

Title	コンピュータ将棋における高精度キラーヒューリスティックの研究
Author(s)	橋本, 隼一
Citation	
Issue Date	2007-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/3583
Rights	
Description	Supervisor:飯田 弘之, 情報科学研究科, 修士

修 士 論 文

**コンピュータ将棋における
高精度キラーヒューリスティックの研究**

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

橋本隼一

2007 年 3 月

修 士 論 文

コンピュータ将棋における 高精度キラーヒューリスティックの研究

指導教官 飯田弘之 教授

審査委員主査 飯田弘之 教授

審査委員 東条敏 教授

審査委員 烏澤健太郎 助教授

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

510078 橋本隼一

提出年月: 2007 年 2 月

概要

本研究は名人を超える将棋プログラムの実現を目標として、 $\alpha\beta$ 法の効率化を図るために既存のキラー手の分析を行い、それらの改良について考察し、新たに高精度のヒューリスティックを提案する。

現在までにキラー手として3種類の手が提案されているが、キラー手の有効性を示す指標を定義してこれをもとに定量的な分析を行ったところ、各キラー手にはそれぞれ異なる性質があることが明らかになった。分析をもとにして既存のキラー手の改良を進める中で手順を考慮したより高精度のキラーヒューリスティックにたどりついたためこれをコンテキストキラーヒューリスティックとして提案する。提案するキラー手は単に探索の効率を上げるだけでなく、静止探索中、あるいは選択的探索における末端付近でも積極的に生成・探索することで、プログラム全体の強化につながることを評価実験の結果から示す。

目次

第1章	はじめに	1
1.1	人工知能とゲーム	1
1.2	研究の目的	1
1.3	論文の構成	2
第2章	ゲーム木探索とキラーヒューリスティック	3
2.1	$\alpha\beta$ 法とカット	3
2.2	ムーブオーダーリング	3
2.3	キラーヒューリスティック	4
2.4	関連研究	5
第3章	既存のキラー手の分析	6
3.1	コンピュータ将棋 TACOS の概要	6
3.1.1	概要	6
3.1.2	各キラー手の実装	6
3.2	キラー手の分析	7
3.2.1	有効率の定義	7
3.2.2	分析項目と実験条件	7
3.2.3	局面の進行に対する有効率の推移	8
3.2.4	先読みの深さに対する有効率の推移	10
3.2.5	有効なキラー手となる指し手のカテゴリの分布	11
3.3	分析結果のまとめと考察	11
3.3.1	Brotherbest の性質	12
3.3.2	Passbest の性質	12
3.3.3	Premovebest の性質	12
第4章	既存のキラー手の高精度化	13
4.1	Brotherbest の改良	13
4.2	Passbest の改良	13
4.3	Premovebest の改良	13
第5章	コンテキストキラーヒューリスティックの提案	17

第 6 章	提案手法の実装と評価	19
6.1	実装上の問題	19
6.2	$n=2$ の場合の実装	19
6.3	自己対戦による評価	20
6.4	次の一手問題による評価	20
6.5	評価実験と提案手法についての考察	21
6.6	キラー手の改良と提案手法のまとめ	21
第 7 章	結論	23
第 8 章	まとめ	24
第 9 章	今後の課題	25
9.1	コンテキストキラーヒューリスティックについて	25
9.2	自然言語処理の利用について	25

図 目 次

2.1	探索中のノードから見た各キラー手の位置関係	4
3.1	進行度に対する有効率の推移	8
3.2	Passbest が有効率 0.91 を示した局面	9
3.3	Passbest が有効率 0.48 を示した局面	9
3.4	深さに対する有効率の推移	10
3.5	カテゴリ別の有効率積算	11
4.1	Premovebest が有効に働かない例 (A 図)	14
4.2	A 図 (図 4.1) からの探索の様子	14
5.1	コンテキストキラーヒューリスティック ($n=2$)	17
6.1	キラー手を格納するハッシュ表	20
6.2	局面に至るまでの手が最善手に影響する割合 (イメージ)	22

表 目 次

2.1	ゲーム非依存の情報利用	5
4.1	A 図からの探索と Premove の変化	15
5.1	A 図から D 図に至るの探索の流れと Contextbest の変化	17
6.1	自己対戦の結果	20
6.2	次の一手問題の結果	21

第1章 はじめに

1.1 人工知能とゲーム

今日、計算機の数値計算能力が人間を大きく上回っていることに異を唱える者はない。それでは計算機が人間の知能を上回っているかといえば、否定的な意見がほとんどであろう。人工知能研究の黎明期から、ゲーム、特にチェスや将棋のような古典的なゲームをプレイすることのできるプログラムの実現はこの分野の試金石とみなされてきた。情報理論の先駆者であるシャノン[1]は1950年にチェスをプレイするプログラムを考案している。ゲームがこの分野で重視される大きな理由は、それらのゲームを指しこなせることが人間の知の象徴としてみなされてきたことによる。実際、1997年にコンピュータチェスDEEPBLUEが当時の世界チャンピオンであったカスパロフを破ったときの衝撃は非常に大きかった。

将棋はチェスによく似たゲームであり、日本ではチェスよりも人気が高い。将棋プログラムが名人に打ち勝つことができれば、カスパロフが敗れて以来の衝撃となるだろうことは想像に難くない。DEEPBLUEの手法を応用することで強い将棋プログラムの開発も可能であるように思われるが、現在コンピュータ将棋のレベルは専門家に肉薄しているものの名人のレベルに達していない。その間にはコンピュータチェスの研究では発見されなかった人間と計算機の差異が隠れているはずである。

本論文の目的は究極的には名人を上回る将棋プログラムの実現である。未来の将棋プログラムが人間の思考をまねたものになるのか、それともチェスのように人間とは全く違うアプローチを取るのかはまだわからない。少なくとも名人を上回る将棋プログラムを実現する過程で、我々は人間の知性に迫ることができるだろう。

1.2 研究の目的

本研究の目標は専門家に勝つことのできるコンピュータ将棋の実現である。そのために既存の手法を改良することが不可欠である。

ゲームをプレイするプログラムはゲーム全体を一つの木とみなし、最善の変化を見つけることで実現される。ゲーム木の探索のアルゴリズムには通常 $\alpha\beta$ 法[2]が用いられておりその効率は指し手をどのような順序で探索するかによって決まる。いわゆる「良い手」を先に探索するほど効率が上がり、理想的には99%以上の探索を行わずに済む[3][4]。ま

た，ある手が良い手であることが探索に先立って分かっているならば，その手に続く変化に対して他の変化以上計算資源を割いて深く調べることができる。

通常どのような手が良い手になりやすいかはゲームに依存した問題であるが，筆者はキラーヒューリスティックと呼ばれる，ゲーム非依存のヒューリスティックに着目した。このヒューリスティックは探索中に得られた結果を利用して良い手を選び出す手法であるがまとまった研究はなされていない。

