修士論文

類似事例を用いた
コンピュータ９路盤囲碁システムの実現

指導教員　：山崎 勝弘　教授

論文提出者氏名　：加藤 恒

学籍番号　：6124960039-2

1998 年 2 月 13 日提出

立命館大学大学院
理工学研究科情報システム学専攻

摘要

コンピュータ囲碁に関する研究は年々盛んになってきており，現在最強クラスのシステムでアマチュア 4 級の棋力を持つプラグラムも登場している．しかし，囲碁は着手にルール的な制約が非常に少なく探索量が膨大となるため棋力を上げるのが難しい分野である．また局面分析も難解で優劣が表現しにくい．そのため，チェスや将棋に代わる人工知能・知識工学の恰好の研究対象とされている．そこで本研究は，コンピュータ囲碁システム構築上の問題点を，実際に 9 路盤を対象としたシステムを実現し，明らかにすることを目的としている．現在，システムはテスト対局を行える段階に来ており，様々な問題点も明らかになってきている．

本システムは，状況判断部，候補手選出部，着手決定部及び事例獲得部からなる．状況判断部では盤面の解析を行うため，攻防単位である連・群にグループ化し，それぞれ死活が判定される．また勢力値，囲っている地，石同士の連結の強さが計算されこれらを候補手選出部に送る．候補手選出部では，実局面とプロ・上級アマの棋譜から抽出した事例との類似度を手の価値に換算する．そして，価値の高い上位 5 手を候補手として選出し，次の着手決定部に送る．そして，選出された候補手を枝にして木探索を行い，最も盤面が良くなる 1 手を着手として決定する．

テスト対局により明らかになった本システムの問題点として，まず，着眼点を絞り込めずに，無差別に可能手について類似度をもとに手の価値を計算してしまい，重要な手を逃すという点が挙げられる．つまり，手の価値を類似性に依存させているために，守るべきところで守らないという問題が起こる．これは石群の周囲状況や死活をもとに緊急度を見極めて急場に着眼させることで解決できると考えられる．また，着手に一貫性がなく不適切な手抜きをしてしまうという問題も明らかになった．そこで，事例自体に手順を記載して流れを切らない着手を行うことを提案している．これにより木探索などに要する時間を省力化できるというメリットもあり，定石ライブラリの組み込みと併せて，事例の見直しの必要性も明らかになった．

目 次

1 はじめに **1**

2 コンピュータ 9 路盤囲碁システム **4**
2.1 状況判断部 6
2.2 候補手選出部 6
2.3 着手決定部 7

3 状況判断 **8**
3.1 石のグループ化 8
3.1.1 連へのグループ化 8
3.1.2 群へのグループ化 10
3.2 死活判定 12
3.2.1 眼形候補 12
3.2.2 死活不明な石群の処理 13
3.3 勢力計算 17
3.3.1 勢力計算 17
3.4 連結度と地計算 19

4 候補手選出および着手決定 **21**
4.1 事例ベースの構築 21
4.2 候補手選出 27
4.2.1 事例検索による候補手選出 27
4.2.2 優先度による候補手選出 29
4.3 着手決定 31

5 実対局によるシステムの評価 **33**
5.1 問題点と強くするための課題 33
5.2 １９路盤囲碁システムの構築へ向けて 36

6 おわりに **38**

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 連の生成 9
4 群の生成 10
5 連絡条件 11
6 眼による死活判定 12
7 眼形候補決定の過程 13
8 １眼判定の様子 14
9 眼形候補による死活判定の全体の流れ 14
10 石連の捕獲 15
11 石の影響力 17
12 盤端からの影響力 17
13 局面 A 18
14 局面 A の勢力計算例 18
15 強連結線と地を囲う境界 19
16 ある局面の勢力図 20
17 地の境界 20
18 事例の形式 21
19 序盤において出現瀕度の高かった形 23
20 中盤において出現瀕度の高かった形 24
21 終盤において出現瀕度の高かった形 25
22 事例の具体例とその獲得時の局面図 26
23 手の価値を求める手順 27
24 手の価値算出の具体例 28
25 優先度による手の生成 29
26 ツギの手 30
27 連絡の手 30
28 大場への手 30
29 ヒラキの手 30
30 着手決定の様子 32
31 重要な着手を逃す例 33
32 着眼点の絞り込み 34
33 受けと手抜き 35
34 着手の流れ 36

35 新たな概念の石のグループ 37
36 局地的な攻防 37

表 目 次

1 獲得された事例数 25
2 死活判定による得点 31
3 “地とアゲハマ” の計算に用いられる重み 32

1　はじめに

現在，ゲームプログラミングの研究は，年々盛んになってきたおり，その背景としては，これらが人工知能・知識工学の格好の研究材料であることが挙げられる．つまり，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である．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おり，この先，研究が進むに連れ，さらに新たな手法が提案されるものと思われる [1]．

最近，日本国内においても，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．棋力的にも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 [2]．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 [3]．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] 一方で，システム全体について評価を行った研究も行われている [6][7]．そこで，本研究では，最初から最後までを通して打てるシステムを全体を構築することにより，コンピュータ囲碁システム構築上のの問題点を明らかにすることを目的する．まず，9 路盤を最初の研究対象とするが，最終的には 19 路盤の囲碁システム構築を目指す．

現在，コンピュータチェスにおいては，人間の世界チャンピオンにも勝利する事も可能なシステムが登場しているが，コンピュータ囲碁ソフトでは最強クラスでアマチュア 4 ~ 5 級程度である．現在世界最強のプログラムと言われる HandTalk（陳志行 , 中国）は FOST 杯世界選手権において数度に渡って優勝するなど，常に上位にランクされているプログラムで，日本棋院からアマチュア 4 級を認定されている．特徴は，自陣を厚くしてから，積極的に戦う場所をみつける打ち方で，布石段階での大局観に優れている．したがって，序盤で他のプログラムに大きな差をつけ，そのまま逃げ切るスタイルである [8]．それに対し，GO 4 + +（Michael Reiss, イングランド, 同 2 位）は，強いプログラム（HandTalk）には，布石で差をつけられるものの，弱いプログラムには安定した成績を残す，かなり優秀なシステムである．これは，基本戦略が群を少なくすることらしく，石の連結を重視しすぎるために，布石段階で遅れをとる傾向にあることに起因している．しかし，石は強くなるた

め，死ぬことはほとんどない [9]．また，日本においても，シルバー囲碁 (山本成辰) [10] や琵琶子 (田中正浩) [11] というプログラムが，前述の FOST 杯の第 3 回大会において，それぞれ第 5 位 ・第 7 位に入賞し，健闘を見せている．

コンピュータ囲碁がコンピュータチェスやコンピュータ将棋 (アマチュア 2 段) に比べて，実現が困難であるのは，以下の 2 つの理由による．

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．

，一般に囲碁は難しいと良く言われるが，その実は，非常に簡単なルールで成り立っているゲームとである．そのため ,, 囲碁には手に対する制約がほとんどなく，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，$(9 \times 9)! \approx 5.79 \times 10^{120}$ 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，一つの目安となる．更に 19 路盤では，$(19 \times 19)! \approx 1.34 \times 10^{769}$となり，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．

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．事例ベース推論とは，過去に培われた経験を，実局面に照らし合わせて有効に活用しようとする手法である．事例ベースを用いた理由は [12][13]，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．これは，碁の上手な人はサーチ (探索) をせず，ほぼ一本道の先読み (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 [12]．

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 200 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．更に，石の繋がり具合を連結度という評価値で分析する．石が強く繋がっているか否かは，囲碁の戦略的な意味合いにおいては非常に重要である．次に，これら石群について，取られる (死ぬ) か取られない (活きる) 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．そして，盤面の勢力の状況と連結度により，”どの領域を囲っているか” を明らかにするために境界線を作り，地の計算を行う．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．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．また，実局面に有効な事例がない場合，その局面に応じた優先度による候補手を生成する．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．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について述べ，第 3章においてどのように局面を分析し，石のグループ化，死活判定，勢力計算，連結度，地計算を行っているかを述べる．また，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る．更に第 5章において本システムの実対局による評価を述べ，そして最後に第 6章において本論文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

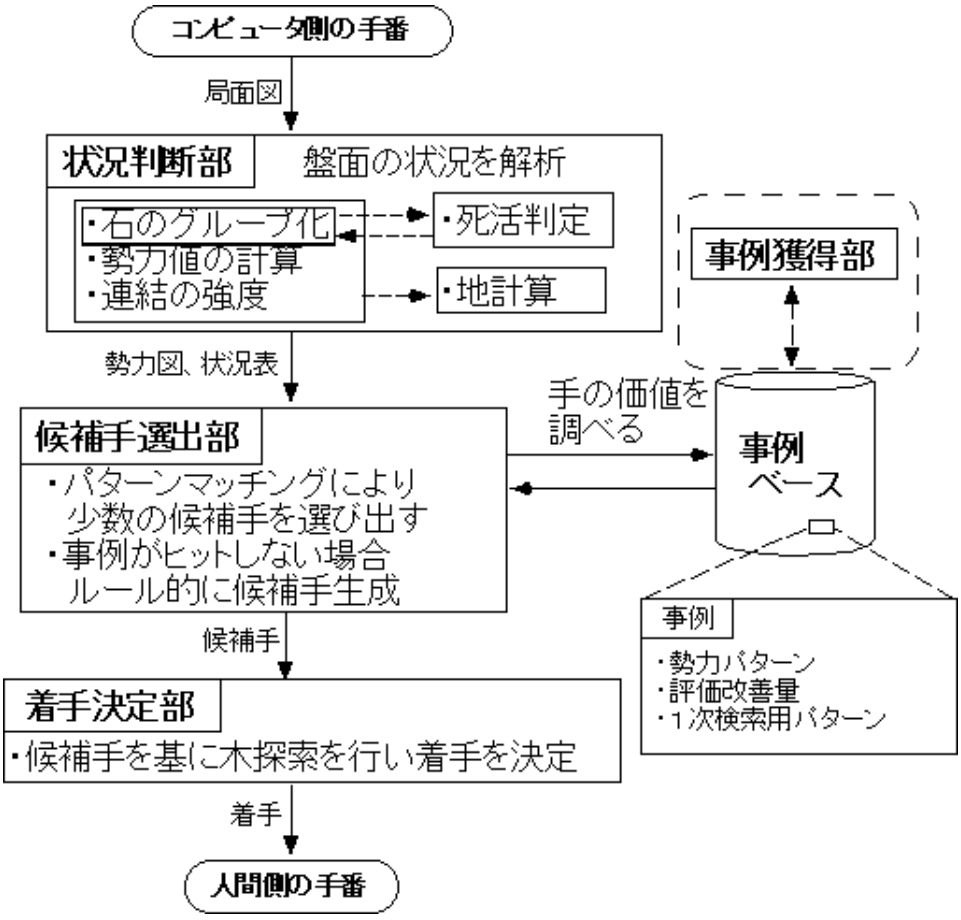

図 1: 囲碁システムの全体構成

本システムは，図 1に示すように，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．まず状況判断部で石のグループ化，勢力値の計算，石の連結強度などの局面の分析を行う．また，これらの情報を用いて死活判定や地計算を行う．次に，プロ／アマ上段者の棋譜から頻出するパタンを事例として用い，その局面で打つべき候補手を数手選び出す．類似事例がヒットしない場合，優先度による手の生成を行う．候補手選出部は，盤面の全ての可能手に対し価値を求め，この価値を基に可能手をソートし，上位数手を着手決定部に送る．最後に，着手決定部において木探索と評価関数により，候補手から最善と思われる手を選択する．

このような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．

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．対局時には “コンピュータ側の思考” の駆動し，更に，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．また，非対局時の事例獲得作業を制御する．

