合議制囲碁プログラムのための多様な知識の集団学習

野 中 翔 平[†1] 中 村 貞 吾[†1]

複数のプレイヤにより多数決を行う合議は，囲碁でも有効であることが示されている．合議は，単一のプレイヤにたいして乱数を加えることで容易に作成することができる．しかし，合議では個々人の能力よりも集団としての多様性が重要であるといわれており，乱数のみでは多様性が少ない．本研究では，合議を行う他のプレイヤの知識を考慮した集団学習を行うことにより，明示的に多様性を持たせた知識を作成し，合議に用いた．更に，学習に用いた棋譜群を棋風により分類することで知識に方向性を持たせた．結果として，無作為な学習標本を元にした集団学習では明示的に多様性を持たせることで勝率を向上させた．また，棋風により分類したプレイヤを用いることで合議による勝率を向上させた．

Ensemble Learning of Diverse Knowledge for Consultation Program of Go

SHOHEI NONAKA[†1] and TEIGO NAKAMURA[†1]

It is effective to make consultation with a number of algorithms to play Go. We can easily construct each algorithm which uses a base evaluation function plus some small random values, but these algorithms have lack of variety. Because the variety of group is more important than the ability of individuals in consultation, we propose a method for increasing the variation of group of algorithms utilizing knowledge based on ensemble learning. Furthermore, we classified learning records of a game of go by style of playing go. We present some experimental results that show our method improves the winning rate.

1. はじめに

将棋プログラムの Bonanza にたいして乱数を用いることで擬似的に複数のプログラムを用意し，それらの多数決により手を決定する文殊が合議プログラムの有効性を示した．この後、合議に対する研究が盛んになり，囲碁にたいしても合議が有効であることが確認されている．

文殊のように乱数による単体プログラムの擬似的な複数化は実装が容易であり，構造のシンプルさといった利点がある．しかし，乱数のみによるプログラムの複数化では個々のプレイヤが似た性質を持つ可能性がある．合議では個々の能力よりも集団としての多様性が重要だといわれている．そこで，将棋プログラムの評価関数を学習する際に，他のプレイヤの学習結果を考慮した集団学習を行うことで多様性を増した合議の有用性が示されている[1)]．

本稿では，UCT を用いたモンテカルロ法による囲碁プログラムにたいし，単純多数決による合議を導入した．モンテカルロ法は乱数を用いた探索法のため，探索を複数回行った結果を単純多数決をさせることで，容易に合議を行うことができる．しかし，先に述べた通り乱数のみによる合議では個々のプレイヤが似た性質を持ち，多様性が少ないという問題がある．そのため，プロの対局による棋譜データを学習棋譜として用意し，集団学習である Bagging 法を用いて複数の学習標本を作成し，個別に学習を行うことで知識を作成した．個々のプレイヤが知識を用いることで多様性を増した．

さらに，学習標本にたいし共有棋譜として全てのプレイヤが共通して持つ棋譜群を用意した．この共有棋譜の割合を変化させ知識を変化させることにより，多様性を変化させた．また，Bagging 法にたいして NCL 法を適用することで，既に学習した他のプレイヤの知識を考慮するように学習を調整した．より，明示的に多様性を持つように知識を調整することで合議制囲碁プログラムの強化を行った．

学習時の調整による多様性だけでなく，知識にたいして方向性を持たせることを目的とし，学習に用いる棋譜群を対局者の情報から棋風により分類を行った．分類された棋風ごとに作成されたプレイヤを混ぜ，合議を行った．

†1 九州工業大学大学院情報工学府
Kyushu Institute of Technology Graduate School of Computer Science and Systems Engineering

2. 関連研究

2.1 モンテカルロ囲碁

囲碁では，駒が黒と白の 2 種類のみであり，打たれた手が後々良い手か悪い手か判断されるなど，途中経過の盤面に対する評価が非常に難しい．そのため，良い評価関数を作成することは非常に難しいといわれている．しかし，囲碁には局面において最善手以外にも良い手が多く，単純なシミュレーションでも容易に終局状態まで到達することができるといった性質がある．これは，乱数を用いてシミュレーションを行うモンテカルロ法と相性がよく，モンテカルロ法を用いたモンテカルロ碁が大きな成果をあげている[2)].

