

KL1の並列処理

2Q-4

— 疎結合マルチプロセッサにおける負荷分配方式の評価 —

米山貢, 杉江衛, 菅谷正弘(日立製作所 中央研究所)

後藤厚宏(ICOT)

1. はじめに

中期PIM(Parallel Inference Machine)では、記述力と並列化オーバーヘッドの点からKL1を核言語として選択した[1]。KL1では、Prolog、Concurrent Prologなどに比して、リダクションの負荷が軽減されているが、これは、並列化に伴うオーバーヘッドをsuspend/resumeに転嫁した結果ともとれる。中期PIMでは、粒度の拡大により、このsuspend/resumeに伴う並列化オーバーヘッドの低減を図る。一方、粒度の拡大は、稼働率の低下をもたらすことが懸念される。そこで、本報告では並列化オーバーヘッドとの関係において稼働率と密接な関係にある負荷分配方式を評価する。評価にあたっては、プロセッサをネットワークで結合したマルチプロセッサを前提とした。

なお、本報告では、全ゴールプロセスに対する投げ出し比率(以下、投げ出し率と略す)で粒度を表現した。

2. 評価対象

評価は、以下に示す4方式に対して行った。

①ランダム負荷分配方式(ランダム分配と略す)。

負荷分配先プロセッサを乱数を用いて決定する負荷分配方式。

②idle優先負荷分配方式(idle優先分配と略す)

無負荷のプロセッサに優先的に負荷を分配し、無負荷のプロセッサがない場合には、乱数を用いて負荷の分配先を決定する負荷分配方式。

③負荷最小優先分配方式(負荷最小分配と略す)

負荷最小のプロセッサに優先的に負荷を分配する負荷分配方式。

④初期ゴールの分配指定と、上記①から③とを組み合わせた負荷分配方式。

3. 評価条件

評価用データは、ハードウェア・シミュレータ[2]を用いて採取した。ハードウェア・シミュレータは、16台のシングルボード・コンピュータから構成

されており、実際に並列動作する条件下でシミュレーションを行なうことができる。

プロセッサ同士を結合するネットワークは、等距離ネットワークを仮定した[3]。

ベンチマークには、Queens、日英翻訳核、Kernelの3プログラムを使用した。

日英翻訳核は、現実のアプリケーションに近い条件での評価を行なうために開発したプログラムであり、構文解析、日英構文変換など自然言語処理の核部分により構成されている。Kernelプログラムは、負荷をプロセッサ間に均一に割当てた条件での評価を行なうためのプログラムであり、16個の6-Queens探索を初期ゴールとしている。

4. 評価結果

図1に、ランダム分配を行なった場合の、投げ出し率と処理時間の関係を示す。同図からは、どのベンチマークにおいても、ランダム分配を行った場合には、投げ出し率10%程度で処理時間が最小になっていることがわかる。同図において、投げ出し率10%以上で処理時間が大きくなっているのは、高投げ出し率によりsuspend/resume回数が増加し、並列化オーバーヘッドが増大したためである。また、投げ出し率10%以下で処理時間が大きくなっているのはプロセッサ間に負荷の偏りが生じ、稼働率が低下したためである。図1の結果は、プロセッサ間の負荷の偏りを小さく抑えることができれば、投げ出し率をより低くすることにより、並列化オーバーヘッドを低減し、処理時間を短縮することが可能性があることを示唆している。

図2は、idle分配、負荷最小分配の2方式の処理時間を、ランダム分配の場合と比較した結果である。同図からは、①投げ出し率の低下とともに、負荷最小分配の効果が大きくなること、②負荷分配方式の違いによる処理時間の差は小さいこと、がわかる。①は、投げ出し率が小さい場合には、分配先に制御を加えたほうがプロセッサ間の負荷の偏りを小さく抑えられることを示唆している。

Parallel Processing of KL1 - Evaluation of Load Dispatching

Strategy on Loosely Connected Multi-processors

M.YONEYAMA¹, M.SUGIE¹, M.SUGAYA¹, A.GOTO²

¹ HITACHI Central Research Laboratory ² ICOT

図3は、Kernelプログラムにおいて、プログラム中に16台のプロセッサへの初期ゴールの割当てを記述した場合としなかった場合の投げ出し率と処理時間の関係を示した。プログラム中にゴールの割当てを記述する方式は、負荷の均一分散を可能ならしめる一方式として取り上げた。図3からは、負荷分配の均一度によっては、投げ出し率は1%程度まで低下させられることがわかる。