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．

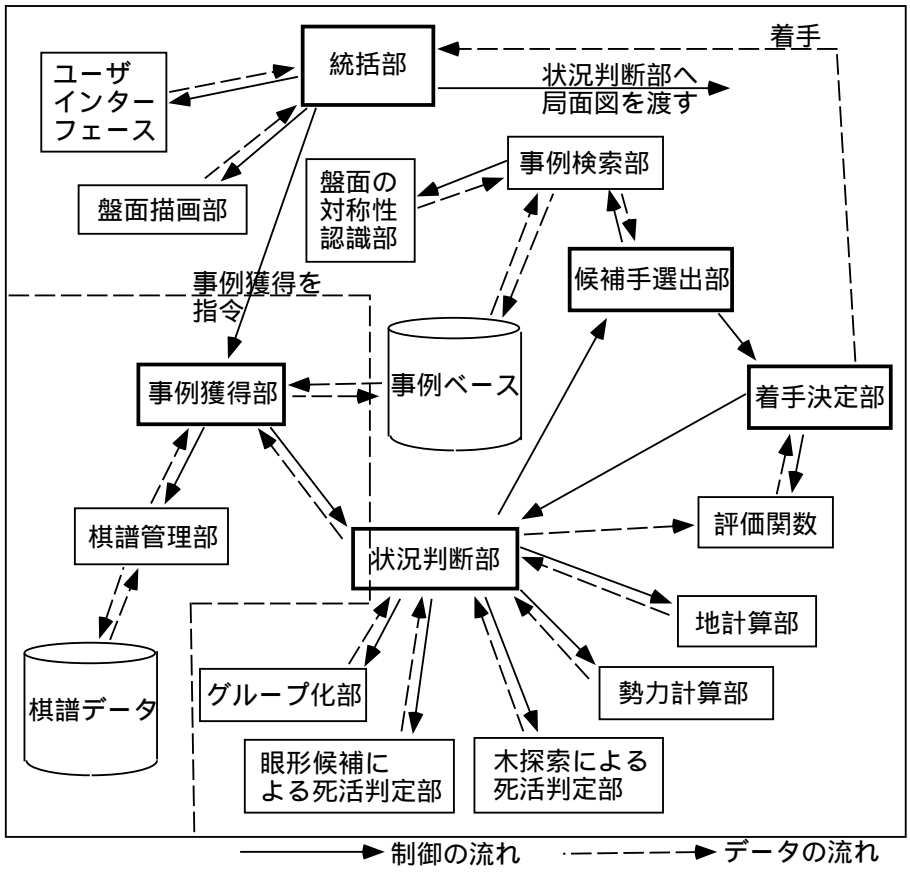

図 2: 囲碁システムの構成要素

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．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．

2.1 状況判断部

．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．また，グループ内の石の繋がりを表す連結強度もここで計算される．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 (活きる) 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．

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，勢力図と局面の各評価値を付した状況表を次の候補手決定部に送り，制御をそちらに移す．

2.2 候補手選出部

候補手選出部は，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，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．

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 200 局を使用することで，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．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．

2.3 着手決定部

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．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，候補手選出部を通り，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，評価関数を計算する部分である．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．そして，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．

3　状況判断

3.1　石のグループ化

正確な状況判断を行うには，黒 / 白石と空点や盤端で構成される格子点の集まりの意味付けを詳細にしたうえで，それぞれの集まりに 「厚い」 弱い」など意味を持たせなければならない．つまり，互いの石の集まりが関連し合うので，隣接の具合や，石の集合間にある空点の領域も，石の集合と同様に局面分析の対象となる．そのために連と群にグループ化にする．

” 連” (あるいは “石連”) とは，“縦横に隣接した同色からなる石，または空点のグループ” を意味し，群を構成する．“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．

3.1.1　連へのグループ化

図 3に，連の例を示す．また，連は切断される可能性のない石の集合であり，それ故に，実際の囲碁においても盤上での攻防の最小の単位であり，同じ連と見なされる石は生死，つまり死活を共にする共同体である．

また，群や周囲との関連付けの為に，図 3の様な項目について調べる一つの連は ,” 連番号” ,” 石の色” ,” 石数” ,” 石の位置” ,” 縦横の接触状況” ,” 接触している連” ,” 同じ群と見なされる連” ,” 所属群” ,,” 呼吸点の位置” の情報を持つ．“連番号” 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 .” 石の色” には，黒 ･白 ･空点のいずれかが割り当てられる .” 石数” は，その意味通り連に所属する石の数であるが，連が捕獲される際には，その損失の大きさを示すパラメータとなる .” 石の位置” には連のメンバー座標である．そして ,” 縦横の接触状況” ,” 接触している連” ,” 同じ群と見なされる連” ,” 所属群” を用いて，この連と周りと関係を記述する .” 縦横の接触状況” は，接触している黒石もしくは白石，空点，盤端の数であり，これと” 接触している連” を組み合わせると，連がどのような封鎖状況にあるか推測できる．また ,” 同じ群と見なされる連” ,” 所属群” の情報をも総合すると，捕獲から免れるかどうか，つまり，「この連は単独で危険そうである」，周りに仲間の連がいて比較的安全である」，というような推測もできる．呼吸点は，連に対して縦横に隣接する空白マスの数であり，言い換えれば ,” 縦横の接触” の空点数である．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．よって ,” 呼吸点の位置” も，その連が捕獲されるかどうか，読む際に必要な情報である．

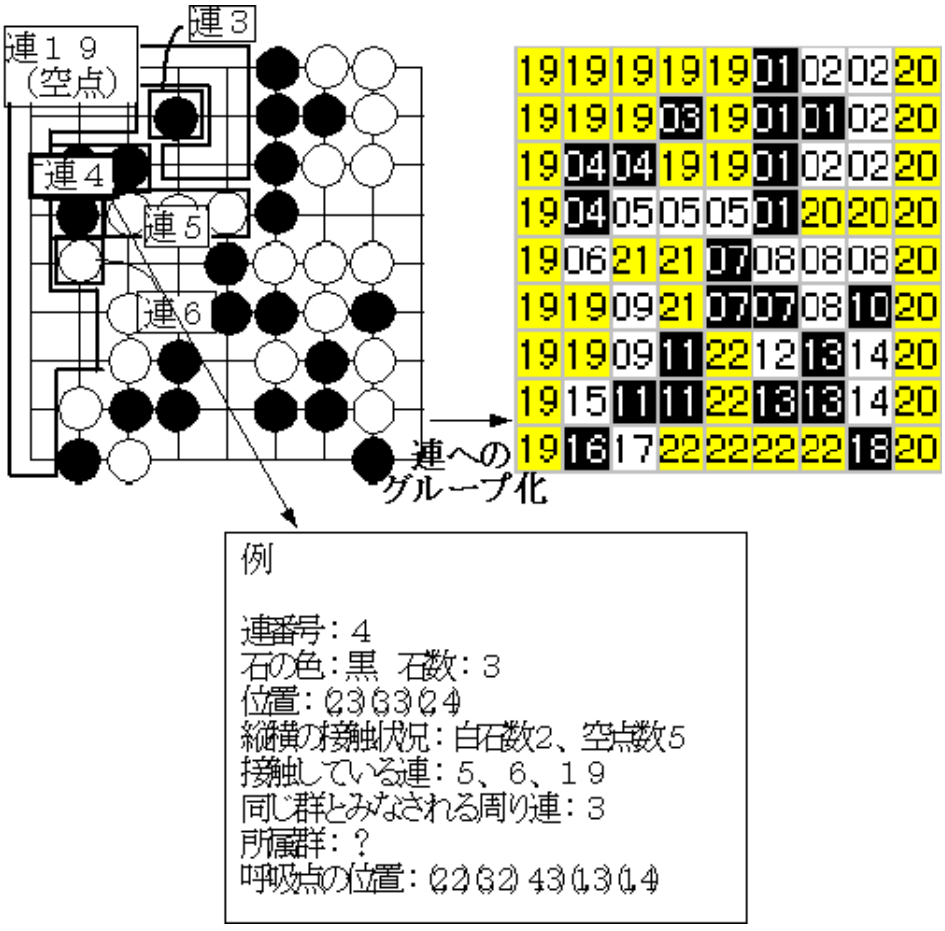
連１９
（空点）
連３
連４
連５
連６
連へのグループ化
例
連番号：４
石の色：黒　石数：３
位置：(2,3)(3,3)(2,4)
縦横の接触状況：白石数2、空点数5
接触している連：５、６、１９
同じ群とみなされる周り連：３
所属群：？
呼吸点の位置：(2,2)(3,2)(4,3)(1,3)(1,4)

図 3: 連の生成

3.1.2　群へのグループ化

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．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4に，群の例を示す．

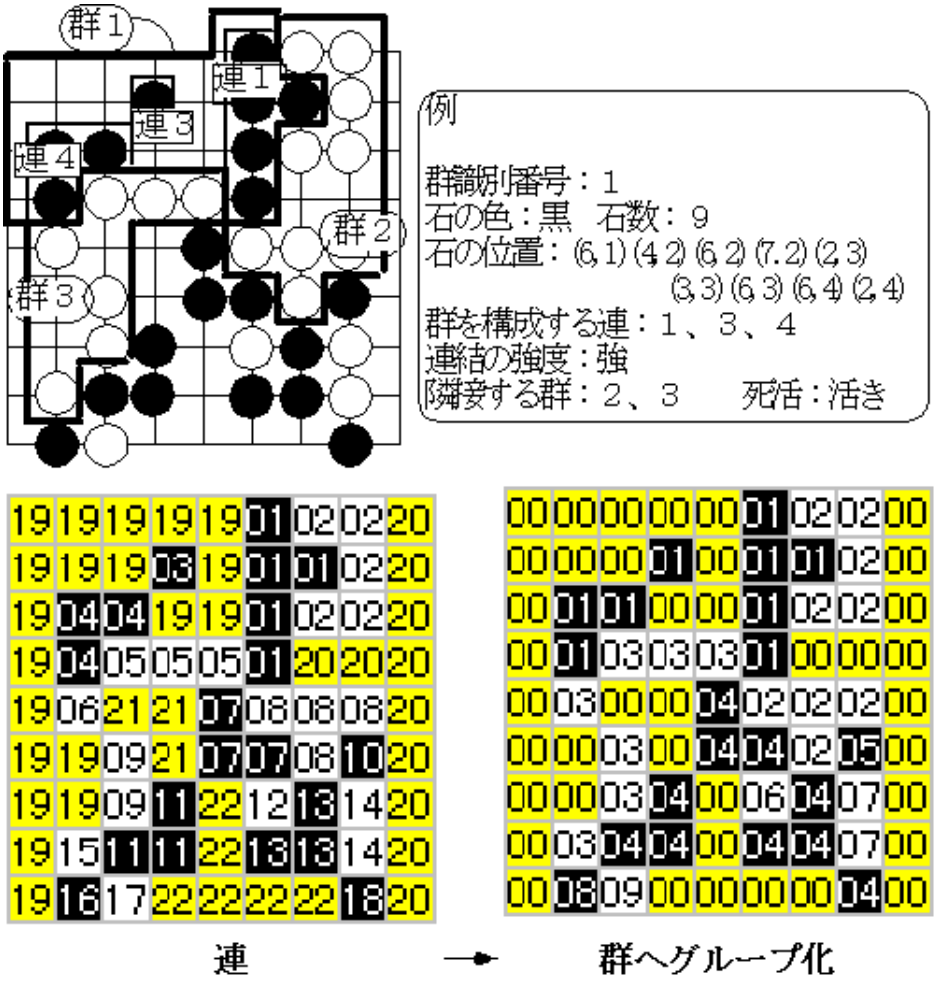

図 4: 群の生成

一つの群は，図4の項目のような情報を持つ．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．”隣接する群” は死活の攻防の相手を知るために必要である．”連結の強度’ は構成する連同士の繋がりの強さであり，言い換えれば，この群の強さや安定度の大きな目安となる．正確な判定方法は，お互いに石を置いていった場合に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 [14]．

囲碁においてよく現れる石のつながり方には，“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．図 5に連絡条件を示す．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．この CP に敵石が幾つ占有しているかで連絡の可否が決定される．

