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

| | |
|---|---|
| Title | コンピュータ将棋における指し手の順序付けによる探索効率化(<特集>ゲームプログラミング) |
| Author(s) | 竹歳，正史；橋本，剛；作田，誠；飯田，弘之 |
| Citation | 情報処理学会論文誌，43(10)：3074-3077 |
| Issue Date | 2002-10-15 |
| Type | Journal Article |
| Text version | publisher |
| URL | http://hdl.handle.net/10119/4573 |
| Rights | 社団法人　情報処理学会，竹歳 正史，橋本 剛，作田 誠，飯田 弘之，情報処理学会論文誌，43(10)，2002，3074-3077．　ここに掲載した著作物の利用に関する注意：本著作物の著作権は（社）情報処理学会に帰属します。本著作物は著作権者である情報処理学会の許可のもとに掲載するものです。ご利用に当たっては「著作権法」ならびに「情報処理学会倫理綱領」に従うことをお願いいたします。 Notice for the use of this material: The copyright of this material is retained by the Information Processing Society of Japan (IPSJ). This material is published on this web site with the agreement of the author (s) and the IPSJ. Please be complied with Copyright Law of Japan and the Code of Ethics of the IPSJ if any users wish to reproduce, make derivative work, distribute or make available to the public any part or whole thereof. All Rights Reserved, Copyright (C) Information Processing Society of Japan. |
| Description | |

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

テクニカルノート

# コンピュータ将棋における指し手の順序付けによる探索効率化

竹歳正史† 橋本剛†
作田誠† 飯田弘之†

本稿はコンピュータ将棋での指し手の順序付けによる探索効率化について述べる．コンピュータ将棋の領域では，以前から指し手の順序付けのための様々なアイデアが用いられてきた．しかし，順序付けのための各方式が探索効率化にどの程度貢献しているのか明らかでなかった．本研究では，代表的な指し手の順序付け方式を将棋プログラム上に実装し，標準テスト問題を探索する実験を行い，各方式の効果を比較評価した．

# Search Efficiency by Move Ordering Techniques in Computer Shogi

Masahumi Taketoshi,† Tsuyoshi Hashimoto,† Makoto Sakuta† and Hiroyuki Iida†

This note describes several exepriments performed to evaluate the search efficiency using move ordering techniques in the domain of computer shogi. We note that these move ordering techniques have been used by the strongest shogi programs but without the detailed outcome of these techniques. We show and evaluate the exprimental performance with the move ordering techniques using a test set of shogi problems.

## 1. はじめに

$\alpha\beta$ 法とその変形である反復深化法は，良い手から展開すると探索の効率が高まる[6]．コンピュータチェスで知られている最も単純な効率化のアイデアは，駒を取る手（capturing）を指し手リストの最初か前方に持ってくる指し手の順序付け（move ordering）である[5]．また，取る駒を価値の高い順番に並べ替えることでも良い結果が得られる[2]．取る駒の価値にかかわらず，相手が直前に動かした駒を取る手も有力な指し手の候補である[5]．これらのアイデアはコンピュータ将棋においても同じように効果があるとされてきた．

しかし，コンピュータ将棋の領域では，チェスで有効とされてきた探索効率化の有効性に関する詳細なデータが知られていない．どのアイデアがどれくらい効果があるか，あるいは，それぞれのアイデアを組み合わせることで得られる相乗効果など明らかではない．本研究では，将棋プログラム TACOS を使用して，これまでよく知られている代表的な探索効率化のアイデアを実装し，評価実験を行いその結果を分析する．

† 静岡大学
Department of Computer Science, Shizuoka University

## 2. 指し手の生成

コンピュータチェスでは全幅探索によるいわゆるしらみつぶし的な方法が成功した（たとえば，文献 1) 参照）．しかし，将棋は持駒再使用ルールのため，中盤から終盤にかけて指し手の数が激増するため，全幅探索型の手法は非実用的である．そのため前向き枝刈りのような選択的探索がコンピュータ将棋では実用的な手法としてとられてきた．このアイデアは，チェスの開発初期にみられた代表的な手法である[3),8)]．このとき好手になる可能性が高い有望そうな手から順番に生成することで探索の効率化をはかることができる．有望そうな手を効率良く生成するためには様々な方法が考えられるが，本稿では将棋というゲームに依存する方法とそうでない方法に大別し，トップレベルの有名将棋ソフトに使用されている手法[9),10)] を比較検討する．

(1) ゲームに依存しない方法
（*The domain-independent criteria*）
(I) ハッシュ表に登録された最善手[5)]
（*best move*）

