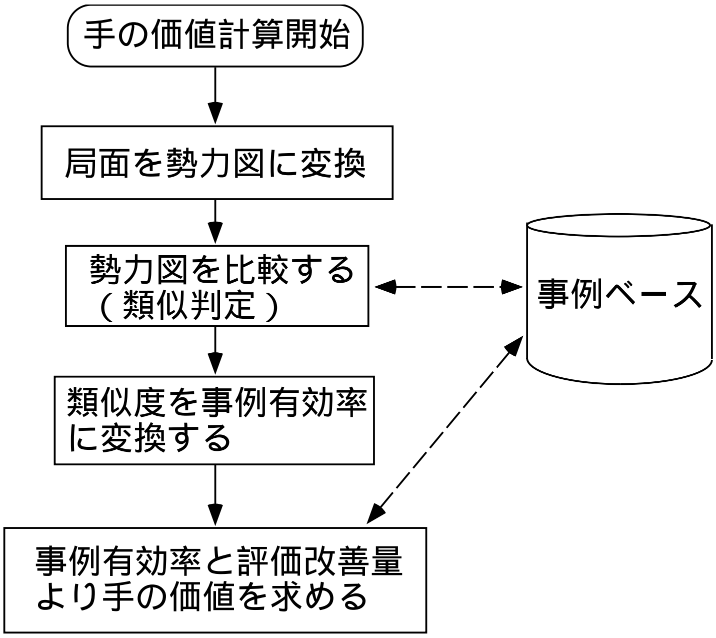


図 33: 手の価値を求める手順

似度を事例有効率に変換するため，以下の計算を行う．

$$\text{事例有効率} = \frac{(\text{事例の勢力パターンの合計値}) - (\text{類似度} \times 2)}{(\text{事例の勢力パターンの合計値})} \times 100$$

結果が負の値を取るときは，結果を 0 とする．よって事例有効率は，0 から 100 までの値を取る．類似度が 0 の場合，事例有効率は 100 となり，類似度が “事例の勢力パターンの合計値” の半分を越えるとき 0 となる．すなわち，事例有効率は高い値を持つほど，より高い類似性を持っており，事例として有効であることを示している．事例有効率の計算をこのようにしたのは，周りに石が少ないときほど，少しの違いがより重要になるようにするためである．すなわち，周りに石が多くパターンが大きな勢力値で埋め尽くされているときは，多少の差は事例の有効性に大きな影響を与えない．しかし，周りに石が少なく勢力パターンの大半が 0 のようなときには，少しの違いが事例の有効性に大きな影響を与えるということである．

最後に，求めた事例有効率に評価改善量をかけて手の価値を求める．評価改善量は事例獲得時に評価関数を使って計算されているので，着手決定と同じ基準に基づいている．

次に実際に可能手の価値を求める様子を図 34に示す．最初に事例有効率を求めるため，可能手の周囲の勢力値が勢力図より取り出される．取り出された勢力図は，事例の勢力パターンと比較される．例えば，勢力パターンの一番左上の部分は，$|6 - 16| = 10$ と計算される．パターン全体について差を求め合計したものが類似度であり，この場合は 64 となる．事例の勢力パターンの合計値は 134 であるので，これと類似度を事例有効率の式に代入すると，

$$\frac{134 - 64 \times 2}{134} \times 100 = 4.48(\%)$$

となる．よって手の価値は，これと事例に示される評価改善量をかけ合わせて，

$$23 \times 4.48 \approx 103$$

となる．この例では，“可能手の局面” の上側にある黒石の位置が事例と異なることと，右側の白石 ‘X’ が及ぼす影響力が事例にはないことが，評価の低さにつながっている．もし仮に，右側の白石 ‘X’ がなければ，事例有効率の計算は，

$$\frac{134 - 40 \times 2}{134} \times 100 = 40.3(\%)$$

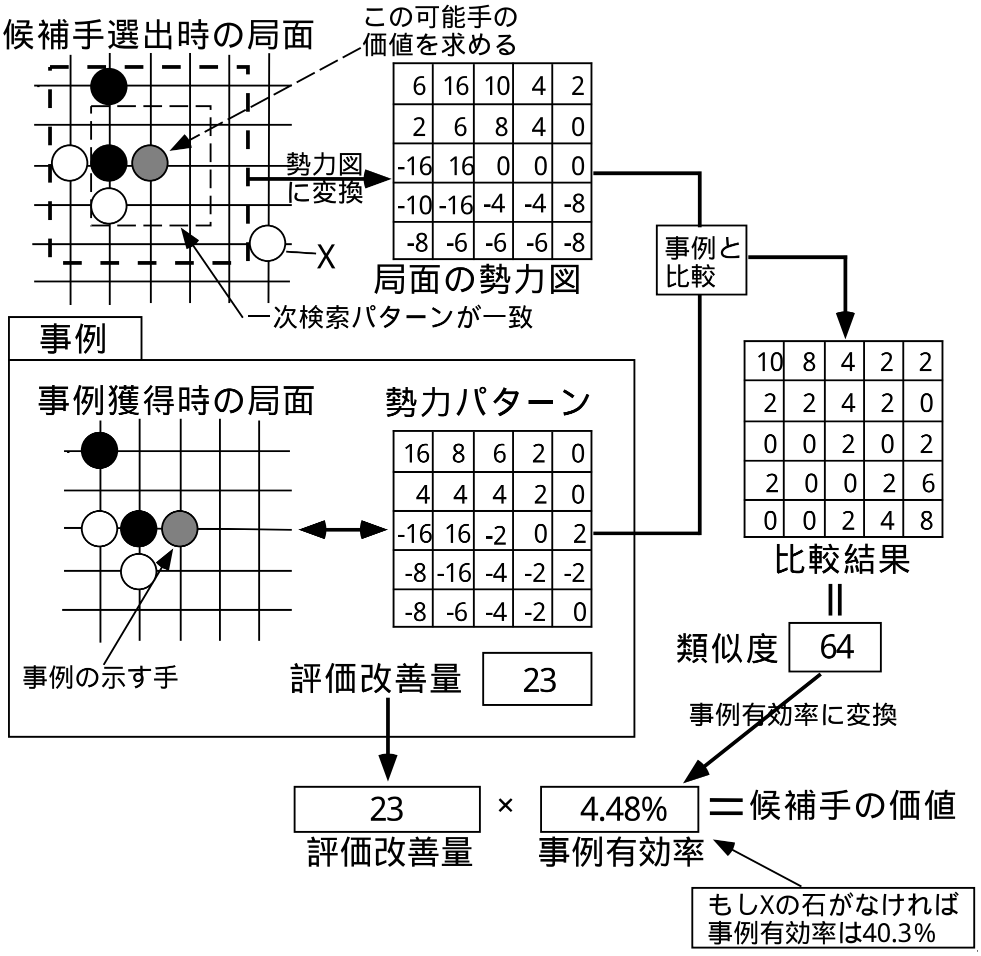
となる．このように，勢力を用いて類似性を比較する方法には，5×5 のパターンの外側の石からの影響力も考慮しているという特徴がある．