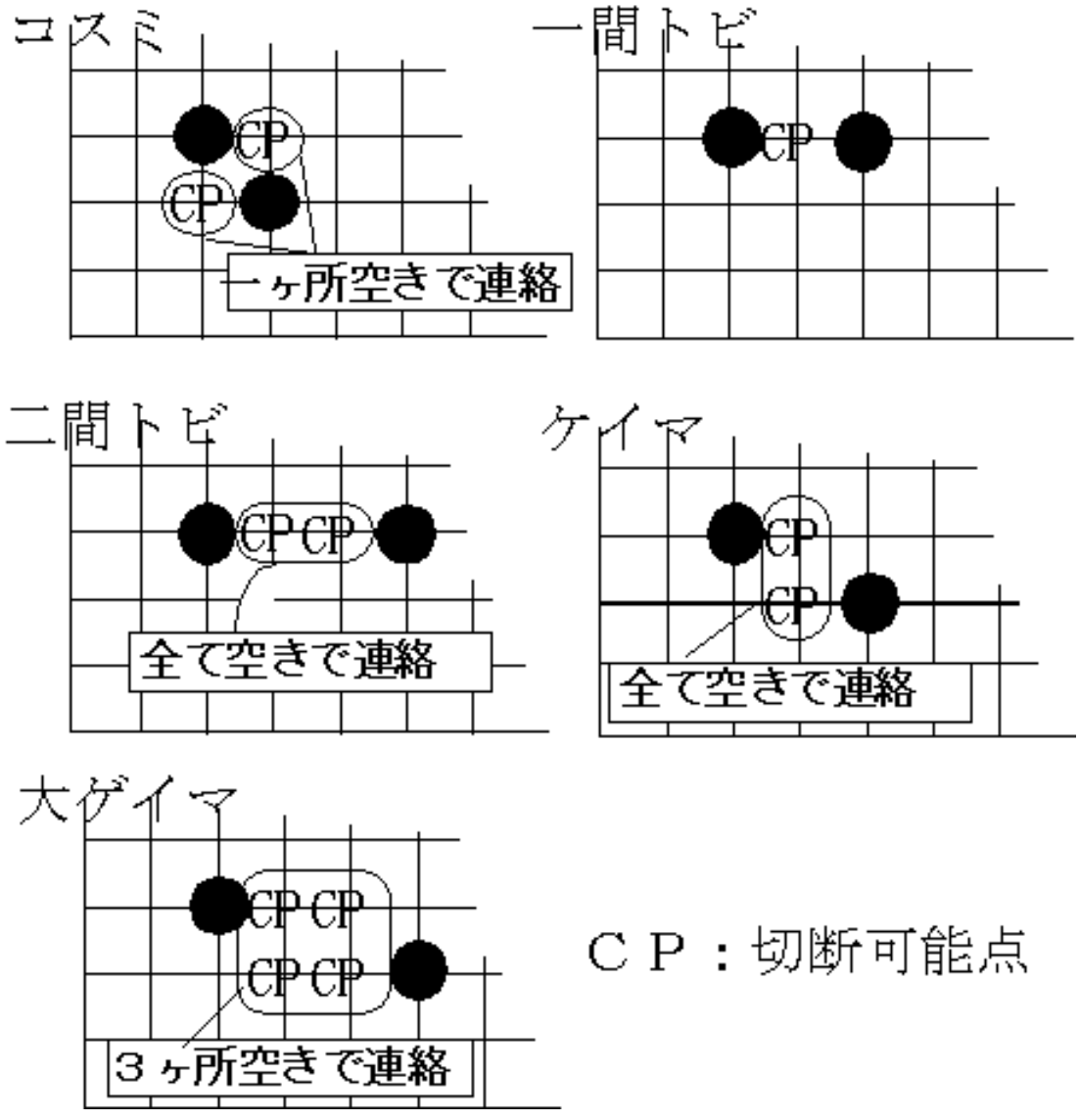

図 5: 連絡条件

3.2 死活判定

3.2.1 眼形候補

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 6にこの様子を示す．

図 6: 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 6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 6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 [14]．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 [15]．

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 7を用いて説明する．

まず，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 7 (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 7 (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

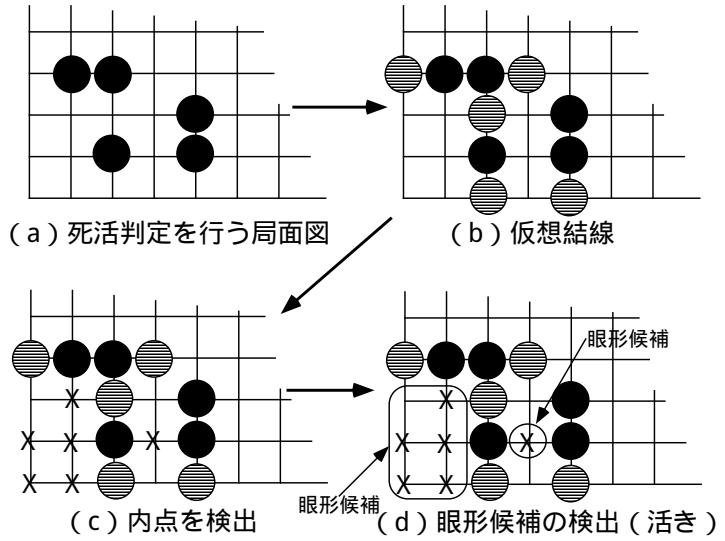

図 7: 眼形候補決定の過程

(図 7 (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．

- 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．

複数の眼形領域を持つ石群の場合，領域の境界による死活の判定で境界が完全であるものは，確実に 1 眼は見込めるわけであるから，残りの幾つか不完全な境界からなる領域の中で，最低 1 つの領域が 1 眼以上持つことが証明できれば，その石群が”生き”であると判定することができる．

一眼判定では，不完全な境界の眼形領域の 1 点ずつごとに，眼であるかどうか調べ，もし 1 つでも眼があれば TRUE と判断されるというような仕組みになっている．

基本的には，調べる点を中心にして，図 8のように中心点から最も近い距離にある 8 つの点 (境界を形成する点) の内，4 つ以上の点を味方石が占拠し，石が占拠していない残りの点が敵石から到達出来ない点であれば，その中心点が眼であるとした．

眼形候補による死活判定の全体の流れを図 9に示す．

3.2.2 死活不明な石群の処理

ここでは，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述べる．

図 8: １眼判定の様子

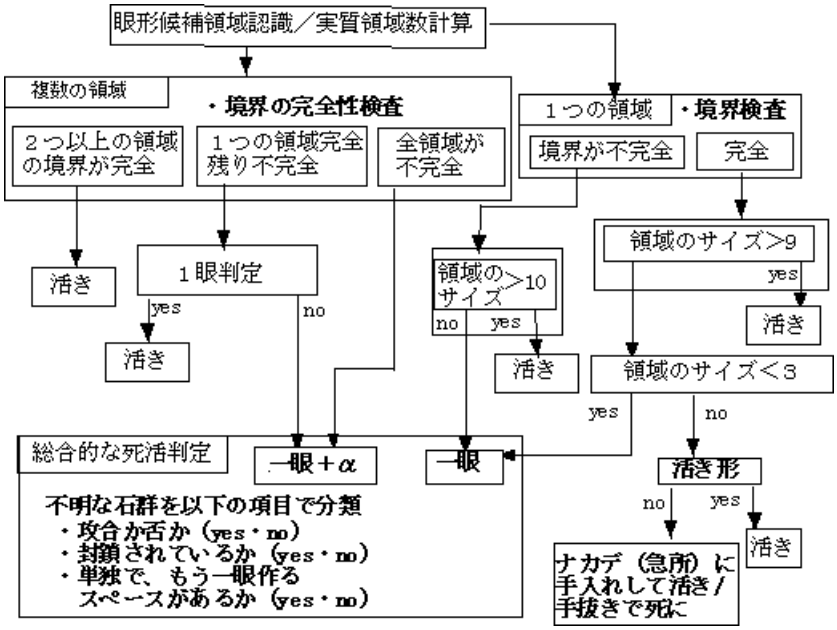

図 9: 眼形候補による死活判定の全体の流れ

図 9に示す死活の一連の解析の流れで不明と判断される石群は，大半が境界が未だ未確定なものであり，将来的には領域のサイズや眼形候補そのものが劇的に増減することが見込まれる石群である．また，これらの不明な石群の中には先手 / 後手の違いで，死活の結果が変わってくる中立的な性質を持つものや，例えば，単眼形でその領域数が 2 以下となってしまい，単独で内部に 2 眼作るのが困難となった限りなく ” 死 ” に近いものも含まれている．このように，一口に ” 死活不明な石群 ” といっても様々で，その危険度は ” 不明 ” ひとまとめにされるものではない．よって，周囲との関係により，その不明な石群が活きるための条件を列挙して検査し，どのような状態で，どの程度の危険であるかを項目別に記述する．以下に，不明な石群に記述する検査項目を述べる．

攻合か否か (yes ・no) … 周りの相手石群が全て ” 活き ” ならば，攻合いとならないが ,” 死に ” または ” 不明 ” な相手石群があれば攻め取りで活きる可能性がある．

封鎖されているか (yes ・no) … 完全に封鎖されている状況なら，単独で活きるしかないが，封鎖されていなければ，包囲から脱出したり，最寄りの味方石群に連絡し，活きることが出来る．

単独で，もう一眼作るスペースがあるか (yes ・no) … 地を拡大するか，または，もう一眼作るためのヒラキを打つのに，充分なスペースがあるか否か．

これらの項目は，重複して活きる可能性 (手段) が否定されると，当然 ” 死 ” に近い ” 不明 ” な石群として表す事が出来る．また，不明ない石群を ” 一眼 ”,” 一眼 + α ” と分けることにより，死の危険度を更に差をつけている．例えば ,” 一眼 ” の領域数 2 以下の閉領域を持つ石群は，検査項目が全て否定されると，もはや完全に ” 死に ” である．逆に ” 一眼 + α ” の場合，検査項目が全て否定されても攻撃側が手抜きをすれば ” 活き ” となれる．

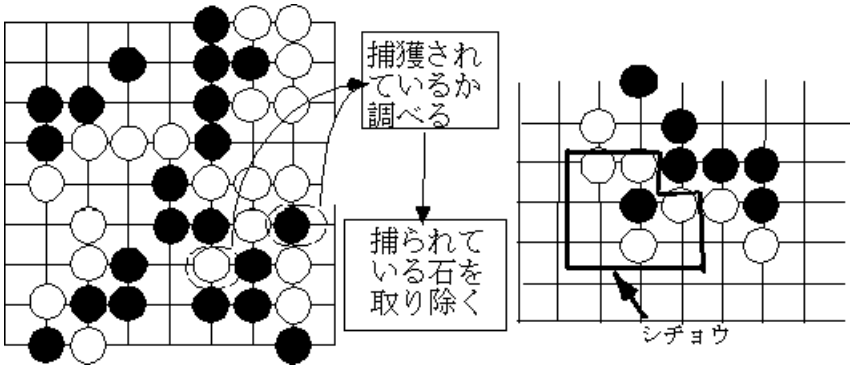

図 10: 石連の捕獲

本システムでは図 9とは別に，連の死活を判定するようにしている．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”

しかし，この方法だけでは，時間もかかり正確性にも欠ける．そこで，囲碁において ” カカエえられている ” といわれるシチョウなどの形を探索の枝狩りに用いている．また，木探索の対象は呼吸点が 2 以下の石連と封鎖され隔離された石連の捕獲についてのものに限定している．