モンテカルロ碁では，作成することが難しいとされている評価関数を，ランダムなシミュレーションによって終局まで打ち終え，最終的な勝敗を取得するプレイアウトを複数回行うことで作成した局面評価を用いている. 単純なモンテカルロ碁では何度もプレイアウトを行い，一番勝率の高い手を選択する．プレイアウト回数が多くなるほど，モンテカルロ法の性質からその勝率の信頼性は向上する．しかし，その分計算時間も必要となる．限られたプレイアウト数では可能な合法手全てにたいして均等にプレイアウトを行うよりも，できるだけ良さそうな手にたいして多くのプレイアウトを行う必要がある．

どの手にプレイアウトを行うかという問題にたいして，現在のモンテカルロ碁では主に UCB applied to Trees(UCT) 探索が用いられている[3)].

2.2 UCT

UCT 探索とは，K 腕バンディッド問題で有効な手法である UCB1 アルゴリズムを木構造に拡張したものである．

UCB1 とは K 腕バンディッド問題から考えられたアルゴリズムである．K 腕バンディッド問題とは，腕が K 本あり，それぞれの腕が独自の確率を持つスロットマシンをプレイする問題である．プレイヤは各腕に対する知識を持っていない状況から，プレイを繰り返しながら各腕に対する知識を増やし，できるだけ当たる確率の高い腕をプレイすることで，報酬の和を最大にすることを目的としている．K 腕バンディッド問題は探索と収穫のジレンマが問題であり，UCB1 アルゴリズムは腕に関する知識を増やす探索と，知識から最善の腕を選ぶ収穫を上手く選択することで，報酬の和が最大になるように腕を選ぶアルゴリズムである．

UCT によって探索であるプレイアウトの数と収穫である勝率のバランスを取る．そうすることで，有効そうな手にたいしてより多くのプレイアウト数を割り当てることができる．

主にモンテカルロ碁は，UCT による木探索部分とプレイアウトによるランダムシミュレート部分で構成されている．

2.3 学習手法

2.3.1 Elo レーティング

モンテカルロ囲碁にたいし，知識を導入するとプレイアウトの質を上げることができることが確認されている．用いられる知識には様々な種類と手法があるが, 8 斤傍の石の配置を特徴とし，その特徴の出現頻度を棋譜データから学習し，探索局面にたいし合法手の確率分布を与えることで，探索時に好手が選ばれる確率をあげる手法が知識作成の容易さと実行速度の面で有効であり，広く使われている[4)].

棋譜からの教師あり学習として，チーム対チームの試合結果に関する予測を行う Bradley-Terry モデルを用いて，学習棋譜中の実際に打たれた手を勝者，他の盤面に現れた合法手を敗者の試合と考えることで，Elo レーティングの一般化を適用した囲碁の教師あり学習がある[5)]．特徴の集合をチームと考え，個々の特徴の Elo レーティングを計算することで，探索局面にたいして合法手の確率分布を持たせることができる．この手法の優れた点として，チームとして考えることで特徴の個数に対するコストが指数的に増えず，複数の特徴を組み合わせることができる．また，従来の特徴の出現率を用いた手法と違い，同時に出現している他の特徴も考慮している．そのため，より正確な評価を行うことができる．

2.3.2 Bradley-Terry モデル

Bradley-Terry モデルは，過去の試合結果から参加者一人一人の強さを推定し，将来の試合結果の確率分布を与える．

Bradkey-Terry モデルのチーム戦における一般化の式は（1）のように表される．

$$P(\text{1-2-3 が勝つ}) = \frac{\gamma_1\gamma_2\gamma_3}{\gamma_1\gamma_2\gamma_3 + \gamma_4\gamma_2 + \gamma_1\gamma_5\gamma_6\gamma_7} \quad (1)$$

この時，各個人 i の強さを正の数値 γ_i で表し，i が強いほど γ_i は大きくなる．強さの推定は γ をパラメタのベクトル，R を過去の結果としたとき，ベイズの公式より（2）で得られる．

$$P(\gamma \mid R) = \frac{P(R \mid \gamma)P(\gamma)}{P(R)} \quad (2)$$

パラメタ γ は $P(\gamma \mid R)$ が最大に成る γ^* を見つけることで推定される．仮想的な結果 R' を事前分布 $P(\gamma) = P(R' \mid \gamma)$ とすることで最適化すると，γ の推

