情報処理学会論文誌　Vol. 50　No. 11　2740–2751 (Nov. 2009)

推薦論文

モンテカルロ木探索によるコンピュータ将棋

佐　藤　佳　州[†1]　　高　橋　大　介[†1]

本論文では，モンテカルロ木探索によるコンピュータ将棋を実現し，その有効性を検証する．モンテカルロ木探索によるゲームの実現は，ゲームプログラミングの分野において現在最も注目を集めているテーマの１つであるが，将棋では今のところ良い結果を得ることには成功していない．本研究では，コンピュータ囲碁で成功した手法を基に，キラームーブの導入など将棋向けの改良を加えたモンテカルロ木探索によるコンピュータ将棋を実現した．次の一手問題による性能評価では，アマチュア初段程度のプログラムに迫る正答数を得ることに成功し，モンテカルロ木探索が将棋においても有効であることを示した．現在のトップレベルの将棋プログラムはプロに迫るまでとなっており，モンテカルロ木探索のみにより従来の手法を単純に上回る棋力を得ることは難しいと考えられる．しかし，序盤の定跡選択や一部の終盤では従来の手法よりも良い結果を得ることに成功し，モンテカルロ木探索の利用により，現在のコンピュータ将棋の性能をさらに改善できる可能性があることを示した．

A Shogi Program Based on Monte-Carlo Tree Search

Yoshikuni Sato[†1] and Daisuke Takahashi[†1]

Recently, Monte-Carlo Tree Search is attracting much attention in game programming. This method has succeeded in Computer-Go, however it has not yet been able to attain good results in Computer-Shogi. We implemented a Shogi program based on Monte-Carlo Tree Search, using techniques proved in Computer-Go with improvements for Computer-Shogi. In the results of solving problems, the number of correct answers that our program found was almost the same as that of about a 1-dan amateur program. Although the strength of top-level Shogi programs is almost as strong as professional players, our method based on Monte-Carlo Tree Search achieved better performance than existing methods in openings and some positions of the endgame. The results of our experiments showed that Monte-Carlo Tree Search has the possibility of improving the performance of Computer-Shogi.

1. はじめに

将棋，チェス，オセロといった思考ゲームをコンピュータで実現する際には，探索と評価関数を用いるのが一般的である．一方，現在注目を集めている手法として，モンテカルロ木探索（Monte-Carlo Tree Search）[1),2)] がある．この手法の最大の特徴は，評価関数が不要となることであり，その設計が困難とされている囲碁では従来の手法を上回る成功を収めた．

コンピュータ将棋は，囲碁に比べ評価関数が設計しやすいことや，近年の機械学習の成功から，従来の手法によりアマチュアトップクラスの性能を持つまでになっている．しかし，将棋においても万能な評価関数の設計は難しく，複雑な終盤や不利な局面では最善手が選択できないことが多い．また，序盤の指し手選択も，定跡データベースとの一致のみを見ている場合が多く，課題の１つといえる．

モンテカルロ木探索は，これらの問題点を解決できる可能性があり，将棋の分野でも注目を集めている[3)–5)] ものの，今のところ良い結果を得ることには成功していない．モンテカルロ木探索によりコンピュータ将棋を実現する際の特有の問題としては，ランダムな指し手の選択によるシミュレーションでは終局条件（詰み）を満たすことが難しい，ゲームの性質として囲碁よりも読みに重点が置かれる，といった点があげられる．

本論文では，モンテカルロ木探索のアルゴリズムとして UCT（Upper Confidence bounds applied to Trees）[6)] を採用したコンピュータ将棋を実現する．さらに，最近のモンテカルロ囲碁で成功した手法を基に，いくつかの将棋向けの改良を行うことで前述した問題点の改善を試みる．また，定跡選択部や探索と評価関数による手法では良い結果を得ることができなかった局面におけるモンテカルロ木探索の有効性について検証を行う．

2. 関 連 研 究

思考ゲームを実現する際の最も一般的な手法は探索と評価関数である．探索は人間でいうところの「読み」，評価関数は「形勢判断」「大局観」といったものに対応する．

コンピュータ将棋の場合にも，多くのプログラムは探索と評価関数に基づいており[7),8)]，現在ではこの手法に基づくプログラムが，アマチュアトップレベルにまで達している[9),10)]．

†1 筑波大学大学院システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba
本論文の内容は 2008 年 11 月のゲームプログラミングワークショップにて報告され，同プログラム委員長により情報処理学会論文誌ジャーナルへの掲載が推薦された論文である．

しかし，この手法も万能とはいえず，いくつか問題点があることが分かっている．この手法の最大の問題点としては，評価関数の設計の困難さがあげられる．将棋の場合にも，人間の知識をコンピュータに正しく反映させることは容易ではなく，複雑な終盤や不利な局面では最善手が選択できないことが多い．また，もう 1 つの問題点としては，人間のように指し手を絞った深い読みができないといった点があげられる．このような問題点のため，現在のコンピュータ将棋は 1 秒間に数百万局面を探索するにもかかわらず，人間のトッププレイヤを上回ることはできていない．

一方で近年，探索と評価関数とはまったく異なる考え方に基づく，モンテカルロ木探索という手法が注目を集めている．この手法の基となる，モンテカルロ法によるゲームの実現は，コンピュータ囲碁では 1993 年から研究されていた[11]．モンテカルロ法による思考では乱数を利用したゲームのシミュレーション（以下 playout）を何度も行い，そのスコア（石の数の差，勝率など）によって局面を評価する．モンテカルロ法によるゲームの最大の特徴は，人間の知識に基づいた評価関数を用いる必要がないことである．そのため，評価関数の設計が特に困難とされていた囲碁において大きな注目を集めた．しかし，相手のミスを期待した手を選択してしまう，ある程度以上 playout の回数を増やしても棋力が頭打ちになってしまう，といった問題があり従来の手法を上回るには至らなかった[12]．

モンテカルロ法に木探索を組み合わせることによりこれらの問題点を大幅に改善した手法がモンテカルロ木探索である．モンテカルロ木探索の出現により，アマチュア級位者レベルであったコンピュータ囲碁は，9 路盤ではプロに勝利するまでの強さになった．モンテカルロ木探索の最も一般的なアルゴリズムとしては，UCT（Upper Confidence bounds applied to Trees）[6] があげられる．UCT は，その時点までに行った playout の回数と勝率に基づき，より有望な指し手（ノード）に多くの playout を割り当て，探索木を成長させていく．Playout 中の指し手の選択は，完全なランダムでは良い性能を得ることは難しく，多くの囲碁プログラムでは，ゲームの知識を利用して確率的に行っている[13)–15]．