図3には10回の試行に対する平均値をプロットしてある。投げ出し率1%の場合、負荷分配方式によっては、試行に対する処理時間の変動が大きかった。これは、投げ出し率を下げすぎることの危険性を示唆している。図4には、Kernelプログラムでゴール割当てを行なった場合の、投げ出し率1%から5%の間での、投げ出し率と処理時間の平均値、分散の関係を示した。同図からは、①ランダム分配では、処理時間の分散に大きな差はないこと、②負荷最小分配では、投げ出し率1%の場合の処理時間の分散は、5%の場合の約20倍であること、③負荷最小分配では、投げ出し率5%の場合でもランダム分配の投げ出し率1%の場合と同程度の処理時間となること、などがわかる。

PIMでは、並列化オーバーヘッド低減のため、負

荷の高閉じ込めを目指している。図4の結果からは、投げ出し率1%程度では、危険が高く、現在の技術レベルでは投げ出し率5%程度にとどめるべきであるといえる。

5. おわりに

疎結合マルチプロセッサにおける負荷分配方式の評価結果について報告した。今後は、本報告の結果に基づき、PIMにおける負荷分配方式の検討を進めていく予定である。

最後に、日頃御指導頂いた、内田俊一ICOT第4研究室長に深謝する。

参考文献

- 1) 「電子計算機基礎技術開発成果報告書-推論サブシステム編-」,昭61.3.
- 2) 米山他:「並列推論マシンPIM-Rのハードウェア・シミュレータ・ハードウェアの方式」,情報処理学会第31回全国大会1C-4,昭60.9.
- 2) 杉江他:「並列推論マシンPIM-リダクション方式の評価」,情報処理学会第33回全国大会,3B-9,昭61.9.

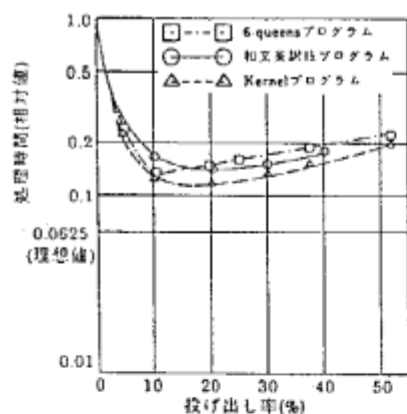


図1. 投げ出し率と処理時間

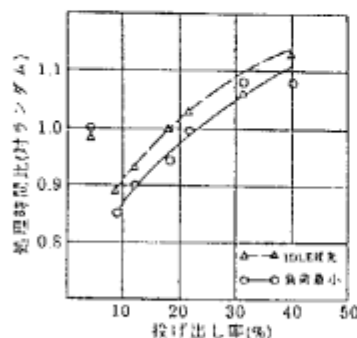


図2. 処理時間比較

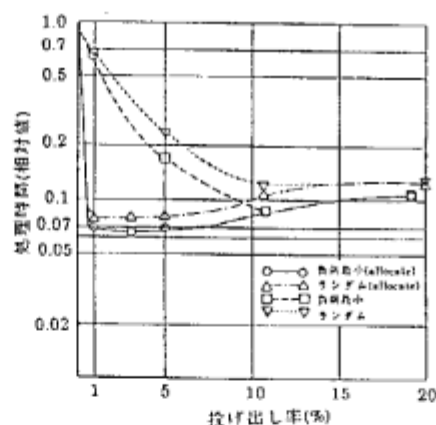


図3. 処理時間の投げ出し率依存性

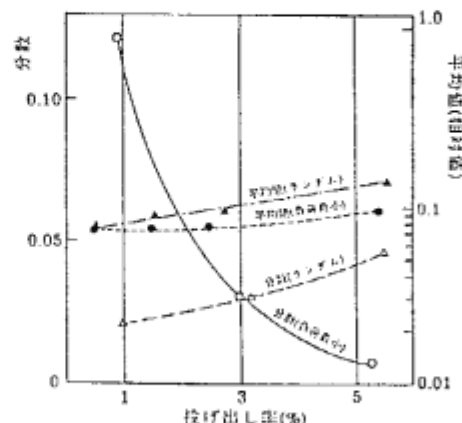


図4. 投げ出し率と処理時間の分散・平均