3.3 勢力計算

3.3.1 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．

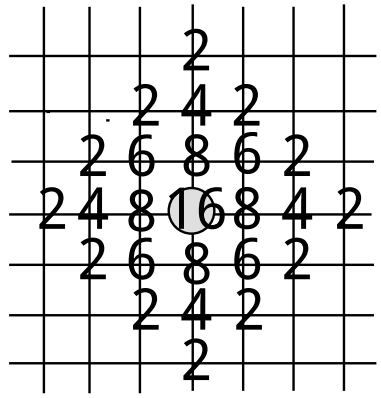

図 11: 石の影響力

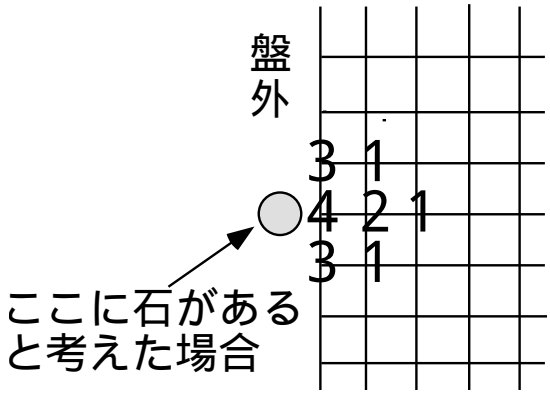

図 12: 盤端からの影響力

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．これは，コンピュータ将棋などでの駒の価値同様，システム製作者によって異なっているのと同じである．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．
- 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．

石の影響力は，黒側に正，白側に負の値を与えている．そして，距離が離れるにつれて減少する．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端を利用する．つまり，盤端は地を囲う味方石と同じ効果を持つ．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 12に示す．盤の端の影響力は，その勢力値が 3 以上のときに計算される．また，盤の端からの影響力は，石の影響力より低くしている．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” の値となるようにしている．以上の影響力を使って，図 13の局面を勢力値計算した例を図 14に示す．

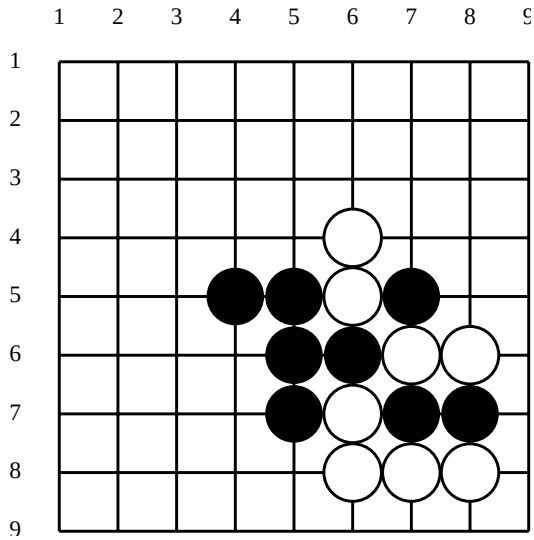

図 13: 局面 A

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	16	16	黒	黒	白	白	-6
0	2	6	16	黒	白	黒	黒	-12
0	0	0	2	-3	白	白	白	-13
0	0	0	0	-7	-16	-16	-16	-12

図 14: 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．

3.4　連結度と地計算

”地”は，囲碁においては死活とともに，形勢を判断するのに最も大切なファクターであり，プレイヤーが目に見えて容易に形勢を判断できると言う点では，むしろ，こちらの方を重要視する実利型プロもいる．”地”とは石，または盤端によって作られた境界結線で囲まれた領域であり，”目 (もく)”と言う単位で数えられる．本質的には，囲碁はこの”地”を何目獲得出来るかを争うゲームであり，コンピュータに対して最も定義しやすい評価値は恐らくこれであり，全ての他の評価値は”地を何目獲得できるか”を基準にするために重みをかけられて換算される．

地として囲まれている領域は，ある程度手が進めば結線を形成する石も多くなり．閉領域となりやすいので正確に計りやすいが，実際の対局者は，まだ手が進まないうちから境界を見込み”大体，どの程度，地が見込めるか”を目算する．この時の”境界”を定義は，曖昧で感覚的なものだが，ある程度近似したものをコンピュータに対し与えることはできる．地を囲う境界は，本質的にはその性質上，先に述べた石群の”連結度”の強いもの選出すれば良いのだが，闇雲に選べば図 15の点線内のように境界を間違える．そこで，本システムの地計算部は連結度と勢力値を上手く使い境界を見込めるようにしている．

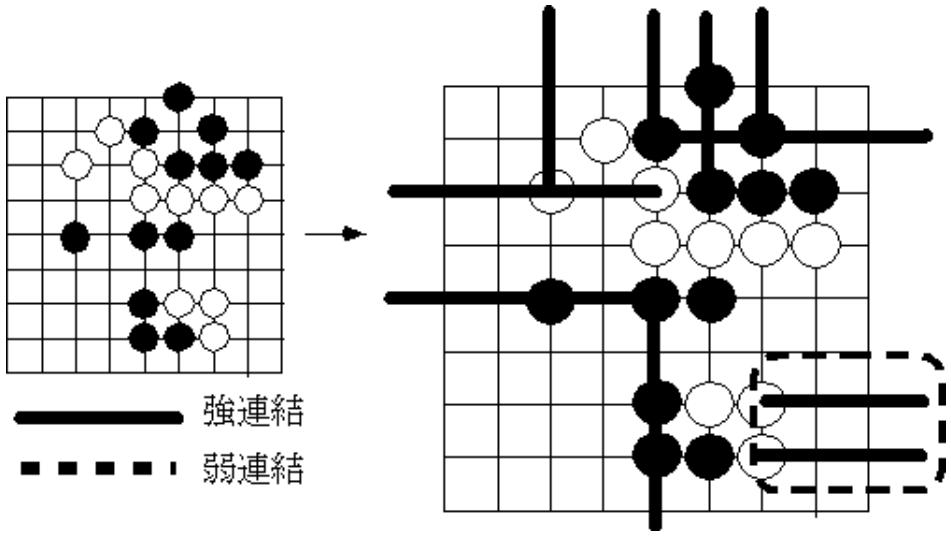

図 15: 強連結線と地を囲う境界

ある局面の勢力値を計算すると図 16にのようになる．ここで，勢力値による境界の見極めを行う際に，まず前提としてあるのは，“勢力値が絶対値１６ならば，そのマスは，その符号の石と同等の働きをする“という原則である．つまり，石自身が置いてあるマスへの影響力は１６ (またはー１６)である為，それらと同等の１６の影響力を受けているマスも石が置いてあると仮定して良いということである．ただし，勢力値１６でも，そのマ

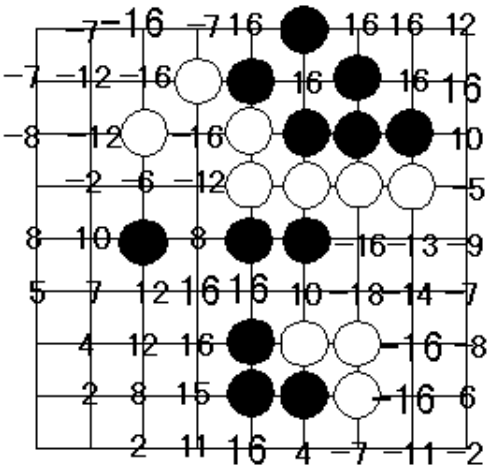

図 16: ある局面の勢力図

ス付近に集中しているだけだったり，形によって，たまたま１６以上の勢力値になったとも考えられる．そこで，強い連結上で勢力値１６なら，石が置いてあるのと同等というようにし，これを最低限の領域を囲う境界ととしている．また，石が置いてあると仮定する

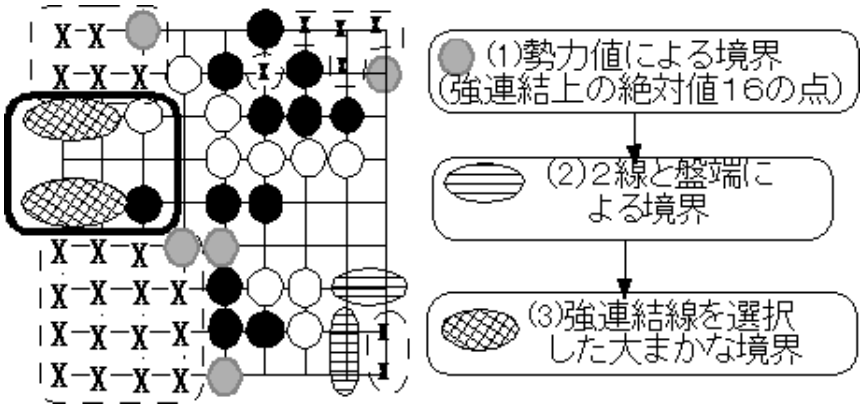

図 17: 地の境界

と，図 17(1) のような境界線も作れる．つまり盤端と (石が置いてあると仮定できる) 勢力値１６ (またはー１６) のマスとが，一間トビの関係なら，領域を閉じる (囲う) のに十分な境界である．この段階までで，領域を完全に閉じている境界を見極められる．後は，領域を完全に閉じていない図 17の太線内の様な例であるが，やはり，盤の端との強連結線を境界とし採用し．地計算する．つまり，要領としては，図 17(1)(2) のように境界を細かく決めることが出来る所を先にやり，曖昧な境界は，図 17(3) のように盤端とで大体のところで境界を作って地を見込み計算をする，という手順となる．

4　候補手選出および着手決定

4.1　事例ベースの構築

本システムで用いる事例の形式を図 18に示す．

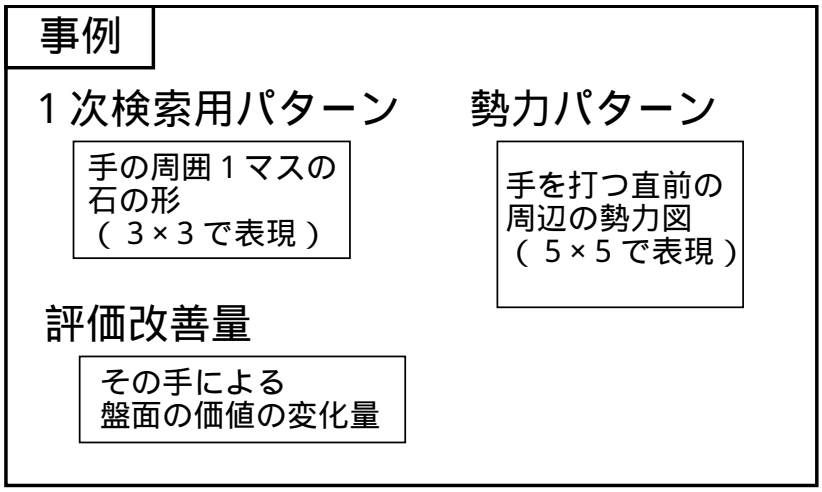

図 18: 事例の形式

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” から構成される．

“勢力パターン” は，勢力値を用い，石の形だけでなく石の力関係も表現することを目的とし，事例に示される手が打たれる前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．このパターンの中心が事例の示す ” 手 ” であり，このように，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” は，事例に示された ” 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，値は大きければ大きいほど，より大きく盤面の価値が良化することを表す．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．一時検索用のパターンであり完全一致を求められので．勢力値を用いて類似性云々を考える必要はない．このパターンは事例の分類に使用され，検索の省力化に役立つ．

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り，局所的に類似しているからと言う理由のみで，その事例が獲得された局面や状況とのギャップを無視して事例をむやみに適

用しないためのそちである．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，候補手に悪手を含ませないようにしている．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．

また，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．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．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 19，図 20，図 21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．

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．

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．事例獲得は，棋譜ファイルの全ての手を対象として行うこととしている．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．

事例は人間との対局から，コンピュータに不足した手を捕捉する目的で獲得できるが，その価値に疑問があり今回のシステムでは実装を見送っている．

先に述べたとおり，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．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，