モンテカルロ法によるコンピュータ将棋の研究としては文献 3) があげられる．この研究では，囲碁プログラム CRAZY STONE[1] のアルゴリズムを将棋へ適用し，モンテカルロ法によるコンピュータ将棋の可能性を考察している．Playout に長手数を要するという問題に対しては，最大手数を 10 程度に制限し，末端で評価関数を利用することにより解決を試みている．しかし，次の一手問題の正答率は 3%程度にとどまっており，有望とはいいがたい結果となっている．

本論文では，探索手法として UCT を採用する．さらに，キラームーブやヒストリーヒューリスティックの利用，囲碁において成功している progressive widening[13] を将棋向けに適用，並列化などを行うことにより性能の改善を試みる．また，playout 部分では最大手数を極端に制限することや評価関数を利用するといったことは行わず，ヒューリスティックの導入により自然な終局の実現を目指す．

3. モンテカルロ木探索によるコンピュータ将棋

本章では，まずモンテカルロ木探索と UCT についての説明を行う．その後，モンテカルロ木探索によるコンピュータ将棋を実現する際に行った，playout 部分と探索部分の改良についてそれぞれ説明する．また，定跡選択におけるモンテカルロ木探索の利用，モンテカルロ木探索と静止探索の併用といった応用的な手法についても説明する．

3.1 UCT（Upper Confidence Bounds applied to Trees）

モンテカルロ木探索の基本的な考え方は次のとおりである．

(1) 初期状態ではランダムに候補手を選び playout を繰り返し行う．
(2) 徐々に（勝率が高い手など）有望な手で多くの playout を行っていく．
(3) Playout の回数が閾値を超えたノードでは探索木を成長させ，playout を開始するノードを 1 段深くする．

この操作を繰り返すことで，有望なノード（指し手）ほど探索木が成長し，深く探索されることになる．ルートでは，勝率の最も良い手，あるいは playout が最も多く行われた手を最善手として選択する．

モンテカルロ木探索の最も一般的なアルゴリズムとしては，UCT があげられる．UCT では，playout を割り当てる子ノードの決定に以下の式，UCB1 を用いる．合法手が K 手存在するとき，playout を行う子ノード $I \in \{1,\ldots,K\}$ は以下の式により決定される．

$$I = \mathop{\mathrm{argmax}}_{i \in \{1,\ldots,K\}} \left\{ X_i + c\sqrt{\frac{2\log n}{n_i}} \right\} \tag{1}$$

X_i はノード（指し手）i を選択した場合の勝率，n，n_i はそれぞれ親ノード，ノード i の訪問回数を示す．c の値は通常は 1.0 を用いる[2]．1.0 より大きくした場合には勝率の低い手に，小さくした場合には勝率の高い手により多くのシミュレーションが割り当てられることになる．

このような式を用いることにより，UCT では有望な手で多くの playout を行い深く探索する，といった手法を実現している．

3.2 Playout の改良

Playout 中の指し手の選択は完全にランダムでは良い性能を得ることは難しく，乱数を利用しつつも，何らかの形でヒューリスティックを導入し，対象とするゲーム（囲碁，将棋）らしい指し手の選択を行う必要があることが経験的に知られている．また，終局までに長手数を要するといった将棋特有の問題点もヒューリスティックの利用により改善することが期待できる．

本論文では指し手の選択に，Elo レーティングを利用した手法[13),16)]を用いた．この手法は，指し手をいくつかの特徴の集合と見なし，プロの棋譜などを基にその選択されやすさを Elo レーティングとして数値化し，その値に基づいて指し手の選択確率を決定するというものである．

指し手の選択法としては，この手法のほかに実現確率[17)]を用いるといった手法も考えられる．この手法は，ある特徴を持つ指し手がプロの棋譜中において，どの程度の割合（確率）で選択されたかという統計情報を利用するものである．実現確率と比較して，Elo レーティングを用いる利点としては，実現確率では指し手単体の重要性（選択されやすさ）を考慮しているのに対し，Elo レーティングでは，他の候補手との比較までモデルに含まれているため，指し手の相対的な重要性がより正確に求められるといった点が考えられる．

3.2.1 Bradley-Terry モデルと Elo レーティング

Bradley-Terry モデルはある試合に関する勝敗を予測するモデルである．あるプレイヤ i が γ_i という正の値（レーティング）を持つとすると，1 から n 人までのプレイヤの中で i （$1 \leq i \leq n$）が勝利する確率は，

$$P(i\,\text{が勝つ}) = \frac{\gamma_i}{\sum_{j=1}^{n} \gamma_j} \qquad (2)$$

と表される．複数プレイヤによるチームでの試合は，チームを組むメンバの強さ γ_i の積をチームの強さとすることで扱うことができる．プレイヤ 1，2，3，4，5，6 が存在し，1-2-3，4-5，6 というチームを組むとき，チーム 4-5 が試合に勝つ確率は，

$$P(\text{チーム 4-5 が勝つ}) = \frac{\gamma_4\gamma_5}{\gamma_1\gamma_2\gamma_3 + \gamma_4\gamma_5 + \gamma_6} \qquad (3)$$

とモデル化される．一般的には，γ_i を $400\log_{10}\gamma_i$ と変換したものを Elo レーティングと呼ぶが，本論文中では，変換前の値 γ_i をレーティングの値として扱う．将棋の指し手には，「駒の損得」，「王手」や「逃げる手」など様々な特徴がある．これらの特徴を個々のプレイヤと見なし，各特徴の選択されやすさをプレイヤの強さ（レーティング）と考えることで，上記のモデルを適用できる．この場合，指し手は複数の特徴からなるチームと考えることができる．このモデルを利用して playout 中の指し手の選択を行うことを考えた場合，指し手の選択確率はチームが勝つ確率と対応する．

各特徴のレーティングは，文献 13) の手法により求めた．この手法では，プロの棋譜を学習データとし，各特徴のレーティングを求める．棋譜中の各局面を 1 つの試合と見なし，実際に指された手を勝ったチーム，指されなかった手を負けたチームと考えることで，実際に指された手のレーティングが高くなるように，各特徴のレーティングを算出する．

