# 並列計算機のPE間通信状況を考慮したタスク配置アルゴリズムの評価

3F－4

須崎 有康　田沼 均　平野 聡　一杉 裕志　Chris Connelly　塚本 享治

電子技術総合研究所
suzaki@etl.go.jp

## 1　はじめに

タスク配置アルゴリズムは並列計算機に複数の並列タスクを効率的に割り当て、空間共有(Space Sharing)型のマルチタスクを実現する。このタスク配置アルゴリズムは連続型と非連続型に分けられる[2]。連続型はタスクが要求するプロセッサ形状で割り当てる。このためタスク内のプロセッサ間通信が他のタスクに邪魔されることなく効率的に行なえるが、プロセッサ形状を変えることができないため、プロセッサ利用率が低い。一方、非連続型はプロセッサ形状を保持しないで割り当てるため、プロセッサ利用率が高い。しかしプロセッサ形状を崩すため、通信経路が長くなったり、他のタスクのメッセージと衝突する場合がある。我々は両者の欠点を解消するため両者を併用したタスク配置アルゴリズムを提案し、できるだけプロセッサ形状を崩さない高いプロセッサ利用率と少ないメッセージの衝突を実現した[3]。本論文では、この方式が異なったメッセージ転送方式(Wormholeや Virtual Cut Through)と異なったメッセージ転送量においての性能を評価する。

## 2　タスク配置アルゴリズム

我々はメッシュ結合並列計算機を対象として、連続型タスク配置アルゴリズム Adaptive Scan(AS)[1] と非連続型タスク配置アルゴリズム Multi Buddy(MS)[2] を融合した方式(AS&MS)を提案した[3]。

プロセッサ上での AS と MS によるタスク配置の例を図1に示す。AS は左下辺から右方向、上方向へとタスクが要求する形状を検索し、最初に割り当て可能となった領域にタスクを割り当てる。更に AS ではプロセッサ形状の転置も許し、要求された形状で領域が見つからなかった場合、形状を転置してもう一度検索を行なう。TASK 3 は転置によって領域が見つかった例である。MS は要求されたプロセッサ数のみに注目し、$2^n \times 2^n$ の正方形の領域を区切って、各正方形の大きさ毎に割当を行なう。例えばTASK 1 では $2 \times 3$ を要求しているが、これは $6 = (2^1 \times 2^1) \times 1 + (2^0 \times 2^0) \times 2$ と解釈され、$2 \times 2$ の領域一つと $1 \times 1$ の領域二つが割り当てられる。MS では領域を割り当てる際にできるだけ大きい領域を崩さない戦略のため、近くの領域が使用可能でも通信距離の遠い断片領域を割り当てる。このため、TASK 1 は分割される。

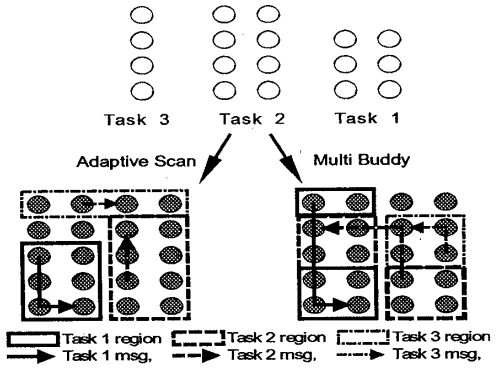


図 1: 連続型と非連続型のタスク配置 ($4 \times 5$PE)

提案した AS&MS では、タスクが投入された際にまず AS で領域探索を行ない、これが失敗した場合に MS に切替える。このためプロセッサ形状保存ができる場合はそのまま領域を割り当て、形状を保存できなくなってはじめてタスクのプロセッサ形状の分割を行なう。これによりプロセッサ利用率を損なわず、割り当て領域をできるだけ近接させることができる。

## 3　タスク配置アルゴリズムの性能

タスク配置アルゴリズムの性能評価はオレゴン州立大で開発された procsimity[2] を改良して行なった。procsimity ではメッセージ転送方式に Wormhole(WH), Virtual Cut Through (VT) 等を備えメッセージ衝突によるタスク処理の遅延をシミュレートできる。また、メッセージパターンも All to All, All to One, FFT, Random, NAS Multigrid 等揃えており、各タスクの性質を変えて性能を評価できる。本論文では、$32 \times 32$ のプロセッサを想定し、千個のタスクを投入間隔を変えてその変化を調べた。タスクの平均処理時間は 1000 unit time とし、平均投入間隔は 100 ～ 1000 unit time とした。この時、計算機の負荷 (load: 平均処理時間／平均投入間

---

The evaluation of task allocation methods under some message transmit conditions

Kuniyasu SUZAKI　Hitoshi TANUMA　Satoshi HIRANO
Yuuji ICHISUGI　Chris Connelly　Michiharu TUKAMOTO
Electrotechnical Laboratory