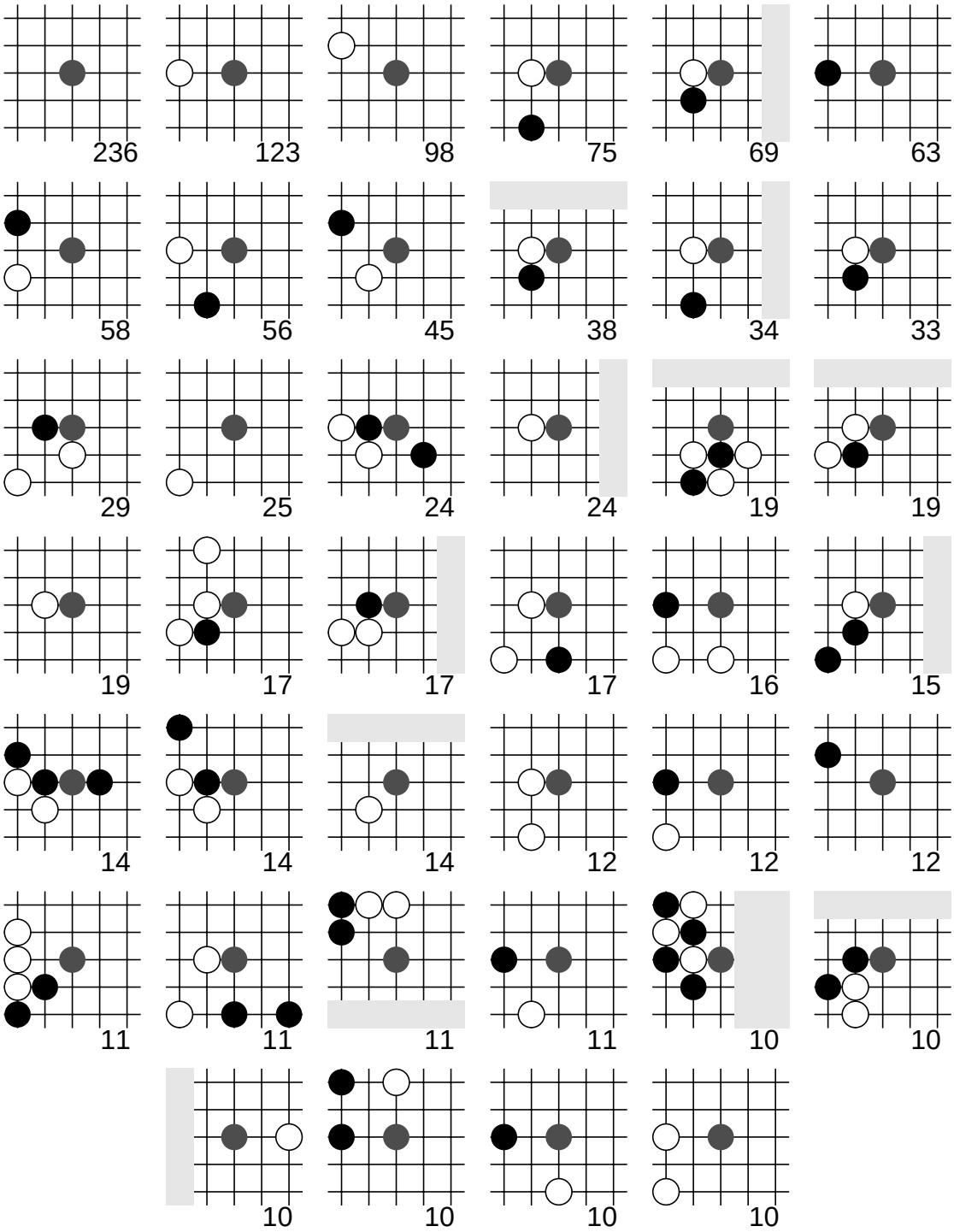

図 19: 序盤において出現瀕度の高かった形

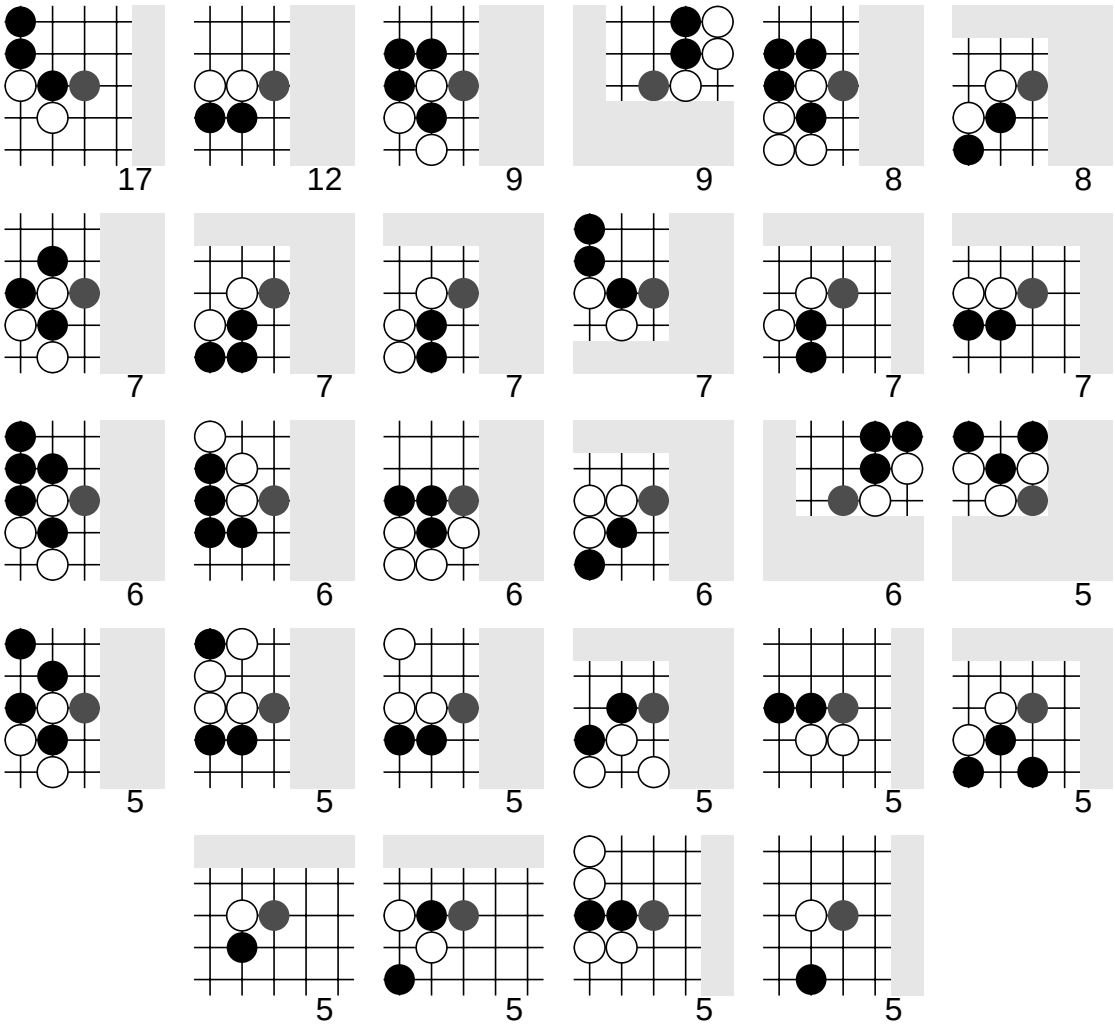

図 20: 中盤において出現瀕度の高かった形

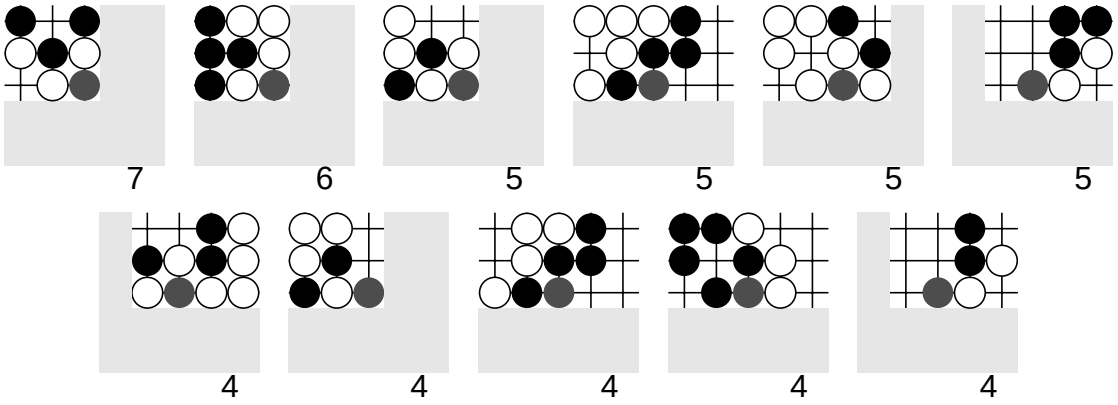

図 21: 終盤において出現瀕度の高かった形

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，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，プロ棋士の羽根泰正九段から提供して頂いた．獲得された事例数を表 1に示す．中盤が一番多く，

表 1: 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

そして序盤，終盤の順になっている．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，及び時間切れとなる場合があったことなどによる．

次に，事例の具体例とそれが獲得されたときの局面を図 22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．“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，”評価の変化項目”により計算される．図では，黒側が勢力範囲を 5 マス獲得し，死活に変化がないことを示している．逆に，白側は勢力範囲が 1 マス減少する．そして，図のように評価改善量が計算される．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．

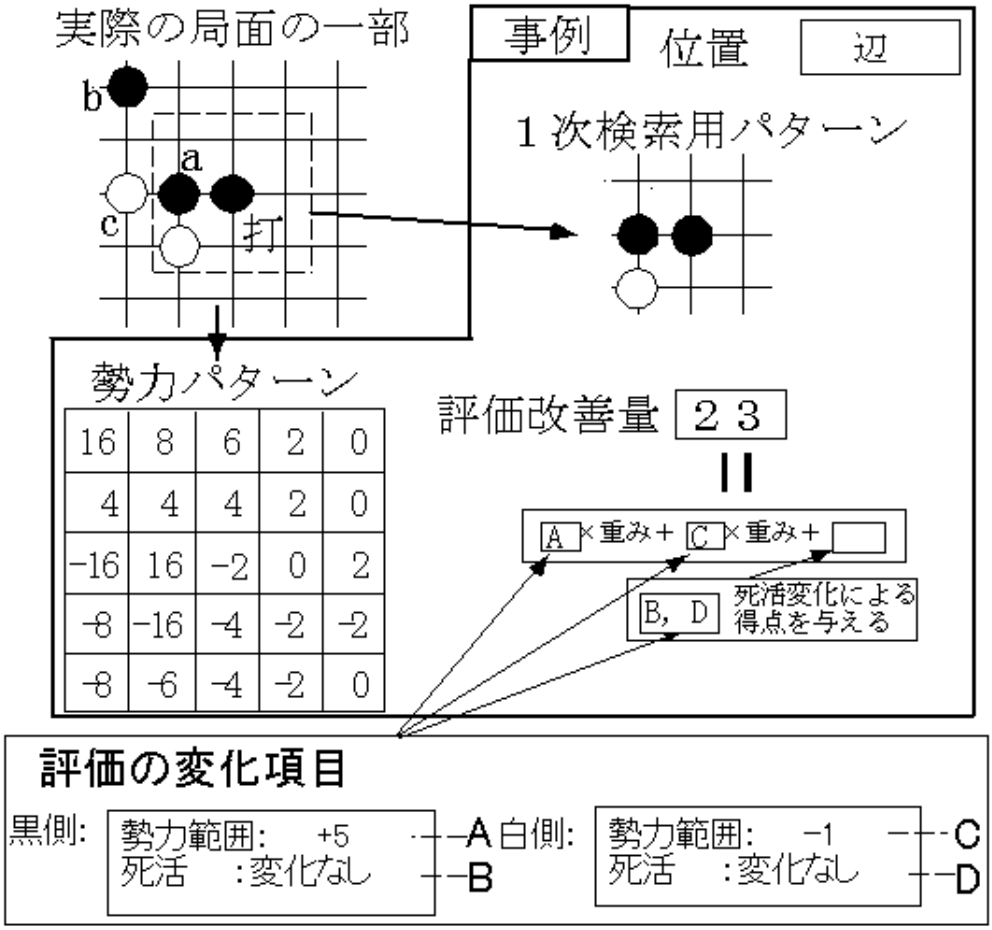

図 22: 事例の具体例とその獲得時の局面図

4.2 候補手選出

4.2.1 事例検索による候補手選出

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 (一次検索による検索対象の限定)