定は $P(R,R' \mid \gamma)$ の最大化となる．$\gamma_1,\ldots,\gamma_n$ を n 人の強さパラメタとし，各個人間の独立な N 回の試合結果 $R_1,\ldots,R_N$ は既知であり，試合結果の確率は（3）で書くことができる．

$$P(R_i) = \frac{A_{ij}\gamma_i + B_{ij}}{C_{ij}\gamma_i + D_{ij}} \tag{3}$$

ここで，A_{ij},B_{ij},C_{ij}, そして D_{ij} は γ_i と独立な係数である．この式より各 $P(R_i)$ をある特定の γ_i の関数として，異なる n 通りの方法で書くことができる．E_j は $E_j = C_{ij}\gamma_i + D_{ij}$ と定義され，最大化する目的関数は（4）となる．

$$L = \prod_{j=1}^{N} P(R_j) \tag{4}$$

L を最大化する反復アルゴリズムとして少数化-最大化を用いる．γ の初期推定値 γ^0 から初め，γ^0 で L を少数化する関数 $m(x)$ を作る．その後 $m(x)$ が最大値を取る γ^1 を求めると，少数化の性質により γ^1 は γ^0 から改善されている．L を γ_i の関数と考え，対数を取り γi を含まない稿を取り除く．また，$B_{ij} = 0$ または $A_{ij} = 0$ なので，最大化すべき関数は（5）に成る．

$$f(x) = W_i \log x - \sum_{j=1}^{N} \log(C_{ij}\gamma_i + D_{ij}) \tag{5}$$

右辺の対数は $x = \gamma_i$ における接戦で少数化され，x を含まない項を取り除くと，最大化する少数化関数は（6）となる．

$$m(x) = W_i \log x - \sum_{j=1}^{N} \frac{C_{ij}x}{E_j} \tag{6}$$

$m(x)$ の最大値は，x について微分し，0 と置いて解いた式（7）で得られる．

$$x = \frac{W_i}{\sum_{j=1}^{N} \frac{C_{ij}}{E_j}} \tag{7}$$

L を最大化するようにパラメタ γ_i を繰り返し更新することで，より精度がます．

2.4 合　　議

2.4.1 モンテカルロ木探索の並列化

モンテカルロ木探索の並列化の代表的手法として，Root 並列化がある[6]．Root 並列化は，探索開始と終了時にのみ情報を共有するため，他の手法に比べオーバーヘッドが少なく有効的な手法である．

Root 並列化では，総和制と合議制がある．総和制による Root 並列化は，探索木の訪問回数の集計を行い，最も多い着手を選択する．合議制では，各プロセスの自身が最善と判断する着手の集計を行い，最も多くのプロセスに選ばれた着手を選択する．

2.4.2 乱 数 合 議

乱数合議とは，複数のプレイヤの判断を統合する合議手法の 1 つであり，単一の思考プログラムを用いて擬似的に複数のプレイヤを作成したい際に有効な実現法である．

複数のプレイヤの生成方法は，個々のプレイヤごとに異なる乱数系列を与えることで，探索局面に対する手の探索を変化させる．こうすることで，各プレイヤのゲーム木探索を異なるものにすることで，各プレイヤが出す結果を異なるものにする．このようにして生成された複数のプレイヤの最終的な判断を多数決や一番評価値が高いものを選ぶ事で決定する．

囲碁プログラムにたいして乱数合議を適用し，単一の囲碁プログラムと対戦させた結果，有意に勝ち越すことが確認されている．

2.4.3 多　様　性

多様性とは，集団の中でメンバーが持つ，問題解決に用いる観点や，問題解決手法の総合的な豊富さである．

集団に属するメンバーが全員同じ観点と解決手法である一様な集団にたいし，多様な集団がより優れていることは明らかである．一様な集団では，一人の出す解が向上する可能性が 0 である．しかし，多様な集団では他のメンバーが更に良い解を出す可能性がある．解の向上が可能であるという特徴が，多様性が一様性に勝る理由である．

