

詰将棋における並列探索法の一考察

3E-9

和田 誠一郎 立松 靖朗 山田 雅之 世木 博久 伊藤 英則
名古屋工業大学

1 はじめに

人工知能の分野で計算機上に知能を実現しようとする試みが盛んに行なわれてきている。しかし、人間の全知能を一度に実現することは不可能に近い「限定した世界」をその対象としている。このような背景で従来からゲームにその世界を限定しその問題解決手法についても盛んに試みられてきた。ここでは、ゲームとして「詰将棋」を取り上げ、並列計算機 AP1000 上にこれを解くプログラムを実現し、その評価について述べる。

2 探索問題としての詰将棋

詰将棋とは、攻方が王手をかけ続け、それに対し受方が逃げる手順を考えていくというパズルゲームの一種である。詰将棋を解くとは、双方が最善をつくり、最長手数で詰みにいたる手順を示すことである。

詰将棋の探索問題は、攻方を OR ノード、受方を AND ノードとすることによる、AND/OR 木の探索問題として一般化できる (図 1)。

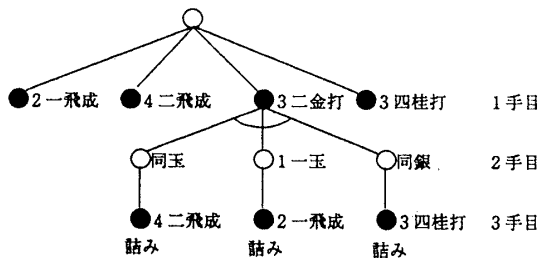


図 1: ゲーム木

詰将棋では、爆発的に広がっていく探索空間をいかに狭めるかが問題になってくる。通常は、評価関数を用いた枝刈りなどを用いて探索すべきノードを絞り込んでいく。しかしながら、このような方法では、評価関数によって解答能力が大きく左右される。また、あらゆる局面に対して有効な優れた評価関数を作ることは実際には非常に困難である。

A study of consideration of a parallel search method in Tsume-shogi game tree
Seiichirou WADA, Yasuo TATEMATSU, Masashi YAMADA, Hirohisa SEKI and Hidenori ITOH.
Nagoya Institute of Technology.
Gokiso-cho, Showa-ku, Nagoya 466, Japan

3 評価関数と並列探索法

3.1 即詰を考慮した評価関数

評価関数には、王手に対する受手の総数（自由度）をその王手の評価値として、この値が小さいほど、詰みに至る可能性が大きくなると判断する。さらに、即詰になりそうな受手はこの数に含めないこととする。例えば、図 2 に示すような局面において、玉は王手 2 三銀に対して 5 つの場所に逃げるのが可能であるため、評価値は 5 となるが、持駒に金があるため D 以外の場所に逃げた場合には、即詰になってしまう。このような場合、評価値を 5 ではなく 1 とする。

その他、桂馬を早く使うようにしたり、合駒の展開は他の受手ノードを展開して、詰みと判断された後にするなどといった工夫もした。

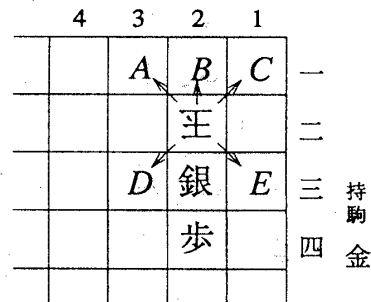


図 2: 即詰を考慮した評価

3.2 ゲーム木全体のノードの評価

ある局面で、複数の王手の各々の評価値を 3.1 に述べた方法で求めた後、MAX/MIN 法を用いてゲーム木全体のノードの評価値を求める [2]。MAX/MIN 法では、ノードの評価値を下位から順に次のように計算する (図 3)。

受方は攻方にとって最も不利な手を打つことより、王手の評価値は一つ下の受手ノードの中で評価の最も大きい値とする。(図 3: 1 手目 参照)

攻方は受方にとって最も有利な手を打つことより、受手の評価値は一つ下の王手ノードの中で評価の最も小さい値とする。(図 3: 2 手目 参照)

