JAIST Repository
https://dspace.jaist.ac.jp/

Title	深層学習囲碁プログラムによる形勢の制御と戦略の演出
Author(s)	施，源
Citation	
Issue Date	2021-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/17154
Rights	
Description	Supervisor:池田 心，先端科学技術研究科，修士（情報科学）

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

修士論文

深層学習囲碁プログラムによる形勢の制御と戦略の演出

SHI　Yuan

主指導教員　　池田　心

北陸先端科学技術大学院大学
先端科学技術研究科
（情報科学）

令和３年３月

Abstract

Computer Go programs have surpassed top-level human players by using deep learning and reinforcement learning techniques. Other than the strength, entertaining Go AI and AI coaches are also interesting directions but have not been well investigated. Some researchers have worked on entertaining beginners or intermediate players. One topic is position control, aiming to make strong programs play close games against weak players. Under such a scenario, the naturalness of the moves is likely to influence weaker players' enjoyment. Another topic is producing various strategies (or preferences), which human players usually have. Some methods for the two topics have been proposed and evaluated for a traditional Monte-Carlo tree search (MCTS) program. However, there are some critical differences between traditional MCTS programs and recent programs based on AlphaGo Zero, such as Leela Zero and KataGo. For example, recent programs do not run random simulations to the ends of games in MCTS, making the existing method for producing various strategies not applicable.

In this paper, we first summarize such differences and some resulted problems. We then adapt existing methods as well as propose new methods to solve the problems, where promising results are obtained.

For position control, the modified Leela Zero can play gently against a weaker player (48% of wins against a weaker program, Ray). A human subject experiment shows that the average number of unnatural moves per game is 1.22, while that by a simple method without considering naturalness is 2.29.

We also propose a new position control method specifically for endgames by using expected territory advantages instead of win rates. KataGo, with the proposed new method, can play gently against a weaker player Ray with 37% of wins in endgames while using our previous position control method is 63%.

Finally, for producing various strategies, a new method is introduced. In our experiments, center- and edge/corner-oriented strategies are produced by our method, and human players use five-level evaluation to evaluate the strategies that -2 mean center-oriented strategies and +2 mean edge/corner-oriented strategies. The average evaluation scores of 13x13 center-oriented is -0.5, 13x13 edge/corner-oriented is +0.7, 19x19 center-oriented is -1.25, 19x19 edge/corner-oriented is +0.85. The results show that human players can successfully identify the strategies.

概要

近年, 深層学習と強化学習の発展によって, コンピュータ囲碁プログラムは人間のトップのプロ棋士よりはるかに強くなった. そのため, 強さに関する研究以外にエンタテインメントや教育に関する研究も注目されているが，現段階ではまだ十分に行われていない．エンタテインメントや初心者と中級者のための教育領域では主に 2 つのトピックが注目されている．1 つ目のトピックは形勢の制御であり, 強いコンピュータ囲碁プログラムを弱化しながら, 不自然な着手を回避することを目的としている．2 つ目のトピックは多様な戦略（棋風）の演出であり，人間プレイヤが用いる戦略をコンピュータ囲碁プログラムに実現することを目的としている．この 2 つのトピックに関する研究は既に伝統的なモンテカルロコンピュータ囲碁プログラムで提案され, 実験で評価された. しかし, 最新の深層学習コンピュータ囲碁プログラムは AlphaGo Zero モデルをベースとしており，伝統的なモンテカルロコンピュータ囲碁プログラムとは大きく異なる．そのため，従来の手法をそのまま使えない恐れがある．例として，深層学習コンピュータ囲碁プログラムでは, モンテカルロ木探索のとき, 終局までランダムにシミュレーションする必要がなく，それを前提とした過去の多様な戦略の演出法は使えない.

本論文では，まず深層学習コンピュータ囲碁プログラムと伝統的なモンテカルロコンピュータ囲碁プログラムの違いを要約し，その違いに生じた問題点を説明する. その後, 発見した問題点について, 解決策を提案し, 実験で評価する.

形勢の制御部分では，新手法を実装した Leela Zero は相対的に弱い相手である Ray に対して十分な手加減を行い，ほぼ五分の対戦成績となった．また，被験者実験によって，着手の自然さを考えない従来の方法では，1 局の平均の不自然な着手数が 2.29 であったが, 新手法の 1 局の平均の不自然な着手数は 1.22 であった．すなわち新手法によって不自然な着手数を減らすことが可能であることを確認した.

さらに，終局の盤面に対して，形勢の制御の新方法を提案した．勝率ではなく地合い差を用いることによって, 終局の盤面での形勢の制御を試みた. 終盤の盤面で提案手法を実装した KataGo を相対的に弱い相手である Ray と対局させたところ，Ray は 100 局中 63 勝であった．従来の手法で Ray は 100 局中 37 勝であったことから，より上手く手加減出来ていることを確認した.

最後に, 多様な戦略の演出を実現するため, 着手の位置によって選択確率に重みづける方法を提案した．また，提案した手法を用いて中央派/実利派の戦略を再現し，再現した戦略が人間プレイヤに認識されるか評価するために被験者実験も行った．被験者実験では再現した戦略を−2（中央派に見える）から +2（実

利派に見える）までの 5 段階で評価させた．その結果，13 路盤中間派が−0.5，実利派が+0.7，19 路中央派が−1.25，実利派が+0.85 と評価されたことから，人間プレイヤが再現した戦略を認識できることを証明した．

目次

第 1 章 はじめに 1
第 2 章 囲碁と囲碁プログラム 3
2.1 囲碁 3
2.1.1 囲碁の基本ルール 3
2.1.2 呼吸点と取り 3
2.1.3 石の死活 4
2.1.4 対局の戦略と段階 5
2.1.5 勝敗判定とコミ 6
2.2 モンテカルロ囲碁プログラム 7
2.2.1 モンテカルロ法の進化 7
2.2.2 モンテカルロ木探索 7
2.2.3 UCT 法 9
2.3 AlphaGo と AlphaGo Zero 9
2.4 Nomitan，Ray，Leela Zero と KataGo 11
第 3 章 先行・関連研究 13
3.1 教育囲碁 13
3.2 形勢の制御 14
3.3 自然な手加減のための先行研究 15
第 4 章 コンピュータ囲碁プログラムの違い 17
4.1 モンテカルロ木探索の仕組みの違い 17
4.2 勝率予測の精度の違い 17
4.3 相手の着手より遠く打つ傾向 18
第 5 章 形勢の制御 20
5.1 従来の手法の問題点 20
5.1.1 有望な着手のみ探索する 20
5.1.2 手抜きをする 20
5.1.3 高勝率時に不自然な手を打ちやすい 21
5.2 提案手法 22
5.2.1 有望でない着手への探索資源の分配 22
5.2.2 前の着手との距離による選択確率の補正 23
5.2.3 高勝率時に用いる手法の改良 24

5.3 実験 24
5.3.1 形勢の制御を評価する 25
5.3.2 自然さを評価する 25
第 6 章 地合い差に基づく形勢の制御 28
6.1 終盤の問題点 28
6.2 アイデアと概念 28
6.3 提案手法 29
6.4 実戦の典型例 30
6.5 実験 31
第 7 章 多様な戦略の演出 34
7.1 問題点 35
7.2 提案手法と概念 35
7.3 実験 36
7.3.1 実験設定 36
7.3.2 強さの変化に関する評価 37
7.3.3 数値実験 37
7.3.4 被験者実験 37
第 8 章 終わりに 39

図目次

図 2.1：石の呼吸点 4
図 2.2：石の取り① 4
図 2.3：石の取り② 4
図 2.4：石の死活 4
図 2.5：隅・辺・中央 5
図 2.6：序盤 5
図 2.7：中盤 6
図 2.8：終盤 6
図 2.9：モンテカルロ木探索の 4 つのステップ 8
図 2.10：AlphaGo Zero のニューラルネットワークの学習 10
図 2.11：AlphaGo Zero のモンテカルロ木探索 11
図 4.1：Ray（横軸）と Leela Zero（縦軸）の勝率推定 18
図 4.2：Ray と Leela Zero の直前の着手との距離の着手数の分布 19
図 5.1：手抜きの例 21
図 5.2：探索集中度の変化例 22
図 5.3：Ray，$\mathrm{Leela}_{\mathrm{A15}}$ と $\mathrm{Leela}_{\mathrm{ABC25}}$ の直前の着手との距離の着手数の分布 26
図 5.4：Ray（黒）対 $\mathrm{Leela}_{\mathrm{ABC25}}$（白） 27
図 6.1：終盤の典型例 30
図 6.3：19 路盤の終盤例 31
図 6.2：13 路盤の終盤例 31
図 7.1：中央派/実利派の例 34
図 7.3：戦略の例 35
図 7.2：13 路盤の碁盤 35
図 7.4：19 路盤の実利派（黒）対中央派（白）の例 38

表目次

表 3.1：形勢の制御方法の例 16
表 5.1：図 5.1 の探索リスト 21
表 5.2：図 5.2 の探索リスト(exploration=0.9) 22
表 5.3：図 5.2 の探索リスト(exploration=10.0) 23
表 5.4：４組の Leela の実験結果（95%信頼区間） 26
表 6.1：$\mathrm{Leela}_{\mathrm{ABC25}}$ の探索リスト 30
表 6.2：KataGo の探索リスト 30
表 6.3：実験結果 32
表 7.1：実験のパラメータ設定 36
表 7.2：中央に（四線と四線以上）着手する平均着手数（95%信頼区間） 37
表 7.3：被験者実験の結果 38

第1章 はじめに

人工知能領域には，囲碁というボードゲームで人間のプロ棋士に超えた実力を達するのは，長期的な目標であった．2016 年，深層学習技術の発展に伴い，AlphaGo［1］と呼ばれるコンピュータ囲碁プログラムが初めてトッププロ棋士に勝利した．そして 2017 年, AlphaGo Zero［2］の公開によって，多くのコンピュータ囲碁も AlphaGo Zero モデルを使用し，人間を超えた強さになっており，これまで強さに関する研究はもう十分に進んだと考えられる．一方，教育とエンタテインメントに関する研究はまだ十分に行われていない．特に，現在の日本には，囲碁を指導することができる指導者の数が不足にしており，人間に指導することができるコンピュータ囲碁に関する研究は重要な課題である．このように楽しませたり指導するのは他のゲームでも不十分だから，囲碁の知見が使えれば世の中の人を幸せにできると考えられる．

教育囲碁とエンタテインメント囲碁に最も重要な課題は形勢の制御である[3][4]．すなわち，盤面上の形勢をいいバランスに維持することが重要である．もしコンピュータ囲碁プログラムが強すぎた場合，弱いプレイヤや初心者プレイヤは簡単に負けるだろう．囲碁のルールには対局前にいくつかの石を盤面上に配置する『置き碁』というハンデキャップがある．しかし九個の石を置いても，現在最新の囲碁プログラムには勝てない可能性が高い．また，多くの石を置くことは人間プレイヤの楽しみを害すると考えられる．そのため，教育囲碁プログラムやエンタテインメント囲碁プログラムは弱い着手を適度に選択すべきである．同時に，明らかな不自然さを人間プレイヤに感じさせる着手を回避すべきだと考えられる．

もうひとつ重要な課題は多様な戦略の演出である[3][4]．人間プレイヤは様々な棋風を持っている．例として隅や辺を重視する実利派，中央を重視する中央派，攻撃的な着手を打つ好戦派，平和的な展開に導く平和派などが挙げられる．そこで，単調な囲碁プログラムより，多様な戦略の演出ができる囲碁プログラムの方が人間プレイヤを楽しませるのではないかと考えられる．

自然な形勢の制御と多様な戦略について，過去池田ら[3][4]は手法を提案したが，この論文は Nomitan という伝統的なモンテカルロコンピュータ囲碁プログラムを使っていた．現在最新の深層学習囲碁プログラムにはそのまま使えない，なぜかというと，深層学習囲碁プログラムがモンテカルロ囲碁プログラムより圧倒的な強さを持っているため，従来の手法より手加減が必要であるからだ．そして人間よりはるかに強いため，人間プレイヤ特に初級者とは違った傾向（相

手の手から遠くに着手するなど）が多く出る．また，伝統的なモンテカルロコンピュータ囲碁プログラムでは，探索を行うとき，終局までランダムにシミュレーションし，領域の計算によって勝負を判断し，その結果を親ノードへのパスに沿って全てのノードを更新する．しかし，深層学習囲碁プログラムでは，探索のシミュレーションを行う時，バリューネットワークから出力した値で勝敗を計算するため，終局までランダムにシミュレーションする必要がない．最後までシミュレーションをする必要がない．故にシミュレーションの最後の盤面評価に工夫するタイプの手法は使えない．

本研究の目的は『深層学習コンピュータ囲碁プログラムを用いることで形勢の制御と多様な戦略の演出がどのように困難になるのか』『その困難をどうすれば解決できるか』を明らかにすることである．そのためにまず，現存する手法を深層学習囲碁プログラムに実装し，どんな結果が出るかを試す．そして試した結果により，従来の手法の問題点を発見し，改良策を提案する．また，多様な戦略について，従来の手法では，モンテカルロ木探索の時，終盤のスコア計算のステップに，中央と辺の石の価値を変えることで，中央派と実利派を実現している．しかし，深層学習囲碁プログラムでは，終盤まで探索を行わないため，従来の手法が使えない．そこで，新しい手法の提案も望んでいる．

なお，本研究の内容は第 41 回ゲーム情報学 (GI) 研究発表会にて発表した“深層学習囲碁プログラムを用いた場合の手加減に関する研究”［5］，2019 International Conference on Technologies and Applications of Artificial Intelligence（TAAI2019）にて発表した“Position Control and Production of Various Strategies for Deep Learning Go Programs”［6］および，Journal of Information Science and Engineering(JISE)にて発表した“Position Control and Production of Various Strategies for Game of Go Using Deep Learning Methods”［7］らを基に整理，加筆したものである．

