

標準タスクグラフセットを用いた データ転送オーバーヘッドを考慮した スケジューリングアルゴリズムの性能評価

山口 高弘, 田中 雄一, 飛田 高雄, 笠原 博徳
早稲田大学理工学部電気電子情報工学科

1 はじめに

マルチプロセッサシステムにおいて、効率のよい並列処理を行うためには、処理すべきタスク集合をどのようにプロセッサに割り当て、どのような順序で実行すれば実行時間を最小にできるかという、実行時間最小マルチプロセッサスケジューリング問題を解決する必要がある。この問題は、一般に要求される任意のタスク処理時間、プロセッサ台数、タスクグラフ形状、プロセッサ間のデータ転送にかかる時間を扱う場合、その時間複雑度は強 NP 困難となることが知られている [1, 2]。

このスケジューリング問題の難しさのため、ヒューリスティックアルゴリズムが多く用いられている [1, 2, 3] が、それらのアルゴリズムの性能は同一基準で評価されていないため、評価結果の客観的な理解が困難であるという問題があった。したがって、性能を公平に評価するためには、さまざまな形状、任意のデータ転送時間などの特徴をもち、誰にでも手に入るタスクグラフを多数用いる必要がある [2]。

そこで本論文では、従来の研究 [3, 4, 5, 6] で使用されているタスクグラフ生成方法を参考にして、スケジューリングアルゴリズムの性能を公平に評価するためのデータ転送付標準タスクグラフセットを提案し公開すると共に、それらを用いてのヒューリスティックアルゴリズム (CP/DT, CP/DT/MISF, DT/CP, DT/CP/MISF, CP/ETF, ETF/CP) [1, 2, 3] の性能評価について述べる。

2 対象とする問題とアーキテクチャ

本論文で扱う実行時間最小マルチプロセッサスケジューリング問題は、 m 台の同一性能のプロセッサを用いて、任意形状の先行制約があり、任意の処理時間を持つ n 個のタスクからなるタスク集合 $T = \{T_1, T_2, \dots, T_n\}$ を処理割り込みなしに並列処理する際に、プロセッサ間のデータ転送オーバーヘッドも考慮して、各タスク及びデータ転送のプロセッサへの割り当てを求める問題である。このタスク集合 T はタスクグラフ $G(T, E)$ (E : 有向エッジの集合) として DAG (無サイクル有向グラフ) で記述される。タスクグラフはそれぞれ 1 つの入口ノードと出口ノードを持ち、すべてのノードは入口ノード、出口ノードから到達できるものとする。

図 1 にタスク数 8 個のタスクグラフの例を示す。ノード内の数字はタスク番号 i 、ノード左側の数字は各タスクの処理時間 t_i を表している。また、図中のアークはタスク間の先行制約を表し、ノード i からノード j へのアークはタスク T_i がタスク T_j に先行して実行されなければならないという半順序制約を表しており、さらにアーク上の数字はノード i とノード j が違うプロセッサに割り当てられた場合のデータ転送時間 c_{ij} を表している。

本論文で対象とするアーキテクチャは、図 2 のように Crossbar Network によって接続された複数のプロセッサエレメント (PE) が、それぞれ分散共有メモリ (DSM) とデータ転送ユニッ

ト (DTU) を持ち、各 PE からアクセス可能な集中共有メモリ (CSM) を持つようなマルチプロセッサシステムである。DTU によるデータ転送は、CPU 上でのタスク処理とオーバーラップして行うことができるものとし、スケジューリング時のデータ転送は DSM から他 PE の DSM へ直接行われるものとする。必要となるデータの総量が DSM 容量を超える場合には、データを CSM へ退避しなければならないが、本論文では DSM の容量は十分で不足は起こらないと仮定する。

