

## FPGA を用いた将棋の高速計算の実現

1 H-4 関 峰伸(筑波大学), 丸山勉(同), 星野力(同), Reijer Grimbergen(電子技術総合研究所)

### 1 はじめに

チェスにおいてはコンピュータが人間のチャンピオンに勝つところまで来たが、将棋においてはまだまだ人間には及ばない。これは将棋はチェスに比べて取った駒を再び使うことができるので、探索量が膨大となり、現在のコンピュータでは先を十分に読む計算能力がないためである。今回、コンピュータ将棋の計算をハードウェア化することにより、計算を並列化、パイプライン処理し、高速化をめざす。将棋の計算方法は未だ発展途上であり、そのアルゴリズムの変化に対応するために、ハードウェア化には書換え可能デバイスであるFPGAを用いた。

### 2 Field Programmable Gate Array

Field Programmable Gate Array(FPGA)は、内部の回路を書換え可能なゲートアレイである。その書き換えは汎用計算機上で、ハードウェア記述言語、CADを用いて設計し、コンパイルし、その構成データをFPGAにダウンロードするだけである。

### 3 将棋の計算方法

将棋のような二人零和完全情報確定ゲーム（他にチェス、オセロ等）は一般にゲームの木を作って次の指し手を探索する。図1はゲームの探索木である。最も上の節点が現在自分の局面であり、そこから出ている枝(a,b,c)がそれぞれ自分が可能な指し手を意味している。その先2段目の節点が次の相手の局面である。そこから出ている枝はその局面での相手の可能な指し手である。これを繰り返して木は順々深くなっていく。図1では2手先まで読んでいる。葉にある数値はその局面をそれぞれ数値で評価した値である。将棋の計算では、この探索木をもとにミニマックス法と呼ばれる手法を用いて最善の指し手を決定する。

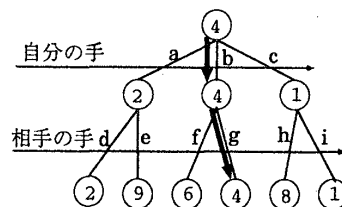


図 1: ゲームの探索木

### 4 パターンマッチングによる候補手生成

我々の将棋プログラムでは、探索木の各枝ごとにパターンマッチングによる候補手の生成を行う。その方法は、局面の状態を表すよう25種類の条件を組み合わせたパターン約3000種類を用い、ある局面とのパターンマッチングを行いマッチしたら、そのパターンに対応する候補手が生成するというものである。図2はその1例で、探索位置を(0,0)とし、E(10),FA(20=0),!JX(20),Ox=(2),E(2-1),E(2-2)の6つの条件で表されたパターンで、マッチした場合R20;飛車を(2,0)に進めるという候補手を作る。

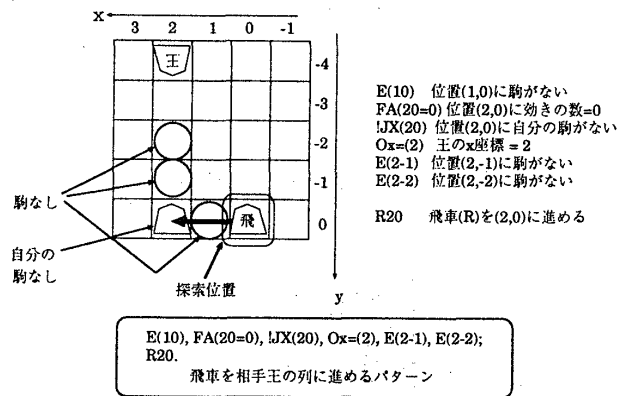


図 2: パターンの例

### 5 パターンマッチング計算のハードウェア化

今回、上に説明したパターンマッチングによる手の生成の部分の計算のハードウェア化を行った。

#### 5.1 パターンの条件を表すデータ構造

ハードウェアでは3に示すように、25種類の条件を用いて構成されているパターンを、いくつかまとめて

High speed computation of shogi with FPGA

Minenobu Seki, Tsutomu Maruyama,

Tsutomu Hoshino & Reijer Grmbergen

University of Tsukuba & Electrotechnical Laboratory

判別できる回路モジュールを10個作成した。そして、その回路モジュールの番号とその条件を判定するのに必要なパラメーターを持つ条件データを組み合わせることによりパターンを構成した。