本論文の構成は次の通りである．第 2 章では本研究で対象とする二人零和有限確定完全情報ゲーム『囲碁』のルール説明と今までのコンピュータ囲碁プログラムの紹介を行う．第 3 章では関連研究について紹介する．第 4 章では伝統的なモンテカルロ囲碁プログラムと深層学習コンピュータ囲碁プログラムの違いとその違いに生じた問題点を説明する．第 5 章では従来の形勢の制御のアプローチの改良と実験について述べる．第 6 章では終盤の形勢の制御の手法と実験について述べる．第 7 章では多様な戦略の演出のアプローチと実験について述べる．最後に第 8 章で結論と将来の課題を述べる．

第2章 囲碁と囲碁プログラム

本章ではまず囲碁と囲碁のルールについて，ウィキペディア[8][9]を参考しながら紹介する．その後，囲碁プログラムの進化の道に沿って，有名な囲碁プログラムを紹介する．

2.1 囲碁

囲碁は 2500 年前に中国で発明した，今までも世界中に大勢の人がプレイするボードゲームだ．2 人のプレイヤが，碁石と呼ばれる白黒の石を，通常 19×19 の格子が描かれた碁盤と呼ばれる板へ交互に配置する．一度置かれた石は，相手の石に全周を取り囲まれない限り，取り除いたり移動させたりすることはできない．ゲームの目的は，自分の色の石によって盤面のより広い領域を確報することである．

2.1.1 囲碁の基本ルール

囲碁は 2 人のプレイヤが領土を競うゲームである．お互いの領土を競うために，2 人のプレイヤのうちの一方が黒い石，他方が白い石を，空白の碁盤から交互に配置する[8]．

2.1.2 呼吸点と取り

盤上の交点に石を置いたとき上下左右に隣接した交点が存在する．石はこの点を使って呼吸をしていると考えることができ，この点を呼吸点と呼ぶ．隣接点に味方の石がある場合，味方の石を通じて呼吸ができ，一つでも呼吸のできる石があれば，その石の全体が呼吸できる[8]．

呼吸点を図 2.1 に示す．黒石 A は×に示した 4 つの呼吸点がある．白石 B は下に隣接点がないため，×に示した 3 つの呼吸点がある．黒石 C は下と右に隣接点がないため，×に示した 2 つの呼吸点がある．白石 D と E は，×に示した 6 つの呼吸点がある．

隣接点が空点であれば，呼吸ができる，つまり生きている．隣接点に相手の石があれば呼吸を邪魔される，上下左右四方向とも相手の石にふさがれると窒息してしまい取られてしまう．隣接している複数の石も，呼吸のできる石が一つも無くなった場合は，その石の全体が窒息し取られてしまう[8]．

石の取りを図 2.2 に示す．白石が 1 に着手すると，黒石 A の呼吸点が全て無

くなり，白番に取られてしまう．黒番の着手 2 と 4 および白番の着手 3 も同じく相手の石を取られる．石が取られた後の局面を図 2.3 に示す．

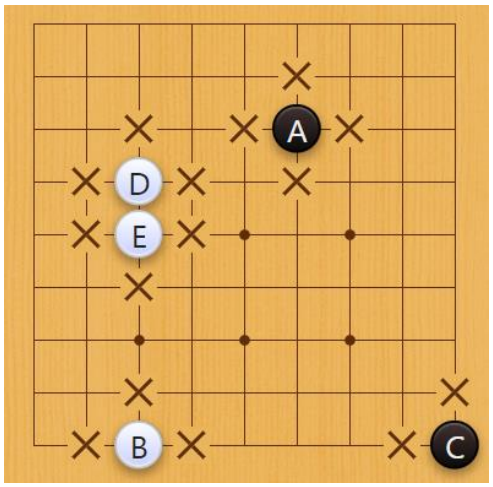

図 2.1：石の呼吸点

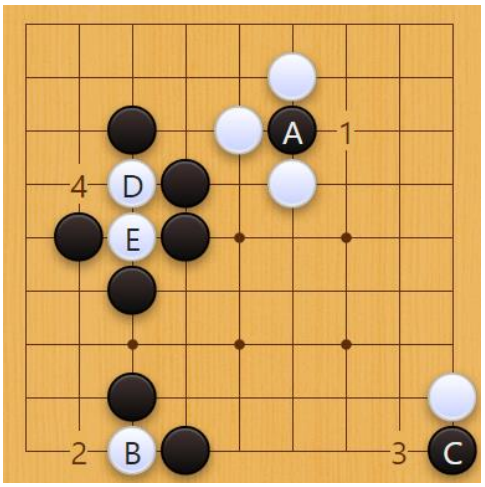

図 2.2：石の取り①

2.1.3 石の死活

囲碁において，『相手に絶対に取られる事の無い石』と『取られても新しく取られない石を置ける石』を活きた石，逆にどうあっても取られる運命にある石を死んだ石と表現し，これらを合わせて（石の集団の）死活と呼ぶ[8]．

『相手に絶対に取られる事の無い石』を図 2.4 に示す．囲碁は基本的にどこに打っても良いのですが，着手禁止点というルールがある．黒の立場から見ると，A や B の場所は石を置いても既に呼吸点がないため，打った瞬間取られる形になってしまう．こういう場所を『着手禁止点』（または『眼』）と呼ぶ．この眼を二つ以上持つ石の一団は相手の着手禁止点を少なくとも 2 箇所以上持つため，周囲の空点の全てに敵石が置かれても取られることはない．このような『絶対に取られることの無い石』のことを活き石と呼ぶ[8]．

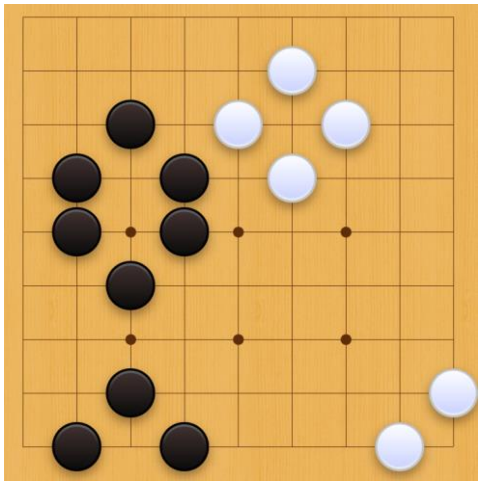

図 2.3：石の取り②

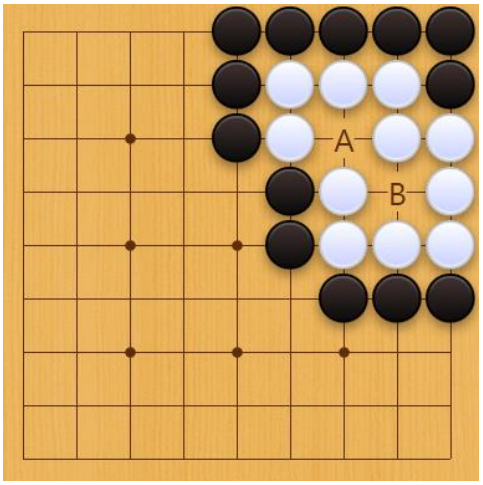

図 2.4：石の死活

2.1.4 対局の戦略と段階

囲碁はだいたい『序盤』『中盤』『終盤』の 3 つに分けられる．『序盤』は初めのころで，石を盤上に散らして，だいたいの陣形を整える．通常，『序盤』ではプレイヤはそれぞれの戦略を考えながら着手する．例として，隅や辺など価値の高い領域を早めに手に入れる『実利派』と碁盤の中央への影響を重視する『中央派』などが挙げられる．以下に序盤の戦略と実戦例を示す．

囲碁の盤面は隅・辺・中央の三部分に分けられる．『隅・辺・中央』の例を図 2.5 に示す．囲碁は石を使い，多い領域を囲むゲームである，隅の領域（1）では 8 個の石で 16 個の交差点（16 目）を囲んだ．辺の領域（2）では 8 個の石で 8 目を囲んだ．中央の領域（3）では 8 個の石で 4 目を囲んだ．この図を見てみると，同じ数の石では，隅で領域を囲むのは一番効率的である．二番目に効率的であるのは辺を囲むことである．中央を囲むことは一番非効率的である．なぜかというと，隅は碁盤の二つの端を使えるため効率が良く，中央は碁盤の端を使えないため石を多く使わないと領域を囲めないからである．そのため，対局の序盤では隅，辺，中央の順番に注目することを囲碁の教育で重視すべきである．一方，隅の効率が中央より良いのは一般論であるが，隅が必ずしも中央の価値より高いとは言えない．なぜかというと，中央で領域を囲むのは難しいが不可能ではないためである．プロ棋士の中でも中央を重視する人が多くいる．

『序盤』の例を図 2.6 に示す，黒番の武宮正樹対白番の趙治勲の対局である．武宮正樹の碁は地よりも中央での展開を重視した『宇宙流』と呼ばれている，一方，趙治勲は序盤から徹底的に実利を稼ぐ『実利派』である．二人の対照的なスタイルの対局は，常に注目を集めた．図を見てみると，黒石が中央重視，白石が隅・辺を重視していることが分かる．

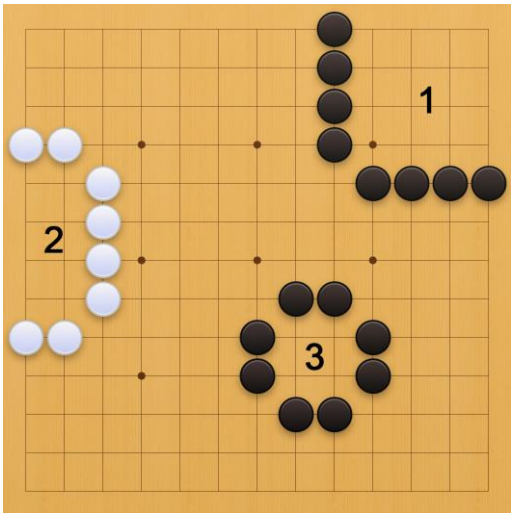

図 2.5：隅・辺・中央

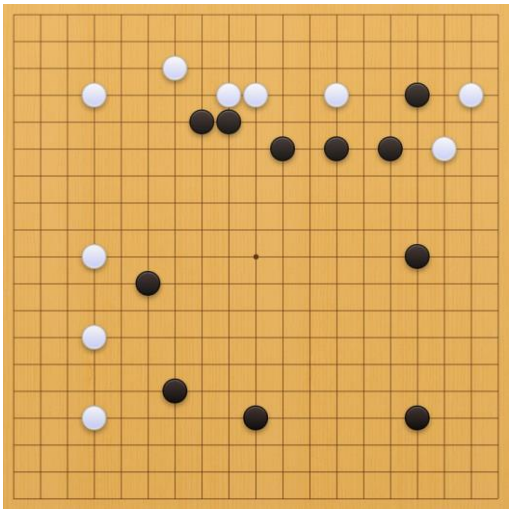

図 2.6：序盤

『中盤』は死活の絡んだ戦いになる．互いに死活がはっきりしていない弱い石を意識しながら打ち進める．『中盤』の例を図 2.7 に示す．黒番と白番も中央の領域を手に入れるため，これから多少戦いになる盤面である．

『終盤』では双方共に死活の心配が無くなり，互いの地の境界線を確定させる段階を目指す．『終盤』の例を図 2.8 に示す．両方のだいたいの領域はもう確定したが，互いの地の境界線をまだ定めていないため，終局まで価値の高い領域を争う．

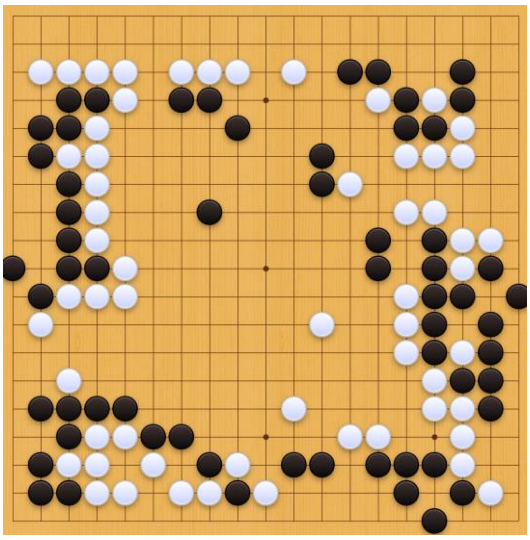

図 2.7：中盤

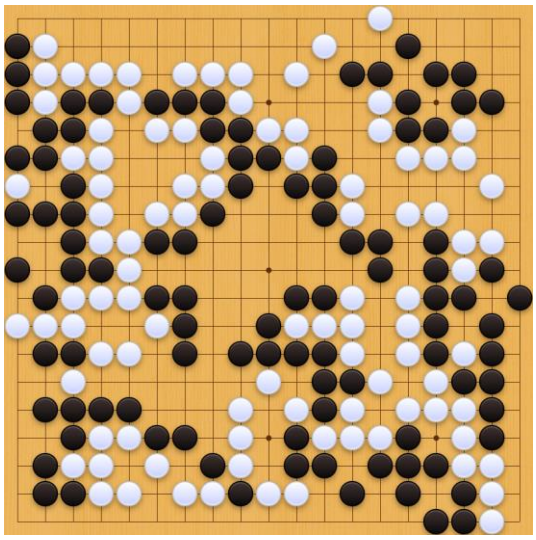

図 2.8：終盤

2.1.5 勝敗判定とコミ

地の一点を『一目』という．地の面積は，交点の数で数え，単位は目である．後述する『コミ』が採用されている場合は，それも計算に含める．双方の地の目数を比較して，その多い方を勝ちとする．

囲碁は先着する権利のある黒番のほうが有利である．そのため，互角の条件の対局（互先）ではコミを白番の地に加えて勝敗を決定する．日本ルールではコミ 6 目半である．日本棋院のデータによる，2002 年から 2007 年にかけてコミ 6 目半で行われた日本棋院公式棋戦での集計では，19702 局で黒番の勝率が 50.59%，白番の勝率が 49.41%とほぼ均等であった[27]．

2.2 モンテカルロ囲碁プログラム

囲碁プログラムでは，探索空間の大きさと評価関数の難しさが起因して，これらのプログラムが人間の初段と互先で戦って勝つのはほぼ不可能という評価をされてきた．しかし，2000 年代後半に入ってモンテカルロ法を導入することにより，アマチュア段位者のレベルに向上したとされた．

2.2.1 モンテカルロ法の進化

囲碁プログラムの評価関数の設計が難しいが，囲碁というゲームでは終局した時点でどちらが勝利したのか簡単に計算可能である．この性質に利用し，1993 年，ランダムな候補手で終局まで対局をシミュレーションし，その中で最も勝率の高い着手を選ぶというモンテカルロ法を応用したアルゴリズムを持つ囲碁プログラムが登場した[10]．当時は，コンピュータの性能が低かったことと，単純なモンテカルロ法はランダムな着手によってプレイアウトを行ったため，従来の手法より弱かった．

2006 年，ゲーム木探索とモンテカルロ法を融合し，勝利の高い着手により多くのプレイアウトを割当てプレイアウト回数が基準値を超えたら一手進んだ局面でプレイアウトを行う『モンテカルロ木探索』を実装したコンピュータ囲碁プログラムが登場し[11]，急速にその手法が広がり他の多くのコンピュータ囲碁プログラムも同様のアルゴリズムを採用するようになり，コンピュータ囲碁プログラムの棋力向上を果たすようになった．2008 年モンテカルロ木探索を採用した『MoGo』がプロ棋士と対戦した，ハンデの無い 9 路盤での対局は 3 局対戦し 1 局に勝利した．通常用いられない 9 路盤であるとはいえ，コンピュータがプロ棋士に公の場で互先（ハンデなし）で 1 勝を挙げたのは史上初のことだった．

2.2.2 モンテカルロ木探索

本節では，『モンテカルロ木探索』のウィキペディア[12]に参考しながら，コンピュータ囲碁プログラムの進化における重要な手法であるモンテカルロ木探索について紹介する．

モンテカルロ木探索は，最も良い着手を選択するため，ランダムサンプリングの結果に基づいて探索木を構築する．ゲームでのモンテカルロ木探索は，最後までプレイしたシミュレーション結果に基づいて構築する．ゲームの勝敗の結果に基づいてノードの値を更新し，最終的に勝率が高いことが見込まれる手を選

択する[12].