1.3 論文の構成

本論文は9章からなる。2章ではゲーム木探索の基本的な概念とキラーヒューリスティックの位置づけについて述べる。3章では有効率を定義して既存のキラー手の性質を定量的に分析し，この結果に基づいて4章でこれらの改良について考察する。5章ではより精度の高いコンテキストキラーヒューリスティックを提案し，その実装方法や評価について6章で述べる。7章で本研究の結論を述べ，8章で論文全体をまとめる。最後に9章で今後の課題について述べる。

第2章 ゲーム木探索と キラーヒューリスティック

2.1 $\alpha\beta$ 法とカット

本論文で扱うゲームとはチェスや将棋などに代表される二人完全情報零和ゲームと呼ばれるものである。この種のゲームは木としてモデル化することができ、これをゲーム木と呼ぶ。

ゲームをプレイするプログラムの最も基本的なアルゴリズムは minimax 法である。このアルゴリズムは局面をルートとして適当な深さまで木を展開し、葉ノードで優劣を評価する (評価関数を呼び出して評価する)。内部ノードでは子ノードのうち自手番にとって最も有利な子ノードの評価値をそのノードの評価値とする。この処理を再帰的に繰り返すことでルート局面の評価値と、ルートから一つの葉ノードへと至る最善応手手順が定まる。最善応手手順の第一手目がこのアルゴリズムによって選ばれた指し手となる。

このアルゴリズムで深さ d まで展開を行う場合、局面で選択できる指し手が平均して b であるとする b^d 回の評価関数呼び出しが必要で、実用的ではない。そこで実際には minimax 法の改良である $\alpha\beta$ 法を用いる。この方法ではウィンドウ (値の範囲) を使って minimax 法と同様の処理を行う。探索中にウィンドウ外の値が得られた場合、ノードで未探索である指し手の探索をすべて打ち切る。これは**カット**と呼ばれ、探索結果に影響しない効率化である。

2.2 ムーブオーダリング

$\alpha\beta$ 法が最も効率よく働くのはすべてのノードで最善の手が一番最初に探索される場合で、評価関数を呼ぶ回数は $b^{d/2}$ になる。これはほとんどの場合、率にして 99% 以上の効率化である。逆に最悪のケースは、すべてのノードで悪い手から順に探索してしまう場合で、このときの評価関数を呼ぶ回数は minimax 法と同じになる。実際は、すべてのノードで最善の手が一番最初に探索が行われるということまずない。探索を効率的に行うには何らかの方法で候補手を並び替え (ムーブオーダリング)、良い手ほど先に探索できるようにする必要がある。

指し手を選択的に探索する場合にはすべての手を生成してから並べ替えるという手順を踏まず、いくつかのカテゴリに当てはまる手だけを逐次的に生成・探索することがある。

このような特定の手のみを生成する手法も広義にはムーブオーダリングと呼べる。

2.3 キラーヒューリスティック

キラーヒューリスティックはすでに探索済みの類似局面の最善手¹を優先的に探索する手法であり、ムーブオーダリングに利用されるヒューリスティックの一つである。この、類似局面の最善手を**キラー手**と呼ぶ。キラー手という言葉がいつ頃から使われているものかは不明である。おそらくチェスなどでとても良い手のことをこの様に呼んでいたのではないかと推測される。ゲームプログラムにおいては直前の相手の手を否定するという意味で反駁手と呼ばれることもある。

ヒューリスティックの多くはゲームに強く依存するものであるが、キラーヒューリスティックはゲームから独立したヒューリスティックである。本論文は将棋を題材として選んだが同様の議論は他のゲームでも当てはまる。

現在までにキラー手として提案されているのは次の三つである。図 2.1 に探索中のノードとの位置関係を示す。

Brotherbest 同じ深さにある直前に探索されたノードで見つかった最善手。

Passbest[11] 親ノードからパス手で遷移したノードで見つかった最善手。

Premovebest[12] 直前の手が同一であるようなノードで見つかった最善手。

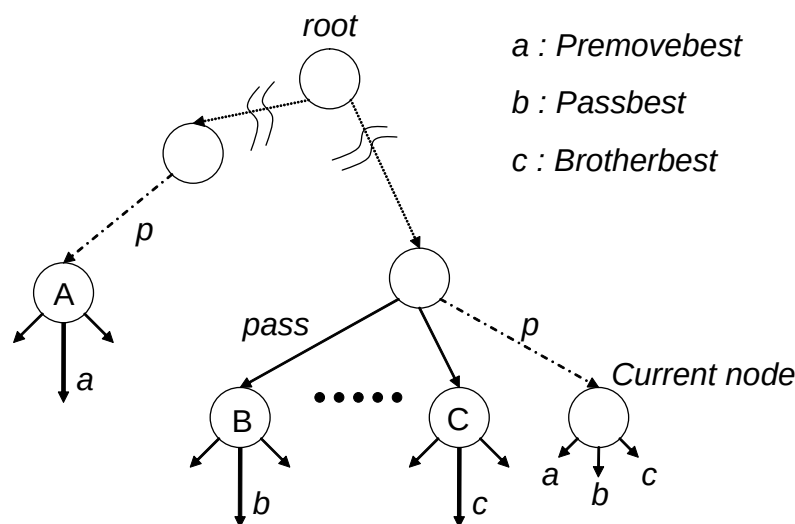


図 2.1: 探索中のノードから見た各キラー手の位置関係
a,b,c:それぞれ局面 A,B,C で見つかった最善手。

¹ カットを発生させた手を含む

Brotherbest と Passbest については探索中の局面と元の局面がゲーム木上で近い位置にあるため、両局面の駒の配置が類似していると考えられる．これに対して Premovebest はやや特殊である．事実 [12] では Premovebest をキラー手とは呼んでいない．ゲーム木上でも探索ノードと距離があるため駒の配置が類似しているという保証もない．しかし両局面に至る直前の手が同一であるということは、**状況が類似**していると考えられる．言語学的な見方をすれば文脈が似ているといえることができる．キラー手の取得に必要なのは駒の配置の類似性ではなく状況の類似性であるからこのとらえ方は妥当であろう．

2.4 関連研究

キラーヒューリスティックについて踏み込んだ研究はない．ここではゲームに関する研究のうち、探索直前までに得られる情報を利用する手法について、分類してみたい．

探索済み局面の最善手を利用する手法はオンライン情報の利用の一つであり広く用いられている．反復深化法は、深化前の最善手を優先的に探索することで効率化が図られている．ヒストリーヒューリスティック [9] やバタフライヒューリスティック [10] は、すべての合法手についてそれぞれが最善手となった回数を残り深さに応じて重みづけして記録しておきオーダリングに利用する手法である．

一方で棋譜の利用など、オフライン情報を利用する手法についても研究が進んでいる．実現確率探索 [5] は専門家の棋譜からどのような手が指されやすいかを統計的に算出し、指し手の選択や探索延長に利用する手法であり、本論文で題材としているコンピュータ将棋 TACOS でも採用している．他にも棋譜を利用した評価関数の最適化 [6] や n-gram モデルを利用した棋譜からの手筋抽出 [7][8] 等がある．

これらを表にまとめると表 2.1 のようになる．このうち 1 の領域については利用できる情報が少なすぎて利用されていない．逆に 2 は多すぎて利用が困難である．

表 2.1: ゲーム非依存の情報利用

	Recorded	Unrecorded
on-line	1	Killer, History, Baterfly, ID
off-line	RPS, N-Gram, Eval-Opt	2

ID:反復進化法, RPS:実現確率探索, Eval-Opt:[6] の評価関数最適化法.