2.5 集 団 学 習

集団学習とは，機械学習手法の 1 つである．複数の判別機を異なる手段により学習することで生成し，複数の判別機による出力を統合することで，最終的な判断を行う．

複数の判別機を用いる利点として，複数の異なる意見を多数決で統合することで単一の判別機では誤ってしまうような場面でも，過半数以上の判別機が誤判別を行わないかぎり避けることができる．また，膨大な学習データでも各プレイヤに割り当て学習することが可能である．さらに，学習を変化させることで個々のプレイヤの欠点を補い合う判別機を用いることができる．

集団学習では異なる複数の判別機をどのように生成するか，生成された複数の判別機の出力をどのように統合し最終的な出力にするかを考慮する必要がある．

2.5.1 Bagging

Bagging は，Boostrap Aggregating の略であり，Boostrap 法により生成された複数の判別機の出力を多数決により統合するアルゴリズムである．

Boostrap 法とは，学習標本から重複ありの無作為な抽出を行う復元抽出により，新たに学習標本を作成する手法である．復元抽出により，元の学習標本から同じ物が複数回抽出される可能性がある一方で，1 度も選ばれないこともある．これにより，少し異なる複数の学習標本を生成する手法である．この生成された複数の学習標本を用いてそれぞれの判別機が学習を行い，その結果生成された複数の出力を多数決により統合する．単純な手法だが，この手法によって多くの場合，単一の判別機よりも高い精度が得られる．これは，学習結果の分散が減ることが挙げられる．この分散とは，判別機の学習に用いる標本の違いが生成される判別機にどの程度影響しているかを表す．

このことから，Bagging は不安定な学習プログラムに有効であるとされている．また，学習用データが少ない場合も，Boostrap 法によりデータサイズを維持したまま複数の学習標本を生成できるため、Bagging 法が有用である．

2.5.2 Negative Correlation Learning

Negative Correlation Learning（NCL）は，最急降下法による学習において，特別な目的関数を用いた学習を行うことにより生成される複数の判別機の多様性に注目した集団学習手法の 1 つである[7]．

この手法では，明示的に学習において判別機が多様になるように調整を行う．

学習の手順として，複数の判別機の出力を（8）のように表現する．

$$\hat{f}(k) = \frac{1}{M}\sum_{i=1}^{M} f_i(k) \tag{8}$$

ここで，M は判別機の数，$\hat{f}_i(k)$ は i 番目の判別機の k 番目の学習標本に対する出力を表す．次に，最急降下法で個々の判別機を学習する際に用いる損失関数を（9）のように定義し，最小化することを考える．

$$e_i(k) = \frac{1}{2}(\hat{f}(k) - c(k))^2 - \alpha p_i(k) \tag{9}$$

$$p_i(k) = (\hat{f}_i(k) - \hat{f(k)})^2 \tag{10}$$

ここで，$c(k)$ は k 番目の学習標本に対する教師例，$p_i(k)$ は判別機間の分散が小さいことに関するペナルティ，α はペナルティの影響度を表す係数である．α が大きくなるほど個々の判別機の精度は落ちるが，判別機の多様性は増すという関係になっている．

以上のように，教師例と近いか判別機間の分散が大きくなることを目指す目的関数を定義し，学習を行うことによって多様性に注目した集団学習を実現している．

3. 本研究の手法

3.1 Elo レーティングへの NCL の適用

Elo レーティングを利用した学習に NCL を適用することで，他のプレイヤが持つ学習結果と離れるように学習を行い，多様性を持たせる．

Elo レーティングにおいて最大化を目指す関数（6）にたいして他のプレイヤの学習との分散である第 2 項を付け加え，NCL を導入した（11）を作成した．

$$h(x_{in}) = \frac{m(x_n)}{N_n} + \frac{\alpha}{2(n-1)}\sum_{m=1}^{n-1}(x_{in} - x_{im})^2 \tag{11}$$