最も単純な方法は，全ての有効な選択肢に，同数ずつプレイアウトの回数を一様に割り振り，最も勝率が良かった手を選択する方法である．これは単純なモンテカルロ木探索と呼ばれる[12].

モンテカルロ木探索の 4 つのステップは以下の手順からなる．

- 選択：根ノード r から始めて，葉ノードのどれかにたどり着くまで，子ノードを選択し続け，たどり着いた葉ノードを l とする．根ノードが現在のゲームの状態で，葉ノードはシミュレーションが行われていないノード．より有望な方向に木が展開していくように，子ノードの選択を片寄らせる方法は，モンテカルロ木探索で重要なことであるが，次の章の所で後述する[12].
- 展開：l が勝負を決するノードでない限り，l から有効手の子ノードの中から c を一つ選ぶ[12].
- シミュレーション：c から完全なランダムプレイアウトを行う．プレイアウトというのは，終局までランダムに着手をすることである[12].
- バックプロパケーション：c から r へのパスに沿って，プレイアウトの結果を伝搬する[12].

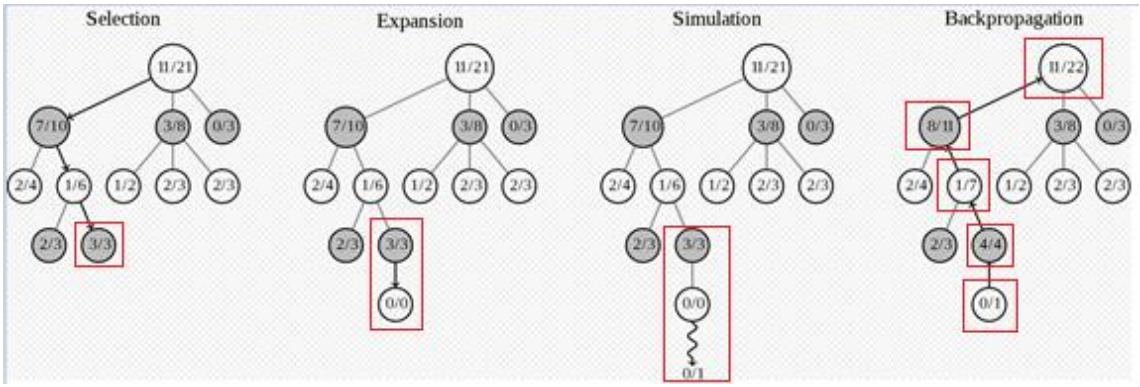

図 2.9：モンテカルロ木探索の 4 つのステップ

『モンテカルロ木探索の 4 つのステップ』は図 2.9 に示した．各ステップの選択を表している．ノードの数字は，そのノードからのプレイアウトの『勝った回数/プレイアウトの回数』を表している．Selection のグラフでは，今，黒の手番である．根ノードの数字は白が 21 回中 11 回勝利していることを表している．裏を返すと黒が 21 回中 10 回勝利していることを表していて，根ノードの下の 3 つのノードは手が 3 種類あることを表していて，数字を合計すると 10/21 になる[12].

シミュレーションで白が負けたとする．白の 0/1 ノードを追加して，そこから

根ノードまでのパスの全てのノードの分母（プレイアウト回数）に 1 加算し，分子（勝った回数）は黒ノードだけ加算する．引き分けの際は，0.5 加算する．こうすることで，プレイヤは最も有望な手を自分の手番で選択することができる．そして計算の制限時間に到達するまで，この 4 つのステップを反復し，最も勝率が高い手を選択する[12].

2.2.3 UCT 法

モンテカルロ木探索の選択のステップで述べた，子ノードを選択する際の難しい点は，何回かのプレイアウトの結果により，高い勝率であるという知識利用とプレイアウトの回数が不足しているので探索することのバランスを取ることである[12].

UCT 法（Upper Confidence Tree）では，探索と知識利用のバランスを取る一つの方法である．UCT は UCB1（Upper Confidence Bound 1）[21]に基づく方法である．この方法はマルチアームバンディット問題でうまく解決できている[12].

UCT 法は以下の式（2.1）に示す.

$$\frac{w}{n} + \mathrm{c}\sqrt{\frac{lnN}{n}} \tag{2.1}$$

各変数は以下のである．w は勝った回数．n はこのノードのシミュレーション回数．N は全シミュレーション回数．c は定数パラメータ.

式を見てみると，第一項の勝率は知識利用である．第二項は探索を表現していて，シミュレーション回数が少ないのを選択するようにする.

2.3 AlphaGo と AlphaGo Zero

前述したモンテカルロ木探索より，コンピュータ囲碁プログラムの強さが大幅に上がったが，19 路盤で人間のプロ棋士に勝つことはまだ遠い目標であった.

近年では計算機の計算能力の増大と深層学習技術の発展によって，コンピュータ囲碁の強さが一層大きく躍進した．特に 2016 年，AlphaGo[1]と呼ばれるコンピュータ囲碁は，当時に世界中で最も強い囲碁棋士 Lee Sedol 氏に勝利した．AlphaGo は現在の盤面の情報を入力し，次の一手を予測する「ポリシーネットワーク」と現在の盤面を評価する「バリューネットワーク」，二つの深層畳込みニューラルネットワークがある．深層学習を用い，二つのニューラルネットワークを人間の棋譜から学習し，そして強化学習で予測精度を向上させる．学習

した深層畳込みニューラルネットワークとモンテカルロ木探索を組合せ，次の着手を決める．

AlphaGo をはじめとして，深層学習囲碁プログラムが主流になっており，特にその後継者の AlphaGo Zero[2]は，完全に人間の知識を用いず，AlphaGo の方策予測ネットワークと局面の価値予測ネットワークを組合せ，一つの深層ニューラルネットワークになっており，一つのニューラルネットワークに現在の盤面を入力し，次の一手の選択確率の分部と現在の盤面の評価を出力する．この深層ニューラルネットワークを自己対戦で強化学習し，AlphaGo の棋力を超えた．

AlphaGo Zero のニューラルネットワークの学習を図 2.10 に示す．

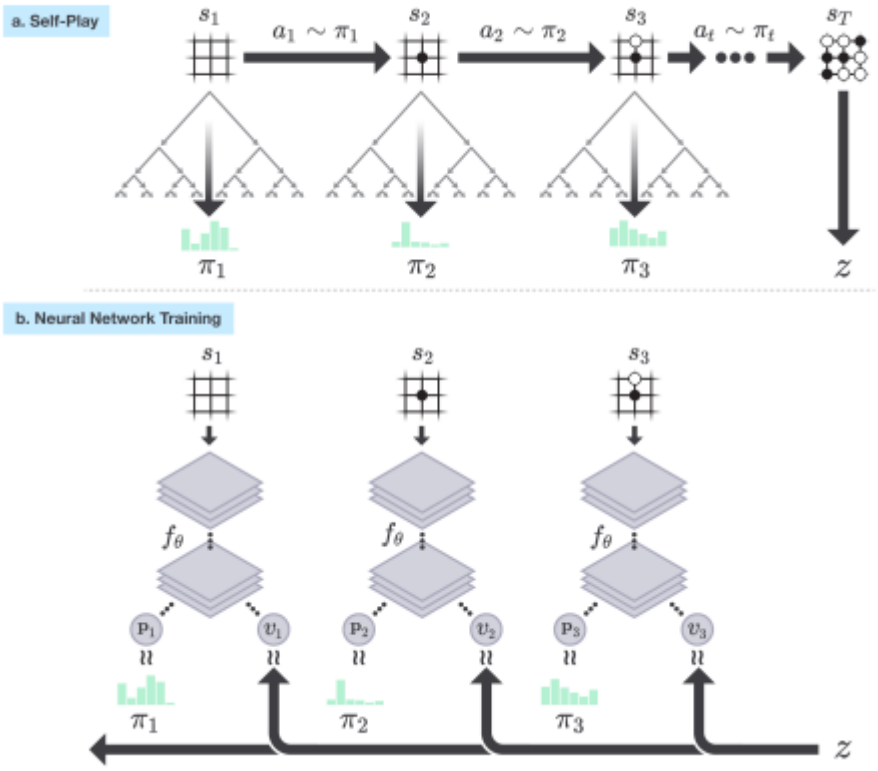

図 2.10：AlphaGo Zero のニューラルネットワークの学習

AlphaGo Zero の学習では，まず自己対戦を行う（a）．現在のニューラルネットワークを使い，各盤面s_iに対して，モンテカルロ木探索で，探索後の選択確率（訪問回数によって確率分布）π_iを計算する．そしてゲームの終局までに進み，対局の結果 z を記録する．

ニューラルネットワークの学習（b）では，勝敗の結果 z と確率分布π_iに従って，各盤面よりニューラルネットワークを更新する．

AlphaGo Zero のモンテカルロ木探索を図 2.11 に示す.

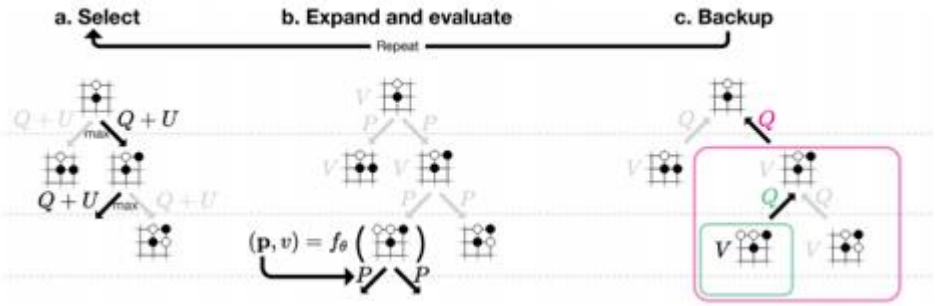

図 2.11：AlphaGo Zero のモンテカルロ木探索

- 選択：根ノードから，葉ノード L にたどり着くまで，評価値『Q＋U』最大の子ノードを選択し続ける．Q は盤面の評価（勝率），U は UCB 値であり，以下の式（2.2）に示す.

$$\mathrm{U} = c_{puct} \cdot \mathrm{P}(s,a) \cdot \frac{\sqrt{\sum_b N(s,b)}}{1+N(s,a)} \qquad (2.2)$$

  P(s,a)は盤面 s で着手 a の選択確率, N(s,a)は盤面 s で着手 a の訪問回数, c_{puct} は定数パラメータ.
- 展開：子ノードを展開し，ニューラルネットワークにより，p と v を得る．p は次の全ての着手の選択確率の分部，v は盤面の評価（勝率）.
- 更新：展開した子ノードの盤面の評価 v により，根ノードへのパスの全てのノードの訪問回数 N と盤面の評価 Q を更新する.

図を見てみると，従来のモンテカルロ木探索に比べ，一つ大きな違いがある．それはニューラルネットワークに得た盤面の評価 v を勝率として更新に使い，終局までのシミュレーションが不要になる点である.

2.4 Nomitan，Ray，Leela Zero と KataGo

本章では，本研究に関わっているコンピュータ囲碁プログラムについて紹介する.

Nomitan[13]と Ray[14]は二つの伝統的なモンテカルロコンピュータ囲碁プログラムである．Nomitan では KGS サーバーでアマチュア 3 段の実力を認定した．Ray では KGS サーバーでアマチュア 2 段の実力を認定した.