4.3 評価関数による着手決定

4.3.1 評価関数

本システムは，盤面の価値を求めるための評価関数を持つ．評価関数の判定項目は，以下の通りである．

- 死活



局面の勢力図

6	16	10	4	2
2	6	8	4	0
-16	16	0	0	0
-10	-16	-4	-4	-8
-8	-6	-6	-6	-8

勢力パターン

16	8	6	2	0
4	4	4	2	0
-16	16	-2	0	2
-8	-16	-4	-2	-2
-8	-6	-4	-2	0

比較結果

10	8	4	2	2
2	2	4	2	0
0	0	2	0	2
2	0	0	2	6
0	0	2	4	8

図 34: 手の価値算出の具体例

  - 死活判定の結果
  - 群の大きさ
- 地とアゲハマ
  - 地
  - アゲハマ
  - 全勢力範囲数

“死活” は，個々の群の生死に対して得点を与えるもので，将来 “地” を形成する群に先行投資する役割を持っている．“地とアゲハマ” は，囲碁の勝敗を決める最終的な観点から見た評価である．“死活” は，“死活判定の結果” と “群の大きさ” の二つの要素から決定される．死活判定の結果は，死活判定部により求められた結果に対し，表 4に応じた得点を与えることで求められる．

表 4: 死活判定による得点

	序盤 （1～20 手目）	中盤 （21 手目～40 手目）	終盤 （41 手目以降）
“生き”	30	60	70
“死に”	0	0	0

死活判定部の結果が不定のときは，生死の可能性の度合に応じて得点を与える．この得点は，図 35で表わされる関数により求められ，表 4に示される生きの場合の得点を上回ることはなく，逆に死にの場合を下回ることはない．

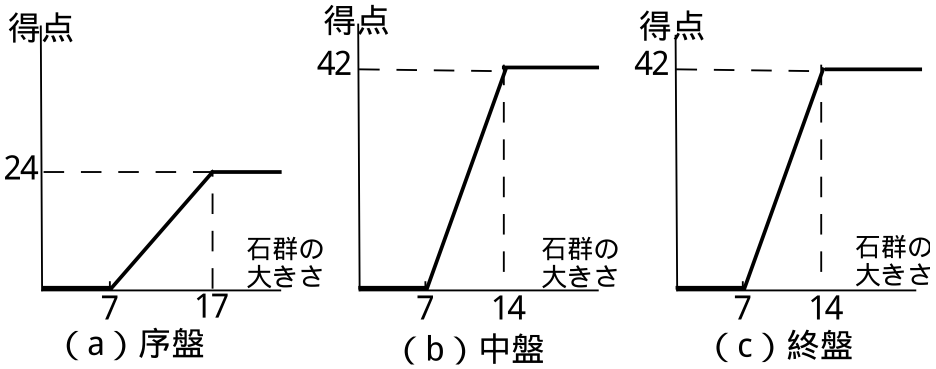


図 35: 死活が不定の群の得点

この関数は，“群の大きさ”，すなわちその群がどれだけの空白地を勢力下に持っているかを入力としている．すなわち，不定の石群に対しては，その大きさを基に生死の可能性を推測して得点を与えている．

“地とアゲハマ” は，地計算部の結果と現在までに取った敵石の数を足すことにより求められる．ただし，対局の前半，特に序盤において，地が確定することは稀であるので，全勢力範囲数を地の予想値として用いる．このようにした理由は，自身の勢力下にある空点のうち，境界が完成したものが地となるからである．しかし，勢力下にある空点は敵の攻撃により地でなくなる可能性もあり，地より価値が低い．そこで，表5の重みをそれぞれにかけている．“地 + アゲハマ” は，

表 5: “地とアゲハマ” の計算に用いられる重み

	序盤（1～20手目）	中盤（21手目～40手目）	終盤（41手目以降）
地 + アゲハマ	1.0	1.4	1.8
全勢力範囲数 − 地	1.0	1.1	1.2

現在までに確定した得点を表している．地を確定する必要があるのは対局の後半であるので，終盤に向かうほどその重みは上昇する．“全勢力範囲数 − 地” は，地として確定していないが勢力下である空点の数である．序盤においては，“地 + アゲハマ” と同じ価値を持つが，対局が進むにつれて，その重みは “地 + アゲハマ” を下回っていく．

以上の二つの項目について味方側について得点を求め，結果を足し合わせる．次に，敵側についても同様に得点を求め，これを味方側の得点から引く．このようにして求められた値が，味方側から見た盤面の評価値であり，評価関数の出力となる．

