

データ駆動計算機における 5C-6 静的負荷分散方式の検討

大塚 喜久 坂井 修一 弓場 敏嗣
(神戸製鋼所) (電子技術総合研究所)

1 はじめに

データ駆動計算機はプログラムの持つ命令レベルの並列性を自然に抽出できることが特徴である。しかし、基本プロセッサ(PE)の台数に対してプログラムの並列性が十分でない場合には、得られた並列性を各PEに効率よく分散する必要がある。

負荷分散は、実行時に行う動的なものと同様に実行以前にコンパイラによって行う静的なものに分けられる。データ駆動計算機では各命令を発火させるためのトークンの対はそれぞれ同じPEに向かわねばならないため、動的な負荷分散ではオーバーヘッドが大きくなる。また、動的な負荷分散では局所的な情報のみを利用するのに対し、静的な負荷分散ではプログラム全ての情報を利用することができる。これらの点から静的な負荷分散方式が有効である。

本問題と類似した問題としてマルチプロセッサシステムにおけるスケジューリング問題がある[1][2]。通常、スケジューリング問題では各タスクを実行するプロセッサと実行順序を決定する。しかし、データ駆動計算機では発火条件によりタスクに対応する命令の実行順序が動的に定まり、実行順序をスケジューラが指定できない。このために、これらのスケジューリング問題とは異なったアルゴリズムが必要となる。

本報告ではループを含むプログラムに対し、PE間の通信時間を考慮に入れたデータ駆動計算機の静的負荷分散方式について検討する。

2 問題の設定

次のデータ駆動計算機を対象とする。

- ① n台のPEがクロスバーネットワークで結合
 - ② 各PEは待ち合わせ機構、実行部を有する
 - ③ タグ付きトークン方式
 - ④ プログラムは全てのPEに重複して格納
- また、次の仮定を置く。
- ⑤ 各命令の実行時間は一定
 - ⑥ 異なるPE間のデータ転送時間は一定

3 特性値の定義

データフローグラフ上の各命令(ノード)の状態を次の3種類のレベルと1種類のフラグで表現する。ここで(高級言語で記述された)プログラムに含まれるループは1重であるとしたが、対応するデータフローグラフではループに属さないノードも存在する。

- ① E-LEVEL: 入口ノードから当該ノードを実行するための最短路長
- ② L-LEVEL1: 当該ノードから出口ノードを実行するための最短路長
- ③ L-LEVEL2: L-LEVEL1と同様
ただしループは1周した路長
- ④ LOOP-FLAG:
n: プログラム中のループに属さないノード

p: プログラム中のループに属するが
グラフ上のループには属さないノード
g: グラフ上のループに属するノード

これらを付けるのは $O(n^2)$ の計算量で可能である。
また、PE間の通信コストは各ノードに対し、次の条件を全て満たすアークの本数として定義する。

- ① 当該ノードへの入力アーク
- ② アーク両端のノードのE-LEVELの差が1
- ③ アーク両端のノードを処理するPEが異なる

4 負荷分散方式

4.1 基本方針

まずグラフにループを含まない場合について考える。負荷分散を行うためには並列実行可能なノード群に対して可能な限り多くのPEを割り付けねばならない。よって、上位のノードのPEへの割り付けが終了しているノード群に対し、できるだけ多くのPEを割り付けることを基本方針とする。

ここで、どのノードにどのPEを割り付けるかを決定する際に下記の判断基準が考えられる。

- T) 通信コストが最小
- E) E-LEVELが最小
- L) L-LEVELが最大

これらのある基準を適用し、複数のノードが選択された場合、さらに他の基準を適用することができる。適用順序によって6通りの組み合わせが存在する。上記の順序で適用する場合をTELと記述する。

4.2 ループ処理

ループの処理方法に関して、次の3点に着目した。

(1) L-LEVELの選択

ループが存在する場合、L-LEVEL1, 2の値は異なる。

(2) ループの区別

ループに属さないノードと属すノードが並列に処理できる場合、これらのノードを対等に割り付けるとループに属するノードに割り付けられたPEの負荷が重くなる。このためにループに属するノードは属さないノードと区別して割り付けを行う必要がある。

プログラムのループとグラフのループは異なるので、ループの区別方法として下記の3方法が考えられる。

L-N) ループを区別しない

L-G) LOOP-FLAG がgのノードをループとする

L-P) LOOP-FLAG がgとpのノードをループとする

(3) pノードへの割り付け

ループ間の並列性がある問題では、LOOP-FLAG がpであるノード列中の複数のノードが発火可能となり、処理はパイプライン的に進む。これらのノードを処理するPEを固定すると、このパイプラインが効率よく働かない。この対策としてpであるノードに関してはループ毎に実行するPEをラウンドロビン式に割り付ける。評価の際にはPEを固定した場合と比較する。