AlphaGo Zero や絶芸など有名なソフトはコードが未公開なので，各地で同じ

アイデアをもとに自作された，研究用に色んなツールが公開されている，プロ棋士も訓練のためによく使っている．本研究では，オープンソースされた二つのソフト，Leela Zero と KataGo を使っている．

Leela Zero[15]は AlphaGo Zero と同様な手法を用い，世界中で最も有名なオープンソースされたコンピュータ囲碁プログラムである．今の盤面での黒石と白石の配置，履歴，そして現在の手番を深層ニューラルネットワークに入力して，次の各着手の選択確率と今の盤面の評価を出力する．そして出力した選択確率と評価を用い，モンテカルロ木探索で次の着手を生成する．

KataGo[16]は同じく AlphaGo Zero の手法を用いたが，二つの特徴がある．一つ目はアルゴリズムの改良で自己対戦の学習効率とコストが AlphaGo Zero より大幅に改良し，家庭用コンピュータでもニューラルネットワークの訓練ができる．二つ目はモンテカルロ木探索後，局目のスコア，つまり両方の領域の差の値が予測できる．人間が囲碁を打つ時，両方の領域を計算しながら，次の着手を考える．そのため，KataGo を使い，人間の考え方と似合う手法が実現できると考えられる．

第3章 先行・関連研究

近年，計算機の計算能力の増大と深層学習技術の発展によって，強いコンピュータ囲碁の研究が盛んになってきている．特に 2016 年 AlphaGo[1]が人間のトップ棋士に勝利した．更に 2017 年 AlphaGo Zero[2]の公開により，世界中で AlphaGo Zero の手法をもとに，多くの人間よりはるかに強いコンピュータ囲碁プログラムが作成された．しかしながら人間よりはるかに強くなったコンピュータ囲碁プログラムには，殆どの人間プレイヤはハンデなしでまともに戦えなくなっている．そのため，十分な強さを持っているコンピュータ囲碁プログラムを如何に応用するかは今後の大きな課題だと考えられる．特に教育囲碁の分野では，囲碁を指導することができる指導者の数が不足しており，更に囲碁教室やプロ棋士の指導碁などの教育にはコストがかかってしまうため，囲碁の普及が非常に難しい状況になっている．このことから，教育囲碁に関する研究では大きな価値があると考えられる．

3.1 教育囲碁

教育囲碁の分野では，Yen らは人間プレイヤによって対戦での囲碁の学習効率について調べた[17]．Yen らは，日本棋院が運営するインターネット囲碁対局サービスである『幽玄の間』でアマチュア級位からアマチュア段位までのコンピュータ囲碁プログラムを多数用意し，幽玄の間の利用者と対戦させた後にアンケート調査を行った．その調査の結果から，人間プレイヤが自分と同じレベルの相手と対戦したときに囲碁の学習効率が一番高いことが示された．

池田らは，人間プレイヤを楽しませる，また人間プレイヤに指導するために，コンピュータ囲碁に必要な要素と技術を以下に列挙した[3][4]．（A）相手モデルの獲得．（B）形勢の誘導．（C）不自然な着手の排除．（D）多様な戦略．（E）投了のタイミングと思考時間．（F）感想戦，検討，おしゃべり，などである．このうち（B）と（C）については 3.2 節で詳述する．

池田らは，（F）感想戦，検討，おしゃべりなどのために，機械学習を用いる囲碁の着手の日本語表現を提案した[22]．池田らは『形』を表現する単語をコンピュータに表現させるために，高段者に着手の名前を入力してもらい，それを訓練データとして教師つき学習を行うことによって，盤面と着手から適切な単語を出力するシステムを提案した．また，機械学習により得られた『形』の日本語表現をプロ棋士に評価してもらい，人間のアマチュア高段者にかなり近い性能を

得られていることが確認できた.

また, 池田らは囲碁における悪手検定の手法を提案した[23]. 高段者に着手が悪手であるかどうかを入力してもらい，それをデータとして教師つき学習を行うことによって, 着手が悪手であるかどうかを出力するシステムを提案した. また，ただの『悪手』を見つけるだけなら簡単だと考えられるが，指導碁が指摘するような悪手は理論的な悪手とは違うことも示した.

池田らは，（D）多様な戦略を演出するために，モンテカルロ木探索の結果に重みをづける手法を提案した[3][4]. モンテカルロ木探索では通常, 終盤までランダムに進めた後，終盤の地合を数え，白石側にコミを加え，勝敗を判断し，勝敗の結果を通過ノードにバックアップすることで訪問回数と勝利数を記録する. その地合を数える部分に，通常一つの交差点を 1 点に数える．多様な戦略を実現するために, 色んな条件で重みをづける方法がある. 例として, 隅と辺の交差点を 1.5 点，中央の交差点を 0.5 点にすると，勝敗が変わり，辺や隅に陣地を多く持つ側が勝ちと判定される，モンテカルロ木探索の結果も変えるなどが挙げられる．しかしこの方法は AlphaGo Zero モデルのコンピュータ囲碁プログラムに使えない．なぜかというと，AlphaGo Zero モデルのコンピュータ囲碁プログラムでは，ニューラルネットワークから得た盤面の評価を勝率として探索に用いるため終盤までシミュレーションする必要がないからである.

3.2 形勢の制御

楽しませる囲碁や教える囲碁のために必要なことはさまざまにあるが，その中でも最も大事なこととして，形勢を適切に制御するということが挙げられる[3][4].

コンピュータ囲碁だけではなくて, 一般的の手加減方法には, 主に二つの戦略がある．a）常に一定の弱さを演出する方法と b）形勢に応じて手加減の度合いを決める方法である．モンテカルロ木探索をベースにしたプログラムでは，a）の戦略を用いるためには相手の強さを知っている必要があり，その上でモンテカルロ木探索の探索回数を下げる，また弱い AI を使う方法がある．b）の戦略はモンテカルロ木探索の結果により，最善手の勝率を基準に盤面の形勢を評価し，その形勢に応じて手加減の度合いを決める方法である.

深層学習コンピュータ囲碁では，探索回数を減少させても十分な強さを持っており, 手加減の程度がかなり厳しいと考えられているため, a)の戦略ではなく b）の戦略について様々な研究が行われた.

伊藤らはコンピュータ将棋における棋力の調整方法を提案した[19][20]. コンピュータ将棋の評価関数を調整し，コンピュータ将棋の従来の評価値が 0

に近ければ近いほどこの方法の評価値が高い値になるように加工した．その上で最もこの方法の評価値が高い候補手を選択することで，現在の盤面で最もコンピュータ将棋の従来の評価値が 0 に近くなる着手を行う．囲碁の場合は，勝率が 50%に最も近づく着手を選ぶという簡単な手法もある．ただし，この方法は非常に悪い着手も打つ恐れがある

Wu らは非常に悪い着手を避けるために，そのような手が打たれる確率を 0 または十分に小さくするような調整方法を提案した[18]．モンテカルロ木探索が終わった後，各候補手の訪問回数をソフトマックスし，着手が選ばれる確率を以下の式（3.1）に示す．

$$\mathrm{P_i} = \frac{N_i^z}{\sum_j N_j^z} \tag{3.1}$$

各変数は以下の通りである．$\mathrm{P_i}$は着手$\mathrm{M_i}$が選ばれる確率，$\mathrm{N_i}$は着手$\mathrm{M_i}$の訪問回数，z は定数パラメータ．式を見てみると，z=0 のとき全ての着手が同じ確率，すなわちランダムに選ばれることが分かる．また，z→∞のとき訪問回数最大の着手が選ばれることが分かる．Wu らは定数パラメータ z の値の調整でコンピュータ囲碁プログラムの強さがコントロールできると示した．また，凄く悪い着手を避けるため，Wu らは訪問回数がある閾値以下の着手を捨てる方法も提案した．ただし，ソフトマックスポリシーでランダムに着手を選ぶのは，同じく悪い着手を選ぶ恐れがある．

池田らはさらに自然さを明示的に意識するため，盤面の形勢における勝率を制御する方法を提案した[3][4]．勝率が低いとき強い着手を選び，勝率が高いとき悪すぎないかつ不自然ではない着手を選ぶ．

勝率の制御と棋力の調整について実現した後に，（C）不自然な着手の排除が重要な課題になる．池田らは不自然な着手を「形が悪い手」，「流れにそぐわない手」，「明らかに損をする手」，「高度すぎる手」などに分類した[3][4]．更に，池田らは不自然な着手を排除するために[3][4]，Bradley-Terry モデル等によって，着手の選択確率という「静的な良さ」と，モンテカルロ木探索の結果によって勝率という「動的な良さ」を両方考えた．その両方の評価値も悪すぎないように，不自然な着手を回避することを狙った．

これらとは別の指摘として，伊藤らはコンピュータ将棋での実験で「強さが一貫性していない」，「意図性を感じない」という不自然なパターンを指摘した[19]．

3.3 自然な手加減のための先行研究

本研究は，池田らの形勢の制御手法を先行研究として行ったことであるため，

本節では池田らの手法を詳しく説明する．

池田らは形勢に応じて手加減の度合いを決める方法[3][4]について，以下の手順からなる勝率制御をする．

1、 用いたプログラムにモンテカルロ木探索を行い，有望な順にソートする．この際，一部の手のみに探索が集中しすぎないように，モンテカルロ木探索の C 値を大きめにする．
2、 1 位の手の勝率と 2 位の勝率差がT_{uniq}以上の場合，唯一の手があると見られる．明らかに悪い着手を打たない為，1 位の手を着手する．
3、 1 位の手の勝率がT_{min}未満の場合，低勝率と見られる．簡単に負けることを避けるため，1 位の手を着手する．
4、 1 位の手の勝率がT_{min}以上T_{max}未満の場合，中勝率と見られる．1 位の手との勝率差がT_{dif}以下の手の中から最も選択確率が高い手を着手する．
5、 1 位の手の勝率がT_{max}以上の場合，高勝率と見られる．勝率差が大きすぎず同時に選択確率が小さすぎない手の中で，勝率を下げるために最も勝率の悪い手を着手する．そういう手が存在しなければ 1 位の手を着手する．

また高勝率の場合の条件には以下のように，勝率差がある程度大きくなっても，選択確率が大きければ認めるような式を用いた．具体的には，勝率差 3%以下かつ選択確率 5%以上．勝率差 4%以下かつ選択確率 10%以上．勝率差 6%以下かつ選択確率 20%以上．勝率差 8%以下かつ選択確率 40%以上．のいずれかを満たした場合に着手の候補に残す

着手	勝率	訪問回数	自然さ
M1	0.60	700	OK
M2	0.60	300	NG
M3	0.59	300	OK
M4	0.50	5	NG
M5	0.30	1	NG

表 3.1：形勢の制御方法の例

表 3.1 に形勢の制御方法の例を挙げる．通常のコンピュータ囲碁プログラムなら，訪問回数最大の M1 の着手を選ぶ．簡単な形勢の制御の方法だと，勝率が 50%に最も近づく M4 の着手を選ぶ，しかし，この着手は不自然な着手である．Wu らのソフトマックスポリシーの方法では，M1，M2 と M3 の着手が選ばれる確率が高いが，手加減無しの着手 M1 と不自然な着手 M2 を選ぶ恐れがある．池田らの方法だと，自然さを含めて考えるため M3 の着手を選ぶ．

第4章 コンピュータ囲碁プログラムの違い

形勢の制御と多様な戦略の既存手法は，伝統的なモンテカルロコンピュータ囲碁プログラムにおいて成功した[3][4]．しかし，伝統的なモンテカルロコンピュータ囲碁プログラムと深層学習コンピュータ囲碁プログラムは強さと仕組みが大きく異なるため，既存手法を直接に使用できない．まず，使用したコンピュータ囲碁プログラムの強さの差があるから，形勢の制御の部分ではより強い手加減が必要である．本章では伝統的なモンテカルロコンピュータ囲碁プログラムと深層学習コンピュータ囲碁プログラムの違いを紹介する．本研究では主に深層学習コンピュータ囲碁プログラムの Leela Zero と KataGo，および伝統的なモンテカルロコンピュータ囲碁プログラム Ray，この 3 つのオープンソースのコンピュータ囲碁プログラムを用いる．

4.1 モンテカルロ木探索の仕組みの違い

深層学習コンピュータ囲碁プログラムと伝統的なモンテカルロコンピュータ囲碁プログラムの最も異なる部分は，モンテカルロ木探索の仕組みが異なることである．伝統的なモンテカルロコンピュータ囲碁プログラムでは，モンテカルロ木探索を行うとき，葉ノードを評価するためにゲームの終局までランダムにシミュレーションし，囲碁のルールにしたがって勝敗を判断する．しかし，現在の深層学習コンピュータ囲碁プログラムでは，バリューネットワークの出力を用いて葉ノードを評価する．すなわち，終局までランダムにシミュレーションを行う必要がなくなり，3.1 節後半で述べた従来の多様な戦略の演出の手法も使えない．そのため，多様な戦略の演出の新手法を第 7 章に提案する．

4.2 勝率予測の精度の違い

深層学習コンピュータ囲碁プログラムは伝統的なモンテカルロコンピュータ囲碁プログラムに比べて勝率予測の精度が非常に高い．図 4.1 は，13 路盤にて行われた 50 局のテスト対局における Leela Zero と Ray の勝率予測の散布図である．赤い三角形は 30 手目の着手であり，対局の序盤の例とする．緑の円形は 60 手目の着手であり，対局の中盤の例とする．黒い四角形は 100 手目の着手であり，対局の終盤の例とする．この図から，赤い三角形と緑の円形の縦軸の幅は横軸の幅よりも広いことが確認できる．これは，Ray にとって小さい優勢/劣勢

であっても，Leela Zero の方が大きく判断することを表している．伝統的なモンテカルロコンピュータ囲碁プログラムでは，特に序盤と中盤のとき，長いランダムシミュレーションを行っているため，現在の盤面にある優勢/劣勢がそのまま最終状態ひいては推定勝率に反映されにくい．

従来の手法は勝率をベースにしており，伝統的なモンテカルロコンピュータ囲碁プログラムと深層学習コンピュータ囲碁プログラムの勝率予測の精度は違うため，手法の調整が必要だと考えられる．したがって，我々は従来の手法のパラメータを調整し，高勝率時に用いる方法の部分を改良する手法を第 5 章に提案する．また，終盤のときに極端な勝率に対して，地合い差を用いる形勢の制御の手法を第 6 章に提案する．