4.3.2 着手決定方法

着手が決定される様子を図36に示す．

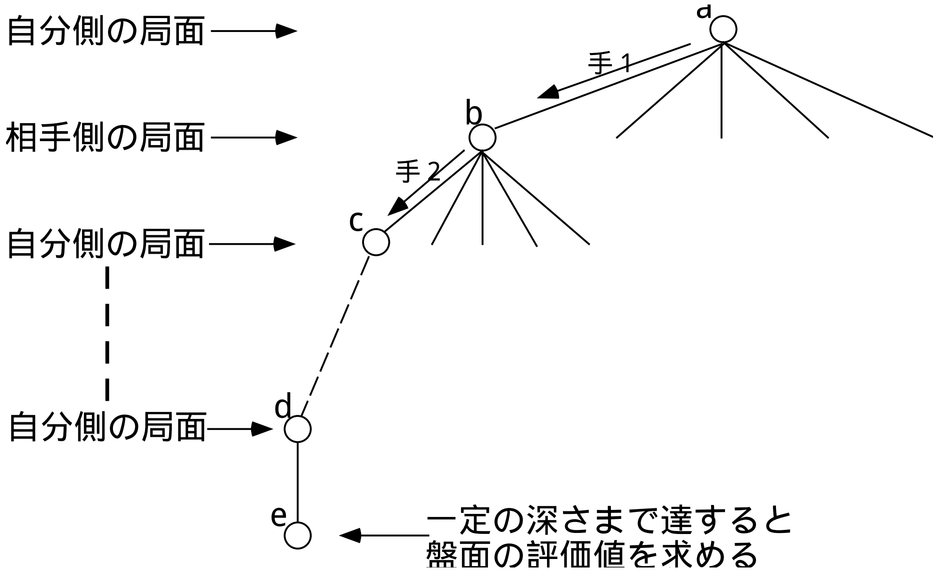


図 36: 着手決定の様子

着手決定部では，候補手選出部によって求められた候補手を枝にして木探索を行い，着手を決定する．最初に，候補手の一つ（ここでは‘手 1’）を打った場合の局面（‘b’ の状態）を生成し，状況判断部に送る．状況判断部は，これを分析し，結果を候補手選出部に渡す．候補手選出部では，この局面と状況表をもとに，候補手を選び出す．この候補手は，コンピュータの敵側（人間側）から見た手であり，‘1’ の手を打った場合に敵が打つと予想される手である．次に敵側がこのうちの一つを打った場合の局面を生成し，同様の処理を続ける．そして，この探索が一定の深さに達したとき（‘e’ の状態），評価関数を呼び出し盤面の評価を求める．そして次に，一つ前の局面に戻り，別の候補手を選択したときの盤面の評価値を求めるという様に探索を続けていく．このようにして，木を構成し，すべての木の末端での評価値を求める．求められた評価値から，各節点においてどの手を選択すればいくらの評価値を得ることができるかを知ることができる．この様子を図 37に示す．評価はすべて味方側から見た値を示しており，高い値ほど評価がよいことを示す．

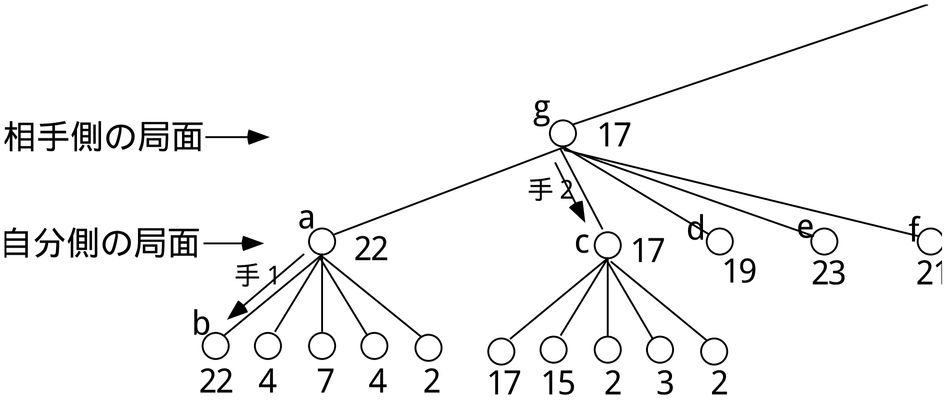


図 37: 木の各節点における評価

‘a’ の状態で，得られる最高の評価値は，‘手 1’ を選んだときの ‘22’ である．同様にして，‘c’ の局面では ‘17’ の評価値を得られることが分かる．このようにして，‘d’ の局面で ‘19’ の評価値，‘e’ の局面で ‘23’ の評価値を，‘f’ の局面では ‘21’ の評価値が得られたとする．次に，‘g’ の局面の評価値を考える．この局面は敵側の手番であるから，最も低い評価値が得られる手が選ばれると考えられる．従って，‘手 2’ を選んだときの ‘17’ が ‘g’ の局面で得られる最大の評価値である．このようにして，一番上の節点（すなわち現在の局面）まで計算を続けると，各手についてどれだけの評価が得られるかが決定されるので，その中から一番高い評価の得られる手を着手として採用する．

ただし，すべての木を構成してから評価を行うのは資源の無駄であるので，実際には，木の構成と評価を同時に行い，不必要となった部分木は次々と削除していく方法をとっている．

5　実験および考察

5.1　実験

5.1.1　実験方法

本システムが，うまく機能するためには，まず候補手が正しく選ばれる必要がある．そこで良質の候補手が選ばれるかを調べる実験を行った．実験は，プロ・上級アマが打った手を候補手に含むかを調べた．

(1) 形が完全に一致するものを候補手とする場合

以下に具体的な方法を示す．

1. 事例獲得に使用しなかった 40 個の棋譜を使用する．
2. 棋譜を一手ずつ再現させ，その度に候補手選出部を呼び出し，候補手を求める．
3. 候補手選出部では，事例と形（5×5 のパターン）部分が完全に一致しているもののみを候補手として選出する．
4. 候補手の中に，棋譜に示される次の手が含まれていれば正答とする．
5. 棋譜内のすべての手について正答かを判定し，結果を序盤・中盤・終盤に分けて集計する．
6. また，1 局面において候補手がどれくらい現れるのかを調べるため，正答の場合の平均候補手数を序盤・中盤・終盤のそれぞれについて求める．
7. 以上の結果をすべての棋譜について求める．

平均候補手数を正答の場合にのみ求めているのは，もし事例の不足のため候補手を生成できないような局面が多数生じた場合に，その影響を受け候補手数が減ってしまうからである．すなわち，事例の不足の影響を極力除いた結果を得るためである．

この実験により，形知識主導型のコンピュータ囲碁システムの有効性を検証すると共に，後の実験の比較材料として用いる．

(2) 事例有効率により候補手を選出する場合

実験方法は，以下の通りである．

1. 候補手選出部では，事例有効率が最も良いもの上位 5 つを候補手とする．
2. 候補手数を 5 手に制限しているので，平均候補手数は求めない．
3. それ以外は，(1) と同様の条件で行う．

本実験は，本システムの事例有効率が正しく機能しているかを調べると共に，候補手を 5 手まで絞った場合にどの程度の割合で正答を含むのかを調べるためである．候補手を 5 手としたのは，候補手選出部の次の着手決定部で木探索を用いているからである．すなわち，探索の爆発を避け，かつ良手を候補手から落さないようにするため，この値を用いた．

