

コンピュータクラスタにおける省エネルギー化のためのスケジューリング手法の研究

高性能コンピューティング学講座 本多研究室

0953019 山下 良

主任指導教員：本多弘樹

1 概要

近年の環境意識の高まりや、電力需要の増大により、コンピュータハードウェアの省エネルギー化への要求が高まっている。これまでの研究でもコンピュータの消費電力を削減するための省エネルギースケジューリングの研究はあった [1][2][3]。しかしながら、これまでの研究では、CPU の電圧制御機構を利用した研究 [1][2]、CPU を動作させるか動作させないかの 2 択の研究 [3] で、コンピュータ全体の電力消費と処理性能の特性に合わせたスケジューリングは行われてこなかった。また、ハードウェアを複数用いたホモジニアス (単一) 環境での研究 [2][3] が主に行われており、データセンタ等の実際の環境に近いヘテロジニアス (異種混在) 環境におけるスケジューリングの研究 [4][5] はあまり注目されてこなかった。ヘテロジニアス環境におけるスケジューリングの研究においてもほとんどは処理性能を向上させることを対象にしており、省エネルギー化のための研究は少なかった。

そこで本研究では、ヘテロジニアス環境のコンピュータクラスタにおいて、コンピュータ内部のパーツごとの消費電力をスケジューリングのパラメータとする。さらに、省エネルギースケジューリング手法の提案により、コンピュータクラスタの省エネルギー化を図ることを目的とする。

本研究において省エネルギー化する上で着目している点は、先行タスクによるコンピュータの処理待ち時間である。コンピュータの処理待ち時間は、処理性能の優れたコンピュータによりタスクが早く処理された結果であると考えられる。

そこで、コンピュータの待ち時間が生じず、なおかつ省エネルギーになるようタスクとタスクを処理するコンピュータの組み合わせをタスクの実行前にシミュレートする。コンピュータの待ち時間が生じると判断された場合、そのタスクを他に比べて省エネルギーであるが遅いコンピュータに割り当てる。タスク割り当てのシミュレーションをタスク処理の実行前に行うことにより、省エネルギー化を図りながら処理性能は落とさないことが可能となる。

2 提案手法

2.1 対象とする環境

本研究では以下のような環境を対象としている。

- それぞれのコンピュータの CPU、メモリ、電源特性と処理性能が異なるヘテロジニアス (異種混在) 環境。
- 扱うタスクが先行制約付きのアプリケーション。
- 各タスクのコンピュータパーツ使用率は分かっている。
- 各パーツの電力使用量は分かっている。
- コンピュータクラスタ内のそれぞれのコンピュータにおけるタスクの処理時間は分かっている。
- CPU の電圧制御機構は利用しない。すべてのタスクはコンピュータハードウェアの最大処理能力で処理するものとする。

2.2 エネルギー削減の余地

本研究では、消費エネルギーを削減できる箇所があるかどうか見極めるため、予備実験としてコンピュータの高負荷時とアイドル時の消費電力を測定した。例として、Xeon 2 GHz、メモリ 2 GB の場合を測定したが、高負荷時は 121 W、アイドル時は 70 W であった。高負荷時とアイドル時の差は大きく、省エネルギー化の余地は大きいと考えられる。

2.3 エネルギー削減方法

本研究では、コンピュータクラスタの省エネルギー化を目的として先行制約付きタスクを扱う。先行制約付きタスクを均一な性能のコンピュータで処理した場合、先行タスク完了までの待ち時間が生ずることがある。

そこで本研究では、先行タスクの完了までのコンピュータの待ち時間をなるべく生じないようにする。先行タスクの完了までの待ち時間が生じる場合、消費エネルギーが大きく、処理性能が高すぎるコンピュータでタスクが処理されたと考えられる。そこで、先行タスクの完了のタイミングを合わせるために、省エネルギーだが処理性能が劣るコンピュータでタスクを処理する方法をとる。(図 1) そのため、結果として消費エネルギーが削減できることになる。

タスクの処理で消費するエネルギーを最小化しようとする場合、処理速度の低下を招くことがある。そこで、本研究のスケジューリング手法では、はじめに処理が一番高速なコンピュータへタスクを割り当てることにして、コンピュータクラスタ内のマスターコンピュータがタスク割り当てシミュレーションを行う。そのシミュレーション中、先行タスクによる待ち時間ができるようであれば、タス