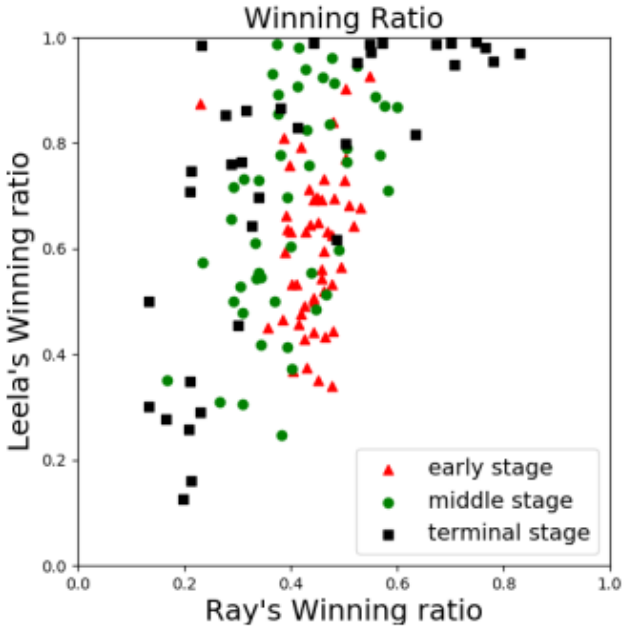

図 4.1：Ray（横軸）と Leela Zero（縦軸）の勝率推定

4.3 相手の着手より遠く打つ傾向

現在の深層学習コンピュータ囲碁プログラムでは，人間の知識を用いず，自己対戦でネットワークを学習させる．そのため，ポリシーネットワークの出力は，人間プレイヤの感覚と多少ずれている．特に，現在の深層学習コンピュータ囲碁プログラムは伝統的なモンテカルロコンピュータ囲碁プログラムに比べて相手の着手から遠い位置に着手する傾向がある．我々は，Leela Zero 対 Ray の 100 対局を用いて，相手の直前の着手とコンピュータ囲碁プログラムの着手間の平均ユークリッド距離を計算した．対局の最初の 60 手では，Leela Zero の相手の

直前の着手との平均ユークリッド距離が 3.17±0.08, Ray の相手の直前の着手との平均ユークリッド距離が 2.64±0.08 であった．このような場合には平均値が代表的な指標とは言えないかもしれないため，Ray と Leela Zero の直前の着手との距離の着手数の分部を図 4.2 に示す．横軸の d は直前の着手とのユークリッド距離を示す．図から，距離が 1 から 2 までのとき，Ray の頻度がが Leela より高い，距離が 4 から 9 までのとき，Leela は倍以上の頻度で着手する，そして，距離が 9 以上のときでは逆転して，Ray の頻度が Leela より高いと確認できる．通常，初心者や中級者にとって，相手からの攻撃や侵略に反応するのは当然なことである．そのため，相手の着手から遠いところに打つことは，初心者にとって『自分の着手が無視されたか，相手の反応が変だ』と思う可能性があり，楽しさを害するリスクがある．そこで，我々は相手の直前の着手とコンピュータ囲碁プログラムの着手間の距離を考慮して，ポリシーネットワークからの出力（選択確率）を補正することを提案する．この手法を第 5 章に提案する．

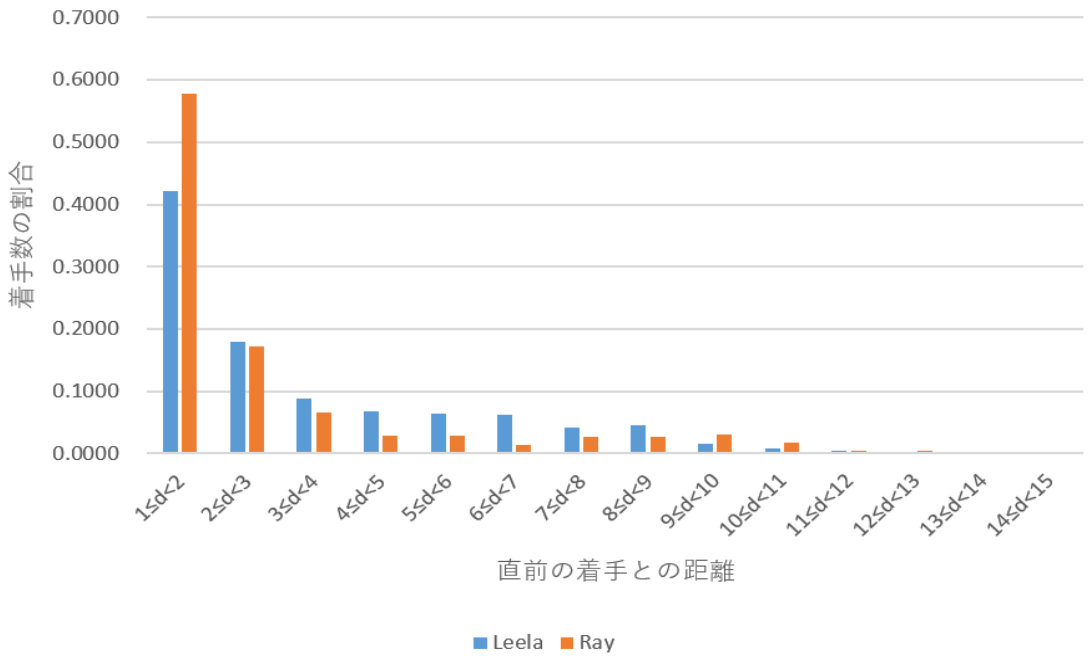

図 4.2：Ray と Leela Zero の直前の着手との距離の着手数の分布

第5章 形勢の制御

5.1 従来の手法の問題点

5.1.1 有望な着手のみ探索する

深層学習コンピュータ囲碁プログラムでは，探索の精度が高いため，モンテカルロ木探索のとき，PUCT アルゴリズムによって，勝率と訪問回数のバランスを考えながらノードを展開する．そのため，有望な着手のみに探索回数が集中し，有望な着手以外の着手の探索回数が全体的に少ない．一方，勝率の計算は，展開した全ての子ノードの平均勝率を計算するため，探索回数が少ない着手の勝率の信頼性が下がる．強さを求めるコンピュータ囲碁プログラムでは最善の着手を選ぶことを目的とするため，有望な着手のみ探索をすることは問題ないが，形勢の制御を目的としたコンピュータ囲碁プログラムでは有望な着手より悪すぎないかつ自然な着手を選ぶため，有望な着手以外の着手もより多く探察することが望まれている．

5.1.2 手抜きをする

手抜きとは，囲碁において，直前の着手に対応せずに，離れた場所に着手することである．

手抜きは不自然な着手として挙げられるが，大別して二種類存在する．(a) 戦いの最中や大きな欠陥を残すような場面で手を抜くこと，(b) 手を抜いても大きな損害が出ないような場合や局所には着手する必要がない場合，思い切って手を抜き，価値が高い場所に先着すること，の二種類である．

（a）の場合には明らかに悪手であるが，（b）の場合は囲碁のテクニックの一つであり，悪手ではない．ただ（b）の場合の手抜きでは初心者プレイヤに対して，その着手の意味が高度すぎて理解できないことが多いため，不自然な着手だと認識しやすい．

手抜きの例を図 5.1 に示す．黒石が C3（13）に着手した，白石の左下の領域を侵略することを狙っている．初心者にとって，白石は D3（a）に着手した方が自然に見えると考えられるが，形勢の制御の手法を実装した Leela Zero は L11（14）に着手した．これは悪い着手ではないが，初心者から見ると，不自然な着手と認識している可能性が高い．

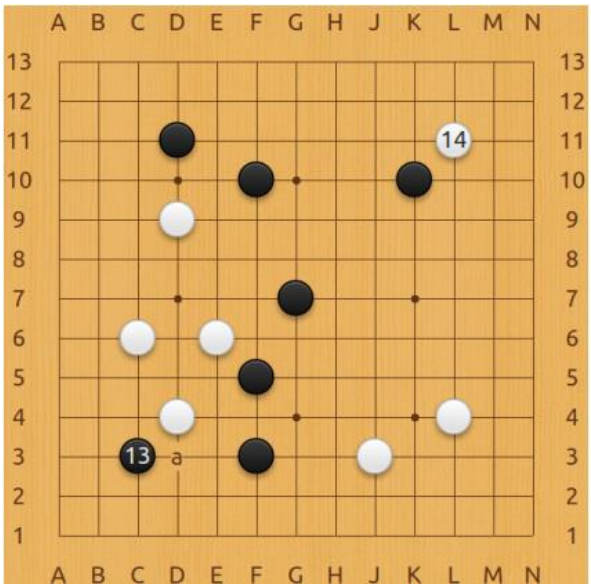

図 5.1：手抜きの例

5.1.3 高勝率時に不自然な手を打ちやすい

従来の手法では，候補手の中で勝率順 1 位の着手の勝率が閾値（0.55）を超えた場合，今の形勢を優勢と判断し，意図的に悪い着手を選ぶ．具体的には，まず候補手の中で以下に示す条件を満たしている全ての着手を抽出する．

（1）$w_1 - w_i < 0.03c$ かつ $p_i > 0.05$．（2）$w_1 - w_i < 0.04c$ かつ $p_i > 0.10$．

（3）$w_1 - w_i < 0.06c$ かつ $p_i > 0.20$．（4）$w_1 - w_i < 0.08c$ かつ $p_i > 0.40$．

ここでw_1は候補手の中で勝率順 1 位の着手の勝率，w_iは候補手の勝率，p_iは候補手の選択確率，c は手加減程度をコントロールする定数パラメータを表す（c の値が大きければ大きいほど，悪い着手を選べるようになっている）．そして抽出した全ての着手の中で勝率最低の着手を選ぶ．

図 5.1 は不自然な手を打ってしまった例である．黒石が C3（13）に着手した後，Leela Zero の探索では，D3（a）の勝率が 69.7%，選択確率が 0.400，L11（14）の勝率が 69.0%，選択確率が 0.139 であった．従来の手法では勝率を下げるため，L11（14）の着手を選んだが，多くの自然度（選択確率）を犠牲にしていることが確認できる．我々は，勝率を少し下げるために多くの自然度（選択確率）を犠牲にすることは不合理だと考える．

座標	勝率	選択確率
D3	0.697	0.400
L11	0.690	0.139

表 5.1：図 5.1 の探索リスト

5.2 提案手法

5.2.1 有望でない着手への探索資源の分配

有望な着手のみを探索する問題を解決するため，悪い着手ももっと探索しなければならないと考えられる．そのため，我々はモンテカルロ木探索の PUCT アルゴリズムの exploration パラメータを大きく設定した[1][2]. Leela Zero のデフォルトの exploration は 0.9 に設定されているが，本研究では exploration が 10 に設定した．同時に，探索の訪問回数が閾値以下の着手を捨てた．なお，次節の評価等では，この 2 つの工夫を合わせて方法 A と呼ぶことにする．

図 5.2 および表 5.2 と表 5.3 を用いて，exploration の変更による探索集中度の変化を説明する．図 5.2 は対局の序盤であり，白石の手番である．この盤面から探索回数 6000 回として，次の着手を探索した．このとき，exploration の値がデフォルトの場合の探索リストを表 5.2 に，exploration の値が 10 の場合の探索リストを表 5.3 に示す．また，それぞれの表に探索回数が 100 回以上の着手のみを載せている．表 5.2 と表 5.3 から，デフォルトの exploration では一つの着手に探索が集中していることを確認できる．また，exploration が 10 の場合では，デフォルトの exploration より多くの着手が探索されていることが確認できる．

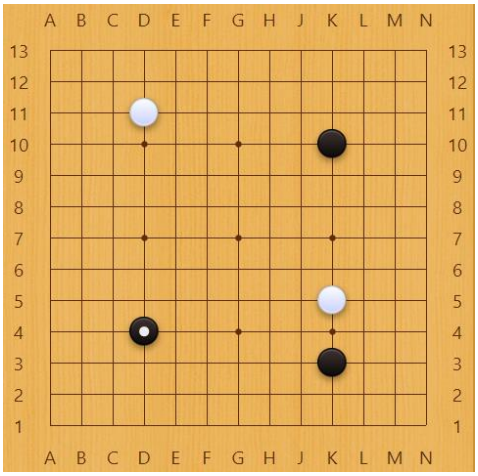

図 5.2：探索集中度の変化例

座標	訪問回数	勝率
H4	133	0.4903
C3	4661	0.4851
L3	724	0.4776
D3	131	0.4698

表 5.2：図 5.2 の探索リスト(exploration=0.9)

座標	訪問回数	勝率
C3	933	0.5119
D3	360	0.4891
L11	234	0.4859
L3	2631	0.4823
C4	227	0.4790
L8	231	0.4652
H3	104	0.4606

表 5.3：図 5.2 の探索リスト(exploration=10.0)

5.2.2 前の着手との距離による選択確率の補正

本節では，前述した『手抜き』という問題点の解決策を提案する．

囲碁は現在の局面さえ見れば理論的には最善手が分かるタイプのゲームであるにも関わらず，棋譜を用いて教師付き学習を行うと，最後の相手の着手との距離が近いほど選択確率が高くなることは以前からよく知られていた[24]．そこで我々は，自己学習によって得られたポリシーネットワークをそのまま用いるのではなく，相手の着手に近い着手ほど選択確率を高くするような補正を施すことを提案する．これによって，勝つために正しい選択確率からは遠ざかってしまうかもしれないが，人間にとっては自然に見える着手が選ばれやすくなると期待できる．なお，次節の評価等では，この工夫を方法 B と呼ぶことにする．

まず，相手の最後の着手の周辺には自分の石があるかどうかを確認する．もし相手の最後の着手の周辺には自分の石がなければ，相手の着手に対して応える必要がないため，手抜きと感じさせる可能性は低く，従って選択確率の補正は行わない．もし相手の最後の着手の周辺には自分の石があれば，相手の着手に応えるため，距離選択確率を導入する．13 路盤の実験では，相手の最後の着手からユークリッド距離が 3 以内の範囲に自分の石があれば，距離選択確率を導入した．

距離選択確率は，相手の最後の着手との距離を用い，ポリシーネットワークから得た選択確率に重みづけをした確率を表す．その方法（重み付づ）は以下の通りである．

- $p_i' = p_i \cdot 1.50 \quad (d_i \leq 2)$
- $p_i' = p_i \cdot 1.25 \quad (2 < d_i \leq 3)$
- $p_i' = p_i \cdot 1.00 \quad (3 < d_i \leq 4)$
- $p_i' = p_i \cdot 0.75 \quad (4 < d_i \leq 5)$