次に，候補手をより絞り込んだ場合の性能を評価するために，候補手を 3 手に制限した場合についても同様の実験を行う．そして，逆に候補手の制限を甘くした場合の性能の変化を見るために，候補手を 7 手とした場合についても同様の実験を行う．また，候補手数が 1，9，11，13 手のときの序盤・中盤・そして終盤における正答率の平均を調べ，さきの結果と合わせてグラフを作成する．

(3) 手の価値を求め候補を選出する場合

手の価値の算出がうまく機能していれば，その結果は事例有効率（類似度）に基づいて候補手を求めた場合より，よい結果が得られるはずである．しかし，パラメータは長い時間をかけて調整を行うべきものであり，いきなり良い結果が得られるとは考えにくい．本実験では，このパラメータが適切に設定されているかを調べる．

事例の評価改善量は，評価関数を用いて計算されているが，本実験では，特別に以下のような評価方法を取っている．

- “死活” 項目については，各群の死活を全て不定とし，“群の大きさ” に基づいて得点を与える．
- “地とアゲハマ” 項目については，全勢力範囲数とアゲハマのみを計算対象とする．

これは，本システムの死活判定部の性能が安定していなかったことと，地計算部が未完成であったことが原因である．しかし，これらの影響が出るのは対局が進んでからであるので，序盤・中盤においては，パラメータが適切であるかの判断をすることができる．実験方法は，候補手選出の方法を除いて，(2) と同じである．

5.1.2 実験結果

形を基に候補手を選んだ場合の実験結果を表 6に示す．

事例有効率を基に候補手を 5 つ上げた場合の実験結果を表 7に，同じく候補手を 3 つ，そして 7 つ上げた場合の実験結果をそれぞれ表 8，表 9に示す．そして，候補手数の変化とともに平均正答率がどのように変化したかを，図 38に示す．

最後に手の価値を用いて，候補手を 5 つ上げた場合の実験結果を表 10に示す．

表 6: 形の完全一致による候補手選出結果

	序盤				中盤				終盤			
棋譜番号	正答	総数	正答率	候補数	正答	総数	正答率	候補数	正答	総数	正答率	候補数
201	9	20	45.0	13.6	3	13	23.1	2.3	0	0	—	—
202	11	20	55.0	13.2	3	20	15.0	3.3	7	17	41.2	1.7
203	9	20	45.0	12.4	0	1	0.0	—	0	0	—	—
204	14	20	70.0	10.8	2	2	100.0	3.5	0	0	—	—
205	7	20	35.0	12.0	3	20	15.0	4.0	0	3	0.0	—
206	12	20	60.0	13.0	2	20	10.0	3.0	1	9	11.1	1.0
207	11	20	55.0	14.1	4	20	20.0	3.5	0	1	0.0	—
208	13	20	65.0	11.3	3	20	15.0	3.0	0	2	0.0	—
209	7	20	35.0	12.9	5	18	27.8	4.0	0	0	—	—
210	18	20	90.0	13.7	3	14	21.4	3.7	0	0	—	—
211	9	20	45.0	12.4	2	20	10.0	3.5	1	5	20.0	2.0
212	12	20	60.0	12.2	0	20	0.0	—	0	11	0.0	—
213	12	20	60.0	14.3	0	13	0.0	—	0	0	—	—
214	10	20	50.0	14.4	8	20	40.0	3.5	2	7	28.6	1.5
215	16	20	80.0	13.2	4	20	20.0	5.5	0	2	0.0	—
216	9	20	45.0	12.7	0	20	0.0	—	0	1	0.0	—
217	7	20	35.0	18.1	4	19	21.1	2.5	0	0	—	—
218	18	20	90.0	15.2	2	20	10.0	2.0	0	4	0.0	—
219	8	20	40.0	13.5	1	13	7.7	4.0	0	0	—	—
220	10	20	50.0	13.4	4	20	20.0	4.8	1	6	16.7	1.0
221	11	20	55.0	14.9	3	20	15.0	3.7	0	5	0.0	—
222	14	20	70.0	12.7	8	20	40.0	3.6	0	10	0.0	—
223	10	20	50.0	11.2	3	20	15.0	2.0	1	5	20.0	1.0
224	6	20	30.0	19.8	1	20	5.0	2.0	0	7	0.0	—
225	8	20	40.0	17.1	4	20	20.0	3.5	1	16	6.2	1.0
226	9	20	45.0	14.7	1	6	16.7	5.0	0	0	—	—
227	6	20	30.0	11.8	5	20	25.0	2.4	1	21	4.8	1.0
228	10	20	50.0	13.2	1	20	5.0	4.0	1	12	8.3	2.0
229	16	20	80.0	12.7	6	17	35.3	2.2	0	0	—	—
230	12	20	60.0	11.9	2	17	11.8	4.0	0	0	—	—
231	13	20	65.0	14.4	3	20	15.0	3.7	0	3	0.0	—
232	10	20	50.0	18.7	3	11	27.3	2.7	0	0	—	—
233	15	20	75.0	10.0	2	15	13.3	2.0	0	0	—	—
234	10	20	50.0	13.5	6	20	30.0	3.7	1	3	33.3	2.0
235	8	20	40.0	13.2	0	20	0.0	—	2	5	40.0	1.5
236	8	20	40.0	12.2	4	13	30.8	3.0	0	0	—	—
237	12	20	60.0	13.8	0	4	0.0	—	0	0	—	—
238	11	20	55.0	11.1	6	20	30.0	4.3	0	4	0.0	—
239	9	20	45.0	15.1	5	11	45.5	1.4	0	0	—	—
240	10	20	50.0	13.4	2	4	50.0	4.0	0	0	—	—
TOTAL	430	800	53.8	13.4	118	651	18.1	3.3	19	159	11.9	1.5