このとき，x_{in} は n 番目のプレイヤのパターン i の評価値，N_n は n 番目の学習棋譜群の局面数，α はプレイヤ間の学習の距離をどれほど重視するかの係数である．この式の最大値は，x について微分し 0 と置いて解いた式（13）となる．

$$c_{in} = \frac{1}{N_n}\sum_{j=1}^{N_n}\frac{C_{inj}}{E_{nj}} \tag{12}$$

$$x_{in} = \frac{c_{in} + \alpha\overline{x_{in}} - \sqrt{(c_{in} + \alpha\overline{x_{in}})^2 - 4\alpha\overline{w_{in}}}}{2\alpha} \tag{13}$$

ここで，C_{ij} は j 番目の局面中でのパターン i の評価値の合計，E_j は j 番目の局面に出現するパターン全ての評価値の合計である．さらに

$$\overline{x_{in}} = \frac{1}{n-1}\sum_{m=1}^{n-1}(x_{in} - x_{im}) \tag{14}$$

$$\overline{w_{in}} = \frac{W_{in}}{N_n} \tag{15}$$

とする．ここで，$\overline{x_{in}}$ は n 番目までのプレイヤのパターン i に対する学習の分散を表し，W_{in} は n 番目の学習標本中でパターン i が打たれた数であり，$\overline{w_{in}}$ がパターン i の使用率を表す．この時，NCL 項が大きすぎ根号中の値が負になり発散しないように，係数 α の値を充分に小さく設定する必要がある．

3.2 棋風による分類

学習に用いる棋譜群を，対局を行ったプロの情報を元に棋風によって分類を行った．

棋風は，実利派，好戦派，模様派と呼ばれる 3 種類に分類した．この棋風による分類を行った棋譜群は，未分類の棋譜群とは別の棋譜群を用いて行った．

また，棋風ごとに棋譜数が異なっており，**表 1** のよ

うになっている．

表 1　棋風ごとの棋譜数

実利派	好戦派	模様派
2,763	1,602	1,006

3.3　単純合議の囲碁プログラム

実験に用いる囲碁プログラムはオープンソースのLibEGoを使用した[8)]．LibEGoは，モンテカルロ法とUCT，8近傍の石の配置とアタリを特徴としたEloレーティングによる知識を用いている．この知識は，UCT探索とプレイアウトの両方において用いられている．また，実行速度が早くオープンソースのため研究に向いている．

LibEGoを元に10プレイヤによる多数決合議を作成した．それぞれのプレイヤはモンテカルロ法に使われる乱数系列，Eloレーティングにより学習された盤面に対する合法手の確率分布を与える知識を変化させることで，多様性を持たせている．

3.4　知識の作成

知識は，プロの対局による棋譜群を元に，Bagging法により作成した．学習は図1のようになる．

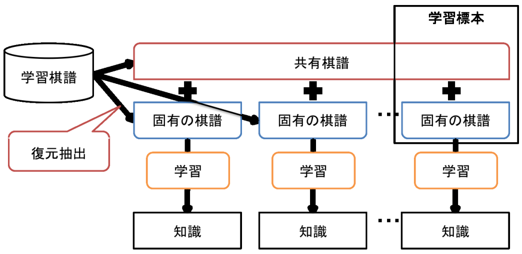

図 1　Bagging 法による学習

棋譜群は10,000棋譜を用意し，復元抽出を行うことで学習標本を作成した．学習標本は，全てのプレイヤが共通して持つ共有棋譜群と個々のプレイヤが個別に持つ棋譜群の2種類の学習標本を組み合わせ，作成している．共有棋譜数は，0，500，1000の3種類を用意し，これに個別に持つ棋譜群を加え1,000棋譜になるように復元抽出した．共有棋譜の割合が多くなるほど各プレイヤ間の知識の多様性が少なくなる．また，共有棋譜数が1,000の場合は全てのプレイヤは共有棋譜からの学習のみとなり，全てのプレイヤの知識が同一となるため，乱数系列のみが異なる合議となる．

さらに，NCLを適用したEloレーティングへ学習式を変更し，作成した学習棋譜にたいして知識の作成を行った．ここで，プレイヤ間の学習の距離をどれほど重視するかの係数 α は 10^{-14} とした．学習は図2のようになる．