手の価値を求める手順と例を，図 23，図 24に示す． 最初に状況判断部で計算された勢

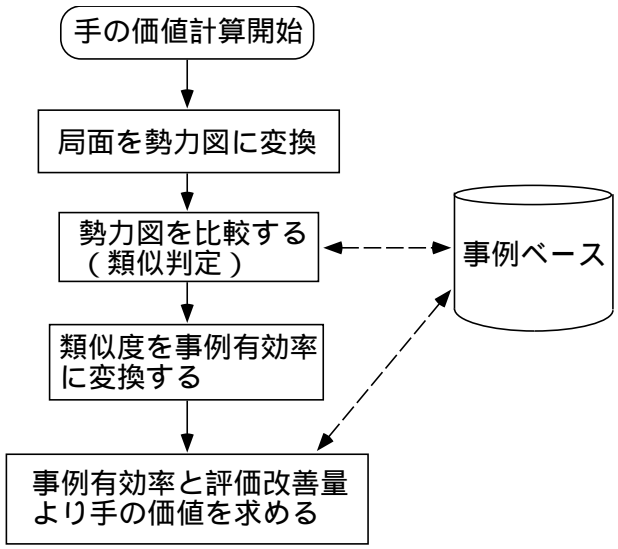

図 23: 手の価値を求める手順

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．図 24では，この 2 つの勢力図の各マスごとの差の絶対値が 64 となり，これが類似度となる．次に，類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．図 24の局面例の場合，事例の勢力パターンの合計値が 134 ，類似度 64 となり，事例有効率は 4.48 となる．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．

そして，この事例有効率に評価改善量を掛けて手の価値を求める．評価改善量とは，事例獲得時に手を打つ前と後で，盤面の価値がどれだけ変化したかを状況判断部を用いて計算したものである．これは，事例に示された手を打つと最大でどれだけの効果があるかを表し，それぞれの事例が固有に持つ数値である．このような過程を，一次検索でヒットした盤面の可能手全てについて行い，価値の高い上位 5 手を求める．

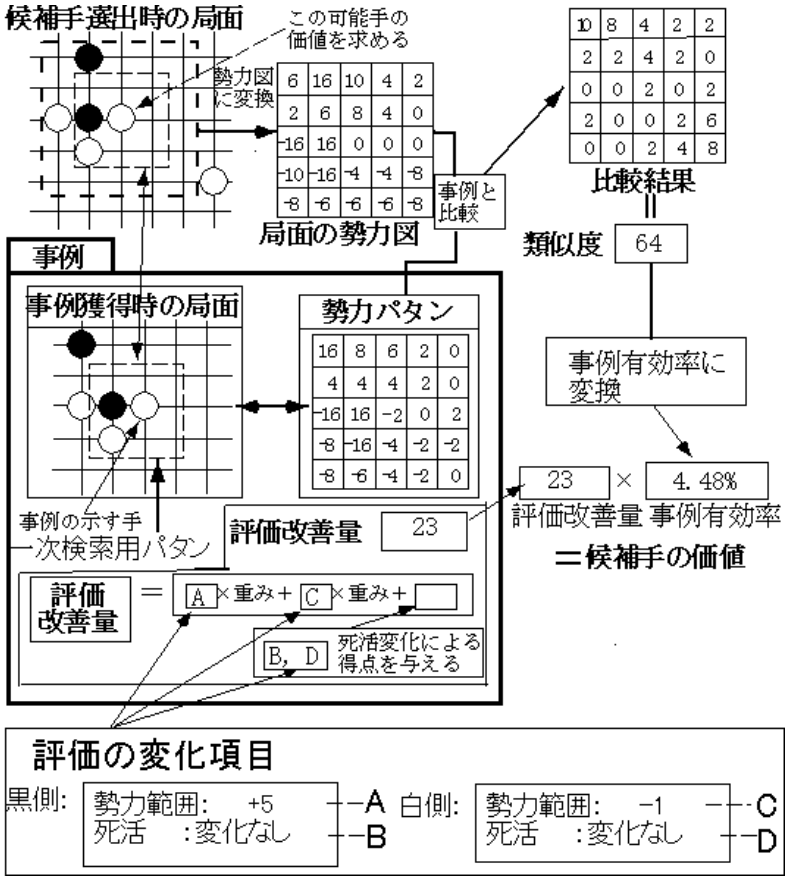

図 24: 手の価値算出の具体例

4.2.2　優先度による候補手選出

事例検索による候補手選出において，有効な事例が見つからない場合も当然考えられる．そこで，本システムでは，このような場合には簡単な戦略に優先度をつけて着眼点を与え，その着眼点に戦略に応じた手を生成するようにしている．図 25は，有効な事例がない場合の候補手生成の優先度を示している．この優先度は，製作者の経験をもとにモデル化したものである．

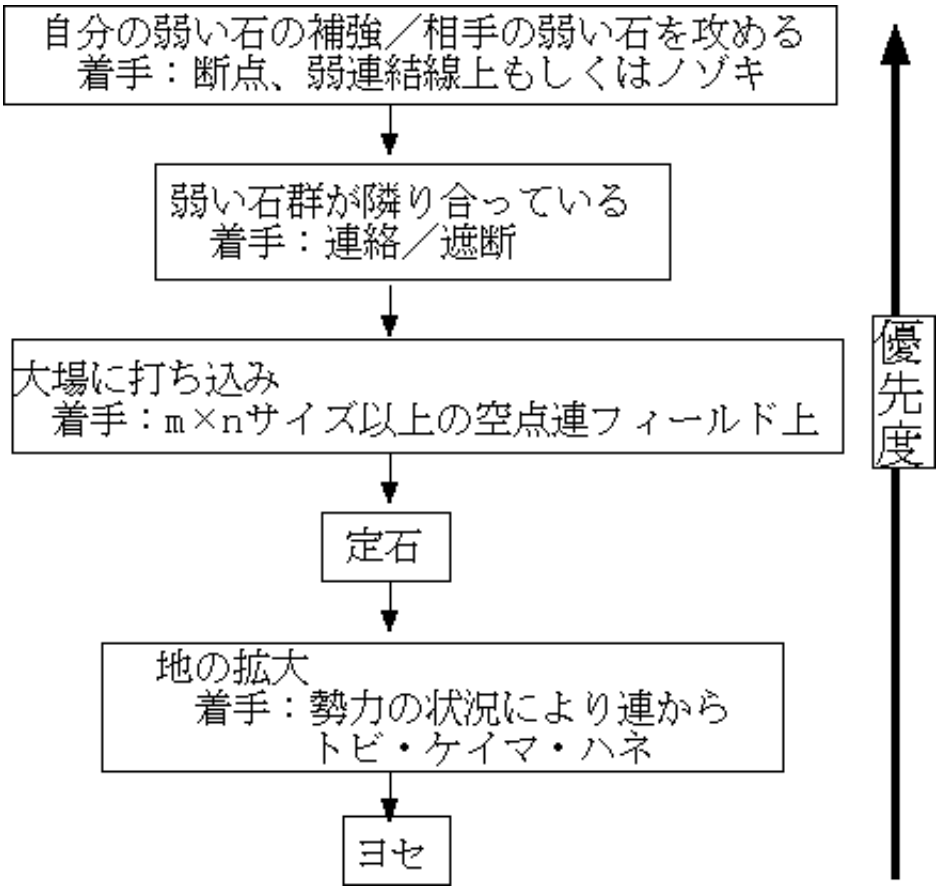

図 25: 優先度による手の生成

まず，自分の弱い石群についての補強に着眼する．石群を補強する手としては，断点や弱連結線上への”ツギ”の一手である．逆に考えると，相手側の弱い石群への攻撃も有効な着手である．これも相手石群の断点や弱連結線への”キリ”,”割り込み”，または”ノゾキ”の手を選出する．

次に複数の石群同士が連絡可能な場所を着眼する．これにより，単独での”活きが”が難しい石群が他の石群と繋がることにより救出されたりする．これも逆から言えば，複数の相手石群の連絡を遮断する一手は有効な着手である．

更に，隅または辺に，ある程度のスペースがあれば，”打ち込み”の手を打つ．特に９路盤では，ただ境界を完成するように打てば，あっと言う間に終局を迎えてしまうので，相手の隅などへの打ち込みは不可欠である．そして，そのスペースがなければ定石的に手を進めるのが自然である．ここで言う”定石”のようなものは，”手筋”と意味合いが強いものも含む．

そして，地の拡大を狙う手として，相手の勢力の範囲を狭めるようにして，”トビ”，”ケイマ”，”ハネ”など，”ヒラキ”的な手を打つ．これは，大まかに言えば石群の補強と同じ意味合いがあるが，実際の囲碁では最もポピュラーな部類に入る着手である．最後に”ヨセ”であるが，本質的には地の拡大を目的としているが，終局間際で”先手”，”後手”の概念があり，戦略的には少々複雑である．これは先手・後手は組み合わせゲーム理論を用いれば解析可能であるが，これは，今までのように正味一手ですむものでなく，一連の流れで，手の交換によって成り立つものである．

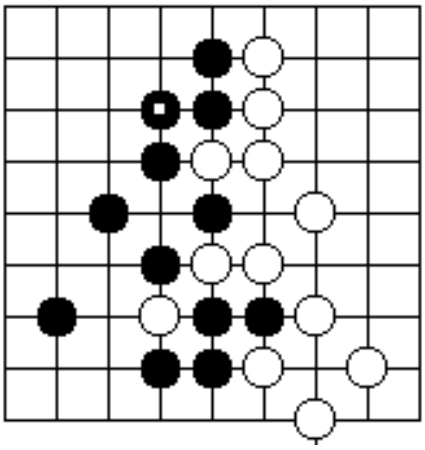

図 26: ツギの手

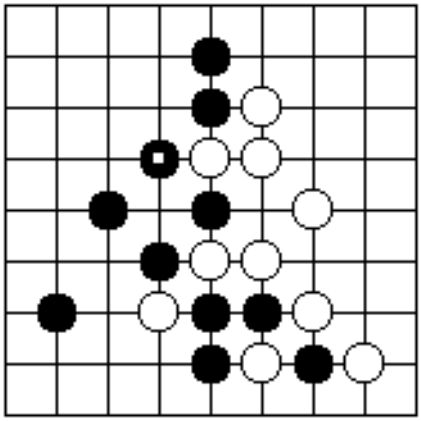

図 27: 連絡の手

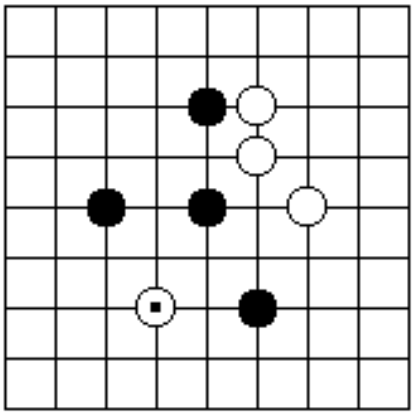

図 28: 大場への手

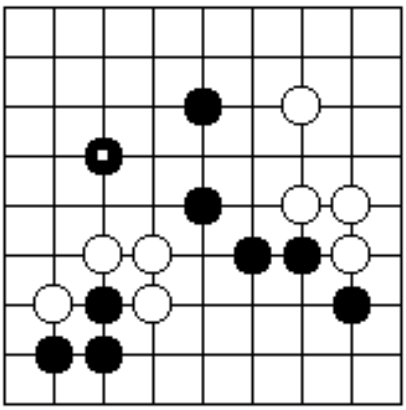

図 29: ヒラキの手

4.3 着手決定

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．

- 死活の結果
- 地とアゲハマ
  - 地，アゲハマ，全勢力範囲数

死活判定の結果は，死活判定部により求められた結果に表 2に応じた得点を与えることで求められる．

表 2: 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)
“活き”	30	60	70
“死に”	-30	-60	-70
“不明”	5 α -5	10 α -10	5 α -15