第3章 既存のキラー手の分析

2章で現在までにキラー手として提案されている Brotherbest, Passbest, Premovebest について紹介した。本章の目的はこれらのキラー手を定量的に分析することである。分析実験にはコンピュータ将棋プログラム TACOS を使用する。キラーヒューリスティックはゲームに非依存なヒューリスティックであるが、将棋のような複雑なゲームでは特に性質の差が現れやすいと考えられるため良い題材である。

本章ではまず TACOS の概要と各キラー手の実装について述べた後、キラー手の有効性の指標として有効率を定義しこれをもとに三つのキラー手を分析する。

3.1 コンピュータ将棋 TACOS の概要

3.1.1 概要

TACOS[13] は実現確率探索をベースにした将棋プログラムである。指し手を比較的細かく分類している点が特徴で、カテゴリ毎に指し手を逐次生成しながら多重反復進化法を利用して探索を進める。評価関数では駒割、駒の位置、玉の安全度、飛び駒の自由度、駒組みなどを評価する。いわゆる静止探索は局面の実現確率に応じて生成する指し手カテゴリを制限することで実現している。また df-pn を利用する詰みルーチンを持つが、基本的にはルート局面でのみ詰み探索を行っている。

3.1.2 各キラー手の実装

提案されているキラー手 (図 2.1) の TACOS における実装について述べる。キラー手の登録はノードの探索終了時 (最善手が判明、あるいはカットが発生した際) に毎回行われる。いずれのキラー手も一手だけ保存しており、最善手が変化した場合には上書きされる。探索時にはキラー手は優先的に生成され、他の手よりも比較的高い確率を与えられて探索される。

Brotherbest 探索の深さをインデックスとする配列を用意しておき、同じ深さにある直前に探索されたノードで見つかった最善手を利用している。ほとんどの場合には直近の兄弟ノードの最善手となるが、探索済みの兄弟ノードが無い場合は直近の従兄弟ノードの最善手を利用する。それもない場合は空である。

Passbest Brotherbest と同様に探索の深さをインデックスとする配列を用意しておき、パス手で遷移したノードで最善手が見つかった場合に手を登録している。

Premovebest 将棋の全ての合法手 (7000 程度) をインデックスとする表を用意しておき、各ノードの探索が終わるごとに、直前の指し手をインデックスとして見つかった最善手を保存しておく。使用時には直前の指し手でこの表を引き、格納されている手を利用する。

3.2 キラー手の分析

3.2.1 有効率の定義

本論文では、あるキラー手が探索中のノードで最善手となるかあるいはカットを起こした場合にそのキラー手は**有効**であると呼ぶ。あるキラー手がどれくらい有効であるかを示す値として次のように有効率を定める。

$$(\text{有効率}) = \frac{(\text{最善手になった回数}) + (\text{カットを起こした回数})}{(\text{合法手であった回数})} \quad (3.1)$$

最善手になる場合とカットが起こる場合では探索効率の面では多少異なるが、キラー手が良い手であるか否かを論じる上では同一視してよい。分母にキラー手の生成回数を利用しないのは、合法手でない場合の判定に必要なコストが、合法手かつ有効でない場合に比べて十分無視できると考えられるからである。

3.2.2 分析項目と実験条件

有効率を指標にコンピュータ将棋プログラム TACOS を用いて各キラー手の性質を分析した。キラーヒューリスティックはゲームに依存しないヒューリスティックであるが、将棋は局面の分岐数が比較的大きいゲームであるため、キラー手の性質の差がよりあらわれやすい題材であると考えられる。

分析は以下の観点で行った。

1. 局面の進行に対する有効率の推移
2. 先読みの深さに対する有効率の推移
3. 有効キラー手となる指し手カテゴリの分布

多くのゲームでは序盤・中盤・終盤という区分けがなされており (1) はそれらを通してキラー手の有効性に変化がないかを見るためのものである。(2) は反復深化法の利用を意識した観点である。TACOS のように指し手を逐次的に生成・探索するプログラムでは、通

常生成するカテゴリの順序は固定されている．探索の深さによって有効率に変動があるならば，カテゴリの順序を動的に変化させることで効率化が図れる可能性がある．(3)はそれぞれのキラー手の傾向（攻めか受けかなど）を掴むことを目的としている．たとえば Passbest は相手がパスに近いような意味のない手を選んだ時に有効になりやすいと考えられ，攻撃色の強いカテゴリが集まると予想される．

実験は TACOS をベースにそれぞれのキラー手を単独で利用するプログラムを用意し，異なる 100 局面を与えて 10 秒間探索させ，式 3.1 で示したそれぞれの回数をカウントするという方法で行った．

キラー手そのものの性質を明らかにするため，他の全ての手よりも前に評価し，深さによる制限は行わず，合法手であるときには必ず読ませることにした．

対象となる局面には，棋譜集 [15] のレーティング 2300 以上同士の対戦棋譜から手数が 30～80 のものをランダムに選んで使用した．

3.2.3 局面の進行に対する有効率の推移

図 3.1 は進行度（どのくらい駒が進んでいるかを表し序盤・中盤・終盤の判定基準となる数値）に対する各キラー手の有効率を局面毎にプロットしたものである（有効率は 10 秒間の探索全体での平均）．

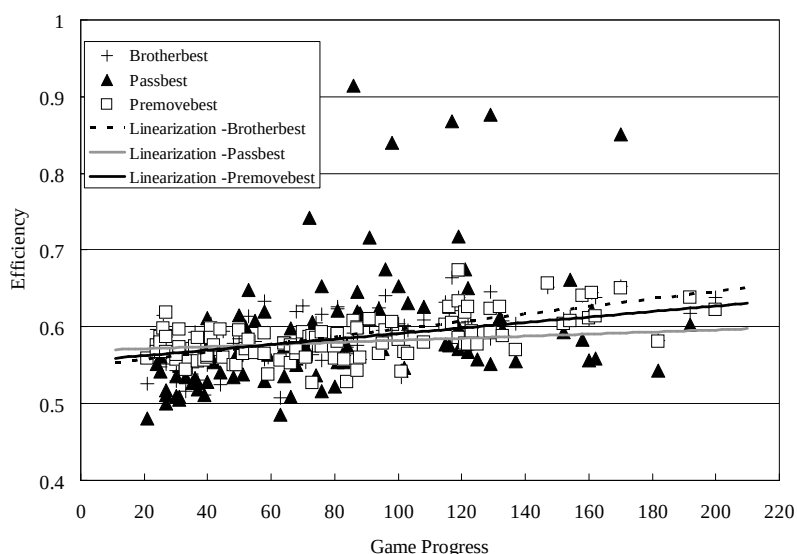


図 3.1: 進行度に対する有効率の推移

点が全体的に広がっていることから，ランダムに選んだ局面が序盤～終盤を網羅しているものであることがうかがえる．また回帰直線から見るといずれのキラー手も平均して 6 割程度の有効率を示している．