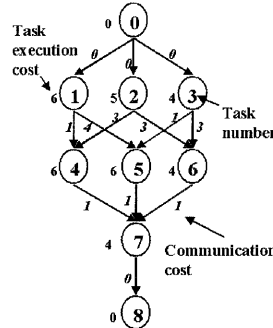


図1: タスクグラフ

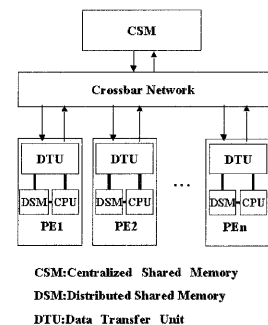


図2: 対象アーキテクチャ

3 データ転送付標準タスクグラフセット

本稿で利用するデータ転送付標準タスクグラフセットは、ランダムタスクグラフとガウス消去計算用タスクグラフで構成されており、今後の Web 公開バージョンでさらに拡充していく予定である。既にデータ転送時間を 0 とした標準タスクグラフセット (Standard Task Graph Set, 以下 STG ver.1) は Web で公開されているが [3], その中のランダムタスクグラフは主に文献 [4, 5] を参考にして、4 種類の方法によるエッジ接続、3 種類の方法によるタスク処理時間、タスク数は 50~1900 までの合計 660 例であった。本論文で用いるデータ転送付標準タスクグラフセットは、タスク数が 50, 100, 300, 500, 700 の STG ver.1 をもとに、タスクグラフの形状は従来の STG ver.1 のものを用い、タスク処理時間を 1000 倍にして、データ転送時間を “1 タスクあたりのデータ転送時間の和 = 平均タスク処理時間 × CCR” (CCR: データ転送比) となるような正規乱数で作成した。本論文では上記の条件で CCR を 0.1, 0.2, 0.5, 0.8, 1.0 とした合計 1500 例のタスクグラフを生成した。そしてガウス消去法に基づくタスクグラフも文献 [6] を参考にして、マトリクスサイズ 5~50 の合計 46 例を生成した。これらのタスクグラフは STG_dt ver. 0.1 として <http://www.kasahara.elec.waseda.ac.jp/schedule/> で公開中である。

4 スケジューリングアルゴリズム

本章では性能評価を行ったスケジューリングアルゴリズム CP/DT, CP/DT/MISF, DT/CP, DT/CP/MISF, CP/ETF, ETF/CP について述べる。

CP/DT [1, 3] は、各スケジューリング時点においてレディタスク集合中の出口ノードからのパス長 (以下 CP 長) が最も長いタスクを選択し、そのタスクをデータ転送時間が最小となるプロセッサ (以下 PE) に割り当てるアルゴリズムである。ここで、同一の CP 長と同一のデータ転送時間を持つタスクと PE の組み合わせが 2 つ以上あれば、直接後続タスク数の多いタスクを優先して割り当てる方法が CP/DT/MISF [1, 3] である。

* Performance Evaluation of Scheduling Algorithms
with Data Transfer Using Standard Task Graph Set
Takahiro Yamaguchi, Yuichi Tanaka,
Takao Tobita, Hironori Kasahara
Department of Electrical, Electronics and Computer
Engineering, Waseda University

DT/CP[3]は、各スケジューリング時点において各レディタスクをアイドルプロセッサに割り当てたときに必要となるデータ転送時間を計算し、データ転送時間が最小となるタスクとPEの組み合わせを優先するが、データ転送が最小となる組み合わせが複数ある場合には、CP長が長いタスクを優先して割り当てるアルゴリズムである。ここで、データ転送時間、CP長が共に等しいタスクが複数あった場合には、直接後続タスク数の多いタスクを優先して割り当てる方法がDT/CP/MISF[3]である。

CP/ETF[2, 3]は、各スケジューリング時点において、レディタスク集合からCP長が最も長いタスクを選択し、そのタスクを各PEに割り当てたときの最早実行可能時間を計算し、最早実行可能時間の最も早いタスクとプロセッサの組み合わせを優先して割り当てるアルゴリズムである。

ETF/CP[2, 3]は、レディタスク集合中の全タスクそれぞれを各PEに割り当てたときの最早実行可能時間を計算し、最早実行可能時間が最小となる組み合わせを優先して割り当てるが、その組み合わせが複数あった場合には、CP長が長いタスクを優先して割り当てるアルゴリズムである。

5 性能評価

本章では、データ転送付標準タスクグラフセットを前章で述べた6つのヒューリスティックおよびFIFOの各アルゴリズムに適用した性能評価結果について述べる。

データ転送付標準タスクグラフセットのタスク数が50, 100, 300, 500, 700のランダムタスクグラフ1500例を各アルゴリズムで2, 4, 8, 10台のPEにそれぞれ割り当てた場合のPE1台での逐次実行時間に対する平均速度向上率を図3に示す。図3のように、PE10台の場合で、ETF/CPの平均速度向上率は4.8倍で、これはFIFOと比べても10.3%優れている。各CCRごとのFIFOの平均スケジュール長(図中1.00に相当)に対する各アルゴリズムの平均スケジュール長の割合を図4に示す。図4のように、CCRが0.1の場合はCP/ETFが平均スケジュール長が0.95と最も短く、CCRが0.2~1.0の場合はETF/CPが0.93, 0.89, 0.88, 0.86と最も短い平均スケジュール長を示している。

この結果から、今回評価したアルゴリズムの中ではETF/CPがPEを多くしたり、CCRが大きい場合にも平均スケジュール長を短くできることが確かめられた。その理由としては、ETF/CPは最も早く実行可能となるタスクとPEの組み合わせを優先するので、CCRすなわちデータ転送時間がタスク処理時間に対して相対的に大きい場合には、CP長を優先して割り当てるアルゴリズムに比べ、データ転送オーバーヘッドを軽減できるからだと考えられる。