3.2.2 用いた特徴

本研究において用いた主な特徴を以下に示す．それぞれの特徴について，さらに詳細な分類をしており，実際には 200 程度の特徴（項目）を用いている．

- 駒の損得（SEE）
- 駒をとる手
- 成る手
- 王手
- 相手の利きのある駒の移動（逃げる手）
- 位置テーブルの値の増減（駒を打つ場合には，駒を打つ位置のテーブルの値）
- 玉の位置，周囲の利き
- 直前および二手前に移動した駒との関係

駒の損得は SEE（Static Exchange Evaluation）により求めた．駒の損得は**表 1** に示す値を用いて計算した SEE の値が，100 点区切りでどの区分に属するかを特徴とした．駒の損得（SEE）の計算は，他の特徴に比べコストがかかるが，将棋において特に重要な特徴と考えられる．

位置テーブルの値は文献 18) の手法により求めたものを用いた．**図 1** は玉が 8 八にいる場合の味方の金の位置による点数である．矢倉囲いや美濃囲いの金の位置の点数が高くなっており，玉から遠くなるほど点数が低くなっていることが分かる．このような表を自玉，相

表 1　駒の交換値
Table 1　Values of pieces.

歩	香	桂	銀	金	角	飛
100	280	300	420	530	620	700
と	成香	成桂	成銀	—	馬	龍
270	320	250	430	—	710	850

−62	−108	−64	−70	−88	−98	−88	−124	−98
−116	−112	−4	4	−22	−28	−6	−74	−94
−110	−46	−50	−26	−22	−2	−28	−44	−76
−64	−40	0	−16	10	−18	−8	−4	−50
−98	−52	−26	−18	−22	−14	−20	−26	−60
−78	−38	−20	−26	−20	−18	−14	−44	−66
−104	−20	2	18	−48	−22	−52	−64	−100
−120	X	46	−22	−2	−34	−52	−80	−162
−116	6	−58	8	−82	−30	−86	−74	−160

図 1　玉が 8 八にいる場合の金の位置による点数
Fig. 1　Points by positions of Gold when own King is at 8h.

手玉がそれぞれ 1 一から 9 九にある場合について作成している．

前述した，位置テーブルの値の増減は，（移動先のテーブルの値 – 移動元のテーブルの値）が，−101 以下，−100〜−11，−10〜−1，0，1〜10，11〜100，101 以上のどの区分にあてはまるかを特徴としている．なお，駒を打つ場合には，単に打つ位置のテーブルの値がどの区分にあてはまるかを特徴としている．

3.3　UCT の改良

UCT の大きな問題点は，訪問回数の少ないノードにおいて効率の良い playout の割当てができず，ランダムに近い子ノードの選択を行ってしまうことである．本論文では，キラームーブやヒストリーヒューリスティックの利用，progressive widening などによりこの問題を改善した．

3.3.1　キラームーブ，ヒストリーヒューリスティックの利用

本論文では，UCT の子ノードの選択にキラームーブおよびヒストリーヒューリスティックを利用した．ともにコンピュータチェスやコンピュータ将棋においてよく用いられている手法で，キラームーブは兄弟ノードの最善手，ヒストリーヒューリスティックはある指し手が最善手であった割合（回数）を示す．

囲碁では，キラームーブやヒストリーヒューリスティックに近い考え方を用いた，RAVE [20] と呼ばれる手法が成功を収めている．

将棋の場合には，ゲームの性質上多くの兄弟ノードでは最善手が同一であることが多く，キラームーブは特に重要な概念であるといえる．実際，多くの将棋プログラムでは，指し手の並べ替えなどにおいてキラームーブを利用しており[7]，これらの知識を利用することは，囲碁と比較した場合においても，高い効果を得ることが期待できる．

本研究では，キラームーブやヒストリーヒューリスティックの値が大きいノードでは，式 (1) における c を 1.0 よりも大きな値に設定する．このようにすることで，訪問回数が少ないうちはキラームーブやヒストリーヒューリスティックの値が大きいノードが優先的に選択されることになる．

3.3.2　Progressive widening

Progressive widening [13] とは，新しいノードを作成する際，1 度にすべての合法手を探索対象とするのではなく，ヒューリスティックを利用して良さそうな指し手から探索対象を広げていくという，枝刈り手法の一種である．

本論文では，未訪問のノードでは式 (1) における勝率の値 X_i の部分に，指し手のレーティングの値 R_i を利用した以下の式を用いることで progressive widening を実現した．

$$U_i = \begin{cases} dR_i + c'\sqrt{\frac{2\log n}{e}} & (n_i = 0) \\ X_i + c\sqrt{\frac{2\log n}{n_i}} & (n_i \neq 0) \end{cases} \tag{4}$$

$$I = \underset{i\in\{1,\ldots,K\}}{\mathrm{argmax}}\ \{U_i\} \tag{5}$$

R_i の値は以下の式により指し手 i のレーティングを 0 から 1.0 の値に正規化したものである．定数 c'，d，e の値は実験的に 0.5，4.0，10.0 とした．

$$R_i = \frac{\text{指し手 } i \text{ のレーティング}}{\text{合法手中の最大のレーティング}} \tag{6}$$

このような式を用いることで，レーティングの高い指し手から順に探索対象を広げていく．単純に駒得する手などレーティングが突出して高い手がある場合には，その手が特に重視されることになる．また，勝率の値 X_i が高い指し手があるうちは，探索対象は広がりにくく，強い枝刈りとなる．逆に現在の探索対象の指し手の勝率 X_i が低い場合には，速やかに探索範囲を広げることになる．

3.3.3　並列化，その他

モンテカルロ木探索では，playout の回数は棋力に大きく影響を及ぼす．playout の部分は独立性が高いことからも，並列化は非常に有効であると考えられる．

本論文では，探索木の部分を共有した状態で，スレッドを用い playout を独立に実行する方法を採用した．この方法では，探索木を更新するタイミングの関係上，単一スレッドで実行した場合とは異なる振舞いをすることになるが，4 CPU 程度の並列化においては十分な性能向上が得られることが知られている[14]．

その他の UCT の改良点としては，探索木の内部で見つけた詰み情報を利用するといったことも行っている．探索中で詰みを見つけた場合には，シミュレーションを行うことなくスコア（勝敗）を求めることができる．また，詰みの情報を親ノードに伝えることで，勝敗が決定したノードでは，以降無駄なシミュレーションを行わないようにしている．モンテカルロ木探索の問題点の 1 つとして，厳密な勝敗の証明に大きなコストがかかるという点があげられるが，探索木の内部で見つけた詰み情報を利用することで，この問題を大幅に改善できると考えられる．