特に目立つのは Passbest であり、9 割近い値を持つ点がいくつかある．Passbest の有効率が最も高い 0.91 を示した局面と、最も低い 0.48 を示した局面を図 3.2，図 3.3 に示す．局面はどちらも先手番である．



図 3.2: Passbest が有効率 0.91 を示した局面



図 3.3: Passbest が有効率 0.48 を示した局面

図 3.2 では後手からの△5五飛～△3九角が見え、先手玉が危うい状況である．先手からは成立するか不明だが▲1四歩～▲1三歩成と迫る手順があり、寄せ合いの最終段階である．Passbest の有効率が 0.8 以上を示した局面は共通してこのような状況にあった．

一方の図 3.3 は双方にとって動きが取りづらい局面である．先手が歩を突けば同歩とされるような局面であり、桂打ちや飛車・角の移動も場所によって相手の受け方は変化する．人間的な感覚でいえば水面下で戦っている局面である．

そもそも Passbest は相手の手がパスの時に得られた最善手であるため、相手の手によって最善手がころころ変わる様な局面では有効に働かない．逆に互いに切り合うような局面では相手の指し手がある程度無視できるため有効性が高くなると考えられる．言い換えれ

ば Passbest の有効率は局面の激しさに比例する傾向があると表現できる。¹二つの局面にはこの傾向が顕著に表れている。

一方で Brotherbest と Premovebest の有効率は 0.6 付近で一定している。直前の手に対応した最善手が含まれる Premovebest が穏やかな序盤から激しい中盤・終盤を通してあまりばらつきが見られないのは興味深い結果である。

3.2.4 先読みの深さに対する有効率の推移

図 3.4 は探索の深さ²に対する各キラー手の有効率（100 局面の平均）をとったものである。実現確率探索を採用しているために有効率が定まらない局面については無視し、有効率が定まった局面が 50 以上あるデータのみを示した。

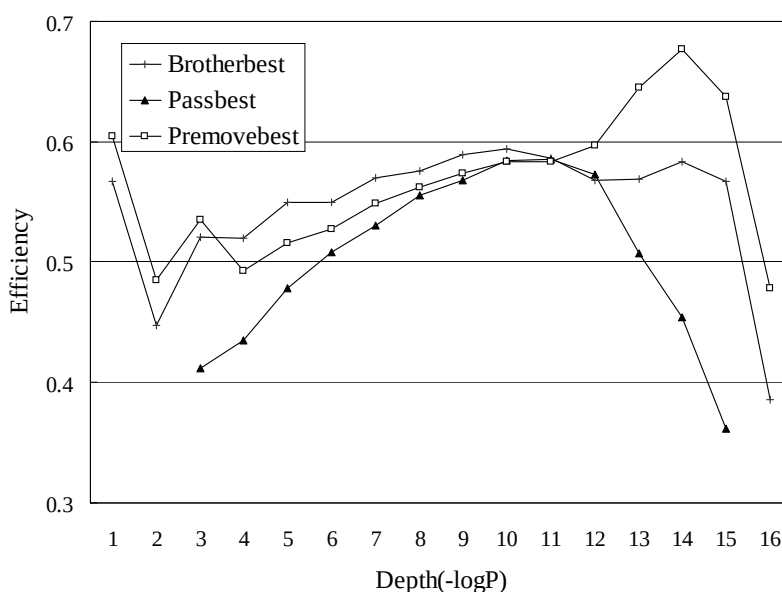


図 3.4: 深さに対する有効率の推移

各キラー手の違いは深さ 11 以降で顕著に表れている。これを超えると Premovebest はそのまま上昇を続け、Brotherbest はその値を保ち、Passbest は減少に転じる。

Premovebest と Passbest の振る舞いは以下のように説明できる。3.1.1 節で述べたとおり、一般的なプログラムが評価関数内で行う静止探索を TACOS ではカテゴリの制限という形にして通常探索の中で行っている。このため探索の末端に近づくに従って、王手を防ぐ手や直前に動いた駒を取る手など直前手と関連の深い手が大きな割合を占めるようになり、同時に局面の激しさはおさまっていくと考えられる。

¹見方を変えれば将棋で言ういわゆる**激しい局面**とは、パスをしたときとパス以外の手を指したときの局面が似ている局面であると考えられる。

²TACOS は実現確率探索を採用しているため、正確には局面の実現確率 p としたときの $-\log_2 p$ 。

3.2.5 有効なキラー手となる指し手のカテゴリの分布

図 3.5 は各キラー手ごとに有効なキラー手が生成された指し手のカテゴリが占める割合を示したその生成元が占める割合を示したものである（100 局面平均）。比較のため、Nokiller としてキラー手を用いない場合のデータを併記した。

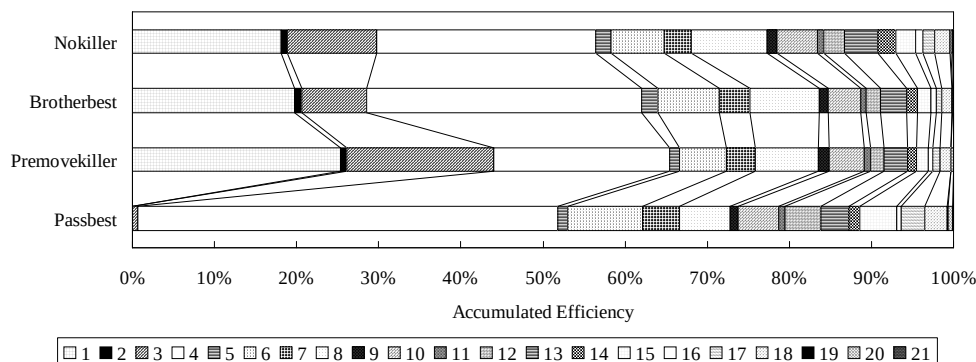


図 3.5: カテゴリ別の有効率積算

割合が少ないものは除いた。各カテゴリの詳細については [13] を参照。1：王手防ぎ，2：詰めろ防ぎ，3：直前に動いた駒を取る，4：最も価値の高い駒を取る，5：得になる駒取り，6：成り，7：損しない王手，8：最も価値の高い駒が逃げる，9：合駒，10：守り，11：白玉への飛び利防ぎ，12：飛び利防ぎ，13：交換，14：相手の最善手を妨害，15：手筋，16：白玉周に利きを増やす，17：駒組み表の値を増やす，18：攻撃，19：相手玉周に利きを増やす，20：絶対値表の値を増やす，21：他