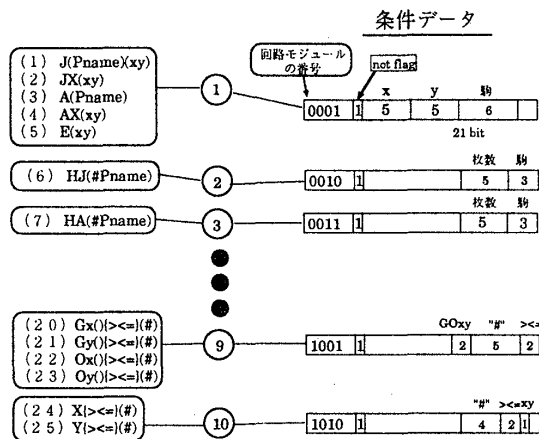


図 3: パターンの条件を表すデータ構造

### 5.2 パターンマッチング処理の流れ (パイプライン構造)

図4に示すように、条件データは check 回路で回路モジュール番号により、並列に配置された10個の回路モジュール群に振り分けられる。そして、各回路モジュールは局面の状態と条件データのパラメータを比較することにより、パターンマッチングを行う。各回路モジュールはパイプライン構造になっていて、1clock に一個の条件を判別できる。

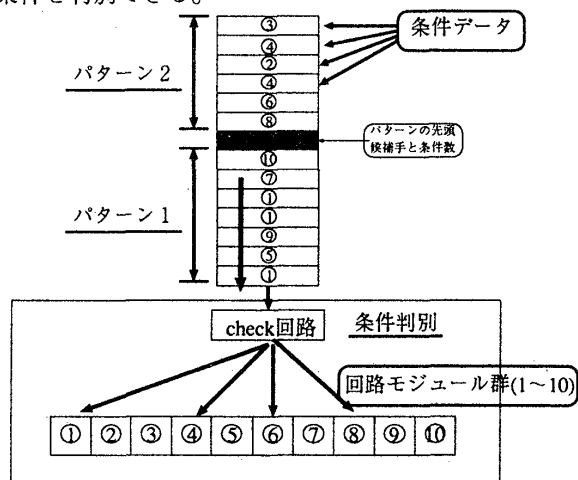


図 4: 回路の構成

### 5.3 並列化

回路モジュール群をその使用率が同程度になるように2分割にし、回路に流す条件データをそれに合わせて

分割し並列にパイプラインに入れる。このことにより回路の使用率を1.54倍にすることができ、全体の処理クロック数を65% (1.54倍の速度向上) に減らすことができた。また同様に、3分割にした場合には44.7% (2.2倍の速度向上) に減らすことができた。

また、1つのFPGA内にパイプラインをX本構成することにより、X倍の速度向上。そして、さらにそのFPGAをY個用いて構成することによりX×Y倍の速度向上が実現できる。

## 6 結果と考察

1局面のパターンマッチングの計算時間をPCと比較した結果を表1に示す。PCはPentium2-400MHzを使用した時の平均値である。FPGAは半分が自分の駒、20%が持ち駒になってる局面の計算をALTERAの100Kゲート相当のチップで、動作速度80MHz、パイプラインを2分割し3ライン構成した時の値である。パイプライン1本のみときは6.71倍の速度向上となる。また、ロジックセルをRAMとして使用できるXILINXのFPGAの場合は、同じゲート数のFPGAを使ってさらに速度向上が可能となる。

表 1: 計算時間

	計算時間	速度比
FPGA(1chip)	27.4us	31
PC	850us	1

## 7 おわりに

今回、FPGAを用いて将棋プログラムにおける候補手の生成を計算することによりPCと比べて高速化できることを示した。また、回路をスケラブルに設計することにより、さらなる高速化ができることを示した。今後は最新の計算方法を取り入れつつ、将棋プログラムを構成する他の計算もFPGAで計算させる予定である。

## 参考文献

[1] 松原 仁 編, "コンピュータ将棋の進歩", 共立出版.  
 [2] Reijer Grimbergen, Hitoshi Matsubara, "Pattern Recognition for Candidate Generation in the Game of Shogi", IJCAI-97.