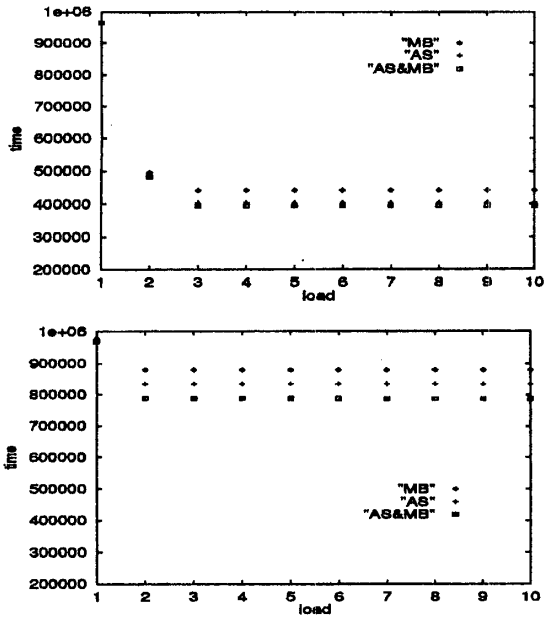


図 2: WH 下での全タスク終了時間 (上図は 8byte メッセージを平均 40 個、下図は平均 80 個)

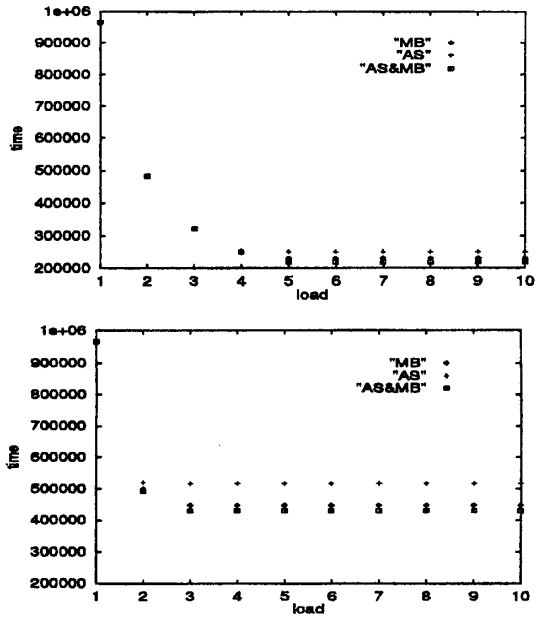


図 3: VT 下での全タスク終了時間 (上図は 8byte メッセージを平均 40 個、下図は平均 80 個)

隔) は 1 ～ 10 となる。タスクは長方形のプロセッサ群を要求し、その一辺は 1 から 32 の一様分布とした。ネットワークの影響を調べるため、WH と VT を対象とした。WH と VT では共に 1byte flit とし、バッファは WH が 1byte、VT が 8byte とした。1 flit のディレイは 3 unit time とした。また本論文では、通信量の影響を調べるためメッセージパターンを All to All に固定し、メッセージ量を変化させた。このメッセージパターンでは、メッセージがブロックされやすいので一般に WH より VT が小さいレイテンシを示すことが知られている。計算機側からの全体性能を示す全タスクの終了時間の負荷に対する変化を図 2(WH) と図 3(VT) に示す。

### 3.1 通信量の影響

図 2より、WH ではメッセージが少ない時 (図 2上) は通信距離が短い AS と AS&MS がほぼ同じ終了時間を示し、MS が若干遅い終了時間を示した。メッセージが多くなった時 (図 2下) は通信距離が短い利点が顕著になった。またプロセッサ利用率の効果が現れ、AS より AS&MS が早い終了時間を示した。

図 3より、VT ではメッセージが少ない時 (図 3上) はプロセッサ利用率の高い MS と AS&MS がほぼ同じ終了時間を示し、AS が若干遅い終了時間を示した。メッセージが多くなった時 (図 3下) はプロセッサ利用率の高い利点が顕著になった。また短い通信距離の効果が現れ、MS より AS&MS が若干早い終了時間を示した。

### 3.2 ネットワークの影響

図 2と図 3を比較すると、レイテンシが大きい WH では終了時間が AS&MS, AS, MS 順位に早いが、レイテンシが小さい VT では AS&MS, MS, AS の順になった。

この結果は WH ではメッセージ転送にかかる負荷が大きいため通信距離が短くなるようなタスク配置が優位になり、VT ではメッセージ転送にかかる負荷が小さいためプロセッサ利用率が高いタスク配置が優位になっているためである。いずれの条件にしても両方の性質を合わせ持つ AS&MS がもっともよい性能を示した。

## 4 おわりに

本論文の性能評価により、連続性と非連続性を組み合わせた方法 (AS&MS) がネットワークレイテンシが小さい計算機 (WH) においても大きい計算機 (VT) においても、またメッセージが多い状況においても少ない状況においても、単独の手法 (AS と MS) よりも短い通信距離と高いプロセッサ利用率の効果を表して優れた性能を示すことがわかった。この結果から AS&MS の手法がどのような計算機環境にも効率良く適応できることが予想される。

**謝辞** 本研究の一部は RWC 計画の一環として「超並列システムアーキテクチャに関する研究」で行なわれたものである。関係各位に感謝する。

## 参考文献

[1] J. Ding and L. N. Bhuyan. An Adaptive Submesh Allocation Strategy for Two-Dimensional Mesh Connected Systems. *ICPP*, 1993.

[2] W. Liu, V. Lo, K. Windish, and B Nitzberg. Non-contiguous Processor Allocation Algorithms for Distributed Memory Multicomputers. *Supercomputing*, 1994.

[3] 須崎, 田沼, 平野, 一杉, Connelly, 塚本. 連続性と非連続性を合わせ持つタクス配置アルゴリズムの提案. 情報処理学会研究会報告 *96-OS-73*, 1996.