

16PE並列計算機「樹葉」における 最大値問題のサイクル数による評価

4L-5

吉川秀之 望月建治 岩崎一彦

千葉大学工学部

1. まえがき

並列アーキテクチャの研究を並列アルゴリズムの研究と関連づけながらおこなうことを目的として、16PEマシン「樹葉」を試作した。従来、並列アルゴリズムの評価は、プログラムステップ数のオーダーで論じられることが多かった。本稿では、最大値問題について実機上でのサイクル数による評価をおこなっている。

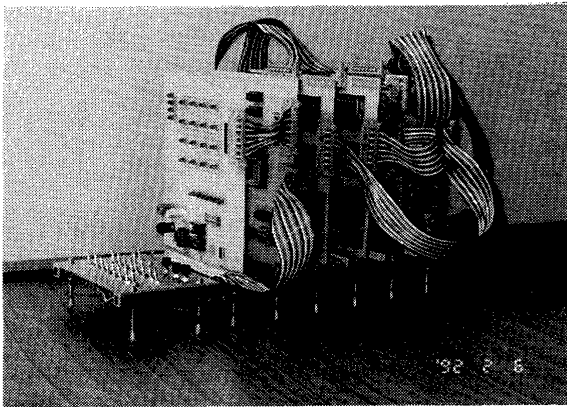


図1. 「樹葉」の外観

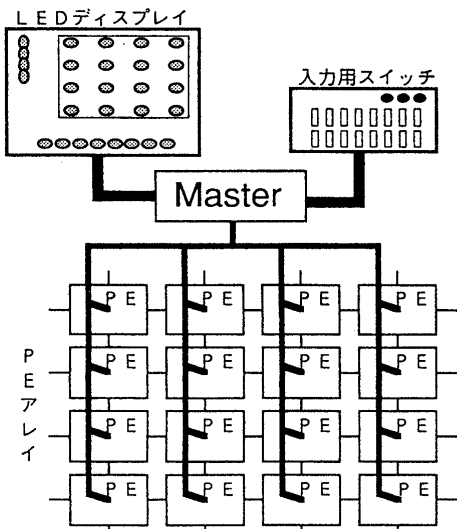


図2. 「樹葉」のアーキテクチャ

2. 16PE並列計算機「樹葉」のアーキテクチャ

並列計算機「樹葉」は、4×4のトラス状に並べられた16台のPEを持つマシンである[1]。各PEには日立製作所シングルチップマイコンH8/330を用いた。アーキテクチャはSPMDあるいはMIMDに分類される。通信方法として、隣接通信、シリアル通信、バス通信の3種類を用意した。図1に「樹葉」の外観を示す。図2は「樹葉」のアーキテクチャの概念図である。16個のPE、PE制御のためのマスタプロセッサ、出力用ディスプレイ、入力用スイッチ群から構成される。

3. 「樹葉」上での最大値問題の評価

3. 1. 隣接通信のみを使ったアルゴリズム

隣接通信のみを使った最大値問題のアルゴリズムをA1とする。N台のPEアレイでは、(通信+比較)の回数は $N^{1/2}$ 回となる[2]。時間量のオーダーは

$$O(N^{1/2})$$

である。A1の総サイクル数を $f_1(N)$ で表すと、

$$f_1(N) = \{ \lfloor w_a/w_r \rfloor \times (T_r + S) + C \} \times N^{1/2}$$

となる。ただし、データのビット幅を w_a 、隣接通信のビット幅を w_r 、隣接通信、同期、比較等にかかるサイクル数をそれぞれ T_r 、 S 、 C としている。「樹葉」をもとに各パラメータを評価する。データ幅は $w_a = 8$ ビットとしてプログラムをおこなった。隣接通信のビット幅 w_r は8である。1回の隣接通信、同期、比較等にかかるサイクル数 T_r 、 S 、 C の値は、「樹葉」上のプログラムから求めると $T_r = 30$ 、 $S = 130$ 、 $C = 30$ となった。以上から

$$f_1(N) = 190 \times N^{1/2}$$

となる。「樹葉」のPE数は16なので $N = 16$ を上式の代入すると、 $f_1(16) = 760$ となる。実際のプログラムのサイクル数は738であったので、近似的に等しい。