表 7: 事例有効率に基づいて候補手を 5 つ上げた場合の実験結果

	序盤			中盤			終盤		
棋譜番号	正答	総数	正答率	正答	総数	正答率	正答	総数	正答率
201	9	20	45.0	9	13	69.2	0	0	—
202	9	20	45.0	7	20	35.0	9	17	52.9
203	11	20	55.0	0	1	0.0	0	0	—
204	16	20	80.0	2	2	100.0	0	0	—
205	8	20	40.0	7	20	35.0	0	3	0.0
206	12	20	60.0	4	20	20.0	5	9	55.6
207	13	20	65.0	5	20	25.0	0	1	0.0
208	11	20	55.0	7	20	35.0	1	2	50.0
209	10	20	50.0	12	18	66.7	0	0	—
210	13	20	65.0	7	14	50.0	0	0	—
211	8	20	40.0	7	20	35.0	4	5	80.0
212	11	20	55.0	10	20	50.0	4	11	36.4
213	15	20	75.0	2	13	15.4	0	0	—
214	12	20	60.0	10	20	50.0	6	7	85.7
215	12	20	60.0	2	20	10.0	0	2	0.0
216	12	20	60.0	5	20	25.0	0	1	0.0
217	7	20	35.0	10	19	52.6	0	0	—
218	16	20	80.0	7	20	35.0	1	4	25.0
219	12	20	60.0	3	13	23.1	0	0	—
220	8	20	40.0	6	20	30.0	1	6	16.7
221	11	20	55.0	6	20	30.0	2	5	40.0
222	11	20	55.0	8	20	40.0	3	10	30.0
223	9	20	45.0	6	20	30.0	4	5	80.0
224	9	20	45.0	0	20	0.0	1	7	14.3
225	11	20	55.0	9	20	45.0	6	16	37.5
226	8	20	40.0	2	6	33.3	0	0	—
227	10	20	50.0	9	20	45.0	12	21	57.1
228	8	20	40.0	5	20	25.0	7	12	58.3
229	14	20	70.0	11	17	64.7	0	0	—
230	9	20	45.0	2	17	11.8	0	0	—
231	9	20	45.0	6	20	30.0	1	3	33.3
232	12	20	60.0	4	11	36.4	0	0	—
233	14	20	70.0	4	15	26.7	0	0	—
234	12	20	60.0	8	20	40.0	2	3	66.7
235	9	20	45.0	3	20	15.0	4	5	80.0
236	10	20	50.0	5	13	38.5	0	0	—
237	11	20	55.0	1	4	25.0	0	0	—
238	11	20	55.0	6	20	30.0	2	4	50.0
239	8	20	40.0	10	11	90.9	0	0	—
240	7	20	35.0	2	4	50.0	0	0	—
TOTAL	428	800	53.5	229	651	35.2	75	159	47.2

表 8: 事例有効率に基づいて候補手を 3 つ上げた場合の実験結果

	序盤			中盤			終盤		
棋譜番号	正答	総数	正答率	正答	総数	正答率	正答	総数	正答率
201	5	20	25.0	6	13	46.2	0	0	—
202	7	20	35.0	3	20	15.0	8	17	47.1
203	8	20	40.0	0	1	0.0	0	0	—
204	15	20	75.0	2	2	100.0	0	0	—
205	5	20	25.0	6	20	30.0	0	3	0.0
206	10	20	50.0	1	20	5.0	5	9	55.6
207	5	20	25.0	5	20	25.0	0	1	0.0
208	7	20	35.0	4	20	20.0	1	2	50.0
209	5	20	25.0	7	18	38.9	0	0	—
210	9	20	45.0	6	14	42.9	0	0	—
211	5	20	25.0	3	20	15.0	4	5	80.0
212	10	20	50.0	6	20	30.0	1	11	9.1
213	11	20	55.0	0	13	0.0	0	0	—
214	8	20	40.0	5	20	25.0	5	7	71.4
215	9	20	45.0	1	20	5.0	0	2	0.0
216	10	20	50.0	1	20	5.0	0	1	0.0
217	6	20	30.0	9	19	47.4	0	0	—
218	12	20	60.0	5	20	25.0	1	4	25.0
219	5	20	25.0	1	13	7.7	0	0	—
220	5	20	25.0	3	20	15.0	0	6	0.0
221	11	20	55.0	3	20	15.0	2	5	40.0
222	8	20	40.0	6	20	30.0	3	10	30.0
223	5	20	25.0	5	20	25.0	3	5	60.0
224	7	20	35.0	0	20	0.0	1	7	14.3
225	8	20	40.0	4	20	20.0	2	16	12.5
226	5	20	25.0	2	6	33.3	0	0	—
227	7	20	35.0	8	20	40.0	3	21	14.3
228	5	20	25.0	5	20	25.0	6	12	50.0
229	11	20	55.0	9	17	52.9	0	0	—
230	6	20	30.0	0	17	0.0	0	0	—
231	7	20	35.0	6	20	30.0	1	3	33.3
232	10	20	50.0	3	11	27.3	0	0	—
233	12	20	60.0	4	15	26.7	0	0	—
234	6	20	30.0	7	20	35.0	1	3	33.3
235	7	20	35.0	1	20	5.0	4	5	80.0
236	6	20	30.0	5	13	38.5	0	0	—
237	5	20	25.0	0	4	0.0	0	0	—
238	8	20	40.0	5	20	25.0	1	4	25.0
239	3	20	15.0	6	11	54.5	0	0	—
240	6	20	30.0	2	4	50.0	0	0	—
TOTAL	300	800	37.5	155	651	23.8	52	159	32.7

表 9: 事例有効率に基づいて候補手を 7 つ上げた場合の実験結果

	序盤			中盤			終盤		
棋譜番号	正答	総数	正答率	正答	総数	正答率	正答	総数	正答率
201	10	20	50.0	9	13	69.2	0	0	—
202	11	20	55.0	10	20	50.0	10	17	58.8
203	13	20	65.0	0	1	0.0	0	0	—
204	16	20	80.0	2	2	100.0	0	0	—
205	8	20	40.0	10	20	50.0	0	3	0.0
206	14	20	70.0	4	20	20.0	5	9	55.6
207	14	20	70.0	6	20	30.0	0	1	0.0
208	14	20	70.0	11	20	55.0	1	2	50.0
209	11	20	55.0	13	18	72.2	0	0	—
210	15	20	75.0	8	14	57.1	0	0	—
211	11	20	55.0	10	20	50.0	4	5	80.0
212	11	20	55.0	11	20	55.0	5	11	45.5
213	16	20	80.0	4	13	30.8	0	0	—
214	14	20	70.0	14	20	70.0	7	7	100.0
215	15	20	75.0	4	20	20.0	0	2	0.0
216	13	20	65.0	8	20	40.0	1	1	100.0
217	7	20	35.0	10	19	52.6	0	0	—
218	16	20	80.0	10	20	50.0	2	4	50.0
219	12	20	60.0	4	13	30.8	0	0	—
220	11	20	55.0	6	20	30.0	2	6	33.3
221	11	20	55.0	6	20	30.0	2	5	40.0
222	12	20	60.0	11	20	55.0	5	10	50.0
223	11	20	55.0	9	20	45.0	4	5	80.0
224	10	20	50.0	1	20	5.0	2	7	28.6
225	11	20	55.0	10	20	50.0	6	16	37.5
226	10	20	50.0	2	6	33.3	0	0	—
227	10	20	50.0	11	20	55.0	13	21	61.9
228	11	20	55.0	6	20	30.0	9	12	75.0
229	16	20	80.0	13	17	76.5	0	0	—
230	10	20	50.0	4	17	23.5	0	0	—
231	13	20	65.0	6	20	30.0	1	3	33.3
232	15	20	75.0	6	11	54.5	0	0	—
233	14	20	70.0	5	15	33.3	0	0	—
234	17	20	85.0	11	20	55.0	2	3	66.7
235	12	20	60.0	5	20	25.0	4	5	80.0
236	12	20	60.0	6	13	46.2	0	0	—
237	13	20	65.0	1	4	25.0	0	0	—
238	12	20	60.0	10	20	50.0	3	4	75.0
239	9	20	45.0	10	11	90.9	0	0	—
240	7	20	35.0	2	4	50.0	0	0	—
TOTAL	488	800	61.0	289	651	44.4	88	159	55.3