他で 2 割近くを占める「王手防ぎ」が Passbest に入っていないのは、王手がかかっているときにパス手を読まないようにしているためである。代わりに半数以上を占めているのが「最も価値の高い駒を取る」である。「手筋」や「攻撃」の割合も比較的多く、全体的に攻撃色の強いキラー手であることが分かる。

Premovebest には「王手防ぎ」と「直前に動いた駒を取る」が比較的多く含まれている。これは直前の指し手に対する最善手が Premovebest となるのだからこの結果は妥当と言えそうだが、半数以上は直前の手と直接関係のなさそう手が含まれている点にも注意が必要である。

Brotherbest は「直前に動いた相手の駒を取る手」が若干少ない他は、Nokiller と似た分布になっている。前節までに Brotherbest には特徴的な性質が見えなかったがここでも同様な結果を見せており、Brotherbest は安定したキラー手と言うことができるだろう。

3.3 分析結果のまとめと考察

3.2.2 節で分析の観点として挙げた以下の三点についてまとめ、それぞれのキラー手の性質について考察する。

1. 局面の進行に対する有効率の推移
2. 先読みの深さに対する有効率の推移
3. 有効キラー手となる指し手カテゴリの分布

(1) について、いずれのキラー手も平均して 6 割程度の有効率を示し、この値はゲームの進行に対しては依存しないことがわかった。

(2) では探索の深さに対しては浅い部分で多少有効率が低く深さが増えるにつれて上昇する傾向が認められたが、末端付近以外での有効率は一定して Brotherbest, Premovebest, Passbest の順になっており、深さに応じて生成の順序を入れ替える必要はなさそうである。

(3) についてはキラー手ごとに特徴がみられた。

3.3.1 Brotherbest の性質

局面の進行度、探索ノードの深さ、有効キラーの分布のいずれについても目立った特色は見られなかった。Brotherbest は三つのキラー手の中で最も書き換えが頻繁に起こっており、そのためプログラムの特徴をそのまま映し出しているのではないと思われる。有効キラーの分布がキラー手を利用しなかった場合とほとんど同じであったこともこの説を裏付けている。

3.3.2 Passbest の性質

図 3.2, 3.3 に示したように局面の激しさが増すほど有効率が上がる傾向が見られた。探索深さに対する変化も、Passbest と局面の激しさに一定の関係があることを示している。有効キラーの分布には攻撃的な指し手が多く含まれ、全体として攻撃の色合いが強いキラー手である。

3.3.3 Premovebest の性質

直前の手に対応した指し手がほとんどであると予想していたが、進行度に対する依存がほとんど見られなかったことと、有効キラーの分布で直前手と直接関係のない手も割合多く含まれていたことから、局面の状況に依存したキラー手であると考えられる。状況に依存していれば穏やかな序盤であれ激しい中終盤であれ一定の有効率を示すことの説明が付き、また有効キラー手の分布が直前手に対応するカテゴリ一色にならなかったのも納得がゆく。

第4章 既存のキラー手の高精度化

前章の分析で既存のキラー手の有効率は平均して6割程度であることがわかった。次の目標はそれらのキラー手を改良し効率を高めることである。

4.1 Brotherbest の改良

Brotherbest は直前の兄弟ノードで見つかった最善手となっているが、「直前」という条件を外せば複数の手を使用することができるため改良が可能であると思われる。なおこの手法は [14] で金沢が述べているものと同様のアイディアで、深さ毎に複数の手 (2,3 手) を保存しておき新たに登録される際にはところてん方式で古いものを押し出してゆく。金沢のプログラムでは効果が見られたようだが TACOS では多少よくはなるものの有意な差が認められなかった。

4.2 Passbest の改良

Passbest については改良が難しいように思われる。探索中の局面に対して、兄弟ノードのうちパスで遷移したノードは一つしか存在しないために該当するキラー手も一つであり、Brotherbest のような複数手の利用は不可能である。反復深化法を利用しているため前回の反復時のパス手を使う方法は考えうるが、すでに前回反復時の最善手を利用して重複する部分が多いと考え実装は見送った。

4.3 Premovebest の改良

Premovebest については改良の余地がある。A 図 (図 4.1) がその例である。

図 4.2 は A 図からの探索の一例を示したものである。A 図から B, C, D の順に探索が進む。表 4.1 にこの探索が進むにつれて△2 二銀に対する Premove がどのように変化するかを示した。

B 図は A から▲2 一飛打△2 二銀打と進んだ局面である。この局面の最善手は▲3 一銀であり、この手が△2 二銀打に対する Premovebest として登録される。



図 4.1: Premovebest が有効に働かない例 (A 図)

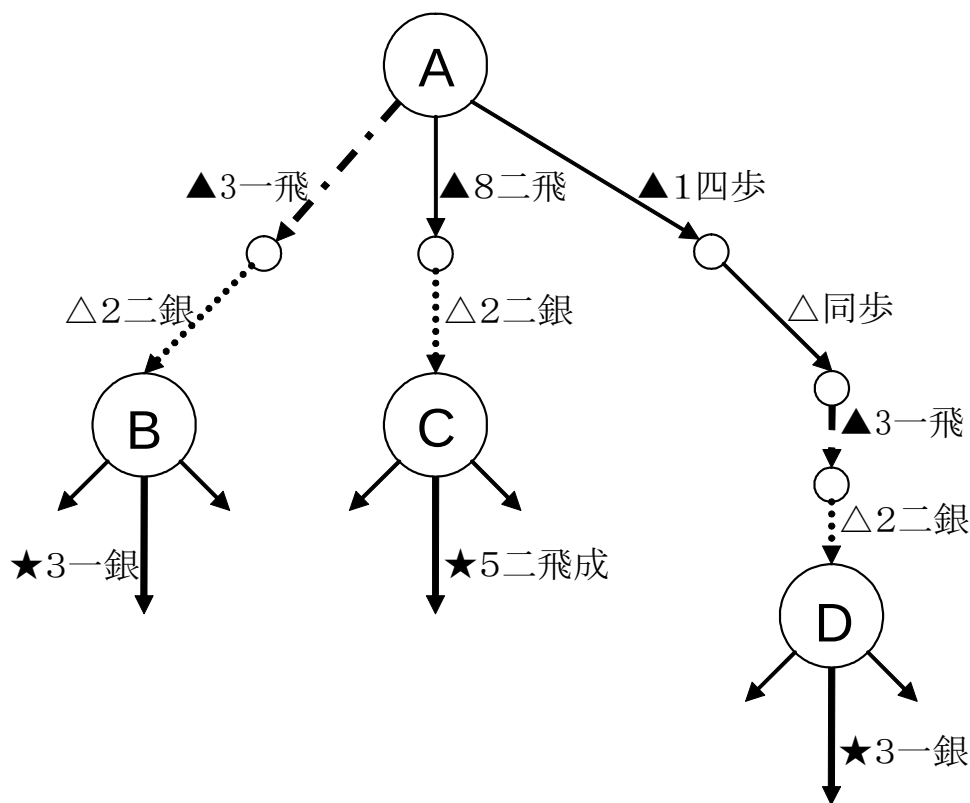


図 4.2: A 図 (図 4.1) からの探索の様子

B,C,D の直前手はすべて同じ. B,D は 2 手前の手も同じ. ★はその局面を探索して見つかる最善手.