3. 2. 隣接通信とバス通信を使ったアルゴリズム

隣接通信とバス通信を使ったアルゴリズムをA2とする。このアルゴリズムでは、N台のPEを $M \times M$ 個のグループに分ける[2]。

N台のPEの場合、(通信+比較)の回数は、隣接通信が $N^{1/2}/M$ 回、バス通信が $M^2 - 1$ 回である。時間量のオーダーは

$$O(N^{1/3})$$

となる。A2の総サイクル数を $f_2(N)$ とすると、

$$f_2(N) = \{ \lfloor w_a/w_r \rfloor \times (T_r + S) + C \} \times N^{1/2}/M + \{ \lfloor w_a/w_b \rfloor \times (T_b + S) + C \} \times (M^2 - 1)$$

と表される。ただし、バスのビット幅を w_b 、1回のバス通信にかかるサイクル数を T_b とする。「樹葉」における各パラメータの値は以下ようになる。バスのビット幅は $w_b = 2$ である。 T_b は実際のプログラムから求めると

Clock Cycle Evaluation of Finding Maximum on Parallel Computer "JUPPA" with 16 PE's

Hideyuki YOSHIKAWA, Kenji MOCHIZUKI, Kazuhiko IWASAKI

Chiba University

$T_s = 80$ である。これらのパラメータから $f_2(N)$ は

$$f_2(N) = 596 \times N^{1/3} - 870$$

となる。ただし、 M は最適とする。

アルゴリズムA2を「樹葉」に適用するため、 $M = 2$ としてプログラミングした。その結果、プログラムのサイクル数は2987となった。 $f_2(16)_{M=2}$ を計算すると $f_2(16) = 2990$ となり、よい近似となっている。

3. 3. サイクル数から見たA1, A2の評価

2つのアルゴリズムのサイクル数の内訳を図3に示す。これは実際のプログラムから求めた。どちらのアルゴリズムも通信とそれに伴う同期が占める割合が高い。

次に、図4に各アルゴリズムのサイクル数とPEアレイの大きさの関係を示す。図にはさらに、シングルプロセッサの場合、「樹葉」のアーキテクチャを改良した場合のA1', A2'のサイクル数A1', A2'も示す。改良点については事項で述べる。

アルゴリズムA2が隣接通信のみのアルゴリズムA1より処理時間が短くなるPEアレイの大きさNについて考える。これは、 $f_1(N) = f_2(N)_{M=2}$ を満たすNである。「樹葉」のアーキテクチャでは $N = 755$ となる。時間量のオーダーはA1が $O(N^{1/2})$ 、A2が $O(N^{1/3})$ であるが、A1とA2を比べると、PE数755台まではバスを使わない方が速い。その理由はアレイが小さい場合、離れたPEとも1回で通信できるバス通信の利点が発揮されないことや、「樹葉」のアーキテクチャではバス通信は隣接通信の2から4倍の時間がかかるためと考えられる。シングルプロセッサ(時間量 $O(N)$)とA1, A2を比較すると、PE数が約100台を越えると並列効果が現れることがわかる。