表 10: 手の価値に基づいて候補手を 5 つ上げた場合の実験結果

	序盤			中盤			終盤		
棋譜番号	正答	総数	正答率	正答	総数	正答率	正答	総数	正答率
201	7	20	35.0	5	13	38.5	0	0	—
202	9	20	45.0	8	20	40.0	10	17	58.8
203	9	20	45.0	0	1	0.0	0	0	—
204	15	20	75.0	2	2	100.0	0	0	—
205	5	20	25.0	9	20	45.0	0	3	0.0
206	9	20	45.0	7	20	35.0	6	9	66.7
207	11	20	55.0	7	20	35.0	0	1	0.0
208	13	20	65.0	9	20	45.0	1	2	50.0
209	9	20	45.0	10	18	55.6	0	0	—
210	12	20	60.0	7	14	50.0	0	0	—
211	10	20	50.0	7	20	35.0	3	5	60.0
212	11	20	55.0	11	20	55.0	6	11	54.5
213	15	20	75.0	5	13	38.5	0	0	—
214	13	20	65.0	12	20	60.0	6	7	85.7
215	13	20	65.0	5	20	25.0	0	2	0.0
216	10	20	50.0	5	20	25.0	0	1	0.0
217	7	20	35.0	13	19	68.4	0	0	—
218	15	20	75.0	6	20	30.0	2	4	50.0
219	11	20	55.0	7	13	53.8	0	0	—
220	10	20	50.0	7	20	35.0	2	6	33.3
221	9	20	45.0	5	20	25.0	1	5	20.0
222	12	20	60.0	7	20	35.0	4	10	40.0
223	8	20	40.0	10	20	50.0	3	5	60.0
224	7	20	35.0	6	20	30.0	2	7	28.6
225	7	20	35.0	7	20	35.0	7	16	43.8
226	6	20	30.0	4	6	66.7	0	0	—
227	10	20	50.0	11	20	55.0	9	21	42.9
228	7	20	35.0	4	20	20.0	9	12	75.0
229	12	20	60.0	11	17	64.7	0	0	—
230	9	20	45.0	7	17	41.2	0	0	—
231	11	20	55.0	8	20	40.0	1	3	33.3
232	11	20	55.0	5	11	45.5	0	0	—
233	14	20	70.0	3	15	20.0	0	0	—
234	14	20	70.0	9	20	45.0	1	3	33.3
235	6	20	30.0	9	20	45.0	4	5	80.0
236	9	20	45.0	5	13	38.5	0	0	—
237	10	20	50.0	0	4	0.0	0	0	—
238	11	20	55.0	6	20	30.0	2	4	50.0
239	11	20	55.0	8	11	72.7	0	0	—
240	9	20	45.0	1	4	25.0	0	0	—
TOTAL	407	800	50.9	268	651	41.2	79	159	49.7