表 4.1: A 図からの探索と Premove の変化

図	探索木中の位置	局面	最善手と Premove
A 再掲		9 8 7 6 5 4 3 2 1 皇 将 と 将 王 将 歩 銀 歩 金 歩 歩 金 玉 金 香 桂 飛 香 一 持駒 歩 桂 銀 金 飛 二 三 四 五 六 七 八 九	最善：- 前：- 後：-
B		9 8 7 6 5 4 3 2 1 皇 将 飛 将 王 将 と 将 王 将 歩 銀 歩 金 歩 歩 金 玉 金 香 桂 飛 香 一 持駒 歩 桂 銀 金 二 三 四 五 六 七 八 九	最善：▲3一銀打 前：- 後：▲3一銀打
C		9 8 7 6 5 4 3 2 1 皇 将 飛 将 王 将 と 将 王 将 歩 銀 歩 金 歩 歩 金 玉 金 香 桂 飛 香 一 持駒 歩 桂 銀 金 二 三 四 五 六 七 八 九	最善：- 前：▲3一銀打 後：▲5二飛成
D		9 8 7 6 5 4 3 2 1 皇 将 飛 将 王 将 と 将 王 将 歩 銀 歩 金 歩 歩 金 玉 金 香 桂 飛 香 一 持駒 歩 桂 銀 金 二 三 四 五 六 七 八 九	最善：▲3一銀打 前：▲5二飛成 後：▲3一銀打

最善：局面を探索して見つかった最善手,

前:探索前に $\triangle 2$ 二銀打に対して入っていた Premove, 後:探索後に $\triangle 2$ 二銀打に対して格納される Premove.

次に B 図から一度 A 図に戻り今度は▲8 二飛打△2 二銀打と進む (C 図) △2 二銀打に対する Premovebest が登録されているがこの手は有効ではなく、最終的に▲5 2 飛成が最善手となって△2 二銀打に対する Premovebest は上書きされる。

さらに探索が進んで A 図から▲1 四歩△同歩のあと、B と同じように▲2 一飛打△2 二銀打と進んだ局面が D 図である。この局面の最善手は B と同じく▲3 一銀打であるが、Premovebest すでに上書きされているため有効に働かない。

このように Premovebest は直前の手のみに依存しているため、重要な手順に対する応手が頻繁に上書きされてしまっていると考えられる。そこで筆者は直前の手のみではなく 2 手前、3 手前の手が一致するノードの最善手を利用することで手順を考慮したキラ一手の実現を図ることにした。Brotherbest の改良で述べたような複数手の利用もこの問題を軽減するには効果があると考えられるが、本質的な問題は解決されないため上述の手法を選択する。

第5章 コンテキストキラーヒューリスティックの提案

Premovebest を一般化した，コンテキストキラーヒューリスティックを提案する．これは前節で述べた Premovebest の問題点を解決するものであり，提案手法では直前の n 手が一致する局面の最善手を利用することで手順を考慮したキラー手となることが期待できる．図 5.1 に $n=2$ の場合の Contextbest と探索ノードの位置関係を示した．「コンテキスト」の名は手順の一致を文章中の文脈の一致になぞらえたものである．このヒューリスティックによって選ばれるキラー手を Contextbest と呼ぶことにする．

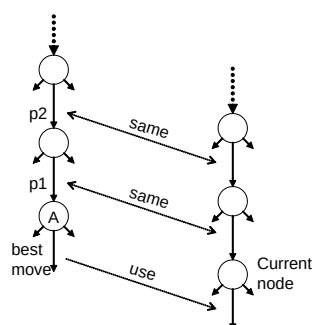


図 5.1: コンテキストキラーヒューリスティック ($n=2$)

A:元局面, p1,p2:指し手. A での最善手 bestmove を探索中の局面でのキラー手として利用する.

表 5.1: A 図から D 図に至るの探索の流れと Contextbest の変化

図	A 図からの指し手	最善手	探索前	探索後
A	(ルート局面)	-	-	-
B	▲ 2 一飛打△ 2 二銀打	▲ 3 一銀打	-	▲ 3 一銀打
C	▲ 8 二飛打△ 2 二銀打	▲ 8 五飛成	▲ 3 一銀打	▲ 3 一銀打※
D	▲ 1 四歩△同歩▲ 2 一飛打△ 2 二銀打	▲ 3 一銀打	▲ 3 一銀打	▲ 3 一銀打

探索前・後はそれぞれ▲ 2 一飛打△ 2 二銀打の手順に対する探索前と探索後の Contextbest に保存されている指し手．表 4.1 と比較されたい．

表 5.1 は前章の A～D 図を提案手法を用いて探索した際の Contextbest の変化である. 表 4.1 と異なり, ※で記したように C 図の探索結果で B 図で得た▲2 一飛打△2 二銀打に対する最善手が上書きされないため, D 図でいきなり▲3 一銀打を評価できる. (▲8 二飛打△2 二銀打に対する最善手は変化する)

第6章 提案手法の実装と評価

この章では前章で提案したコンテキストキラーヒューリスティックの $n=2$ の場合について実装しその評価を行う。

6.1 実装上の問題

単純な方法で Contextbest を登録参照しようとする問題が生じる。Premovebest の場合は合法手の総数 (7000 程度) と同じ大きさの表を用意しておけばよかったが、 n 手の一致を正確に確認するには合法手の総数の n 乗分の表を用意しなければならない。これは相当な大きさになってしまい、プログラムのほかの部分のパフォーマンスを落としかねない。表の大きさを抑えるために次の二通りの実装を行った。

1. 合法手数と同じ大きさの配列を用意し、直前の手と 2 手前の手からハッシュ値を得る方法
2. 4 手分の配列を合法手数分用意し、一つの直前手に対して 2 手前の手を最大 4 つまで登録する方法

(1) は簡潔な方法でコンテキストキラーヒューリスティックが実現できるが、(2) のほうが直前手・2 手前手の一致を完全に確認でき精度が高いと思われる。二つの実装の自己対戦ではほとんど差が見られなかったため、以下ではより簡潔な (1) の実装について述べる。