・ $p_i' = p_i \cdot 0.50$ $(5 < d_i \leq 6)$
・ $p_i' = p_i \cdot 0.25$ $(6 < d_i \leq 7)$
・ $p_i' = p_i \cdot 0.10$ $(7 < d_i)$

p_iはポリシーネットワークから得た選択確率，d_iは相手の最後の着手とのユークリッド距離，p_i'は距離選択確率を表す．

選択確率が補正されることにより，勝率が少し悪い場合に，着手候補から除外される可能性が減る，つまり着手される可能性が上がる．

5.2.3 高勝率時に用いる手法の改良

本節では，前述した『高勝率時に不自然な手を打ちやすい』の場合の問題点の解決策を提案する．

従来の方法では高勝率の時，前述した通り，勝率を少し下げるために多くの自然度（選択確率）を犠牲にした問題がある．このことから，着手を選択する際には選択確率も考慮すべきだと考えられる．以上を踏まえ，高勝率のときに勝率だけではなく選択確率も含めて着手を選択する新手法を提案する（以下方法 C で示す）．

新手法もまず候補手の中で従来の手法の条件を満たしている全ての着手を抽出する．そして抽出した着手の評価値を以下の式（5.1）により計算する．

$$\text{Gain} = (w_{max} - w_i) + \alpha \cdot p_i \qquad (5.1)$$

各変数について，Gain は着手の評価値，w_{max}は勝率 1 位の着手の勝率，w_iは着手の勝率，p_iは着手の選択確率，αは調整できる正の定数パラメータを表す．そして計算した各着手の評価値に基づく，評価値最大の着手を選ぶ．

この評価値では，$(w_{max} - w_i)$がどれだけ勝率を下げられるかを示し，p_iがどれだけ自然かを表しており，それをαで重みづけ加算した場合に，一番高い（良い）着手を選ぶことが望まれている．

図 5.1 の例では，αが 0.1 のとき，D3（a）の評価値 Gain は 0.040，L11（14）の評価値 Gain は 0.021 である．従って，新手法は D3 を選ぶようにする．

5.3 実験

本節では，改良した形勢の制御の手法は従来の形勢の制御の手法よりどれだけ優れているかを『形勢の制御』と『着手の自然度』の二つの方向で検証する．

本節の実験では，二つのコンピュータ囲碁プログラム Leela Zero と Ray を 13 路盤で用いた．Leela Zero では，13 路用のネットワークがないため，用いているのは我々が 2 週間訓練した，アマチュア高段レベルのネットワークであり，探索回数は各着手 6000 回とした. Ray は伝統的なモンテカルロコンピュータ囲碁プログラムである. その実力はアマチュア初段レベルであり, 探索回数は各着手 60000 回とした.

5.3.1 形勢の制御を評価する

我々は，まず訓練した Leela Zero の強さを評価した．形勢の制御の手法を実装していない場合，Leela Zero 対 Ray は Leela Zero が 30 戦全勝であった．この結果から今回訓練したネットワークを用いた Leela Zero は Ray より明らかに強いと考えられる．

次は従来の形勢の制御の手法を評価した．各設定は以下の通りである．T_{uniq}=0.08c，T_{dif}=0.03 c，T_{min}=0.35，T_{max}=0.55 と設定した，ここの c は手加減パラメータであり，ここでは 1.5 に設定した．従来の手法と異なり，5.2.1 節で述べた方法 A を実装した．以下，この手法を$\mathrm{Leela}_{\mathrm{A15}}$とする．この設定で Ray と Leela Zero を 500 局対戦させた，Ray は 500 局中 183 勝であった．この結果より，形勢の制御の手法は効果があるが，手加減の程度が不十分であると考えられる．

Leela Zero をより弱化するために手加減パラメータ c を 2.5 まで上げる，その上で不自然な着手を防ぐため，4.2.1 節の距離選択確率（方法 B）と 4.2.2 節の高勝率時の改良方法（方法 C）を実装した（以下この手法を$\mathrm{Leela}_{\mathrm{ABC25}}$とする）．このとき，方法 C の α は 0.25 に設定した．この設定で Ray と Leela Zero を 500 局対戦させた．その結果，Ray は 500 局中 238 勝であった．$\mathrm{Leela}_{\mathrm{ABC25}}$は明らかに$\mathrm{Leela}_{\mathrm{A15}}$より弱いと考えられる．

5.3.2 自然さを評価する

着手の自然さを無視すれば，強い人間プレイヤやコンピュータ囲碁プログラムは簡単に形勢を制御できる．しかし，弱いプレイヤは不自然な着手により勝利することを望んでいない．

そのため，着手の自然さを評価するために被験者実験を行った．被験者には三つのバージョンの Leela と Ray が対局した棋譜を見せ，それぞれの着手を評価させた．用いたバージョンは$\mathrm{Leela}_{\mathrm{A15}}$，$\mathrm{Leela}_{\mathrm{ABC25}}$と$\mathrm{Leela}_{\mathrm{navie}}$である．$\mathrm{Leela}_{\mathrm{navie}}$は方法 A を実装した上で，勝率が 50%に最も近い着手を選ぶ方法で

ある．

実験は，9 人の被験者を用いて行った．被験者の棋力はアマチュア 8 級からアマチュア 8 段まで様々である．実験はランダムに抽出した三つのバージョンの Leela Zero 対 Ray の対局各 5 局（対局最初の 60 手）を用い，各被験者は 2 時間で棋譜を見て評価した．

Leela のバージョン	Ray の勝率	不自然な着手数
Leela Zero	0.00	—
$\mathrm{Leela}_{\mathrm{navie}}$	—	2.29±0.45
$\mathrm{Leela}_{\mathrm{A15}}$	0.37±0.04	1.27±0.28
$\mathrm{Leela}_{\mathrm{ABC25}}$	0.48±0.04	1.22±0.28

表 5.4：4 組の Leela の実験結果（95%信頼区間）

実験の結果は表 5.4 で示す．着手の自然度の部分では，$\mathrm{Leela}_{\mathrm{navie}}$と我々の二人の手法の差は明確である．また$\mathrm{Leela}_{\mathrm{A15}}$と$\mathrm{Leela}_{\mathrm{ABC25}}$の不自然な着手数にはあまり差がないが，$\mathrm{Leela}_{\mathrm{ABC25}}$の方が強度の手加減をしていることを考えれば，手法 B と手法 C によって，不自然な着手が増えることが防げている．

また，100 局（対局最初の 60 手）中，$\mathrm{Leela}_{\mathrm{ABC25}}$の相手（Ray）の直前の着手とのユークリッド距離は 2.33±0.06 であり，Ray（2.65±0.08）と$\mathrm{Leela}_{\mathrm{A15}}$（3.16±0.10）より小さいことによって，手法 B は有効であることを証明した．また，Ray，$\mathrm{Leela}_{\mathrm{A15}}$と$\mathrm{Leela}_{\mathrm{ABC25}}$の直前の着手との距離の着手数のヒストグラムを図 5.3 に示す．横軸の d は直前の着手とのユークリッド距離を示す．$\mathrm{Leela}_{\mathrm{ABC25}}$は，距離が 1 から 6 未満まではアマチュアの着手頻度と同じくらいの値にできているが，距離 6 以上になるとアマチュアよりもかなり低い頻度でしか着手していないことが分かる．このヒストグラムの比較を行ったのは調整や実験のあとであって，本来ならばここまで遠くの着手の選択確率を下げることは避けるべきだったっかもしれない．

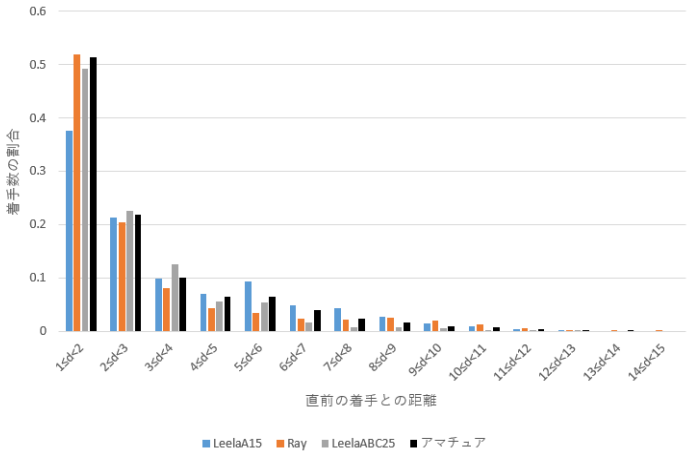

図 5.3：Ray，$\mathrm{Leela}_{\mathrm{A15}}$ と $\mathrm{Leela}_{\mathrm{ABC25}}$ の直前の着手との距離の着手数の分布

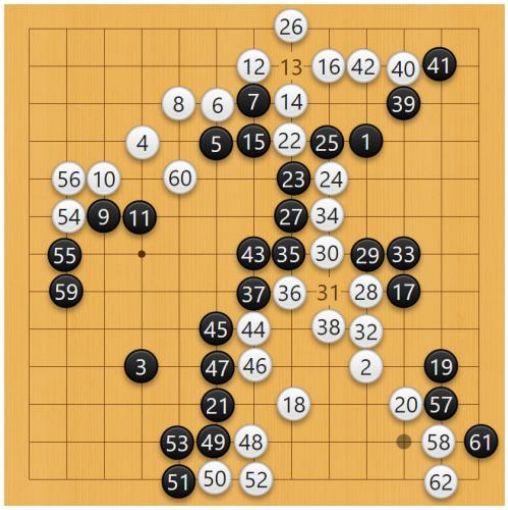

図 5.4：Ray（黒）対 $\mathrm{Leela}_{\mathrm{ABC25}}$（白）

Ray 対$\mathrm{Leela}_{\mathrm{ABC25}}$の対局の例は図 5.4 で示す．白石は特に不自然な着手を打たずに，平和的に終盤へ導いていた，そして最後の結果は黒石が半目勝ちである．

第6章 地合い差に基づく形勢の制御

本章では，形勢の制御の一つの大きな問題点『終盤の形勢の制御』について説明する．またこの問題を解決するため，地合い差に基づく形勢の制御の手法を提案し，実験で手法を検証する．

6.1 終盤の問題点

終盤のとき，形勢の制御には一つ大きな問題がある．人間の指導者が指導碁をするとき，もし下手側がミスしていないまま終盤に至り，終盤のときも両方の領域の差が小さい場合，上手が負けてあげることが普通である．例として，対局が終わる前に，半目勝ち（囲碁の中で一番小さい勝利）の着手と半目負けの着手があると，人間指導者は簡単に半目負けの着手を選び，下手に負けてあげることが挙げられる．

しかし，コンピュータ囲碁プログラムでは適当に負けてあげることが難しい，なぜかというと着手の勝率の差が大きすぎるからである．特に深層学習コンピュータ囲碁プログラムは高い精度を持っているため，極端な勝率が出やすい．例として，終局直前に半目勝ちと半目負けの着手がある場合，前者は勝率 99%，後者は勝率 1%などと大差があるものと評価されることが多い．よってこの場合，勝率に基づく手加減手法では，90%もの勝率低下を招く手は打てずに，半目勝ってしまうことが予想される．そのため，我々は地合い差（領域の差）に基づく方法を提案する．

6.2 アイデアと概念

人間プレイヤでは，囲碁を打つとき，両方の領域を計算し，領域差によって形勢を判断し，戦略と着手を決める．例として，『自分の領域が相手の領域より 5 目多いから，安全な着手をする』や『自分の領域が相手の領域より 2 目少ないから，積極的な着手をする』などが挙げられる．もしコンピュータ囲碁プログラムも領域の差を計算できるなら，形勢の制御と教育囲碁にとって，大きな進歩になる可能性がある．

我々が使っている AlphaGo Zero モデルのコンピュータ囲碁プログラム Leela Zero は，少なくとも実験に用いたバージョンでは地合い差を計算できないため，この問題にはもうひとつのオープンソースされたコンピュータ囲碁プログラム『KataGo』を用いる．KataGo では，モンテカルロ木探索を行うとき，同じく勝率，選択確率，訪問回数などの値を出力する．その上で，(a) 両方の領域の地

合い差の予測『scoreLead』.（b）両方の領域差の標準偏差『scoreStdev』.（c）各交差点が黒や白に属する確率『ownership』なども出力される．そして今回の手法は主に scoreLead を用いる．

本手法では地合い差を用い，適度な範囲内に地合い差を維持しながら，悪すぎないかつ自然な着手をすることを目標とする．勝率を用いてしまうと 1 目差が極端な差として評価されてしまうが．地合いを用いれば 1 目差がそのまま 1 目差（微差）として評価できる．

6.3 提案手法

本手法（以下 KataGo+で示す）は，対局の終盤だけで用いる．どんな盤面が終盤であるかを後の節で説明する．本手法の手順は以下の通りである．ここで p_i は着手 i の選択確率，δ_i は着手 i の地合い差，γ は定数パラメータであり，本手法の全てのパラメータは調整できるものとする．

1. 候補手を選択確率順にソートし，上位 20 手を抽出する．
2. もしある着手の選択確率が 0.9 以上であれば，その着手をする．
3. もしある着手の選択確率が 0.01 以下であれば，その着手を捨てる．
4. 各候補手の地合い差 δ_i を計算し，最大の地合い差を δ^*にする．
5. もしある着手が条件 $\delta_i<\delta^*-5$ を満たしていれば，その着手を捨てる．
6. 各候補手の評価値 s_i を計算し，評価値最大の着手をする．評価値の式は以下の通りである：

   (a) もし$\delta_i < -10$，$s_i = \frac{p_i}{\gamma^{-10-\delta_i}}$．劣勢の盤面であるため，より劣勢になる手を低く評価する．

   (b) もし$-10 < \delta_i < -4$，$s_i = p_i$．適当な範囲であるため，自然な（選択確率が高い）着手をする．

   (c) もし$\delta_i > -4$，$s_i = \frac{p_i}{\gamma^{4+\delta_i}}$．少し劣勢または優勢であるため，適切な範囲よりも勝ちに近づくような手を低く評価する．