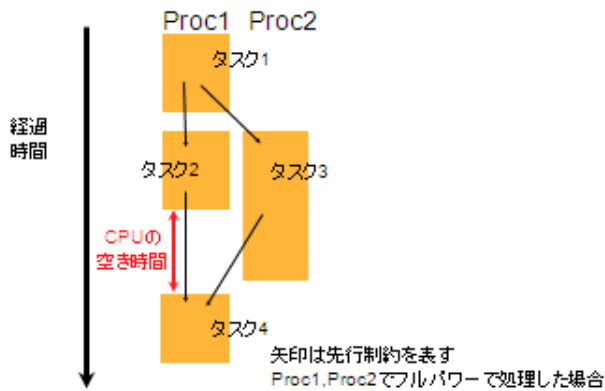


図 1: エネルギー削減箇所

タスクの割り当て方を変えて省エネルギーだが処理速度が劣るコンピュータに割り当てる方法を取ることにする。マスターコンピュータは、タスクの並び替えパターンを何度も繰り返し、均一にタスクを割り当てた場合より早く処理でき、なおかつ消費エネルギーが削減できることを確認した後、タスクの処理を実行する。

3 進捗状況

まず初めに消費電力削減の余地があるかどうかを確認するためにアイドル時と高負荷時のコンピュータ全体の電力使用量を測定した。

現在は各パーツの消費電力を把握するための環境を整えたところである。測定にはクランプセンサを用い、デジタルアナログ変換ボードを通して、コンピュータへ取り込む。取り込むためのプログラムは Linux 用サンプルで付属したものを Windows へ移植して使用することとした。

現在測定を行っているのは電源からロジックボードで至る 3.3V、5V、12V の各線である。電流を測定して、それぞれの電圧をかけることで消費電力を算出している。表 1 に示した数値は Nano1.6GHz、メモリ 2 GB、400 W 電源の構成のコンピュータを対象として測定した際のアイドル時のものである。

表 1: 消費電力測定結果

測定項目	消費電力 [W]
+3.3V ライン	6.50
+5V ライン	17.98
+12V ライン	39.25
-12V ライン	0.48

4 今後の方針

これからも予備実験としての電力測定を続け、各パーツごとの消費電力の最大値と最小値を割り出す予定である。その後は、様々なパーツを組み合わせた際のコンピュータ全体の消費電力を予測できるようにする。それらデータを元に今回の研究の肝となる省エネルギースケジューリ

ング手法の実装、検証を行っていく予定である。

5 関連研究

スケジューリングを工夫して消費エネルギーを削減する方法は少ないながらも従来より行われている。文献 [3] ではデータセンタにおいて、これまでのコンピュータの利用情報を元に今後の利用状況を予測して、最小限度のコンピュータのみ起動する方法を提案している。その他に、CPU の電圧制御機構を利用して省エネルギー化を図っているものでは、文献 [1][2] がある。文献 [2] ではホモジニアス環境、文献 [1] ではヘテロジニアス環境を対象としており、先行制約付きタスクを扱っている。

文献 [4][5] ではヘテロジニアス環境を対象としているが、消費エネルギーについての言及はなく、先行制約付きではないタスクを対象としている。

6 まとめ

ヘテロジニアス環境のコンピュータクラスタで先行制約付きタスクを扱う省エネルギースケジューリング手法を提案する。今回の発表ではスケジューリングによるコンピュータクラスタの省エネルギー化の余地があることを示した。現在はそれぞれのコンピュータパーツの消費電力測定を行い、省エネルギースケジューリング手法を考案している状態である。今後はスケジューリング手法の実装と検証に力を入れていく予定である。

参考文献

- [1] Young Choon Lee et al. "Minimizing Energy Consumption for Precedence-constrained Applications Using Dynamic Voltage Scaling", Zomaya, 9th IEEE/ACM International Symposium on Cluster Computing and the Grid(CCGrid), 2009, Pages 92-99
- [2] K.H.Kim et al. "Power Aware Scheduling of Bag-of-Tasks Applications with Deadline Constraints on DVS-enabled Clusters", International Symposium on Cluster Computing and the Grid IEEE/ACM, 2007, pages 541-548.
- [3] Dusit Niyato et al. "Optimal Power Management for Server Farm to Support Green Computing", International Symposium on Cluster Computing and the Grid IEEE/ACM, 2009, pages 84-91.
- [4] H.Topcuoglu et.al "Performance-Effective and Low-Complexity Task Scheduling for Heterogeneous Computing", IEEE Trans. Parallel Dist. Systems, vol.10, no.8, 1999, pages 795-812.
- [5] Bozdog.D et al. "A task duplication based bottom-up scheduling algorithm for heterogeneous environments" Parallel and Distributed Processing Symposium, 2006, pages 132-143.