また、実アプリケーションタスクグラフの例として、マトリクスサイズが10~40の31例のガウス消去法に基づくタスクグラフを2, 4, 8, 10台のPEにスケジューリングした場合の、PE1台での逐次実行時間に対する平均速度向上率を図5に示す。図5のように、PE10台の場合で、ETF/CPの平均速度向上率は4.24倍で、これはFIFOと比べても69.3%優れているなど、今回評価したアルゴリズムの中ではETF/CPが最も優れた性能を示している。この理由としては、ガウス消去法のタスクグラフのCCRが1.0程度で比較的大きいため、ETF/CPがデータ転送による遅延の少ないスケジュールを導くためだと考えられる。

6 まとめ

本論文では、ヒューリスティックアルゴリズムを公平に評価するためのデータ転送付標準タスクグラフセットSTG_dt ver. 0.1を提案し、そのタスクグラフセットを用いたヒューリスティックアルゴリズム(CP/DT, CP/DT/MISF, DT/CP, DT/CP/MISF, CP/ETF, ETF/CP)の性能評価の結果、データ転送比CCRが小さい場合にはCP長の長いタスクを最も優先して割り当てるアルゴリズムが有効であるが、CCRが大きい

場合には、先行タスクからのデータ転送量が最も少なくなる割り当てを優先するアルゴリズムのほうがデータ転送の影響によるタスク処理の遅延の影響を受けにくく、ETF/CPが最良の性能を示すことが確かめられた。

今後、実アプリケーションに基づく多種類のタスクグラフやさらに大規模なタスクグラフの追加によるデータ転送付標準タスクグラフセットSTG_dt ver. 0.1

(<http://www.kasahara.oscar.elec.waseda.ac.jp/schedule/>)

の充実をはかり、それらを用いてヒューリスティックアルゴリズムや最適化アルゴリズムの評価を行っていく予定である。

本研究の一部はNEDOアドバンスド並列化コンパイラプロジェクト及び学術振興会未来開拓事業により行われた。

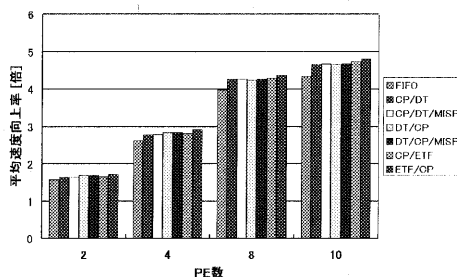


図3: 平均速度向上率(ランダムタスクグラフ)

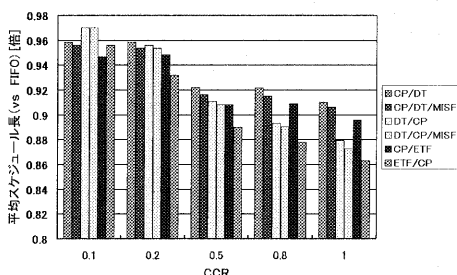


図4: 平均スケジュール長(ランダムタスクグラフ)

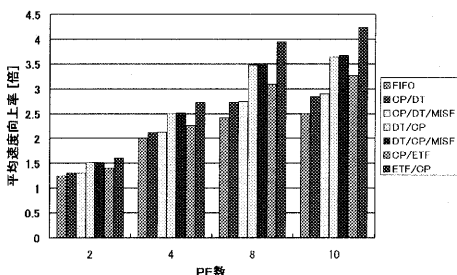


図5: 平均速度向上率(ガウス消去法タスクグラフ)

参考文献

- [1] 笠原 博徳: “並列処理技術”, コロナ社 (1991).
- [2] Kwok, Y. and Ahmad, I.: “Benchmarking the Task Graph Scheduling Algorithms”, IPPS Proc.'98 (1998).
- [3] 飛田 高雄, 笠原 博徳: “標準タスクグラフセットを用いたマルチプロセッサスケジューリングアルゴリズムの性能評価”, JSPP2000, pp.131-138 (2000).
- [4] Yang, T. and Gerasoulis, A.: “DSC: Scheduling Parallel Tasks on an Unbounded Number of Processors”, IEEE Trans. Parallel and Distributed Systems, Vol.5, No.9, pp.951-967(1994)
- [5] Almedia, V. A. F., Vasconcelos, I. M. M., Árabe, J. and Menascé, D. A.: “Using Random Task Graphs to Investigate the Potential Benefits of Heterogeneity in Parallel Systems”, Proc. of Supercomputing '92, pp.683-691(1992)
- [6] Cosnard, M., Marrakchi, M., Robert, Y. and Trystram, D.: “Parallel Gaussian Elimination on an MIMD Computer”, Parallel Computing, Vol.6, pp.275-296(1993)