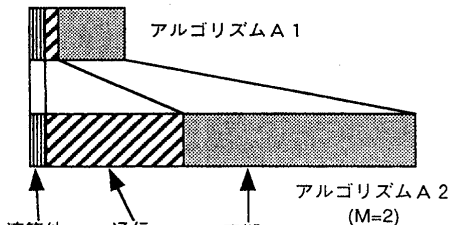


図3. 最大値問題アルゴリズムA1, A2のサイクル数の内訳

4. 「樹葉」のアーキテクチャの改良

「樹葉」のアーキテクチャの改良を考える。改良点として以下の2点を考えた。(1)バス幅を2ビットから8ビットに拡張する。 $w_b = 8$ となる。(2)同期をハードウェア化し、同期にかかる時間を短縮する。実機では $S = 130$ であるが、 $S = 3$ になると仮定した。改良後のアルゴリズムA1, A2をA1', A2'とする。また、それぞれの総サイクル数を $f_1'(N)$, $f_2'(N)$ とすると、

$$f_1'(N) = 60 \times N^{1/2}$$

$$f_2'(N) = 139 \times N^{1/3} - 110$$

となる。ともに実行時間が3分の1以上減っている。改良後も「樹葉」のアーキテクチャではPEが121台まではバス通信使用の効果はない。シングルプロセッサと比べた場合、このアーキテクチャの変更によりPE10台程度から並列効果が現れる。

上記以外の改良点として、ID番号比較の高速化が考

えられる。「樹葉」のプログラムはSPMDを意識して書かれているため、分岐時に自分のID番号の比較が何回かおこなわれる。これをID番号により分岐先が変わるジャンプ命令を作り高速化する。これにより、 $C = 30$ が $C = 20$ 程度に抑えられる。また、「樹葉」では隣接およびバス通信はポートに直接値を書き込むことで実現していた。隣接通信専用のハードウェアを追加することにより $T_s = 30$ が $T_s = 14$ 程度に、バス通信専用のハードウェアを追加することにより $T_s = 80$ が $T_s = 10$ 程度に改善できる。その結果、

$$f_1^*(N) = 37 \times N^{1/2}$$

$$f_2^*(N) = 67 \times N^{1/3} - 33$$

となる。この時、並列化の効果が見れるのはPE4台からと予想される。

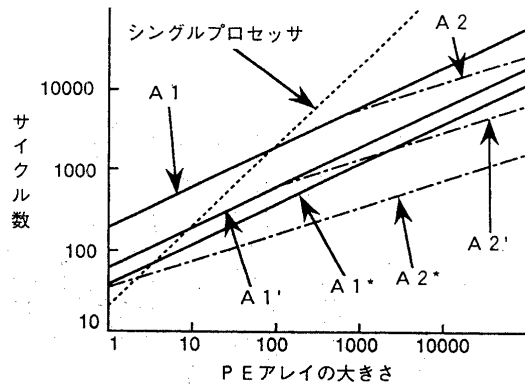


図4. 16PE並列計算機「樹葉」における最大値問題のPEアレイの大きさとクロックサイクル数の関係

5. まとめ

16PEを用いた並列計算機「樹葉」上での最大値問題を実際のサイクル数で評価した。その結果、アーキテクチャを評価する上でサイクルレベルのアルゴリズム評価が有用であることが再認識された。

今後の課題としては、単純多角形内の最短経路木問題や構文解析など、より複雑な問題の評価があげられる。最大値問題については、本稿ではバス転送に要する時間はコンスタントとした。PE数の関数と考えて評価することも今後の課題である。また、64PE並列計算機の試作にも取り組んでいく。

この分野は並列アーキテクチャ、並列アルゴリズム、フォールトトレランス、ソフトウェア、デバッグ、実装技術など多岐にわたる内容を含んでいる。そのため、かえって統合的な研究が少ないのではないかとと思われる。今後はこれらをグローバルに捕らえた研究が必要である。

謝辞 H8/330に関し御教示頂いた日立製作所赤尾泰主任技師、岩下裕之主任技師に感謝します。

文献 [1]吉川, 望月, 岩崎: "16PE並列計算機「樹葉」における最大値問題およびバイトニック・マージ・ソート", 信学技報, COMP92-7(1992). [2]梅尾博司: "超並列計算機アーキテクチャとそのアルゴリズム", 共立出版(1991).