3.4　モンテカルロ木探索を用いた定跡選択

現在多くのプログラムは，定跡選択の部分に力を注いでいるとはいいがたく，定跡データベース（ここでは実践譜の集合とする）との一致のみで選択しているプログラムも多い．著者が開発したプログラム「遠見」，「棋理」では，定跡データベース中のある候補手 m が選択される確率を以下の式により決定している（以下確率による定跡選択と呼ぶ）．

$$P(m\,\text{が選択される}) = \frac{m\,\text{が指された回数}}{\text{現局面がデータベースに存在する数}} \tag{7}$$

この手法は，多くのプログラムで用いられていると考えられる一般的な手法といえるが，データベースの指し手を絶対的に信頼し，コンピュータ自体はまったく思考していないため，悪手を選択してしまうことも多い．また，プログラムがあまり得意でない展開も選択してしまうといった問題点も存在する．

本論文では，モンテカルロ木探索を用いた定跡選択を実装し，その有効性を検証した．モンテカルロ木探索を用いた定跡選択部の手順を以下に示す．

(1)　定跡データベースに含まれる各棋譜を 1 回の playout とした UCT の探索を行う．ただし，子ノードを生成する際にはすべての指し手を生成するのではなく，定跡データベース中に存在する指し手のみを生成する．

(2)　(1) で構成した探索木の上で，通常の UCT による探索を行う．

3.5　静止探索との組合せ

将棋は，囲碁と比べて読みが重視されるゲームである．モンテカルロ木探索は，終局までの playout を何度も行いながら，徐々に探索木を成長させていくという性質上，駒の取り合いなどの直線的な読みは苦手と考えられる．このような問題から，モンテカルロ木探索では，特に思考時間が短い場合，駒の損得を正しく評価できない可能性がある．

本研究では，この問題の改善策として，静止探索[21] を用いた駒の損得評価を行うことを検討する．静止探索との組合せでは，駒の価値は表 1 を用い，モンテカルロ木探索の評価がほぼ同等の指し手では，駒損しない手を優先して選択する．

4. 実　　験

実験結果は，特に断りがない場合には，以下に示す環境で行ったものである．

- CPU Quad-Core Xeon X5355 2.66 GHz × 2（8 コア使用）
- メモリ 2 GB

4.1　指し手の Elo レーティングを利用した際の playout の性質

3.1 節で述べた指し手の特徴のレーティングを**表 2** に示す．レーティングの算出には文献 13) の手法を用いた．名人戦の棋譜約 300 局を学習用データとし，各特徴のレーティングを算出した．

レーティングの値が大きいほどより選択されやすい特徴となる．駒の損得や直前の駒の取り返し，王手などは特に強い特徴となっており，妥当なレーティングが算出できているといえる．位置テーブルの値の増減では，飛角より金銀などの小駒の方が，玉との位置関係が重視されていることが分かる．

表 3 に指し手の選択法による playout の性質の違いを示す．予測率は評価用の棋譜 200 局において実際に指された指し手に割り当てた確率の平均，終局率は平手の初期局面にお

表 2　指し手の特徴の Elo レーティング
Table 2　Elo ratings of move patterns.

特徴	詳細	γ_i			
駒の損得（SEE）	飛程度の得	20.04	位置テーブルの値の増減	歩	1.04 – 2.19
	角程度の得	14.37		香	0.16 – 0.90
	金程度の得	9.54		桂	0.57 – 2.25
	銀程度の得	6.08		銀	0.60 – 3.00
	桂香程度の得	3.89		金	0.33 – 2.22
	歩程度の得	2.55		角	0.66 – 1.45
	損得なし	1.11		飛	0.38 – 1.00
	歩程度の損	0.47		馬	0.78 – 1.49
	桂香程度の損	0.32		龍	0.66 – 0.99
	銀程度の損	0.10		その他の成駒	1.05 – 2.21
	金程度の損	0.06	位置テーブルの値（打つ場合）	歩	0.50 – 2.21
	角程度の損	0.05		香	0.24 – 0.97
	飛程度の損	0.02		桂	0.17 – 1.66
駒を取る手	取り返し	12.75		銀	0.19 – 1.35
	その他	1.88		金	0.28 – 1.45
王手		7.55		角	0.16 – 1.17
成る手		1.47		飛	0.22 – 1.89
相手の利きのある駒の移動	歩	1.13 – 1.27	玉移動（位置，周囲の利き）	通常時	0.30 – 1.01
	香	1.90 – 3.19		王手時	0.57 – 4.21
	桂	1.70 – 2.59	直前に移動した駒の移動		0.90 – 1.55
	銀	2.20 – 2.60	直前の駒との位置関係		0.89 – 1.97
	金	4.38 – 4.78	二手前の駒との位置関係		0.94 – 1.86
	角	1.77 – 5.12	直前の位置に戻る（序盤のみ）		0.10 – 0.87
	飛	6.85 – 23.68	…		…
	馬	3.26 – 9.98			
	龍	8.53 – 14.62			
	その他の成駒	0.82 – 1.65			

表 3　指し手の選択法による playout の性質
Table 3　Differences of playout's characteristics according to the method of move selection.

指し手の選択	予測率	終局率	速度
rating なし（完全乱数）	0.037	0.193	約 3,500 回/秒
rating あり	0.172	0.905	約　900 回/秒

表 4　次の一手問題の結果
Table 4　Results of solving problems.

用いた手法（カッコ内はスレッド数）	正答数	除詰将棋
MC/UCT(1)	6 / 98	6 / 86
MC/UCT(8)	12 / 98	11 / 86
MC/UCT(8) + rating	41 / 98	38 / 86
MC/UCT(8) + rating + killer	49 / 98	45 / 86
遠見	60 / 98	48 / 86
棋理	71 / 98	59 / 86

いて playout を 10,000 回行ったとき，256 手以内に終局した割合を示す．速度は平手の初期局面において 1 秒間に実行できる playout 回数を計測したものである．実験環境は Xeon X5355（1 コアのみ使用）とした．

速度は 4 分の 1 程度となっているものの，予測率が大きく向上していることが分かる．また，終局率の向上から，終局条件を満たすことが難しいという将棋特有の問題点を改善できていることが分かる．

なお，以降の実験において，終局しなかった playout は 0.5 勝として扱っている．評価関数を用い，ヒューリスティックにより勝敗を決定する[3] といった手法も考えられるが，評価関数が不要というモンテカルロ木探索の利点を生かすため，本研究ではそのような手法はとっていない．

