

ハッシュテーブルを用いた
コンピュータ将棋の探索の効率化

1 R-7

細江正樹, 久保田聡, 瀬野訓啓, 飯田弘之, 小谷善行
(東京農工大学 工学部 数理情報工学科)

1. はじめに

Sun ワークステーション上の Quintus-Prolog を用いて将棋プレーシステム「shouchan」を作成し、現在も研究を行っている。

現在、局面の探索には $\alpha\beta$ 法を用いて行っているが、十分な探索が行われているとは言えない。実力を上げるには、探索の深さを深くし、たくさんの局面を読むことが大切である。将棋やチェスなどの二人零和完全情報ゲームでは、局面の探索時においては手の生成よりも静的評価の方に多くの時間を費やす。そのため、さらに何らかの知識を用いて不必要な局面を探索しないようにする工夫が必要である。

2. 効率化の方法

「shouchan」が探索中に生成するゲーム木を調べると、局面の探索中に同一局面が出現するときがある。その際に同じ局面を再度評価し直すのは無駄である。そこで同じ局面が出現したときに再評価し直さないために、ハッシュテーブルに、一度探索した局面の情報を格納し、同一局面が出現したときにハッシュテーブルの情報を利用することで探索の高速化をはかることができる。

まず、探索中に新しい局面に到達するとハッシュテーブルにその局面の情報があるかどうかを調べる。もし十分な情報があれば、その局面以下の探索の必要はない。その局面を末端局面としてハッシュテーブルから評価値を取り出し、その局面の探索は終了する。もし情報がないか充分ではない場合は、新しいエントリを作って情報を保存する。

評価値計算を省略することによって、少ない時間で局面の探索をすることができる。

ハッシュ値は、将棋盤上の四十枚の駒の位置(盤上あるいは持駒)をキーとし、Piece-Squareテーブル[2]を用いて計算している。この方法を用いた理由は、局面が進んだり戻ったりしたときにハッシュ値の再計算が簡単に行えることである。

2.1 Piece-Square テーブル

Piece-Square テーブルは、81の将棋盤の各マスと持駒に対して、成駒を含めて14種類(王将・飛車・角行・金将・銀将・桂馬・香車・歩兵・龍・馬・成銀・成桂・成香・と金)の各駒ごとに、乱数を記録したテーブルである。先手と後手で二通りあるので、全ての駒の、位置と種類をあらわすのに必要なパターン数は2272個になる。

Piece-Square テーブルは次のような Prolog の述語であらわされる。

```
ps_table (駒種,座標,所有者,状態,乱数値).
駒種:'OU', 'HI', 'KA', 'KI', 'GI', 'KE',
     'KY', 'FU'
座標:11 ~ 99、持駒の場合はその数
所有者:先手 (sente) または後手 (gote)
状態:生 (nama)、成り (nari)、持駒 (mochi)
乱数値は1 ~ 32767の値をとる。
```

2.2 局面のハッシュ値の計算

次に、局面のハッシュ値の計算方法について述べる。40枚の各駒に対して、Piece-Square テーブルから乱数値を求める。求められた40個の乱数値に対して排他的論理和(+で表す)の計算をすることによってハッシュ値を求める。

例として、図1の局面を元にして実際に局面のハッシュ値を求めてみる。

	5	4	3	2	1	
				王		一■
			将			二持
			銀			三駒
				歩		四金
						五

図1 例題局面

図1の局面におけるハッシュ値は次のようになる。

- 2 二王 2 6 7 7 7
- (+) □ 3 三桂 3 1 5 1 6
- (+) ■ 3 四銀 8 8 0 5
- (+) ■ 2 五歩 1 8 8 7 6
- (+) ■ 持駒 金1枚 2 3 1 8 2

8 9 1 4

2.3 局面変更にともなうハッシュ値の変更の方法

現在局面のハッシュ値に対して、移動元の駒の乱数値と移動先の駒の乱数値の二回の排他的論理和をとることで新しい局面のハッシュ値を計算することができる。駒を取ったり持ち駒を打ったりする手の場合は、取られたり打ったりした駒に対して同様の処理をすれば良い。