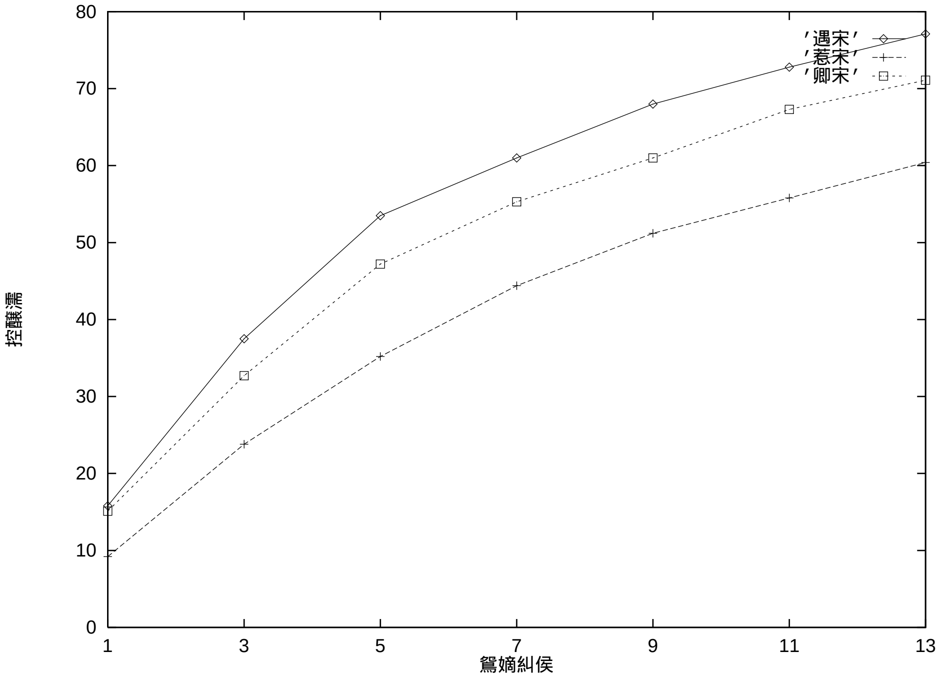


図 38: 候補手数による平均正答率の変化

5.2 考察

(1) 形の完全一致による候補手選出

実験前の予想では，事例を獲得した棋譜が 200 個であったので，完全に一致することは少なく比較的小さな値になり，終盤に向かうにつれて減少すると考えていた．しかし実際には，序盤の平均正答率は 53.8%であった．これは，序盤においては，石の密度が低く，出現する形がかなり固定されることからこのようになったと考えられる．しかし序盤に対して，中盤・終盤ではそれぞれ 18.1%，11.9%と急激に平均正答率が悪くなっている．これは，中盤・終盤においては石の密度が高く，パターン内に多くの石が現れ，多くの形が現れたためと思われる．すなわち，本方式でシステムを動かすには，事例数が不足であると判断できる．中盤・終盤における平均候補数がそれぞれ 3.3 手，1.5 手であることが，このことを裏付けている．しかし序盤においては，平均候補数は 13.4 手もあり，次の着手決定部にとって候補が多過ぎ探索の爆発が予想されるので，なんらかの方法で候補を絞る必要がある．次に各棋譜ごとについて結果を見てみると，まず序盤であるが，正答率にばらつきがある．これは，一度事例にない局面になると，あとはずっと事例にない局面が続きやすいことが原因である．また，候補数にもばらつきがあり，最大のもので最小のものの約 2 倍となっている．しかし，候補数が多いからといって正答率が上がっているかというとそのような関係はみられない．中盤においても，同様の傾向が見られるが，これは事例の不足の影響が大きいと考えられる．終盤については，棋譜番号 202 において正答が 7 もあり，他の棋譜の結果と比べて飛び抜けている．これについては，コウ争いにより同じ場所で複数回事例と一致していると考えられたが，実際はそうでなく，地を確定するため境界に打つ手がほとんどで

あった．よって事例が十分あれば，他の棋譜においてもこの程度の正答数を出せる可能性がある．結論として，形主導の候補手選出方法は何らかの工夫さえ行えば十分に有効であり，序盤においては候補手を絞り込み，中盤・終盤においては類似判定をうまく行い，ある程度の候補を常に提示できるようにすればよいことが分かる．

(2) 事例有効率を用いた候補手選出

最初に，候補手数を5つ上げた場合について検討する．予想としては，類似判定を行っているので特に中盤・終盤において，形の完全一致による候補手選出の場合に比べて正答率が上がることが考えられたが，逆に候補手を5つまで絞り込むので特に序盤において正答率の減少が心配された．結果は，序盤で53.5%，中盤で35.2%，そして終盤で47.2%となり，形の完全一致の場合に比べて序盤でほぼ同じ，中盤で約2倍，そして終盤では大幅に上昇し47.2%となった．事例有効率では勢力値で類似性を判定するとはいえ，形が基本であり，“形が完全一致の場合”で平均13手もある候補手から5手を選ぶので，単純に考えれば正答率は二分の一以下に減少しそうなものである．これについては，事例の勢力判定において5×5の外側の情報が勢力値の形で勢力パターンに現れたため，5×5の範囲で見ると同じ類似性をもつ13手に順序をつけることができ，うまく5手を選びだせたということである．他にも，事例有効率の計算式が有効に働いたとことも原因の一つである．実際，事例有効率を求める前の類似度（各マス目ごとの差の絶対値の合計）で本実験を行ってみると各棋譜ごとに正答率が2~3減少してしまった．中盤・終盤においては，予想通り正答率を上昇させることができた．すなわち，中盤・終盤においてはやはりすこしだけ形が違うという基本の形からの派生型が多く現れるため，類似判定を行うことは非常に有効であると考えられる．

次に候補手数を3手にした場合の結果についてみる．当然，候補手数が5手の場合に比べて正答率が減少することが予想されたが，どの程度下がるのかが注目された．結果は，序盤・中盤・終盤すべてにおいて正答率が約3割減少した．全てにおいて等しく3割減少したのは，候補数が5手から3手に減少，すなわち4割減少したからであり，4割でなく3割しか減少していないのは，事例有効率により候補手の順序づけがある程度正しくなされているからではないかと思われる．

候補手数を7手にした場合は，候補手が5手の場合に比べて，正答率は，序盤で1.14倍，中盤で1.26倍，終盤で1.17倍となった．候補手数が1.4倍になっていることと候補手が3手の場合の結果を合わせて考えると，妥当な結果といえる．

次に，候補手を1手から13手まで変化させたときの正答率の変化を表した図38のグラフについて考える．これを見ると，候補手数が小さいときほど，候補手数を増加させたときの正答率の上昇率が大きいことが分かる．他には，全ての候補手数において一番正答率のよいのは序盤であり，その後は終盤・中盤の順になっていることが分かる．形の完全一致の場合には，序盤・中盤・終盤の順であり，中盤と終盤の順が逆になっている．序盤の正答率が一番良いのは，盤上の石の密度が低く，形が限定されるからである．そう考えると，終盤より中盤の方が正答率がよくなりそうだが，そうはなっていない．この理由は，おそらく中盤の事例数が不足しているからだと考えられる．すなわち，中盤には多種多様な種類の手が要求され，多くの事例が必要とされるが，終盤に要求される手は，地の確定のため境界に打つ手やコウ争いなどに限定されるため少数の事例でも候補手をうまく決定できたのだと思われる．

(3) 手の価値による候補手選出

本実験の結果は，評価改善量を算出する評価関数のパラメータに大きく依存しており，どのような結果になるか予想がつかなかった．結果は，事例有効率により候補手を 5 手選ぶ場合に比べて，正答率が序盤で-2.6%の減少，中盤で 14.3%，終盤で 2.5%の増加となった．中盤で少し増加しているが，事例有効率より計算量を必要とする方法であるので，期待通りの性能とはいえない結果になった．中盤で大きく上昇したのは，中盤の事例が不足したために事例有効率の方法において性能が出なかったことが原因ではないかと考えられる．

手の価値を求めることの目的は，形は完全に一致していないが人の目から見て価値のある手を，他の完全に一致してはいるが余り価値のない手より，高く評価できるようにしようというものである．正しく価値判断ができているかは，本研究の結果から本システムのパラメータでは不適切であることがわかる．これについては，時間をかけてパラメータを変更し必要ならば新しい概念を導入する必要がある．では，評価ができた場合に正答率をあげる余地があるのかを分析するため，再び図 38を見てみる．図を見れば分かるように，候補手数が増えれば正答の割合が上がり，その増加の割合は候補手が少ない時ほど急激である．これは，事例有効率の高い順に可能手を並べたとき，実際に打たれた手の多くは上位の位置を占めるが，必ずしも 1 ~ 5 番といった最上位には含まれないことを表している．いいかえれば，候補手数を絞ろうとすればするほど正答率は減少するので，何らかの方法により実際に打たれるような価値ある手を上位に動かし，形は似ているが価値の低い手を後ろに下げなければ，良質の候補手は得られないといえる．よって，候補手の数を小さく押える必要があるときには，手の価値により候補手を求める方法は必要であると考えられる．