4.2　次の一手問題の正答数による評価

次の一手問題の正答数による性能評価を行った結果を**表 4** に示す．問題としては，文献 22), 23) の 98 題を用いた．解答時間は 1 問 30 秒とした．

MC/UCT は単純なモンテカルロ木探索（カッコ内はスレッド数），rating は playout および progressive widening に Elo レーティングを利用した場合，killer はキラームーブ，ヒストリーヒューリスティックを利用した場合を示す．参考として，探索と評価関数に基づくプログラムの正答数を併記した．「遠見」，「棋理」はそれぞれアマチュア初段程度，三段程度の棋力を持つ[⋆1]．「遠見」，「棋理」は詰探索ルーチンを持っているため，詰将棋を除いた問題 86 題の正答数による比較も行った．「遠見」は 1 スレッド，「棋理」は 8 スレッドでの実行結果である．

正答数の向上から，各手法がモンテカルロ木探索によるコンピュータ将棋において有効な改良となっていることが分かる．

モンテカルロ木探索は，詰将棋を除いた問題では探索と評価関数による初段程度のプログラムに迫る正答数を得た．詰将棋などの問題では，正確な読みが必要となるためモンテカルロ木探索には適していないと考えられる．

なお，以降の実験において，「モンテカルロ木探索」と表記した場合には，表 4 中の「MC/UCT(8) + rating + killer」を用いたことを示す．

表 5　「遠見」との対局結果
Table 5　Results of games with Tomi.

用いた手法	勝率
モンテカルロ木探索	0.04
モンテカルロ木探索+静止探索	0.32

4.3　「遠見」との対局による評価および静止探索の効果

探索と評価関数によるプログラム「遠見」との対局により，モンテカルロ木探索によるプログラムの評価を行った．結果を**表 5** に示す．思考時間はともに 1 手 10 秒，対局数は 200 局とした．

モンテカルロ木探索は，次の一手問題の正答数では「遠見」とほぼ同等であったにもかかわらず，連続対局の結果では大きく負け越していることが分かる．

その理由として，モンテカルロ木探索の問題点として小さな駒損が多いことがあげられる．特に序中盤の歩損などは，playout の勝敗には結び付きにくく，意味のない歩の突き捨てなどを指してしまうことが多い．また，不利になるほど，相手のミスによる逆転に期待した悪手を指しやすくなるといった傾向もある．

一方，次の一手問題のような tactical な局面では，多少の駒損などは問題にならないことが多い．このようなことから，次の一手問題の正答数による評価と対局結果による評価に大きな差が出たと考えられる．

⋆1 世界コンピュータ将棋選手権や floodgate（http://wdoor.c.u-tokyo.ac.jp/shogi/）での成績，次の一手問題の正答数から推定．

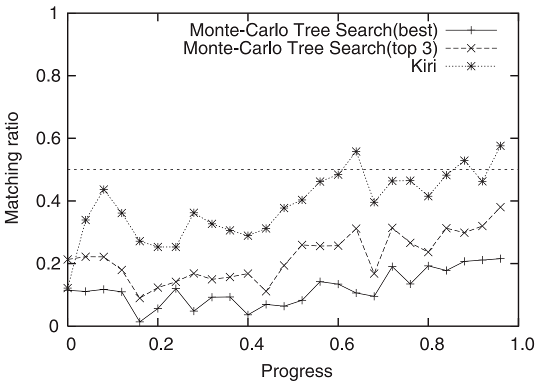

図 2　プロの棋譜との一致率（1 手 1 秒の場合）
Fig. 2　Matching ratio with game records by professional players (1 second per move).

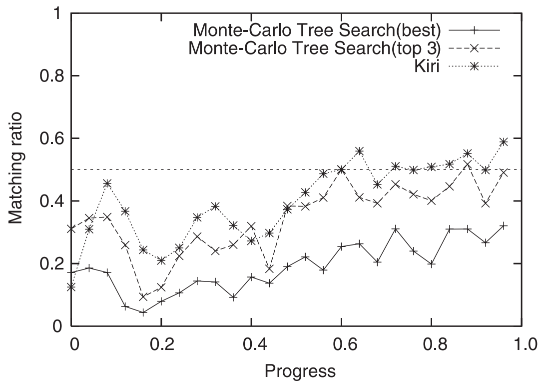

図 3　プロの棋譜との一致率（1 手 10 秒の場合）
Fig. 3　Matching ratio with game records by professional players (10 seconds per move).

モンテカルロ木探索+静止探索は，モンテカルロ木探索に静止探索を組み合わせ，評価値がほぼ互角の場合には駒損しない手を選択するようにしたものである．表 5 の結果より単純なモンテカルロ木探索よりも勝率を改善できていることが分かる．

しかし，依然初段程度のプログラムに達することはできておらず，モンテカルロ木探索により単純に従来の手法を上回ることは難しいと考えられる．

4.4　プロの棋譜との一致率

モンテカルロ木探索を用いた場合のプロの棋譜との一致率を局面の進行度別に計測した．k 手で終局する棋譜中の n 手目の局面の進行度は以下の式で示す値としている．

$$\text{進行度}\ (k,n) = \frac{n}{k} \tag{8}$$

図 2 および**図 3** に進行度別のプロの棋譜との一致率を示す．棋譜中の指し手が，モンテカルロ木探索の最善手と一致した割合，上位 3 手に含まれていた割合，参考として「棋理」の最善手との一致率も示している．

図 2 は思考時間を 1 手 1 秒とした場合の一致率を示したものである．モンテカルロ木探索の一致率が非常に低くなっていることが分かる．これは 1 手 1 秒程度では十分な playout を行うことができていないためであると考えられる．一方，探索と評価関数によるプログラムでは思考時間が短い場合にも，ある程度の一致率を得ることができているといえる．

図 3 は思考時間を 1 手 10 秒とした場合の一致率を示したものである．1 手 1 秒の場合に比べ，モンテカルロ木探索の一致率が大きく向上していることが分かる．一定時間内により多くの playout を行うことができれば，性能はさらに向上すると考えられる．

序盤から終盤にかけて徐々に一致率が向上している理由としては以下のような点が考えられる．

・序盤の場合には，そもそも最善手を 1 つに決定することが難しい．
・終局（詰み）が近いほど時間あたりに行える playout の回数が増加する．
・終局が近いほど指し手の善悪が playout の結果（勝敗）に反映されやすくなる．

また，モンテカルロ木探索がプロの棋譜と一致した局面において，「棋理」がその手を選択した割合は約 0.58 となった（思考時間は 1 手 10 秒）．一致率の差から考えると低い値となっており，モンテカルロ木探索と探索と評価関数による手法は得意とする局面に違いがあると考えることができる．

4.5　モンテカルロ木探索を用いた定跡選択

モンテカルロ木探索を用いた定跡選択の評価として，定跡選択に他の手法を用いた場合との自己対局を行った．対局数は 200 局とし，定跡打ち切り後の思考部には「棋理」を用いた．定跡選択部，通常の思考部ともに思考時間は 1 手 2 秒とし，定跡データベースには，プロやアマ高段者の棋譜約 3 万局のうち手番側が勝ちの棋譜を用いた．なお，定跡選択において，完全に同一の手順が選択された場合には，その対局は無効とし，200 局の中にはカウントしていない．

表 6　定跡選択にモンテカルロ木探索を用いたプログラムの対局結果
Table 6　Match results of a program using Monte-Carlo Tree Search for opening book selection.