6.2 $n=2$ の場合の実装

直前の 2 手が一致する局面の最善手を利用する手法を示す。図 6.1 が実装に用いたハッシュ表の様子である。表中でのインデクスは直前 2 手のハッシュ値であり、これには指し手のコード (各指し手に 6480^1 以下の数値を割り当てたもの) の排他論理和を取ったものを用いた。

¹駒の数 $\times$ 移動先のマス数 $\times$ 成り・不成り = $40 \times 81 \times 2$ 。実際の指し手の総数はこれを下回る。

hash value	Move
0	
	$\vdots$
$p1.code() \oplus p2.code()$	bestmove
$\vdots$	$\vdots$

図 6.1: キラー手を格納するハッシュ表

p1: 直前手, p2: 2 手前の手, code(): 指し手に対応する数値, bestmove: 見つかった最善手

6.3 自己対戦による評価

Contextbest を利用するプログラムと利用しないプログラムとで 300 回の自己対戦を行い性能を評価した. 思考時間は一手 10 秒とし, 初期局面として横歩取りや振り飛車対居飛車, 矢倉戦などの序盤局面を 150 用意して各局面について先後を入れ替えて 2 局ずつの対局を行った.

表 6.1: 自己対戦の結果

Context	Normal
171	129
数字は勝数.	

表 6.1 がその結果で 40 勝以上勝ち越しており, 有意水準 0.01 で強くなったことが確認できた. また強さの指標を表すレーティング値でみると 50 強²の改善が見られた. 現在の TACOS のレーティングは 2450 程であり, 名人のレーティング値が 3000 程度であることを考えると, 提案手法によって 1 割程度近づいたことになる.

6.4 次の一手問題による評価

Contextbest を利用するプログラムと利用しないプログラムで次の一手問題を解かせて性能を評価した. 問題には初段向けの問題 [16] と高段者向けの問題 (4 段)[17] にある各 100 問を用い, 思考時間は 1 手 10 秒以内とする. 表 6.2 がその結果であり, 解ける問題数が増えている. 探索速度は若干遅くなっているが誤差のとして無視できる程度であり, 提案した実装方法によってプログラム全体のパフォーマンスが落ちることはなかった.

² 3^n 勝 1 敗でレーティング差が $200 \times n$ となる式を使用. 勝率を $p = \frac{171}{300}$ として $200 \times \log_3 \frac{p}{1-p} = 51.3$.

表 6.2: 次の一手問題の結果

	初段向け	4 段向け
Normal	63(105.2)	40(101.4)
Context	67(102.0)	43(100.1)

数字は正当数. () 内は探索速度で単位は千ノード/秒.

6.5 評価実験と提案手法についての考察

キラークューリスティックは基本的には探索の効率を上げるものであり、本来は探索結果に影響しない。自己対戦と次の一手問題でプログラム性能の向上が見られた理由としては反復深化法との関係によるものがまず考えられる。提案したキラークュー手によって探索の効率が上がったため、指定時間内により深く読むことができるようになり大幅な性能の向上につながったと考えられる。

また TACOS でのキラークュー手の扱い方にも理由があると考えられる。今日、複雑なゲームを扱うプログラムでは選択的探索を用いる場合が多く、探索の末端付近では駒を取るなどの単純な手に偏りがちである。TACOS ではキラークュー手に関しては末端付近でも生成するようにしているため、提案した手法によって通常は読み落とされるような重要な手順も読むことができるようになり性能の向上を後押ししたと考えられる。

実験で用いた Contextbest は直前の 2 手を利用するもの ($n=2$) であったが一般の場合について少し考えてみたい。

もし仮に n が初期局面からの手数に一致すれば、Contextbest が得られた局面と探索中の局面は一致することになる。すべての局面には必ず一つの最善手が存在すると考えるなら、直前の一手を利用する Premovebest の方法は局面を一次近似して比較したものであり、2 手以上を利用する提案手法は二次、三次の近似で局面の一致・不一致を判断するものであると言えるだろう。

事前に行った実験では 2 手前の手のみが一致するノードの最善手を利用する方法では、Premovebest に比べて性能の悪化が見られた。この結果から、局面に至るまでの手が局面での最善手の決定に与える影響は手数がさかのぼるにつれて小さくなっていくと考えられる (図 6.2)。そのため、 n の値は探索全体のパフォーマンスを考慮に入れて決定する必要がある。

6.6 キラークュー手の改良と提案手法のまとめ

3 章の分析をもとに既存のキラークュー手の高精度化について検討した。Premovebest が局面の状況にあったキラークュー手であることに着目し、これを拡張した直前の n 手が一致したノードの最善手を用いるコンテキストキラークューリスティックを提案した。

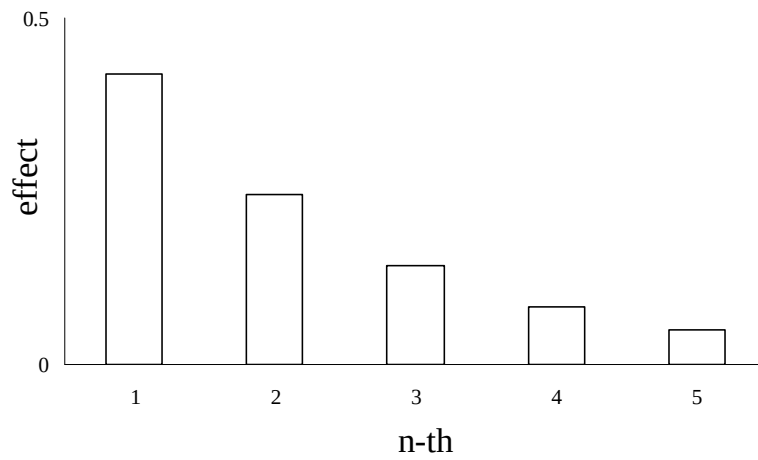


図 6.2: 局面に至るまでの手が最善手に影響する割合（イメージ）

特に直前の2手が一致したノードの最善手を用いる場合について実装し，改良前のものとの評価実験を行ったところ自己対戦，次の一手問題のどちらでも良い結果が得られた．特に自己対戦では300戦171勝129敗という素晴らしい結果が得られたが，これらの性能向上は単に探索の効率化によるものだけではなく，末端近くでもキラー手を探索していることで提案した手法によって通常は読み落とされるような重要な手順も読むことができるようになったためと考えられる．

第7章 結論

提案されている3種類のキラー手について有効率を指標にして分析したところ、それぞれのキラー手性質は異なることがわかった。Brotherbest はプログラム全体の性格を反映しやすい性質があり、Passbest は局面の激しさと関連性があり、Premovebest は局面の状況に沿った手を含みやすい。なおこれらのキラー手の有効率はゲーム全体を通して6割程度であった。

それぞれのキラー手の高精度化を試みたが Brotherbest と Passbest に関しては改良が難しかった。Premovebest に関しては直前の n 手が一致するノードの最善手を利用する様に拡張し、コンテキストキラーヒューリスティックとして提案した。このキラー手を利用したプログラムと改良前のものとの自己対戦は300戦171勝129敗で大きく勝ち越し、提案手法の有効性が裏付けられたと言える。