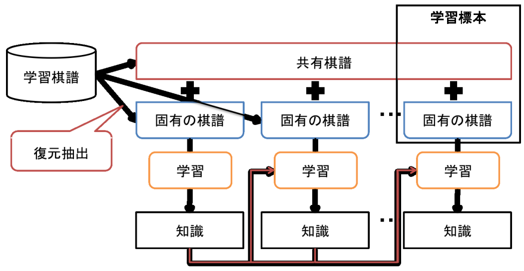

図 2　Bagging 法に NCL を適用した学習

さらに棋風で分類した学習棋譜については，共有棋譜数0でそれぞれの棋風ごとに復元抽出を行うことで学習標本を作成した．

4.　実　　験

4.1　実 験 方 法

各プレイヤはプレイアウト数を20,000回とした．対局は9 × 9の盤面で中国ルール，コミ6．5として行った．

4.2　実 験 結 果

4.2.1　知識の有用性

知識を用いることの有用性を確かめることを目的とし，知識の有無による実験結果を**表 2** に示す．

対局相手は，知識有りと全ての合法手に対して同一の確率分布を持つ知識無しに関して，単一プログラムと合議プログラムの2種類を行った．単一プログラムは1人対1人，合議プログラムは10人による合議を行っている．対局は白番50回，黒番50回の計100回を行った．

表 2　知識を適用した対戦結果

知識有りのプログラム	知識無しに対する勝率 (%)
単一プログラム	90
合議プログラム	82

表2の対戦結果は危険度1%における100戦の二項検定から有意に勝ち越していると言える．このことから，知識を用いることが有効であることがわかる．

4.2.2　合議の有用性

合議の有用性を確かめることを目的とし，単純多数決による合議のプログラムと，単一プログラムによる実験結果を**表 3** に示す．

対局相手は，合議プログラムと単一プログラムに関して，共に知識有り，無しの2種類を行った．対局は

白番 50 回，黒番 50 回の計 100 回を行った．

表 3　合議と単体による対戦結果

合議プログラム	単一プログラムに対する勝率 (%)
知識無し	74
知識有り	65

表 3 の対戦結果は危険度 1%における 100 戦の二項検定から有意に勝ち越していると言える．このことから，合議を用いることが有効であることがわかる．

また，知識無しによる勝率から，知識無しの場合でも合議を行うことで勝率を向上することができている．

4.2.3　多様性を増した合議

未分類の学習標本にたいして Bagging 法を用いた合議と，Bagging 法にたいして NCL を適用した合議による実験結果を**表 4** に示す．

対局相手はオープンソースの GnuGo を Level0 で用いた[9]．これ以降の実験では，対戦相手は Gnugo を用いて行った．対局は白番 150 回，黒番 150 回の計 300 回を行った．

表 4　Bagging と NCL の実験結果

共有棋譜数	Bagging の勝率 (%)	NCL の勝率 (%)
0	63.4	68.5
500	65.2	68.6
1,000	61.6	65.3

表 4 の共有棋譜数が 1,000 の Bagging はプレイヤ間の乱数のみが異なる合議である．この乱数のみの合議は，知識の差異による多様性を持つ他の全ての合議に対して勝率が低く，合議では集団の多様性が重要なことが示せる．

また，個別に学習を行う Bagging に対し明示的に多様性を増した NCL は全て勝率が向上しているため，未分類の学習標本にたいして NCL が有効であることがわかる．

4.2.4　棋風による分類を行った合議

棋風により分類した学習標本にたいして Bagging 法を用いた合議と，Bagging 法にたいして NCL を適用した合議による実験結果を示す．

まず，実利派のみ，好戦派のみ，模様派のみの棋風を統一した合議の実験結果を**表 5** に示す．対局は白番 200 回，黒番 200 回の計 400 回を行った．

表 5　棋風を統一した合議の実験結果

	実利派	好戦派	模様派
Bagging	70.9	65.5	68.78
NCL	64.75	65.48	67.68

表 5 では，未分類である表 4 の共有棋譜数 0 と比較すると，棋風により分類した方が勝率が向上している．しかし，棋風を統一した合議にたいしての NCL は Bagging に比べ好戦派，模様派と変化が少なく，実利派は勝率が低下している．