比較手法	勝率	評価値
Probability	0.61	−2.01
Maximum Frequency	0.57	+8.10
Search	0.58	−68.24

結果を**表 6** に示す．比較手法の Probability は確率による定跡選択，Maximum Frequency は定跡データベース中で最もよく現れた指し手を選択する手法，Search は通常の探索を行い定跡データベース中に存在する手の中での最善手を選択する手法である．

実験の結果，いずれの手法と比較した場合にもモンテカルロ木探索を用いた定跡選択が勝ち越した（すべて有意水準 5%の二項検定で有意）．通常の探索部分や評価関数との相性の関係もあるため，モンテカルロ木探索を用いた定跡選択がただちに優れているといいきることはできないが，有力な手法の 1 つと考えられる．

評価値は，定跡が打ち切られた時点での局面における「棋理」の評価関数の値の平均である．定跡が打ち切られた時点での評価値は，歩の価値が 100 点であることを考慮すると，ほぼ互角といえる．勝率と比較すると，序盤の優劣を評価関数で正しく判断することの難しさが示されているといえる．

また，定跡選択のアルゴリズムに求められる要件として，ある程度選択される指し手にばらつきがあることがあげられる．実験として，モンテカルロ木探索を用いた定跡選択を初期局面から乱数の初期値を変えて 100 回行ったところ，32 パターンの異なる手順を選択した．この結果から，モンテカルロ木探索による定跡選択では，ある程度ばらつきのある指し手の選択を実現できていると考えることができる．さらにばらつきを持たせたい場合には，毎回の指し手の思考時間を乱数により変更するといった手法も考えられる．

4.6　現在のコンピュータ将棋の問題点とモンテカルロ木探索の利点

前節までの実験結果では，定跡選択では一定の成果をあげたものの，通常探索部分での性能では探索と評価関数によるプログラムには及ばないという結果になった．しかし，探索と評価関数による手法にも，いくつか問題点があることが分かっており，そのような局面ではモンテカルロ木探索が有効である可能性がある．

本節では，2007 年 3 月に行われた渡辺竜王対 Bonanza の対局[24] を例に，現在のコンピュータ将棋がかかえる問題点とモンテカルロ木探索の利点について述べる．この対局は，中盤までは互角の形勢に持ち込んでいたものの，終盤に典型的にコンピュータが苦手とする局面に入り，Bonanza が敗れた．

図 4　渡辺竜王 対 Bonanza（局面 1）
Fig. 4　Watanabe vs. Bonanza (1).

図 4 において，Bonanza は▲ 2 四歩を選択した．しかし，この局面は攻め合いでは先手に勝機はなく，実際この手が敗着となった．

この局面の最善手は▲ 2 七香でこれならまだ難しかったとされている[25]．相穴熊のような局面では，正しい局面の評価をするためには，特定の手順について深く読む必要がある．探索と評価関数による手法では，基本的に一定の深さで探索を打ち切るため，このような局面は苦手といえる．

表 7 は，図 4 に今回実装したモンテカルロ木探索によるプログラムに解答させたときの結果を示す（思考時間は 1 手 60 秒）．評価値はモンテカルロ木探索の勝率を示しており，読み筋は訪問回数が最大の手順を示したものである．

渡辺竜王が最善手として示した▲ 2 七香を選択できていることが分かる．また，Bonanza が最善手として選択した▲ 2 四歩の評価値は 0.301 となっており，この手順の攻め合いでは勝ちにくいことが評価できている．

図 5 における，渡辺竜王の指し手は△ 3 九龍である．しかし，Bonanza をはじめとした多くのプログラムでは，この手を正しく評価することはできていない[26]．この局面も，従来の手法を用いて正しく評価するためには相当深く読む必要がある局面である．

表 8 は，図 5 の局面を今回実装したモンテカルロ木探索によるプログラムに解答させた結果である．△ 3 九龍を選択することはできていないものの，僅差で上位 3 手に含まれて

表 7　モンテカルロ木探索による図 4 の局面の評価
Table 7　Evaluation of Fig. 4 using Monte-Carlo Tree Search.

オーダ	指し手	評価値	読み筋
1（最善手）	▲ 2 七香	0.471	△ 2 六金▲同香△ 7 九飛▲ 3 八金打
2	▲ 3 七馬	0.412	△ 2 四歩▲ 2 七香△ 2 六金▲同香
3	▲ 3 七銀打	0.398	△ 2 六金▲ 2 七香△ 3 九竜▲同銀
…	…	…	
21	▲ 2 四歩	0.301	△ 2 六金▲ 2 三歩成△同金右▲ 8 一馬
…	…	…	

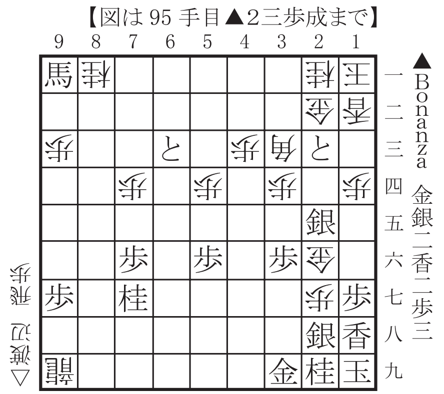

図 5　渡辺竜王 対 Bonanza（局面 2）
Fig. 5　Watanabe vs. Bonanza (2).

表 8　モンテカルロ木探索による図 5 の局面の評価
Table 8　Evaluation of Fig. 5 using Monte-Carlo Tree Search.