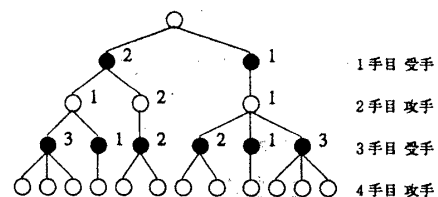


図 3: MAX/MIN 法に基づくノードの評価

3.3 並列優先探索

ここでは、ゲーム木全体のノードを3.2で述べた方法で評価した後、詰みに至る手順を並列に探索する。並列度 n で探索する場合は、評価の良い順に最大 n 個の王手葉ノード以降の探索をホストからセルと呼ばれる各プロセッサに対して要求する、BFSP[1]を用いた(図4)。

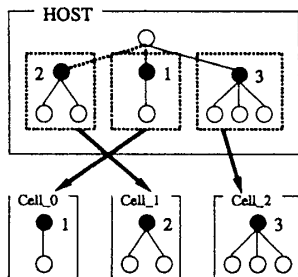


図4: ホストからセルへの探索要求

しかし、この方法では、セル数が多くなるにつれてセル稼働率が低下して並列効果が期待できない。そこで、並列効果を上げるために、並列度 n に対して王手葉ノードの数が少ない場合、ホストにおいて新たにその下の王手葉ノードを展開し、セルに探索を要求する方法を取った。例えば、図5のような探索木を考える。文献[1]の探索方法では最初に要求する探索の数は2つとなるが、本システムの方法では、評価値の良い、つまり図中の右側の王手葉ノードを追加探索することにより、要求する数が6つとなるため、並列効果が期待できる。もし、それでも探索を要求する王手葉ノードがセルの並列度 n に満たない場合は、 n を満たすまで王手葉ノードを展開してから探索要求する。

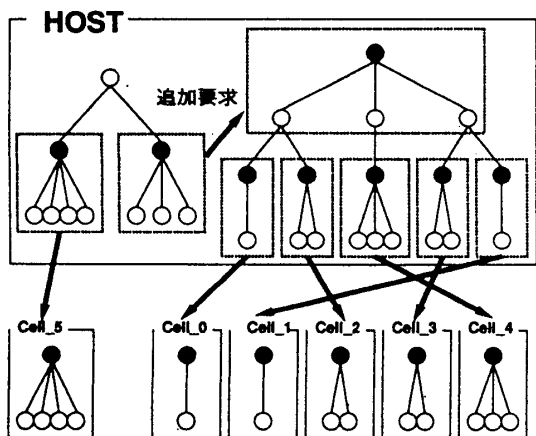


図5: 王手葉ノードの再展開

次にセルは、3.1、3.2で述べた評価関数を用い、横型の最良優先探索を行う。セルは、

1. 王手以下が詰みであると判断できるまで、
2. 王手以下が逃れ(不詰)であると判断できるまで、

3. 生成部分木の大きさが定められた量に達するまで、
 4. ホストから処理中断のメッセージが到着するまで、
- のいずれかの条件を満たすまで処理を続ける。1,3の場合は、部分木を、ホストにある全体木に連結する。

4 結果

「続・詰むや詰まざるや」第46番55手詰を文献[1]において並列度16で解いた時のセル稼働率の推移(図6)と、本システムを用いた時のセル稼働率の推移(図7)である。



図6: 文献[1]のセル稼働率の推移



図7: 本システムのセル稼働率の推移

このグラフは面積が大きくなればなるほど、セルの利用率が高くなることを示し、セル稼働率は文献[1]63%から本システム74%までの向上が見られた。また、これにより内藤国雄の「九級から一級までの詰将棋」において探索時間が79秒から9%の向上が見られた。

5 おわりに

BFSPを拡張した本システムにおいて、文献[1]に比べセル稼働率の向上、また処理時間の短縮が実測された。また、「続・詰むや詰まざるや」において解答率が7%向上した。今後、さらなる解答率の向上を図ってきたい。

参考文献

- [1] 笠田 洋和, 松波 功二, 世木 博久, 伊藤 英則, '詰将棋におけるゲーム木の並列探索とその評価', 情報処理学会論文誌, vol.36, No.11, pp.2531-2539(1995)
- [2] 小谷 善行, 吉川 竹四郎, 柿木 義一, 森田 和郎, 'コンピュータ将棋', サイエンス社, 1990, ISBN 4-7819-0581-1
- [3] 内藤 国雄, '九級から一級までの詰将棋', 成美堂出版, 1992, ISBN4-415-04647-9
- [4] 門脇 芳雄, '続・詰むや詰まざるや', 平凡社, 1978, ISBN4-582-80335-0