6.4 実戦の典型例

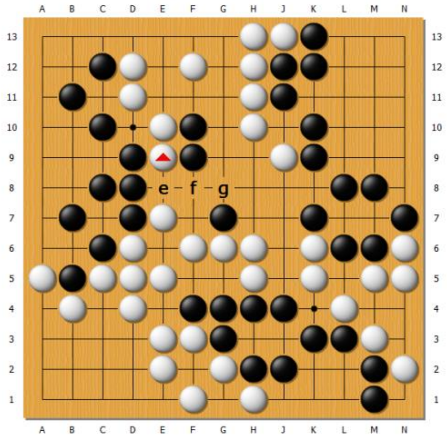

図 6.1：終盤の典型例

図 6.1 を用いて地合い差に基づいて着手を決める新手法の例を示す．白石は E9（赤い三角形）に打った盤面である．Leela zero はこの局面で，黒石の最も良い着手は G8(g) であり, 58.2%の勝率で少し優勢だと判断した. 一方, KataGo は黒石が F8（f）に打てば，83.6%の勝率で優勢だと判断した．

着手	勝率	訪問回数	距離選択確率
G8（g）	0.582	871	0.0095
F8（f）	0.572	1188	0.0445
L5	0.384	133	0.0043
E8（e）	0.352	3429	1.1176

表 6.1：$\mathrm{Leela}_{\mathrm{ABC25}}$ の探索リスト

$\mathrm{Leela}_{\mathrm{ABC25}}$の探索リストを表 6.1 で示す．Leela は G8（g）の着手が最高の勝率を持っていると判断した．最も自然な着手（最高の選択確率）は E8（e）であるが，勝率差が大きすぎるため，$\mathrm{Leela}_{\mathrm{ABC25}}$ではこの着手を選ばない．

着手	勝率	地合い差	選択確率	評価値
E8（e）	0.235	-1.234	0.4755	0.0699
G8（g）	0.516	+0.527	0.1410	0.0061
J8	0.538	+0.605	0.1307	0.0053
F8（f）	0.836	+1.862	0.1094	0.0018

表 6.2：KataGo の探索リスト

KataGo の探索リストは表 6.2 で示す．地合い差を見ると，E8 がこの中では最も損な手であり，F8 に比べ 3 目ほど損することが分かる．一方，それによって E8 が最も適切な範囲（[-10,-4]）に近づける手になっており，選択確率の高さ

もあいまって評価値 s_i は他の手の 10 倍ほどの値となっている．従って，KataGo+では E8 が着手される．この手は少なくとも初中級者にとっては不自然な手ではない．

6.5 実験

地合い差に基づく手法を評価するため実験を行った．まず，終盤の状態の盤面を用意し，その盤面から$\mathrm{Leela}_{\mathrm{ABC25}}$対 Ray と KataGo+対 Ray の対局を行った．その対局の結果により，形勢の制御と着手の自然さを評価した．$\mathrm{Leela}_{\mathrm{ABC25}}$では第 5 章で用いたネットワークを用い，探索回数は各着手 6000 回とした．KataGo+では GitHub でのネットワーク[25]を用い，探索回数は各着手 6000 回とした．Ray の探索回数は各着手 6000 回とした．

実際，どのような盤面の状態を終盤と呼ぶかは面白いトピックである．我々は以下の条件を満たしている棋譜を収集した：(1) 最大の地合い差がある閾値以下．本実験では 13 路で最大の地合い差が 5 以下と 19 路で最大の地合い差が 12 以下と設定した．(2) 地合い差の予測の標準偏差が 10 以下と設定した．(3) 着手をパスした場合，盤面の地合い差の変化値がある閾値以下．本実験では 13 路で変化値が 7 以下と 19 路で変化値が 5 以下と設定した．

(1) は大きな地合い差を出ないようにするための条件である．地合い差が大きければ，どんな方法でも自然に負けてあげられないため必要だと考えた．(2) の条件を満たしていた盤面は平和的な盤面だと考えられる．(3) の条件を満たした盤面はパスをしても大きな差が出ない，または現状手抜きができないような激しい戦いが起きていないため，死活問題などの心配もなく，大きな価値の着手もないと考えられる．

13 路盤と 19 路盤の終盤例は図 6.2, 図 6.3 で示す．両方も平和な盤面であり，終盤の段階に入っている．

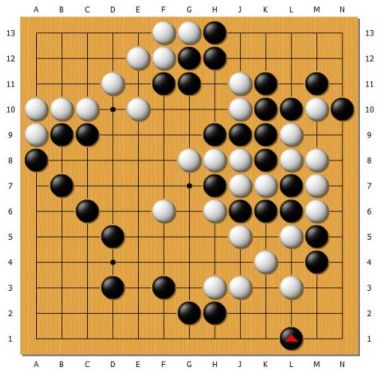

図 6.2：13 路盤の終盤例

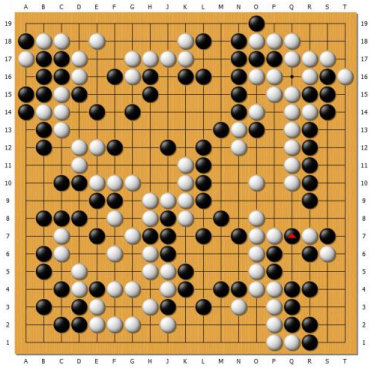

図 6.3：19 路盤の終盤例

我々は 13 路盤の終盤状況の盤面を 10 局収集し, 各盤面から$\mathrm{Leela}_{\mathrm{ABC25}}$対 Ray と KataGo+対 Ray の組み合わせで各 10 局対戦を行った（合計 100 局）．結果を表 6.3 に示す．ここで，p_{Leela}は Leela Zero のネットワークから出力した選択確率，p_{KataGo}は KataGo のネットワークから出力した選択確率を表す.

プログラム	$\mathrm{Leela}_{\mathrm{ABC25}}$	KataGo+
対 Ray の勝利数	63 勝（63%）	37 勝（37%）
p_{Leela}の算数平均	0.3551	0.3553
p_{KataGo}の算数平均	0.3742	0.4565
p_{Leela}の幾何平均	0.2398	0.2473
p_{KataGo}の幾何平均	0.1538	0.3008
p_{Leela}<0.05 の着手数	223（9.70%）	164（9.34%）
p_{KataGo}<0.05 の着手数	552（24.15%）	154（8.50%）

表 6.3：実験結果

『対 Ray の勝利数』より，KataGo＋は$\mathrm{Leela}_{\mathrm{ABC25}}$に比べ上手く負けていると考えられる．すなわち，指導碁の視点から見ると，KataGo＋が優れていると考えられる．特に，13 路盤ではベースとなっているコンピュータ囲碁プログラムの KataGo の強さが我々の訓練した Leela Zero より明らかに強い（我々は簡単な実験で, KataGo と Leela Zero を対戦させた. ハンディキャップなしの場合, KataGo が 20 戦全勝であり, 2 石のハンディキャップ+7.5 のコミの対局の場合, KataGo が 20 戦 14 勝であった）ため，この結果は新手法が形勢の制御の部分で効果があることを示している.

同時に, 我々は二つの方法の自然さも比較した. 着手の自然さは選択確率で評価される．二つのコンピュータ囲碁プログラムを比較すると KataGo+は$\mathrm{Leela}_{\mathrm{ABC25}}$より自然さが高いことが確認できる. また, 我々は選択確率が 0.05 以下の着手数をカウントした. 不自然な着手は回避すべきことであるが, 形勢の制御のために必要な場合もある．選択確率が 0.05 以下の着手数を二つのコンピュータ囲碁プログラムの間で比較すると KataGo+の着手数は$\mathrm{Leela}_{\mathrm{ABC25}}$より少ないことが確認できる.

また，19 路の終盤の盤面を 3 局収集し，KataGo+対 Ray を各 10 局対戦させた（合計 30 局）．$\mathrm{Leela}_{\mathrm{ABC25}}$は 19 路でのパラメータの調整をしていないため用いていない. その代わりに, γ=3 と γ=5 の設定における結果の違いを比較した. γ というパラメータは, 形勢の制御のために, どれだけ選択確率を犠牲することを示すパラメータである．γ=3 のとき，KataGo+は 30 局中 29 局勝利してしまった．γ=5 のとき，KataGo+は 30 局中 17 局勝利した．また，着手の自然さを表す指標として選択確率が 0.05 以下の着手の数もカウントした．γ=3 のとき

選択確率が 0.05 以下の着手は 66 手，γ=5 のとき選択確率が 0.05 以下の着手は 117 手であった．この結果は形勢の制御と着手の自然さがトレードオフであることを示している．また，γ の値は相手のレベルによって調整する必要があるため，如何に調整するのか，着手の自然さにどれほど影響するのか，ということは一つの重要な将来の課題である．

第7章 多様な戦略の演出

本章では，本研究のもう一つの課題である『多様な戦略の演出』について紹介する．囲碁では，通常いくつかの戦略（棋風）が存在する．例として，(1) 中央派/実利派 (2) 好戦派/平和派 (3) 楽観派/悲観派などが挙げられる．これらは極端になりすぎれば勝ちにくくなるが，それでもプロ棋士を含め多くのプレイヤになんらかの棋風があり，対戦または観戦の際に楽しみの一つとなっている．したがって，囲碁プログラムにこれら棋風を再現させることには価値がある．

(1) 中央派/実利派は，通常序盤のときに用いられる戦略である．図 7.1 で示したように，対局が始まった段階で，黒石が早々中央に向けて，領域を築くことを狙っている．一方，白石は三つの隅の領域を取って，実利を手に入れた．

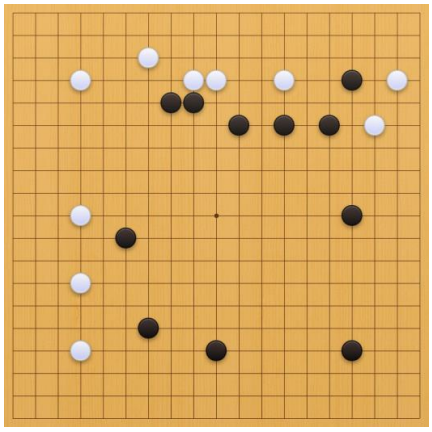

図 7.1：中央派/実利派の例

(2) 好戦派/平和派は，対局の全段階で用いられる戦略である．好戦派は常に戦闘を追求し，戦闘で利益を得ることを狙っている．平和派は，無用な戦いを避け，堅実な着手で自分の領域を守る戦略である．

(3) 楽観派/悲観派も，対局の全段階で用いられる戦略である．囲碁は両方の領域の大きさを比較し，大きな領域を手に入れた者が勝者である．そのため中級者以上のプレイヤは常に盤面上の領域を計算している．楽観派のプレイヤは，両方の領域を計算するとき，領域の境界線が曖昧なところには，自分の領域を本当の状況より多く計算し，優勢だと思いながら安全な着手やぬるい着手を打つ傾向がある．逆に悲観派のプレイヤは，本当の領域より少なく計算し，劣勢だと思いながら攻撃的な着手やリスクが高い着手をする傾向がある．

本研究は，(1) 中央派/実利派を対象として，前章の形勢の制御の手法をベースとした新手法を提案する．また，(2) 好戦派/平和派を対象とした手法については今後の課題として第 8 章に軽く説明する．

7.1 問題点

従来の方法では，モンテカルロ木探索を行うとき，葉ノードを評価するためにゲームの終局までランダムにシミュレーションし，囲碁のルールにしたがって勝敗を判断する．ここで勝ちと負けの計算ルールを変更することによって，伝統的なモンテカルロコンピュータ囲碁プログラムで多様な戦略の演出を実現した[3][4]．しかし，深層学習コンピュータ囲碁プログラムでは，バリューネットワークの出力を用いて葉ノードを評価する．すなわち，終局までランダムにシミュレーションを行う必要がなくなり，そのため，従来の多様な戦略の演出の手法を使えない．

7.2 提案手法と概念

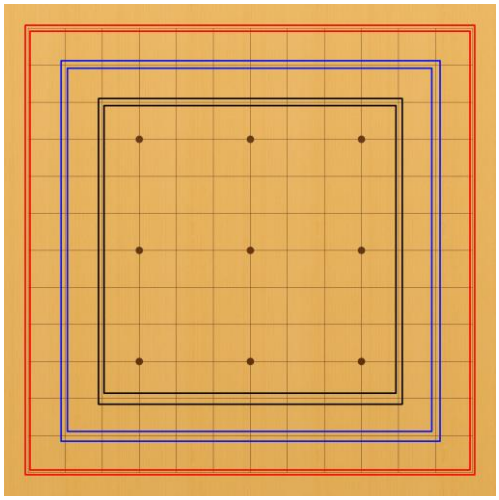

図 7.2：13 路盤の碁盤

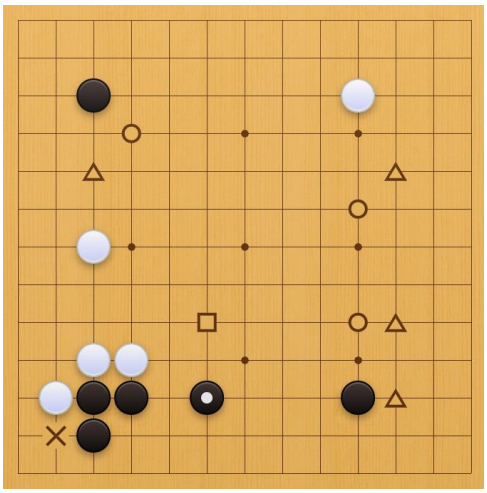

図 7.3：戦略の例

本手法の目的は，深層学習コンピュータ囲碁プログラムでも動くようにすることである．ポリシーネットワークから得た着手の選択確率に盤面上の位置によって重みづけを行うことで，選択確率を上下させて，打たれやすさを制御することによって，中央派と実利派の戦略を実現することである．

まず，囲碁の用語の説明のために，碁盤（13 路盤）を図 7.2 に示す．囲碁では，碁盤の一番外側をまわっている部分（赤い線）を『一線』，その内側をまわっている部分（青い線）を『二線』，次の内側をまわっている部分（黒い線）を『三線』と呼び，以下『四線』『五線』なども同様に定義される．

戦略の例を図 7.2 に示す．今は白石の手番であり，候補手をマークした．人間知識を用いると，丸と四角形のマークは中央派の着手と判断でき，三角形と×の

マークは実利派の着手と判断できる．この例を見ると，中央派の着手が四線と四線以上になり，実利派の着手が三線と三線以下になることが確認できる．そのため，我々は石の位置によって重み付ける考え方から，新手法を提案する．

提案した重み付ける手法を以下の式（7.1）に示す．

$$p_i'' = p_i' \cdot w_{l_i} \tag{7.1}$$

p_i' は 5.2.2 節で述べた相手の手との距離によって補正された選択確率であり，不自然な着手を減るため用いた．l_i は着手が盤面上の何線にあるかを示す値である．w_{l_i}は戦略と位置にベースするパラメータである．

このように補正したことによって，戦略と合う着手の選択確率を大きく補正する．そして，形勢の制御の手法にしたがって，勝率と選択確率のバランスがよい着手を選択することで，打たれやすくなる．

7.3 実験

戦略の演出の手法を評価するために実験を行った．7.3.1 節では実験設定と強さの検証を示す. 7.3.2 節では被験者実験で人間プレイヤの評価と良い例を示す.