オーダ	指し手	評価値	読み筋
1（最善手）	△ 7 九飛	0.622	▲ 2 二と△同角▲ 3 八金打△ 2 八歩成
2	△ 5 九飛	0.607	▲ 2 二と△同玉▲ 4 九歩△ 2 八歩成
3	△ 3 九龍	0.599	▲ 2 二と△同角▲ 3 九銀△ 2 八金
…	…	…	

おり，有力な手として評価できていることが分かる．

モンテカルロ木探索では，playout が直線的な読みの働きをするため，このような局面では探索と評価関数による手法よりも良い結果が得られることが多いと考えられる．

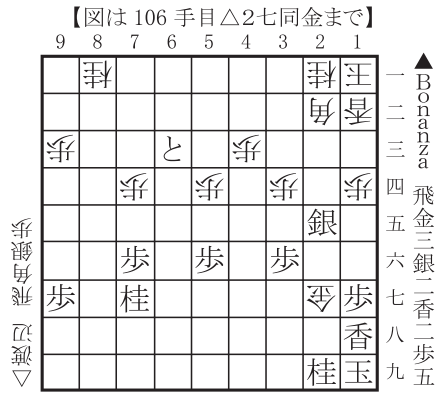

図 6　渡辺竜王 対 Bonanza（局面 3）
Fig. 6　Watanabe vs. Bonanza (3).

表 9　モンテカルロ木探索による図 6 の局面の評価
Table 9　Evaluation of Fig. 6 using Monte-Carlo Tree Search.

オーダ	指し手	評価値	読み筋
1（最善手）	▲ 2 八金	0.334	△ 4 八飛▲ 3 九金△ 4 七飛成▲ 2 七金
2	▲ 3 七銀	0.326	△ 2 六銀▲ 4 八銀打△ 8 二角▲ 3 四銀
3	▲ 3 九銀	0.315	△ 2 六飛▲ 2 八銀打△同金▲同銀
…	…	…	

図 6 は，すでに先手が敗勢の局面であるが，現在の将棋プログラムではほぼ互角の評価をしてしまう[26]．この局面では，先手の持ち駒の多さ，後手玉に王手がかからないことを評価することの難しさなどから，評価関数により正しい形勢判断を行うことが難しい局面といえる．

表 9 は，図 6 の局面をモンテカルロ木探索によるプログラムに解答させた結果である．評価値（勝率）が 3 割程度となっており，かなりの劣勢と評価していることが分かる．

評価関数を設計する際の大きな問題点として，ある程度複雑なゲームでは，局面によって評価すべき項目が大きく異なるということがあげられる．たとえば，図 6 のような局面では，駒の損得よりも手番や相手玉に王手がかかるかといったことを重視しなければならない．多くの将棋プログラムでは，局面の進行度により評価関数のパラメータの値を変化させるなどの手法をとっているものの，あらゆる局面に対応できる評価関数の設計は困難である．

モンテカルロ木探索では，駒の損得や働き，手番や玉の危険度など多くの項目を個別に評価することはなく，シミュレーションの勝率というゲームの知識に依存しない評価指標を用いるため，より汎用性のある局面の評価が可能である．

5. 今後の展望

本論文では，コンピュータ将棋ではモンテカルロ木探索により従来の手法を単純に上回ることは難しいものの，部分的には十分に有効であることを示した．今後コンピュータ将棋にモンテカルロ木探索を利用する場合，従来の手法と併用し，その利点を生かすことが必要となると考えられる．本章では，モンテカルロ木探索の定跡選択部分における利用，通常探索との併用について今後の展望を述べる．

5.1 モンテカルロ木探索を用いた定跡選択

実験の結果から，定跡選択部分ではモンテカルロ木探索の一定の有効性が示されたといえる．問題点としては，通常の思考部（探索と評価関数）との相性が考慮されていない点があげられる．将棋の序盤は確実な最善手を求めることは困難といえ，現実的にはプレイヤ（プログラム）がより勝ちやすい手順を選択することが目標となる．そのため通常の思考部との相性を考えることは重要であり，大きな課題といえる．

改善策の1つとしては，playout 中の指し手の選択に，プログラムの評価関数の値を用いることが考えられる．この場合，実行時間が問題となるが，定跡選択部のように指し手が絞られている局面では数千～数万程度の少ない playout 数でもある程度良い指し手の選択が可能であると考えられる．

また，本論文中では定跡データベース中に存在しない手はまったく生成していないが，この方法では，一致するデータベースが少ない局面では，良い手を見逃してしまう可能性がある．このような問題を解決するため，データベースに存在する指し手のほかに，ヒューリスティックにより有力と考えられる指し手を生成することなども有効と考えられる．

5.2 従来からの手法による思考との併用

モンテカルロ木探索は単純に探索と評価関数による手法を上回ることは難しいと考えられるものの，4章で示したように，局面によっては非常に有効といえる．モンテカルロ木探索単独で用いるのではなく，従来からの手法である探索と評価関数による手法と併用することは棋力の向上に有効である可能性がある．以下に具体的な例を示す．

- 通常の探索において評価値が安定しないとき，モンテカルロ木探索を利用する．
- 通常の探索とモンテカルロ木探索を同時に利用し，評価が大きく違う場合には探索の深さ，思考時間を延長する．

通常の探索において，時間をかけるほど評価値が下がるなど結果が安定しないことがある．このような状況は，深さを基準に探索を打ち切るという通常の手法では，正しい評価を行うために必要な深さまで読むことができていないために起きていると考えられる．しかし，モンテカルロ木探索では，playout が直線的な読みの働きをすることが期待できるため，探索の深さを十分にとることが難しい状況において従来の手法より有効に働く可能性がある．

また，現在のコンピュータはマルチコア化が進んでおり，従来の手法とモンテカルロ木探索の両手法を同時に行うことも可能である．まったく異なる手法による思考を同時に行い，参考にすることは棋力向上に有効である可能性が高いと考えられる．

6. おわりに

本論文では，モンテカルロ木探索によるコンピュータ将棋を実現した．囲碁において成功した手法に加え，チェスや将棋で古くから用いられているキラームーブやヒストリーヒューリスティックといった考え方を利用することにより，その性能を改善できることを示した．

次の一手問題による性能評価では，単純にモンテカルロ木探索を利用した場合の正答数は非常に低いものとなっているのに対し，本論文中の手法を適用した場合には，初段程度のプログラムに迫る正答数を得ることに成功した．

有用性という点では，現在のコンピュータ将棋はアマチュアトップに匹敵する強さにまで達しており，モンテカルロ木探索によるプログラムが従来の手法を単純に上回ることは難しいと考えられる．しかし，序盤の定跡選択や一部の終盤では，従来の手法を上回る有望な結果を得ることに成功し，コンピュータ将棋においてモンテカルロ木探索の部分的な利用が棋力の向上に有効となりうることを示した．

現在のコンピュータ将棋は，ほぼすべてのプログラムが同様のロジックに基づいており，かかえている問題点もある程度共通しているといえる．モンテカルロ木探索は，これらの問題点を改善するコンピュータ将棋の新たな実現法として，今後がおおいに期待される手法であると考えられる．

参考文献

1) Coulom, R.: Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search, *Proc. 5th International Conference on Computers and Games*, Turin, Italy (2006).