このように，対局が進むにつれて “活き” と “死に” の差が増加するようにしている．死活判定部の結果が不明のときは，生死の可能性の度合に応じて得点を与える．つまり，この得点は，図 9で検査される項目に応じて活きる可能性を点数化する．α には，その石群が活きる可能性のある項目数を代入し計算する．このように，不明の石群の得点は，表 2に示される活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．また，不明な石群は，特に終盤においては，包囲網を脱出したり単独で二眼作るスペース的な余裕がなく，活きる事が出来る可能性が低いので得点を抑えている．

“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 3の重みをそれぞれに掛けている．“地 + アゲハマ” は，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．

表 3: “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

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．

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 1 を打った場合の局面を仮に生成し，状況判断部を呼び出して仮の局面の解析を行う．次に，コンピュータの敵側 (人間側) から見た立場で候補手を，この局面と状況表をもとに，通常の候補州選出と同様に選び出す．更に，敵側がこのうちの一つを打った場合の局面を生成し，また同様の処理を行う．そして，この探索が一定の深さに達したときに，評価関数を呼び出し盤面の得点を求める．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よう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． 木の構成は，各節点ごとに評価を同時に行いながら，不必要となった部

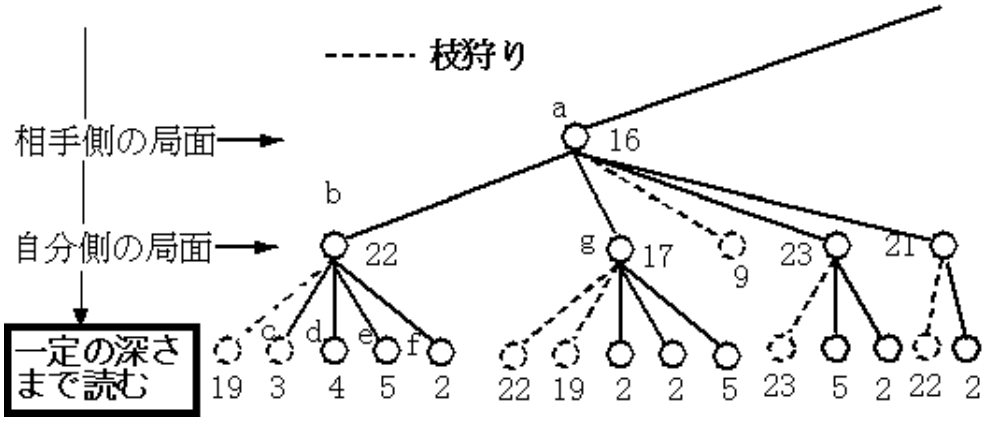

図 30: 着手決定の様子

分木は次々と削除していく方法をとっている．この様子を図 30に示す．得点はコンピュータ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．逆に言えば，敵側の手番では当然最も低い評価値が得られる手が選ばれる．

5　実対局によるシステムの評価

5.1　問題点と強くするための課題

現在，コンピュータ 9 路盤囲碁システムはテスト対局を行えるところまできているが，テスト対局をを行うにつれて問題点も明らかになってきている．

(1) 着眼点の絞り込みを行っていない

本システムは，実際の着手を決定するまでに類似事例を用いて，その局面に”良く似た形”が現れたらそれを候補手とし，これをパターンマッチングにより実現している．しかし一方で，候補手の選出を類似度をもとにした手の価値の優劣に頼り，それを可能手全てについて考えているため、重要な場所に着眼する事が出来ず類似性は低いが現局面において重要な手が候補手となる上位 5 手に含まれない．この様子を図 31に示す．図 31の局面で

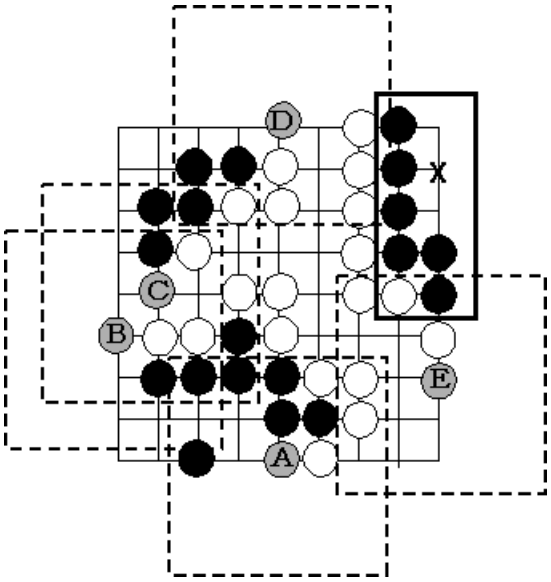

図 31: 重要な着手を逃す例

は，ある程度経験のある人間のプレイヤーならば太線内の石群を救済するためにＸに着手するのが最も重要であると判断するが，本システムでは，Ｘの手入れの手よりも他のＡ～Ｅの手の方が類似度が高くＸが候補手として選出されない事態となってしまう．本システムでは事例有効率　(類似度をもとに変換）に，”この事例を用いることによりどれだけ盤面が改善されるか”を表す評価改善量を考慮に入れるため，これを掛け合わせるという手段を取っている．しかし，評価改善量は数値的な大小しか意味を持たず，また，事例有効率と掛け合わせるにしても，基本的に評価改善量の良いものが事例として蓄積されるため，

差が付きにくく手の価値の優劣を決定するにも有効に機能していないようである．この問題を改善するにはパターンマッチングによる候補州選出の前に，手入れの必要な石群を見極め，どの石群に着眼するかを決定する必要がある．この様子を図 32に示す．

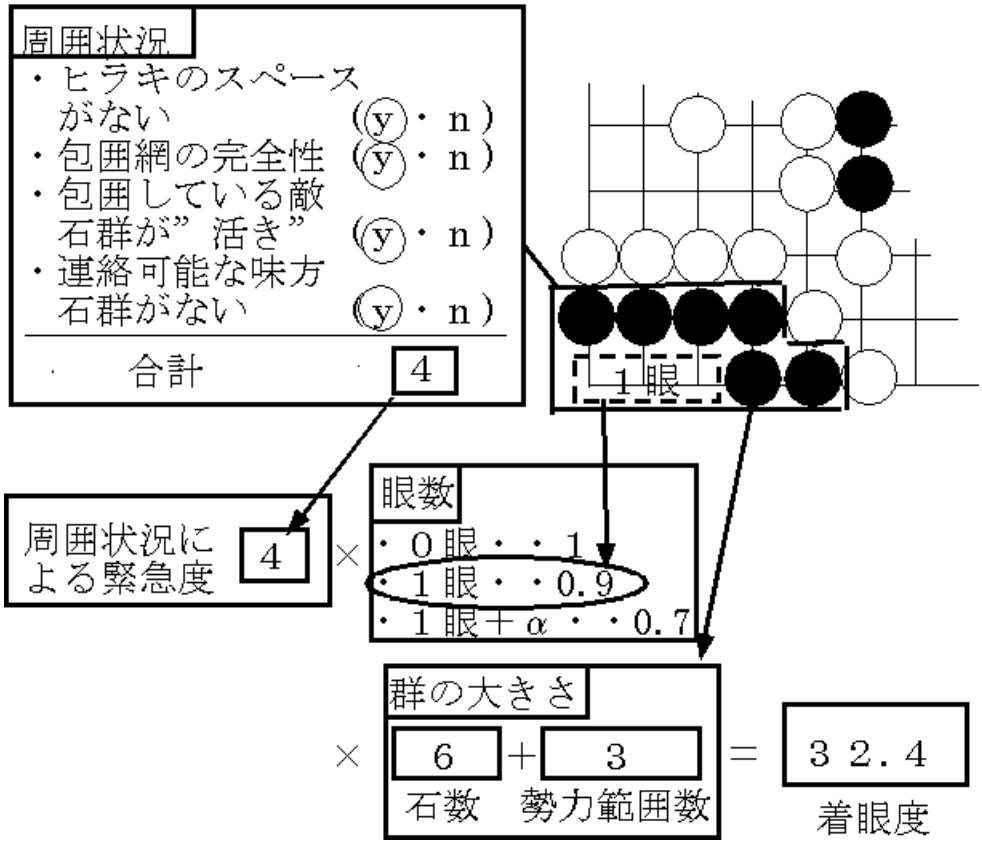

図 32: 着眼点の絞り込み

着眼点は，その石群の死活の緊急度をもとに決定される．具体的にその石群と包囲している敵石群，また連絡可能と考えられる味方石群との状況 ･関係を点数化して，どれだけの急を要する (急場とされる) 石群であるを決定する．更に，その石群の眼数に応じて係数を掛け緊急度に差をつける．また，この石群がウチアゲられた (捕獲された) 場合どれだけ被害を被るかを考慮に入れるため”群の大きさ”を掛ける．この結果が”着眼度”となる．これが最も高い石群を着眼点とする．

(2) 着手に一貫性がない

もう一つの大きな問題が相手に対する”受け”が出来ずに流れに沿った着手が行われないという点である．これを図 33に示す．図 33の局面では，☆の手に対しては左の１～３の

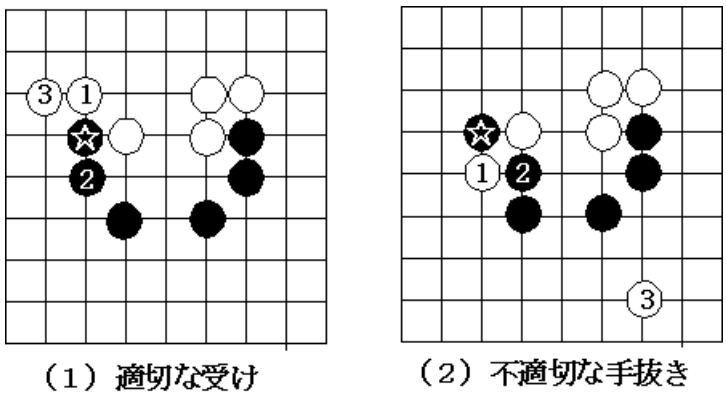

図 33: 受けと手抜き

手順の手の交換で一段落というのが相場である．しかし本システムでは，わざわざ 1 と自らキリにいってるのに，いきなり手抜きをし，一連の着手の交換をせずに各石を自ら窮地に陥れてしまう．これも原因として考えられるのは候補手選出を実局面と事例形の類似性に主眼を置いているためで，更に，事例自体には一手分だけの情報のみ記載されているのも，この問題点の大きな要因である．これを改善するには，やはり事例に一手だけでなく一段落するまでの手順を記載する手段が有効だと思われる．そのため，現在 5× 5 サイズで表現されている事例を，m× n の事例に応じたサイズに変更しなければならない．

以上のの問題点を考慮すると，候補手 ~ 着手決定の流れは図 34のようになる．m× n パターンは，各々の事例が固有のサイズを持ち，一手だけでなくその後の手順まで表す．そのため，この事例を用いるためには，図 34のとおり現時点の局面がm× n パターンの手順中か否かの判断を行わなければならない．また，局面として複数のm× n パターンの手順中である状況も考えられる．この場合，手順の最後まで仮打し，盤面の評価の良い方を着手とする．この際，パターンに手順が記載されているため木探索を行う必要はない．