TACOS を実験に用いたことで、末端付近でのキラー手の振る舞いも確認することができた。先読みの深さに対する有効率の推移から、末端付近で Premovebest の有効率が増すことがわかり、また Contextbest が大きな成果を上げたことも提案したキラー手がやはり末端付近で有効に働いていることの証左と言える。全幅探索を基本にしているプログラムであっても内部で静止探索を利用しているいればその静止探索の中で Premovebest や Contextbest が有効に働くと予想される。

本稿で提案した手法は局面にいたる指し手に着目したヒューリスティックであり、ある局面での最善手がそれ以前の指し手と深い関連があるようなゲームにおいては将棋に限らず有効であると考えられる。

第8章 まとめ

名人を超えるコンピュータ将棋の実現に向けて、探索アルゴリズムの基本となっている α β 法の効率化を目標に、高精度キラーヒューリスティックの実現を試みた。

はじめに提案されている3種類のキラー手について、有効率を用いて定量的な比較・分析を行った。その結果、各キラー手の有効率は局面の進行度に対してほとんど依存せず、いずれも0.6程度で推移することが確認された。分析結果から各キラー手にはそれぞれの特色があり、Brotherbestは探索アルゴリズム全体の性格を反映しやすいこと、Passbestは攻撃的で局面の激しさと関連が深いこと、Premovebestは局面の状況に沿った手が含まれ易いことを示した。

次にこの分析結果を元に既存のキラー手の改良を試み、局面に至る手順が同一であるノード最善手を利用するコンテキストキラーキラーヒューリスティックを提案した。2手分の手順を考慮するものを実装し、改良前のものと比較実験を行ったところ自己対戦、次の一手問題ともに性能の向上が見られた。特に自己対戦の結果は300戦171勝129敗と大きく勝ち越した。この結果は提案手法の利用と相まって、探索の末端付近でもキラー手を探索していることが好影響を及ぼしていると思われる。

コンテキストキラーヒューリスティックは手順を考慮したヒューリスティックであり、ある局面での最善手がそれ以前の指し手と深い関連があるようなゲームにおいては将棋に限らず有効であると考えられる。

第9章 今後の課題

9.1 コンテキストキラーヒューリスティックについて

提案した手法ではハッシュ表上で衝突があった場合に単純な上書きを行っているが、よりカットを起こしやすい手を残す方が効率が良いと考えられる。また $n=3$ 以上の場合については、6.6 節で示した過去の手が局面の最善手に影響する割合を考慮したハッシュ関数にすべきである。

9.2 自然言語処理の利用について

今日、ゲームプログラムに関する基本的なアルゴリズムはほぼ出そろっている感があり、今後の発展は2.4 節で示したようなオンライン・オフライン情報の効率的な利用によってもたらされると予想される。これらの情報の利用はゲームを木とみなしているままでは難しく、文章とみなすことが鍵になると思われる。事実、実現確率探索や n -gram モデルを利用した手筋の抽出などが成果を上げており、本稿で提案したヒューリスティックもゲーム木の中に現れる文脈に注目したものである。提案手法をより良いものにしていくためにも自然言語処理との関連性を追求していきたい。

謝辞

本研究は北陸先端科学技術大学情報科学科飯田研究室で開発されている，コンピュータ将棋 TACOS を利用して行った．同研究室の橋本剛講師には研究テーマの決定から始めて論文の書き方までお世話になった．改めてお礼申し上げたい．

TACOS グループの長嶋さんには TACOS の理解に躓いたときにたびたびお世話になった．同グループの村田君との議論の中で生まれた話が本研究の底流にある．A.Cincotti と J.Kloetzer のおかげで筆者の英語は大分ましになった．朝倉さんと他の飯田研究室のメンバーにも感謝したい．

最後に飯田弘之教授のイマジネーションと行動力に敬意を表したい．氏に巡り合えてこの研究室に迎えられたことに心から感謝する．

参考文献

- [1] Shannon, Claude E: “Programming a Computer for Playing Chess”, Philosophical Magazine, Ser.7, Vol. 41(314), 1950.
- [2] D.E.Knuth and R.W.Moore: “An analysis of alpha-beta pruning”, Artificial Intelligence, Vol. 6, No. 4, pp.293-326, 1975.
- [3] Judea Pearl: “The Solution for the Branching Factor of the Alpha-Beta Pruning Algorithm and its Optimality”, Communications of the ACM, Vol. 25, No.8, pp.559-564, 1982.
- [4] Jaap van den Heric: “The delicate fight”, The 10th Game Programming Workshop 特別記念講演, pp.63-67, 2005.
- [5] Yoshimasa Tsuruoka, Daisaku Yokoyama and Takashi Chikayama: “Game-tree Search Algorithm based on Realization Probability”, ICGA Journal, Vol. 25, No.3, pp.145-152, 2002.
- [6] 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」, The 10th Game Programming Workshop 招待講演, pp78-83, 2006.
- [7] 大槻 知史: 「n-gram 統計からの『必然手の抽出』」, The 10th Game Programming Workshop, pp.89-96, 2005
- [8] 村田 朋紀, 橋本 剛, 長島 淳: 「棋譜情報からの手筋自動抽出とその利用」, The 10th Game Programming Workshop, pp.17-24, 2006
- [9] J. Schaeffer: “The History Heuristic and Alpha-Beta Search Enhancements in Practice”, IEEE Transactions on Pattern Analysis and Machine Intelligence 11, pp.1203-1212, 1989.
- [10] D. Hartmann: “Butterfly boards”, ICGA Journal 11(2-3), pp.123-132, 1999.
- [11] 山下 宏: 「YSS – そのデータ構造, およびアルゴリズムについて」, コンピュータ将棋の進歩 2 (松原 仁編著), 共立出版, pp.112-142, ISBN: 4320028929, 1998.

- [12] 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 情報処理学会, pp.817-822, 2005.
- [13] 橋本 剛: 「将棋プログラム TACOS のアルゴリズム」, アマトップクラスに迫る コンピュータ将棋の進歩 5 (松原 仁編著), 共立出版, pp.33-66, ISBN: 4320121546, 2003.
- [14] 金沢 伸一郎: 「金沢将棋のアルゴリズム」, コンピュータ将棋の進歩 3 (松原 仁編著), 共立出版, pp.15-26, ISBN: 4320029569, 2000.
- [15] 久米 宏: 「最強の棋譜データベース将棋倶楽部 24」, 成甲書房, ISBN: 4880861693, 2004.
- [16] 週刊将棋編: 「初段の終盤」, 毎日コミュニケーションズ, ISBN: 4839905584, 2001.
- [17] 週刊将棋編: 「四段の終盤」, 毎日コミュニケーションズ, ISBN: 4839905711, 2001.