Static Load Scheduling in Data Flow Machine

Yoshihisa Otsuka¹, Shuichi Sakai², and Toshitsugu Yuba²

¹ Kobe Steel, Ltd. ² Electrotechnical Laboratory

5 負荷分散方式の評価

5.1 基本方針の検討

評価には電総研で開発中のSIGMA-1 [3] のソフトウェアシミュレータを用いた。PEが2台の場合を例として検討を行う。

TEL, TLEを用いると、図1のようにデータフローグラフがサブグラフに分割できる問題ではサブグラフA, Bに各プロセッサが割り付けられ、処理も速い。

LTE, LETを用いると、並列度の変化の大きな問題ではグラフの微妙な差異で結果は大きく異なる。図2のグラフを例にとると、Lを最優先した割り付けでは、C, Dの各ノード群にはほぼ同一PEが割り付けられる。点線のアークが無い場合はCのノード群はDのノード群の影響を受けないので処理は速いが、点線のアークがある場合は逆に遅くなる。これらの例を共に最適に割り付けるにはノード群の依存関係及び各ノードが実行可能となる時刻を検出する必要があるが、これらは用いているE-levelなどの特性値では判別できない。実行時刻の推定ではE-LEVEL順に実行が進むと考えるのが自然であり、サブグラフへの対応のとれるTELが基本方針として良いと考えられる。以下では基本方針としてTELを用いる。

上記の結果を図3に示す。

5.2 ループ処理方法の評価

ループ処理方法の評価として階乗の計算および図4のグラフを用いた測定結果を図5に示す。前者はループ間並列性がない例であり、後者はある例である。

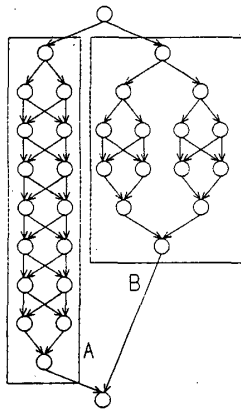


FIG. 1 GRAPH-1

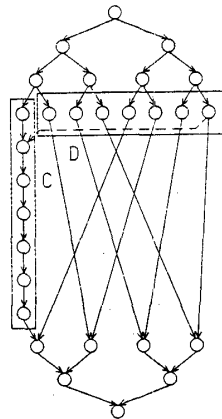


FIG. 2 GRAPH-2

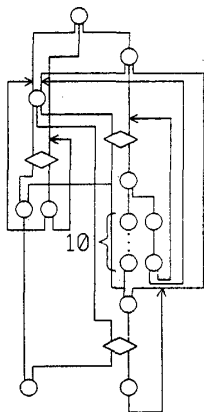


FIG. 4 GRAPH-3

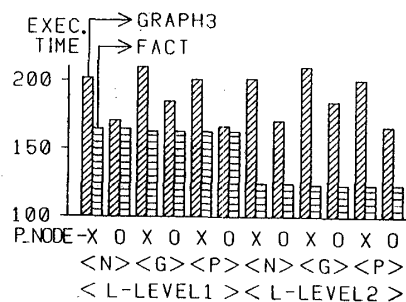


FIG. 5 EVALUATION OF LOOP STRATEGY

階乗の結果よりL-LEVEL2がよく、また図4のグラフの結果よりpノードの割り付け方法を採用したことによる効果が判る。これらの結果のみではL-NとL-Pの差は少ないが、ループに並列な路のある問題ではL-Pの方がすぐれていることは明らかである。

5.3 提案方法の評価

複雑な問題として、1次元熱伝導方程式の陽解法による計算を例として評価を行った。結果を図8に示す。比較のために乱数を用いて負荷分散を行った場合を示す。また乱数を用いて速度が飽和した状態から最適な割り付けを行った場合の推定値を併せて示す。

この結果から本方法によりよい割り付けが行えていることがわかる。

6 あとがき

ループを含むプログラムに対し、PE間の通信時間を考慮に入れた静的な負荷分散方式の提案を行った。またシミュレーションにより本方法は精度が良い方法であることを示した。

研究の遂行にあたり、御指導、御討論いただいた電子技術総合研究所柏木電子計算機部長および計算機方式研究室の方々へ感謝致します。

参考文献

- 【1】 Kasahara, H. and Narita, S.: Practical Multi-Processor Sheduling Algorithms for Efficient Parallel Processing, IEEE Trans. Comput., Vol. C-33, No. 11, pp. 1023-1029 (Nov. 1984)
- 【2】 Tokoro, M. Jagannathan, J. R. and Sunahara, H.: On the Working Set Concept for Data-Flow Machines, Proc. of 10th Annu. Symp. on Comput. Arch., pp. 90-97 (Jun. 1983)
- 【3】 Shimada, T. Hiraki, K. and Nishida, K.: Evaluation of a Prototype Data Flow Processor of the SIGMA-1 for Scientific Computation, Proc. of 13th Annu. Symp. on Comput. Arch., pp. 226-234 (Jun. 1986)

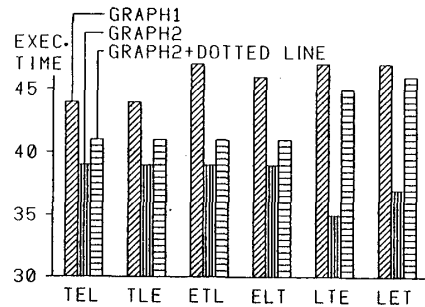


FIG. 3 COMPARISON OF BASIC STRATEGY

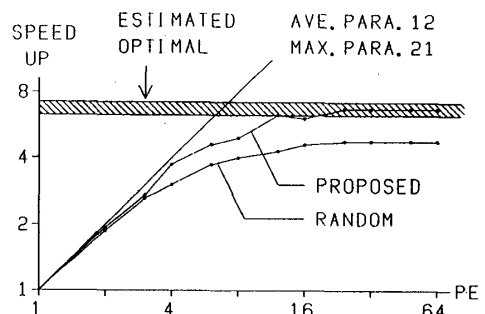


FIG. 6 EVALUATION OF PROPOSED METHOD