7.3.1 実験設定

我々は，13 路盤と 19 路盤の両方で実験を行った．13 路盤と 19 路盤のどちらにおいても$\mathrm{Leela}_{\mathrm{ABC25}}$をベースに実装し，探索回数は各着手 6000 回とした．13 路では第 5 章で用いたネットワークを用いる. 19 路盤では人間プレイヤの棋譜から訓練したネットワークを用いる[26]．実験に用いるw_{l_i}の設定は表 7.1 を示す. このパラメータは予備実験によって適切と思われるものを選んだ. また，同じネットワークを用いているオリジナル Leela を相手役として用意した，以下標準 Leela で示す.

	13 路中央派	13 路実利派	19 路中央派	19 路実利派
$l_i \leq 2$	w_{l_i}=0.50	w_{l_i}=2.00	w_{l_i}=0.25	w_{l_i}=2.00
l_i=3	w_{l_i}=0.50	w_{l_i}=2.00	w_{l_i}=0.50	w_{l_i}=1.50
l_i=4	w_{l_i}=2.00	w_{l_i}=0.50	w_{l_i}=1.50	w_{l_i}=0.75
l_i=5	w_{l_i}=2.00	w_{l_i}=0.50	w_{l_i}=1.75	w_{l_i}=0.50
l_i>5	w_{l_i}=2.00	w_{l_i}=0.50	w_{l_i}=2.00	w_{l_i}=0.25

表 7.1：実験のパラメータ設定

7.3.2 強さの変化に関する評価

一般に，このような確率補正の工夫を行うことで強さや手加減の上手さが損なわれる可能性はある．そこで，多様な戦略の演出の手法の強さを評価するため，実利派対 Ray と中央派対 Ray の対局を 100 局回行った．実利派対 Ray の結果は，Ray が 50 局中 23 勝であり，中央派対 Ray の結果は，Ray が 50 局中 22 勝であった．これは 5.3.1 章の結果『Ray は 500 局中 238 勝であった』とあまり差がないため，提案手法で生成した戦略は強さに害することがないと示された．

7.3.3 数値実験

まず数値から，戦略の演出ができるかどうかを確認するため，13 路盤の棋譜 60 枚(対局最初の 40 手)と 19 路盤の棋譜 60 枚（対局最初の 60 手）を用意した．60 枚の棋譜の内訳は，実利派対中央派の棋譜が 20 枚，実利派対標準 Leela の棋譜が 20 枚，中央派対標準 Leela の棋譜が 20 枚である．これらの棋譜を用いて，各戦略が中央（四線と四線以上）に打つ着手数の平均を計算した．その結果を表 7.2 に示す．

	中央派	標準 Leela	実利派
13 路盤	10.80±0.72	8.83±0.96	3.88±0.75
19 路盤	18.68±1.01	13.43±1.24	11.33±1.09

表 7.2：中央に（四線と四線以上）着手する平均着手数（95%信頼区間）

この表から，中央派は中央に着手する数が多く，実利派は中央に着手する数が少ない．標準 Leela の中央に着手する数は中央派と実利派のおよそ中間である．この結果は各設定も戦略にしたがって着手をすることが証明できた．そして 13 路盤の実利派以外では，各戦略の着手数の間の差が有意である（$p<0.015$）．

7.3.4 被験者実験

続いて，人間プレイヤは本手法の戦略を認識できるかどうかを検証するため，被験者実験を行った．実験は，10 人の被験者を用いて行った．被験者の棋力はアマチュア 1 級からアマチュア 6 段まで様々である．実験では 13 路盤の棋譜 15 枚（対局最初の 40 手）と 19 路盤の棋譜 15 枚（対局最初の 60 手）を用意した．15 枚の棋譜の内訳は，実利派対中央派の棋譜が 5 枚，実利派対標準 Leela の棋譜が 5 枚，中央派対標準 Leela の棋譜が 5 枚である．一人の被験者は，各設定（3 通り）と各サイズ（2 通り）の組み合わせを 1 枚ずつ，計 6 枚を評価す

る．そして各試合について，黒番・白番のそれぞれに，−2（中央派に見える）から +2（実利派に見える）までの 5 段階評価を行ってもらった．その結果を以下の表 7.3 に示す．

	中央派	標準 Leela	実利派
13 路盤	-0.50	-0.10	＋0.70
19 路盤	-1.25	+0.65	＋0.85

表 7.3：被験者実験の結果

この表から，中央派の棋譜は負の点数つまり中央派と認識され，実利派の棋譜は正の点数つまり実利派と認識された．どちらの戦略も良く認識されていることが示された．ただし，19 路盤の標準 Leela の評価が 19 路盤の実利派に近いことから，19 路盤の標準 Leela は元々実利派に見えると考えられる．また，19 路盤の結果が 13 路盤より優れることも示された．序盤は戦略が判断しやすく，19 路盤の序盤が 13 路盤より長いことが原因であると考えられる．

図 7.3 に 19 路盤で実利派対中央派の例を示す．この棋譜から，提案手法が戦略に沿って着手をすることが確認できる．根拠として，白の 10 手目と 14 手目は中央の領域を目的とする典型的な着手であり，黒の 11 手目，21 手目，31 手目は隅・辺の実利を目的とする着手であることが挙げられる．また，白の 18 手目，30 手目，32 手目と 38 手目も中央派のプレイヤの典型的な着手であるなどが挙げられる．

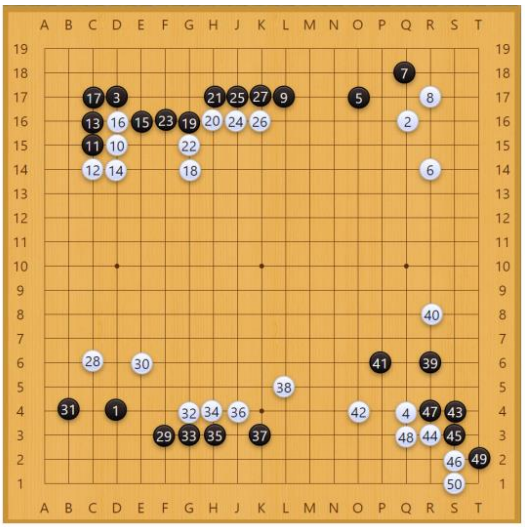

図 7.4：19 路盤の実利派（黒）対中央派（白）の例

第8章 終わりに

本研究では『深層学習コンピュータ囲碁プログラムを用いることで形勢の制御と多様な戦略の実現がどのように困難になるのか』『その困難をどうすれば解決できるか』を明らかにすべく，深層学習コンピュータ囲碁プログラムと伝統的なモンテカルロコンピュータ囲碁プログラムの違いを比較し，その違いによって深層学習コンピュータ囲碁プログラムに従来の手法を用いるとどのような問題点があるかを発見し，問題点に対する解決策を提案した．

まず，形勢の制御の部分では 5.1 節で三つの手法を紹介した．紹介した手法を深層学習コンピュータ囲碁プログラム Leela Zero に実装し，相対的に弱い伝統的なモンテカルロ囲碁プログラム Ray と互角な対局が可能であることを示した．そして被験者実験で，紹介した手法は自然さを考えない単純な方法より不自然な着手を抑えることを示した．

さらに，終盤では勝率が極端な値になってしまって従来の手加減手法がうまく動作しない課題に対し，地合い差を用いる新しい手法を導入し，従来手法に比べて終盤の逆転勝ちを減らしてかつ自然さが向上できることを実験により示した．

多様な戦略の演出の部分では，石の位置によって，着手の選択確率に重みづける手法を提案した．被験者実験で人間プレイヤが提案手法の戦略を認識できることを示した．

今後の展望として，まずは，アマチュアまたは指導者の着手と似た着手を打てるため，パラメータを更に調整する必要がある．特に本手法は手抜きの問題を解決するため，直前の着手との距離による選択確率を補正するが，補正後の $\mathrm{Leela}_{\mathrm{ABC25}}$は距離 6 以上になるとアマチュアよりもかなり低い頻度でしか着手していない．また，第 6 章で提案した地合い差に基づく形勢の制御の手法は，今の段階では終盤の部分のみを用いている，将来は従来の形勢の制御の手法と組み合わせ，実用化を目指している．最後は中央派と実利派以外の戦略の実現を目指す．例として，7 章冒頭部分で述べた好戦派と平和派，楽観派と悲観派などが挙げられる．特に，好戦派と平和派では，6.2 節で説明した盤面上の交差点が黒や白に属する確率である ownership を用いることで実現が可能だと考えられる．例として，相手の石に属する確率が大きい場所に着手すると攻撃的な着手だと認識する可能性が高いことや，自分の石に属する確率が大きい場所に着手すると平和・防御的な着手だと認識する可能性が高いことなどが挙げられる．

参考文献

[1] Silver, David, et al. ”Mastering the game of Go with deep neural networks and tree search.” nature 529.7587 (2016): 484-489.

[2] Silver D, Schrittwieser J, Simonyan K, et al. Mastering the game of go without human knowledge[J]. nature, 2017, 550(7676): 354-359.

[3] 池田 心, Viennot Simon, モンテカルロ碁における多様な戦略の演出と形勢の制御: 接待碁 AI に向けて, 情報処理学会, ゲームプログラミングワークショップ 2012 論文集, Vol.2012, No.6, pp. 47-54, 2012-11-09.

[4] Kokolo Ikeda,Simon Viennot,Production of Various Strategies and Position Control for Monte-Carlo Go - Entertaining human players,IEEE Conference on Computational Intelligence and Games(CIG2012), pp. 71-78, 2012-09.

[5] Shi Yuan，Fan Tianwen，Li Wanxiang，池田 心，深層学習囲碁プログラムを用いた場合の手加減に関する研究，情報処理学会 第 41 回ゲーム情報学(GI)研究発表会，2019-3.

[6] Tianwen Fan, Yuan Shi, Wanxiang Li and Kokolo Ikeda, Position Control and Production of Various Strategies for Deep Learning Go Programs, 2019 International Conference on Technologies and Applications of Artificial Intelligence (TAAI 2019), 21st November 2019.

[7] Yuan Shi,Tianwen Fan,Wanxiang Li,Chu-Hsuan Hsueh and Kokolo Ikeda,Position Control and Production of Various Strategies for Game of Go Using Deep Learning Methods,Journal of Information Science and Engineering, accepted.

[8] 囲碁[https://ja.wikipedia.org/wiki/囲碁]. （アクセス:2021/01/04）.

[9] コンピュータ囲碁[https://ja.wikipedia.org/wiki/コンピュータ囲碁].（アクセス: 2021/01/04）.

[10] Brügmann, Bernd. Monte carlo go. Vol. 44. Syracuse, NY: Technical report, Physics Department, Syracuse University, 1993.

[11] Kocsis, Levente, and Csaba Szepesvári. "Bandit based monte-carlo planning." European conference on machine learning. Springer, Berlin,

Heidelberg, 2006.

[12] モンテカルロ木探索[https://ja.wikipedia.org/wiki/モンテカルロ木探索]. （アクセス:2021/01/04）.

[13] Nomitan[http://www.jaist.ac.jp/is/labs/ikeda-lab/rs_nomitan.html] （アクセス:2021/01/04）.

[14] Ray[http://computer-go-ray.com] （アクセス:2021/01/04）.

[15] Leela Zero[https://github.com/leela-zero/leela-zero] （アクセス:2021/01/04）.

[16] Wu, David J. "Accelerating self-play learning in Go." arXiv preprint arXiv:1902.10565 (2019).

[17] Yen, S.J., Chen, Y.L., Lin, H.I.: Scaffolding learning for the Novice Players of Go. In: 2019 International Conference of Innovative Technologies and Learning(ICITL 2019), LNCS 11937 (2019)

[18] I.-C. Wu, T.-R. Wu, A.-J. Liu, H. Guei, and T. Wei, “On strength adjustment for mcts-based programs,” in Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 1222–1229.

[19] 仲道隆史，伊藤毅志：人を楽しませる接待将棋システム、2014 年度人工知能学会全国大会,1E5-OS-23b-5in,(2014).

[20] 仲道隆史，伊藤毅志：将棋 AI における棋力の調整が不自然さを与える影響、ゲームプログラミングワークショップ 2014, P-16 (2014).

[21] Auer, Peter, Nicolo Cesa-Bianchi, and Paul Fischer. "Finite-time analysis of the multiarmed bandit problem." Machine learning 47.2-3 (2002): 235-256.

[22] Kokolo Ikeda, Simon Viennot and Takanari Shishido, Machine-Learning of Shape Names for the Game of Go, 14th International Conference Advances in Computer Games (ACG2015), pp.247-259, 2015-07.

[23] Kokolo Ikeda, Simon Viennot and Naoyuki Sato, Detection and Labeling of Bad Moves for Coaching Go, IEEE Conference on Computational Intelligence and Games (CIG2016), pp.395-401, 2016-09.

[24] R. Coulom, “Efficient selectivity and backup operators in monte-carlo tree search,” in International conference on computers and games. Springer, 2006, pp. 72–83

[25] “KataGo のネットワーク” https://d3dndmfyhecmj0.cloudfront.net/g104/models/b20c256-

s447913472-d241840887.zip,（アクセス:2021/01/26）

[26] “Leela Zero のネットワーク” https://sjeng.org/zero/best_v1.txt.zip,（アクセス:2020/12/21）

[27] “コミ 6 目半の対局手番別勝率 ― 上半期集計 <1> ― ” http://archive.nihonkiin.or.jp/match/2007/07/6_2.html（アクセス:2021/01/05）