3 種類の棋風にたいして NCL を適用したが，勝率が向上しているものは無い．また，棋風によって勝率が変化しており，実利派が特に高いという結果になった．

次に，棋風ごとのプレイヤを混ぜた合議の実験結果を**表 6** 示す．

棋風ごとの割合は，4，3，3 の平均型と 6，2，2，の重点型の 2 種類を行った．対局は白番 200 回，黒番 200 回の計 400 回を行った．

表 6　棋風を混ぜた合議の実験結果

実利:好戦:模様	Bagging の勝率 (%)	NCL の勝率
4:3:3	61.63	68.36
3:4:3	67.61	65.49
3:3:4	65.08	63.36
6:2:2	67.18	69.79
2:6:2	63.79	62.53
2:2:6	65.09	69.18

表 6 では，Bagging の平均型と重点型では，棋風により平均型が高い，重点型が高いと別れる結果になった．また，平均型の場合でも 4 人にどの棋風が来るかにより勝率が変化している．

さらに，表 5 と比較すると，棋風を混ぜた合議よりも棋風を統一した合議の方が実利派，模様派において勝率が高いことがわかる．

Bagging にたいして NCL を適用した場合，勝率が向上したが，全てが向上しているという結果にはならなかった．

4.2.5　合議に用いたプレイヤの分析

未分類の学習棋譜による共有棋譜数 0 と，棋風を統一した合議に用いた 10 プレイヤの単体での勝率の平均を**表 7** に示す．

対局は各プレイヤ白番 200 回，黒番 200 回の計 400 回を行った．

表 7　棋風ごとの実験結果

	未分類	実利派	好戦派	模様派
Bagging	56.42	60.91	59.04	59.04
NCL	56.65	62.09	60.09	59.48

表 7 と表 5，表 6 を比較すると，単一では平均 60%程の勝率だが，合議を行うことで勝率が向上していることがわかる．

また，未分類と棋風による分類の勝率を比較する

と，棋風による分類を行った知識では単体での勝率が2,3%程上昇していることがわかる．このことから，モンテカルロ囲碁に用いられる知識としては，未分類による知識にノイズがかかったものより，棋風の分類等により知識に対するノイズが除去されたものの方が向いているのではないかと推測される．

4.2.6　合議の人数による変化

合議を行う人数によって勝率がどのように変化するのかについて実験を行った．未分類の共有棋譜数 0 の合議の人数を 1 から 10 人まで変化させた実験結果を**表 8**，図 3 に示す．

対局は白番 200 回，黒番 200 回の計 400 回を行った．

表 8　合議を行う人数による実験結果

合議人数	知識無し	Bagging の勝率	NCL の勝率
1	30.35	58.40	59.20
2	23.33	54.33	52.90
3	31.48	62.85	55.23
4	34.15	59.58	58.53
5	36.45	65.80	63.18
6	32.33	65.23	63.18
7	33.25	64.45	64.40
8	32.13	59.53	60.83
9	30.38	63.20	60.38
10	36.40	63.40	68.50

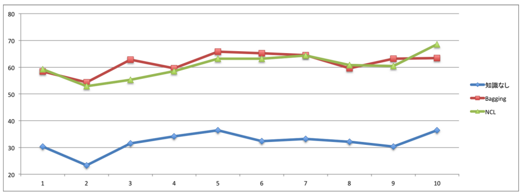

図 3　合議を行う人数による実験

次に，棋風の分類を行った合議の人数を 1 から 10 人まで変化させた実験結果を表 8，図 4 に示す．

対局は白番 200 回，黒番 200 回の計 400 回を行った．

表 9　棋風ごとの合議を行う人数による実験結果

合議人数	実利派	好戦派	模様派
1	59.2	58.1	58.1
2	57.2	56.8	59.6
3	63.7	60.9	62.8
4	61.8	61.4	64.6
5	68.8	68.8	61.9
6	66.2	68.6	62.2
7	66.7	63.1	62.6
8	63.7	69.0	63.0
9	65.5	66.9	67.8
10	70.9	65.5	68.9