この方法を用いれば、新しいハッシュ値を求めるのに少なくとも二回の排他的論理和の計算をするだけでよく、非常に簡単に行うことができる。

2.3.1 局面変更にともなうハッシュ値の変更

将棋の指し手には、進む手・駒を取る手・持ち駒を打つ手の三種類の手がある。実際の局面変更のアルゴリズムを図1の局面をもとに説明する。

2.3.1.1 進む手

図1の局面で▲2四歩という手を指す場合を考えてみる。

単純に進むだけの手の場合は、現在局面のハッシュ値に対して

(1) 移動元の駒に対する乱数値

(2) 移動先の駒に対する乱数値

の二回の排他的論理和の計算で求めることができる。

図1の局面で▲2四歩という手を指す場合のハッシュ値は、

現在局面のハッシュ値	8 9 1 4
(+) ▲2五歩	1 8 8 7 6
(+) ▲2四歩	3 8 0 7

2 6 0 3 3

になる。

2.3.1.2 駒を取る手

次に図1の局面で▲3三銀成という手によって駒を取る手を指す場合を考えてみる。

駒を取る場合は、進んだ駒だけではなく取られた駒に対する処理が必要になる。現在局面のハッシュ値に対して、

(1) 移動元の駒に対する乱数値

(2) 移動先の駒に対する乱数値

(3) 移動先にある取られる駒に対する乱数値

(4) 取る駒の駒種を持駒に持っている枚数に対する乱数値(無ければ0)

(5) (4)で求めた枚数+1) 枚の持駒に対する乱数値

の五回の排他的論理和の計算で求めることができる。

図1の局面で▲3三銀成という手を指す場合のハッシュ値は、

現在局面のハッシュ値	8 9 1 4
(+) ▲3四銀	8 8 0 5
(+) ▲3三銀成	6 4 0 7
(+) □3三桂	3 1 5 1 6
(+) ▲持ち駒 桂0枚	0
(+) ▲持ち駒 桂1枚	7 4 1 6

3 2 3 4 0

になる。

2.3.1.3 持駒を打つ手

最後に図1の局面で▲2三金という手を指す場合を考えてみる。持ち駒を打つ手なので、通常の処理に加えて手を指す側の持ち駒の処理をする必要がある。現在局面のハッシュ値に対して、

(1) 打つ駒の駒種を持駒に持っている枚数に対する乱数値

(2) 移動先の駒に対する乱数値

(3) (1)で求めた枚数-1) 枚の持駒に対する乱数値(無ければ0)

の三回の排他的論理和の計算で求めることができる。

図1の局面で▲2三金という手を指す場合のハッシュ値は、

現在局面のハッシュ値	8 9 1 4
(+) ▲持ち駒 金1枚	2 3 1 8 2
(+) ▲2三金	3 0 4 5 3
(+) ▲持ち駒 金0枚	0

3 7 5 3

になる。

2.4 ハッシュテーブル(転換表)

探索中の局面の情報を保存しておくハッシュテーブルは、次のような Prolog の述語で表される。

h_table (ハッシュ値, 評価値, 探索の深さ, 手番),
ハッシュ値:局面のハッシュ値

評価値:ハッシュ値で表される局面の評価値

探索の深さ:探索開始局面からの深さ

手番:Piece-Squareテーブルの手番と同様

ハッシュテーブルのエントリはPiece-Squareテーブルの乱数値の範囲と同じである。ハッシュテーブルは局面の探索が始まると初期化され、探索中の局面の評価値が計算されると次々にassertされる。

2.5 同一局面が末端局面になるための条件

探索中に同一局面が見つかった場合、次の条件を満たしていれば末端局面とみなし、ハッシュテーブルの情報を利用することによって、その局面以下の探索を省略することができる。

・手番が同じ。

・開始局面からの深さが以前出現した局面の深さと同じかそれよりも深い。

条件を満たさなかった場合は、ハッシュテーブルの情報が不十分ということで、探索をやり直さなければならない。

3.まとめ

本システムで実際に対局させてみた結果、定跡が切れた直後や駒組みの段階では同一局面の出現はあまり見られなかったが、中盤から終盤にかけての取り合いの多い局面で同一局面が多く出現し、探索時間を従来のものより最高で約1割ほど短縮することができた。

参考文献

[1]実近憲昭:ゲームと探索, 電子通信学会誌, Vol.65 No.4, 1982, pp.405-412

[2]Levy, Newborn:How Computer Play Chess, COMPUTER SCIENCE PRESS, pp.153-190, 1991