反復深化法の過程によって得られた最善手を使用する．ここでいう最善手とは前回探索時における最善手である．

（II） キラー手[3]（*killer move*）
チェスのような複雑なゲームの探索木では，兄弟ノードは似た局面になることが多いので，それに対する最善手も同じになる可能性が高い．キラー手として枝刈りを多くおこした指し手を登録する方法[7]や，枝刈りを起こした回数を記録して並べ替えに利用するヒストリーヒューリスティクス[4]という手法もあるが，ここでは兄弟ノードでの最善手をキラー手として使用する．

（2） ゲームに依存する方法（*heuristics*）

（III）直前に指された駒を取る手[5]（*just capture*）
相手が直前に指した駒をすぐに取る手．その駒が取れないときは生成しない．

（IV）最も価値の高い駒を取る手[5]（*highest capture*）
取ることが可能な駒の中で最も価値の高い駒を 1 つだけ取る手．取れる駒がないときは生成しない．

（V） 最も価値の高い駒が逃げる手[5]（*escape move*）
取られそうな駒の中で最も価値の高い駒が逃げる手．本稿では，相手駒によって当たりになっている味方駒の可能な動きをすべて逃げる手とした．

## 3. 実　　験

将棋プログラム TACOS に深さを閾値とした反復深化法を実装し，$\alpha\beta$ 法による全幅探索で深さ 4 まで探索を行う．評価関数は最もよく知られている山下の手法[9]を元に作成した．

### 3.1 有望そうな手の比較

探索効率化を定量的に評価するために，指し手の生成手法の組合せにより探索したノード数の増減を比較する．目的を持った指し手を順に生成して探索した後は，残りの合法手をすべて生成する．すべての合法手について，仮評価によるソーティングなどは行っていない．実験にはコンピュータ将棋用の問題として知られている文献 9) に収録されている問題局面すべてで探索を行った．

比較する指し手の組合せと実験結果を**表 1** に示す．ここで ratio は何も実装しない場合のノード数との比を表している．

また表 1 のグラフを**図 1** に示す．

## 4. 考　　察

### 4.1 ゲームに依存しない方法

ハッシュ表に登録された最善手（I）とキラー手（II）をそれぞれ単体で実装した場合を比べると，意外にもキラー手は最善手の 4 分の 1 のノード数になりかなり効果が大きいことが分かる．理由としては，探索の行われていない末端では最善手がハッシュ表に登録されていないのに対して，キラー手は兄弟ノードの探索が 1 つ以上済んでいるすべての探索ノードで適用されるため，全体としての効率化の度合いが大きいと考えられる．この 2 つを組み合わせるとキラー手単体に比べてノード数はさらに約 53%に減少しかなり効率が上がっている．

### 4.2 ゲームに依存する方法

直前に指された駒を取る手（III）と最も価値の高い駒を取る手（IV）はそれぞれ文献 10) で棚瀬と金沢が示している似た意味を持つ手法であるが，単体で実装した場合を比べると最も価値の高い駒を取る手の方が有効であった．両方の駒を取る手を組み合わせてみると（III，IV）探索の効率は落ちることがなく，わずかにノード数が減少した．これに最も価値の高い駒が逃げる手（V）を加えると，やはりわずかではあるがノード数が減少し，何も実装しないバージョンに比べ約 97%効率化できた．

### 4.3 すべてを実装

最後にすべてを実装したバージョン（I, II, III, IV, V）を見ると,（I, II）だけのバージョンに比べてさらに約 21%の効率化に成功しており，これらを組み合わせることがより効果的であることを示している．

## 5. ま と め

本研究ではハッシュ表に登録された最善手，キラー手，有望そうな指し手の生成を実装し，その効果を評価した．これらの方法は単体で使用するより，それぞれ組み合わせることでより高い探索の効率化が可能である．これらを実装していく過程で，どこでコストと効率化のバランスがとられるのかという分岐点が判明すれば，今後将棋プログラムを作成し改良していく際に大きな指標となるだろう．

**表 1** 指し手の組合せと結果
Table 1 The combination of moves and results.

| methods | Nodes average | ratio |
|---|---|---|
| 何も実装しない（none） | 7625436 | 1.000 |
| *The domain-independent criteria* | | |
| ハッシュ表に登録された最善手（I） | 1413184 | 0.185 |
| キラー手（II） | 303338 | 0.040 |
| ハッシュ表に登録された最善手，キラー手（I,II） | 161100 | 0.021 |
| *heuristics* | | |
| 直前に指された駒を取る手（III） | 3583041 | 0.470 |
| 最も価値の高い駒を取る手（IV） | 282011 | 0.037 |
| 直前に指された駒を取る手，最も価値の高い駒を取る手（III,IV） | 263420 | 0.035 |
| 上記の駒を取る手に最も価値の高い駒が逃げる手を加えたもの（III,IV,V） | 251934 | 0.033 |
| *all* | | |
| すべて実装（I,II,III,IV,V） | 127152 | 0.017 |