(3) 序盤の長考

本システムは，序盤に必要以上に長考してしまうという問題点を持っている．これは，序盤から状況判断 ~ 類似度をもとにした手の価値による候補手選出 ~ 木探索による着手選出という一連の処理を行っているためである．更に，布石段階ではあまり意味のない連の捕獲の正否も読んでいるために，余計に処理が重くなってしまう．また，本システムの通常の着手決定過程をそのまま用いると，手の価値が類似度に依存しているため、序盤から ” キリ ” や ” ワリコミ ” 等の戦いを挑む手をいきなり打ってしまう．通常囲碁では，序盤は ” 布石と呼ばれる空き空白地の占有やそれに付随する ” ツケノビ ” に代表されるような定石が打たれる．

これらの問題を解決するには，序盤はパターンマッチングを行わずに，単純に”トビ”,”ヒラキ”などを定石的に打つ定石ライブラリを組み込む必要がある．また，これを盤面サイズ9× 9のパターンとして，先に述べたm× nパターンの一部として扱うことを考えている．

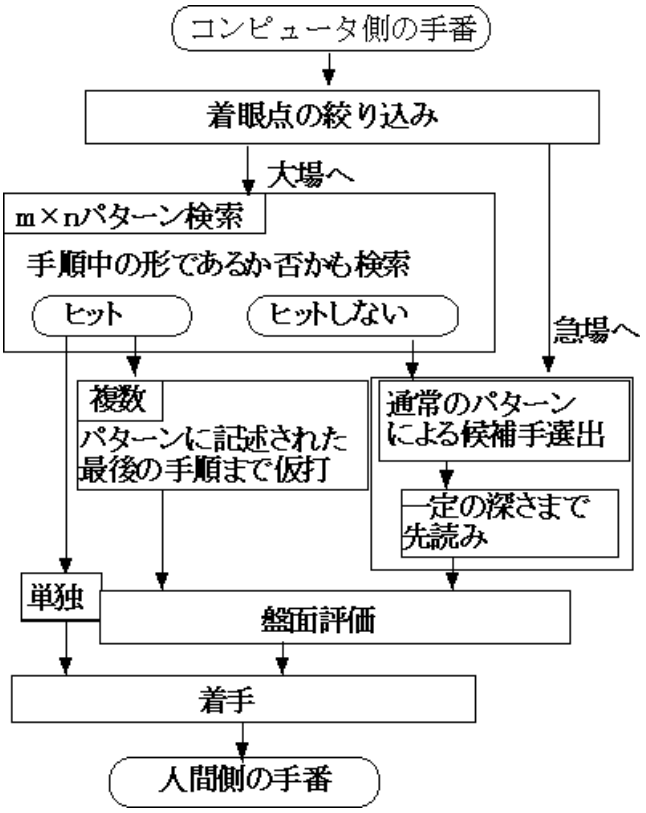

図 34: 着手の流れ

5.2 １９路盤囲碁システムの構築へ向けて

１９路盤システムの実現へ向けては，まず現在使用している5× 5サイズの事例を１９路盤に応じたサイズへと変更を迫られることも考えられるが，実際のところは事例のサイズが可変であるほうが，より類似性を正確に判定出来そうである．また，１９路盤は当然９路盤より更に大きいサイズであるため，現在の攻防単位である”群”だけでは局所的過ぎて上手く盤面を解析できないことが予想される．例えば，図35のように黒が大きく囲い”模様を張る”という状況を表現するには，個々の群や連だけでは不十分で，距離の離れた複数の群と連を組み合わせて新たな概念の石の集合を生成する必要がある．

図35のような石の集合を扱うことが可能ならば，これに着眼し黒の模様を”消し”に行く★のような手を候補手に加えることも可能である．また，システムに与える攻防の局面

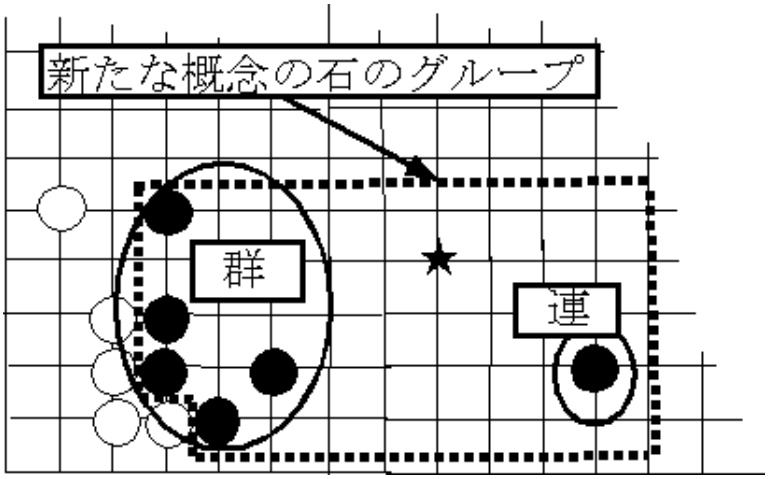

図 35: 新たな概念の石のグループ

の切り出しも適切なものとなる．これに対して，図 ref19-2 のように局地的な攻防の切り出しを行う必要もある．こちらはむしろ９路盤システムで用いているいたような局所的な形知識のパターンとの類似度と先読みで着手を決定する手法が適していると考えられる．

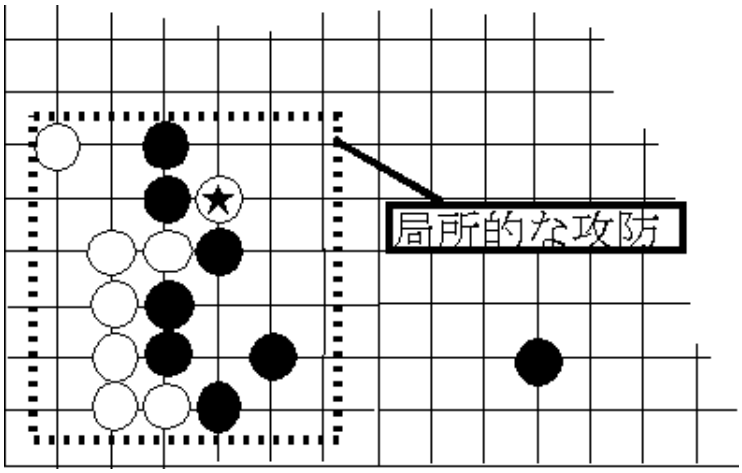

図 36: 局地的な攻防

また，１９路盤サイズのシステムの場合，広大な盤に対し序盤から無駄な探索を行うわけにはいかないので，定石ライブラリの組み込みは必須である．幸い，囲碁 (１９路盤) というゲームには一般化された”定石”が多々存在するので，特に問題なく組み込み可能であると考えられる．”手筋”に関しても同様である．

6　おわりに

本論文では，類似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候補手選出に用いられる事例ベースの構築，および死活判定や勢力計算，地計算などの評価値を計算する状況判断部，パターンマッチングと局面の優先度による候補手選出部，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，本システムの実対局によるテストを行ない，問題点の明確化及び改善策と１９路盤システムに向けての課題を述べた．

コンピュータ 9 路盤囲碁システムの問題点としては，やはり候補手選出のまずさが最たる例として挙げられる．これは，類似事例を候補手選出に利用する際に，類似度が大きなウェートを占めるパラメタとなっているために，ただ単に” 良く似た形” が候補手として選ばれ，本当に重要な手が候補手に含まれないという問題であった．つまり，本システムは重要な局面を着眼・絞り込みを行なうことをせずに，無差別に類似性の高い手を検索対象としており，これが適切な着手を妨げている．これを解決するための手法として，状況判断を更に詳細にし，局面の優劣をはっきりしなければならないが，現状の評価値では限界があり，まず死活判定によって危険な石群を救出するために，急場に着眼する戦略の実装を目指すべきである．更に高度な” 戦略 / 着眼ルーチン” を導入するには，何らかの新たな概念の評価値を導入する必要がある．また，現在実装されている各パラメータの調整も必要であろう．

また，システムが流れを切らない” 受け” の着手をするためには，現在使用している 5 × 5 サイズの事例だけでは不足である．そのため，今後は一手だけでなく，その後の手順まで記した m × n サイズの事例を収集 / 生成していく必要がある．ただし，これまで 5 × 5 サイズの事例のように，棋譜ファイルから自動的に頻出する形を得ることは困難であり，手作りで地道に作り上げていくしかない．

この他，本システムには，ヨセやコウ，セキに関する機能が実装されておらず，特に終盤のシステムの弱さの大きな要因となっている．ヨセは，先手・後手の概念があり，ただ単にパターンで記述できないので少々複雑な処理が必要である．しかし，ヨセの解法は David Wolfe らによって組み合わせゲーム理論を用いることにより確立されている [16]．また，コウについても定式化が成されているが [17]，本システムは現在，千日手を打たないというルールのみの実装にとどまっている．また，序盤の長考を回避するために，定石ライブラリの実装も必要である．

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，本システムのインターフェース部を担当して頂きました黒柳良太郎 氏，西山章彦 氏に心より感謝致します．

更に，9 路盤棋譜を御提供頂きました日本棋院所属プロ棋士の羽根泰正 九段に心より感謝致します．

参考文献

[1] 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).

[2] 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).

[3] Jay Burmeister and Janet Wiles : Accessing Go and Computer Go Resources on the Internet, Game Programming Workshop, pp.75-84 (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] 陳志行 ： HandTakK (手段対局), The Profile(The Third FOST Cup World Open Computar Go Championship), 科学技術融合振興財団, pp.1 (1997).

[9] Michael Reiss :GO ４ + +, The Profile(The Third FOST Cup World Open Computar Go Championship), 科学技術融合振興財団, pp.2-5(1997).

[10] 山本成辰：シルバー囲碁, The Profile(The Third FOST Cup World Open Computar Go Championship), 科学技術融合振興財団, pp.44-45(1997).

[11] 田中正浩 ： 琵琶子, The Profile(The Third FOST Cup World Open Computar Go Championship), 科学技術融合振興財団, pp.17(1997).

[12] 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).

[13] 北村 佳己 ：事例を用いたコンピュータ 9 路盤囲碁システムの実現，立命館大学大学院理工学研究科情報システム学専攻修士論文 (1996).

[14] 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 (1996).

[15] 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).

[16] Elwyn Berlekamp，David Wolfe原著，吉川，小林，石原訳: 囲碁の算法 ~ ヨセの研究, 株式会社トッパン (1994).

[17] 菰田和史：コンピュータ囲碁におけるコウの形式化，九州大学工学部情報工学科卒業論文 (1996).

[18] DAVID B.BENSON : Life in the Game of Go, INFORMATION SCIENCES, pp.17-29(1976).

[19] 河田汎 :コンピュータ囲碁実験，Game Programming Workshop, pp.1-7 (1995).

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

[21] 小島 琢矢, 吉川 厚, 植田 一博, 永野 三郎：囲碁名人のモデル構築に向けて ~ 認知科学と人工知能の融合の検証，1997 年度日本認知科学会，冬のシンポジウム資料集 「ゲームと認知科学」， pp.25-31 (1997).

[22] 小島 琢矢, 植田 一博, 永野 三郎：生態学アナロジーを用いた囲碁パタン知識の獲得，Game Programming Workshop, pp.133-140 (1996).

[23] 小島 琢矢, 植田 一博, 永野 三郎：囲碁の棋譜データからのパタン、定石、詰碁知識の獲得，Game Programming Workshop, pp.9-18(1997).

[24] 近藤繁，井狩一紀，清水賢資 :詰め碁プロセッサ Gokyo，Game Programming Workshop, pp.1-8 (1997).

[25] 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 回全国大会講演論文集 (2) ,3E-4 , pp.59-60 (1996) .

[26] 加藤恒, 山崎勝弘：事例を用いたコンピュータ 9 路盤囲碁における死活判定，情報処理学会第 53 回全国大会講演論文集 (1) ,2M-3 , pp.145-146 (1996) .

[27] 加藤恒, 山崎勝弘：類似事例を用いたコンピュータ 9 路盤囲碁システム，情報処理学会第 56 回全国大会講演原稿 (1998) .2S-1(発表予定），