合議の人数を変化させている表 8，**表 9** から，合議

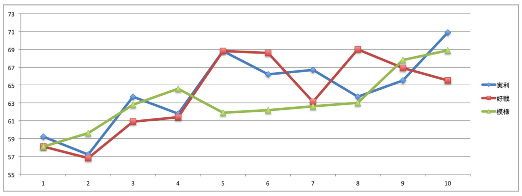

図 4　棋風ごとの合議を行う人数による実験

の人数を増やすことで勝率が向上していくことがわかる．しかし，人数が増えた場合でも勝率が低下している場合もある．

また，表 8 の知識なしとその他の知識を用いた合議について比較を行うと，知識を用いた合議の方が人数が増えた際の勝率の向上が高いことがわかる．

5. おわりに

本研究では，プレイヤが用いる知識にたいし，多様性を増すように調整を行い，囲碁の合議に用いる手法について示した．

まず，乱数のみが異なる合議にたいして，知識の差異により多様性を増すことで勝率を向上させることができた．囲碁の合議においても，多様性が重要であり，知識による多様性が有用であることを示すことができた．

さらに，知識の学習において既に学習した他のプレイヤの知識を考慮した集団学習を行うことにより，より勝率を向上させることができた．

また，学習棋譜にたいして棋風による分類を行い知識を作成した．この棋風を混ぜた合議では，多様性による勝率の向上という点では期待した結果ではないが，未分類の知識を用いた合議に比べ勝率を向上させることができた．

最後に，合議に用いたプレイヤの分析から，単体の勝率において未分類の学習棋譜による知識より棋風による分類を行った知識を用いたプレイヤの方が勝率が上回る結果となった．モンテカルロ囲碁に用いる知識において，棋風などにより分類を行い洗練ことで方向性を持たせることで探索が上手く行くのではないかと推測される．合議に用いられた各プレイヤの勝率が向上したことにより，合議を行った際の単体に対する勝率の向上が上昇したと推測される．

合議の人数増加による勝率の遷移では，人数を増やすことにより勝率が上昇している．しかし，人数を増やした場合でも勝率が下がる場合があり，安定していない．

今後の課題として，棋風の分類により合議がどのように変化したのかについて調査する必要がある．

知識の学習にたいして調整を加えることで多様性を増すだけでなく，学習に用いる棋譜群を棋風により分類することで，より明示的に知識にたいして方向性を持たせることを目的とした．この方向性を持つ知識を用いたプレイヤを混ぜた合議を行うことで，より多様性が増し勝率が向上することを期待した．しかし，実験結果は棋風による知識を持つプレイヤの混ぜる割合が，平均型よりも重点型の方が勝率が高く，さらに棋風を統一した合議の方が勝率が高いという結果になった．最初に述べた，合議における多様性が増すことの重要性という観点からは真逆の結果となっており，棋風ごとの知識についてより調査する必要がある．

また，今回用いた知識は 3×3 の 8 近傍の石の配置とアタリを特徴としているが，棋風によって特徴が出やすい，出にくいといった問題が無いかについて考慮する必要がある．

その他にも，現在勝率を見ることで多様性による合議の勝率向上の成否を確認をしている．勝率という観点のみでは運という要素も強く，その他の観点から成否について判断を行っていく必要がある．

参 考 文 献

1) 鈴木洋平，三輪誠，金田康正:“合議のための多様な将棋プレイヤの集団学習”，GPW2012，pp.17-24（2012）.
2) 清 愼一，山下 宏，佐々木 宣介:“コンピュータ囲碁の入門”，共立出版，2005.
3) P. Auer, N. Cesa-Bianchi, and P. Fischer:“Finite-time analysis of the multiarmed bandit problem”，Machine Learning，47(2/3):pages 235-256，2002.
4) Sylvain Gelly, Yizao Wang, Remi Munos, and Olivier Teytaud:“Modification of UCT with patterns in Monte-Carlo Go”, Technical Report 6062，INRIA, France，2006.
5) Remi Coulom:“Computing Elo Ratings of Move Patterns in the Game of Go”，ICGA Computer Game Workshop,（2007）.
6) 副島佑介, 岸本章宏, 渡辺治. “モンテカルロ木探索の root 並列化とコンピュータ囲碁での有効性について”, GPW2009, pp. 27-33, (2009).
7) Y.Liu，X.Yao:“Ensemble learning via negative correlation”，Neural Networks，vol.12，no.10，pp.1399-1404,（1999）.
8) libEGO，http://www.mimiw.edu.pl/lew/hg/libego/，2011.
9) GnuGo，http://www.gnu.org/software/gnugo/gnugo.html，2011.