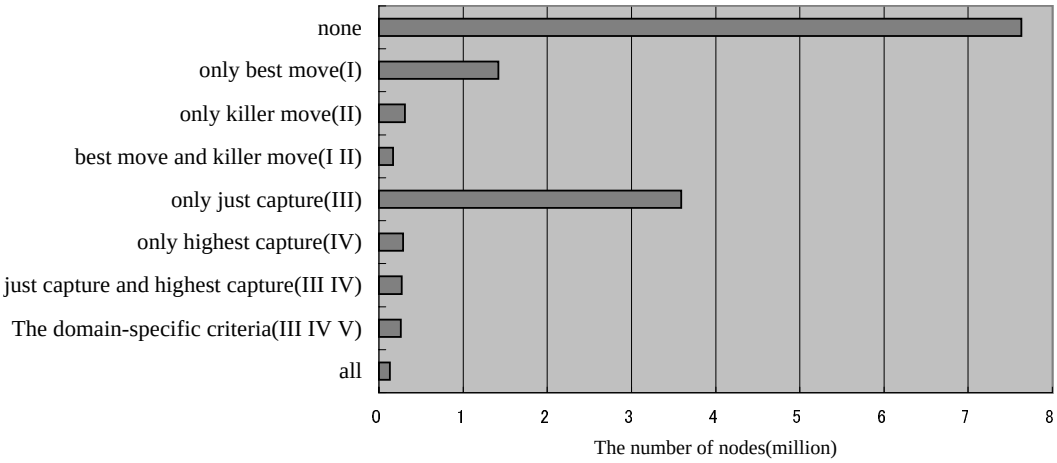


**図 1** 各手法を用いた場合のノード数の比較
Fig. 1 Number of nodes for each method

## 参 考 文 献


1) Campbell, M., Hoane, A.J. and H.F-H.: Deep Blue, *Artificial Intelligence*, Vol.134, No.1-2, pp.57–83 (2002).
2) Bettadapur, P.: Influence of ordering on capture search, *ICCA Journal*, Vol.9, No.4, pp.180–188 (1986).
3) Gillogly, J.J.: The Technology Chess Program, *Artificial Intelligence*, Vol.3, pp.145–163 (1972).
4) Schaeffer, J.: The History Heuristic and the Performance of Alpha-Beta Enhancements, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.11, No.11, pp.1203–1212 (1989).
5) Heinz, E.A.: Scalable Search in Computer Chess: Algorithmic Enhancements and Experiments at High Search Depths, Ph.D thesis, Department of Computer Science, University of Karlsruhe, published from Computational Intelligence, ISBN 3-528-05732-7.
6) Knuth, D.E. and Moore, R.W.: An Analysis of Alpha-Beta Pruning, *Artificial Intelligence*, Vol.6, No.4, pp.293–326 (1975).
7) Levy, D.N.L. and Newborn, M.: *How Computers Play Chess*, Computer Science Press, New York (1991). 小谷善行（監訳）：コンピュータチェス，サイエンス社 (1994).
8) Slate, D.J. and Atkin, L.R.: Chess 4.5 — The Northwestern University chess program, *Chess Skill in Man and Machine*, 2nd ed., Frey, P.W. (Ed.), pp.82–118, Springer, (1983).
9) 松原　仁（編著）：コンピュータ将棋の進歩 2, 共立出版 (1998).
10) 松原　仁（編著）：コンピュータ将棋の進歩 3, 共立出版 (2000).


(平成 14 年 2 月 21 日受付)
(平成 14 年 9 月 5 日採録)

**竹歳　正史**

2001 年静岡大学情報学部卒業．現在，静岡大学大学院情報学研究科．

**作田　　誠**

2001 年静岡大学大学院博士後期課程修了．博士（工学）．現在，同大学院研究生．

**橋本　　剛**

1994 年京都大学農学部卒業．1996 年東京大学大学院理学系研究科在学中に中国雲南民族学院へ留学．1997 年東京大学を中途退学し，台湾文化大学に留学．1998 年静岡大学大学院理工学研究科博士後期課程入学．2002 年同修了．博士（工学）．現在，学術振興会特別研究員として静岡大学工学部システム工学科に在籍．主にコンピュータ将棋等ゲームの研究，開発や生物進化のモデリングに取り組んでいる．

**飯田　弘之**（正会員）

1962 年生．1975 年大内延介九段に師事．将棋プロ棋士六段．1994 年東京農工大学大学院博士後期課程修了．博士（工学）．オランダリンブルグ大学コンピュータサイエンス学科客員研究員，科学技術振興事業団・博士研究員（電子技術総合研究所勤務），オランダマーストリヒト大学客員教授等．現在，静岡大学情報学部助教授．ゲーム情報学，環境問題等の数理モデリング等に興味を持つ．