6　おわりに

本論文では，事例を用いたコンピュータ 9 路盤囲碁システムの全体構成とそのデータ構造，候補手選出に用いられる事例の形式，およびシステムの主要部分である事例獲得部，状況判断部，候補手選出部，そして着手決定部がどのように実装されているかを具体的に述べた．また，本システムの性能を調べるため，候補手選出部において様々な方法を試し，結果の比較を行った．

システム全体を通して見るとコンピュータ囲碁システムの問題点は，各局面の優劣を定義し評価関数により表現することが困難であることにあり，これが全てに悪影響を与えていることが分かった．具体的にいうと，評価関数がうまく設定できているかの影響を最も受けるのは，着手決定部である．しかし，それ以外にも候補手選出部もこの影響を強く受けている．理由の一つは，候補手選出部で手の価値を求めるときに使用される評価改善量が評価関数を基に決定されているためである．もう一つの理由は，着手決定部が誤った着手を決定するために，対局が進むにつれて事例獲得時には現れなかった局面が生じてしまい，そのため適切な事例が見つけられず，候補手選出部がうまく機能しなくなるためである．本システムは序盤において候補手 5 手の中に 53.5%の割合で実際に指された手を含む性能を持っているが，それにも関わらず弱いのは，特に後者の理由によるところが大きい．

評価関数の問題を解決するには，着手決定がうまく機能しない局面をひとつひとつ抽出し，正しく着手が行われるように評価関数のパラメータを少しずつ改善していき，必要ならば新しい概念を導入すればよい．しかしこの方法には，新しくパラメータを変更したために，以前の方法ではうまくいっていた局面がおかしくなるというサイドエフェクトの問題があり，非常に時間と根気を必要とする．これを避けるため，何らかの方法により計算機に自動的に各局面でうまく機能するパラメータを選ばせる方法をとることが考えられる．これには，ニューラルネットワークを用いた研究などが行われているので，将来的にはこれらの研究の成果を採り入れていくことも考えられる．しかし，当面はパラメータの設定はシステム作成者の経験に基づいて行う必要があると考えるので，まずはこれを支援するシステムを作成する必要がある．またこれに関して，現在事例の評価改善量を求める部分が評価関数のパラメータに依存しているので，パラメータの変更があると事例を再学習させる必要があるという問題があり，これを改善しなければならない．これについては，事例に事例獲得時の局面を保持させ，パラメータが変更された時に評価改善量部分だけを再計算させる方法が考えられる．他には，評価改善量の代わりに，評価の変化した項目を保持させ，パラメータから独立の値にすることが考えられる．例えば，“勢力範囲が 3 マス広がり，死活が不定からほぼ生きに変化した” などということを記録し，事例として使用されるときに計算により評価改善量に変換するなどが考えられる．

他の今後の課題としては，候補手選出部の性能を向上させることがある．これには，候補手の質の向上と処理速度の向上の二点が考えられる．候補手の質の向上では，まず事例数を増やした場合について実験を行う必要がある．他には，評価改善量算出の方法を変えることが考えられる．また，類似判定を行うパターンの大きさを今より大きくすることも考えられるが，9 路盤を対象とする限り，この大きさで十分であると考えている．しかし，研究対象を 19 路盤にするならば，パターンの大きさは今より大きくとる必要がある．速度については，類似判定の方法が繁雑なため，形が一致しているかのマッチングだけに比べて多くの時間が必要とされる問題がある．これ

については，プログラムの高速化だけでなく，処理の途中で類似性の無いことが判明すると処理を切り上げる，新しく石が置かれたために価値が変わったであろう可能手だけを再計算の対象とするなどが考えられる．しかし，候補手選出部の速度を向上を目指すあまり，候補手選出の性能を落すことだけは避けなければならない．候補手選出の性能が落ちれば，候補手数を減らすことができず，候補手選出部に続く着手決定部で探索の爆発が生じてしまうからである．

謝辞

本研究の機会を与えて下さり，貴重な御指導を頂きました山崎勝弘 教授に深く感謝の意を表します．また，本システムのグループ化および死活判定の部分を担当して頂きました加藤恒 氏に心より感謝致します．

参考文献

[1] 斉藤康己：コンピュータ囲碁研究, 人工知能学会誌, Vol.10, No.6, pp.860-870 (1995).

[2] Jay Burmeister and Janet Wiles : Accessing Go and Computer Go Resources on the Internet, Game Programming Workshop, pp.75-84 (1995).

[3] 松原仁：最近のゲームプログラミング研究の動向, 人工知能学会誌, Vol.10, No.6, pp.835-845 (1995).

[4] 清慎一，川嶋俊明：「局所パターン」知識主導型の囲碁プログラムの試み, Game Programming Workshop, pp.97-104(1994).

[5] 近藤繁，清水賢資，吉川竹四朗：詰碁における無条件死にの判定, Game Programming Workshop, pp.28-36 (1995).

[6] 丸山真佐夫，吉田真，丸山啓，並木美太郎，高橋延匡：囲碁対局プログラムと囲碁の知識記述言語 “爛柯” の研究, Game Programming Workshop, pp.66-75(1994).

[7] 丸山真佐夫，丸山啓，吉田真，早川栄一，並木美太郎，高橋延匡：囲碁の知識記述言語 “爛柯” を用いた囲碁対局システム, Game Programming Workshop, pp.18-27(1995).

[8] 斉藤康己，吉川厚： 囲碁に関する認知的研究, Game Programming Workshop, pp.44-55 (1994).

[9] 実近憲昭： 囲碁対局プログラムの中の死活ルーチンの設計, Game Programming Workshop, pp.105-114(1994).

[10] Thomas Wolf : The program GoTools and its computer-generated tume go database, Game Programming Workshop, pp.84-96 (1994).

[11] 加藤恒：コンピュータ 9 路盤囲碁システムにおける死活判定, 立命館大学理工学部情報工学科卒業論文 (1995).

[12] 北村佳己，山崎勝弘：事例を用いたコンピュータ 9 路盤囲碁システムの実現, 情報処理学会第 52 回全国大会講演原稿 (1996)（発表予定）.