2) 美添一樹：モンテカルロ木探索—コンピュータ囲碁に革命を起こした新手法，情報処理，Vol.49, No.6, pp.686–693 (2008).
3) 橋本隼一，橋本　剛，長嶋　淳：コンピュータ将棋におけるモンテカルロ法の可能性，第 11 回ゲームプログラミングワークショップ，pp.195–198 (2006).
4) 伊藤毅志，新沢　剛：モンテカルロ法を用いた 5 五将棋システム，情報処理学会研究報告，GI，[ゲーム情報学]，Vol.2007, No.62, pp.1–6 (2007).
5) 佐藤佳州，高橋大介：モンテカルロ法によるコンピュータ将棋の実現，情報処理学会第 70 回全国大会講演論文集，2U-4 (2008).
6) Kocsis, L. and Szepesvári, C.: Bandit Based Monte-Carlo Planning, *Proc. 15th European Conference on Machine Learning* (*ECML*), pp.282–293 (2006).
7) 鶴岡慶雅：将棋プログラムの現状と未来，情報処理，Vol.46, No.7, pp.817–822 (2005).
8) 橋本　剛：コンピュータ将棋，オペレーションズ・リサーチ：経営の科学，Vol.52, No.1, pp.5–9 (2007).
9) 鶴岡慶雅：最近のコンピュータ将棋の技術背景と激指，情報処理，Vol.49, No.8, pp.982–986 (2008).
10) 棚瀬　寧：棚瀬将棋の技術背景，情報処理，Vol.49, No.8, pp.987–992 (2008).
11) Brügmann, B.: Monte Carlo Go. http://ideanest.com/vegos/MonteCarloGo.pdf (accessed 2009-1-11)
12) Yoshimoto, H., Yoshizoe, K., Kaneko, T., Kishimoto, A. and Taura, K.: Monte Carlo Go Has a Way to Go, *Proc. 21st National Conference on Artificial Intelligence* (*AAAI-06*), pp.1070–1075 (2006).
13) Coulom, R.: Computing Elo Ratings of Move Patterns in the Game of Go, *Computer Game Workshop*, Amsterdam, Netherlands (2007).
14) Gelly, S., Wang, Y., Munos, R. and Teytaud, O.: Modifications of UCT with Patterns in Monte-Carlo Go, Technical Report RR-6062, INRIA (2006).
15) 山下　宏：モンテカルロ法の彩について．http://www32.ocn.ne.jp/~yss/cgf20080412.html（参照 2009-1-11）
16) 金子知適：兄弟節点の比較に基づく評価関数の調整，第 12 回ゲームプログラミングワークショップ，pp.9–16 (2007).
17) 鶴岡慶雅：将棋プログラム「激指」，アマ 4 段を超える—コンピュータ将棋の進歩 4, 松原　仁（編），pp.1–17, 共立出版 (2003).
18) 保木邦仁：局面評価の学習を目指した探索結果の最適制御，第 11 回ゲームプログラミングワークショップ，pp.78–83 (2006).
19) Auer, P., Cesa-Bianchi, N. and Fischer, P.: Finite-time Analysis of the Multiarmed Bandit Problem, *Machine Learning*, Vol.47, pp.235–256 (2002).
20) Gelly, S. and Silver, D.: Combining Online and Offline Knowledge in UCT, *Proc. 24th International Conference on Machine Learning*, pp.273–280 (2007).
21) Beal, D.: A generalised quiescence search algorithm, *Artificial Intelligence Journal*, Vol.43, No.1, pp.85–98 (1990).
22) 松原　仁：コンピュータ将棋の進歩 2，共立出版 (1998).
23) 将棋タウン棋力判定問題集．http://www.shogitown.com/school/judge/judgetop.html（参照 2009-1-11）
24) 大和証券杯ネット将棋公式ホームページ．http://www.daiwashogi.net/（参照 2009-1-11）
25) 渡辺　明：渡辺明ブログ．http://blog.goo.ne.jp/kishi-akira/（参照 2009-1-11）
26) 山下　宏：FPGA で将棋プログラムを作ってみるブログ．http://blog.livedoor.jp/yss_fpga/（参照 2009-1-11）

(平成 21 年 2 月 10 日受付)
(平成 21 年 9 月 11 日採録)

推　薦　文

この論文は，ゲーム木探索の新しいアルゴリズムであるモンテカルロ木探索について，種々の技術の組合せが将棋を対象とした場合にも有効に働くことを示したものである．これまでモンテカルロ木探索は，局所性の高い囲碁のような対象が適していると信じられていたため，一手が全局的に影響することの多い将棋での有効性を示したことには大きな意義がある．よって，論文誌の推薦論文としてふさわしいものである．

（ゲームプログラミングワークショッププログラム委員長　岸本章宏・金子知適）

佐藤　佳州（学生会員）

1985 年生．2008 年筑波大学第三学群情報学類卒業．現在，筑波大学大学院システム情報工学研究科博士前期課程．人工知能，思考ゲームに興味を持つ．

高橋　大介（正会員）

1970年生．1991年呉工業高等専門学校電気工学科卒業．1993年豊橋技術科学大学工学部情報工学課程卒業．1995年同大学大学院工学研究科情報工学専攻修士課程修了．1997年東京大学大学院理学系研究科情報科学専攻博士課程中退．同年同大学大型計算機センター助手．1999年同大学情報基盤センター助手．2000年埼玉大学大学院理工学研究科助手．2001年筑波大学電子・情報工学系講師．2004年同大学大学院システム情報工学研究科講師．2006年同助教授，2007年同准教授．博士（理学）．並列数値計算アルゴリズムに関する研究に従事．1998年度情報処理学会山下記念研究賞，1998年度，2003年度情報処理学会論文賞各受賞．日本応用数理学会，ACM，IEEE，SIAM各会員．