

変動する負荷の並列性を考慮したマルチコアスマートフォンの電力制御方式とその評価

村上 岳生[†] 野呂 正明[†] 上和田 徹[†] 松本 達郎[†]

富士通研究所[†]

1. はじめに

近年、マルチコア CPU を搭載したスマートフォンが普及しつつある。マルチコア CPU では性能と消費電力のバランスを取るため周波数を制御する手法¹⁾、コア間の負荷分散で消費電力を抑える手法²⁾、負荷に合わせてコアを ON/OFF する制御手法がある^{3) 4) 5)}。このうち、コア数の制御方法には、CPU の利用率に基づくもの^{3) 4)}、OS のカーネルにおけるスケジューラの実行するプロセスの実行キュー長に基づくもの⁵⁾がある。

本研究では、負荷の並列性が容易に観測できることからスマートフォンにおいて多く利用されている、実行キュー長によるコア数制御方式⁵⁾の電力性能を改善するため、プロセスおよびスレッドのコア間での偏りに着目した制御を追加した。本稿では、その制御方式および、その電力性能の評価結果について報告する。

2. 従来技術とその課題

ここでは、従来の実行キュー長平均値に基づく制御方式⁵⁾および、現在のスマートフォンにおけるプロセスやスレッド間での負荷の偏りについて説明する。

2.1 実行キュー長に基づくコア数制御方法

実行キュー長に基づいて CPU のコア数を制御する方式⁵⁾では、load average (実行キュー長) および、全体の処理量に基づいてコア数を制御しているが、主として実行キュー長を用いる。稼働中のコア数で処理がさばけない時に実行キュー長は長くなるため、実行キュー長が事前に定められたしきい値を超えた場合にコア数を増加させる。

2.2 コア間での負荷の偏り

図1はスマートフォンの地図アプリ (1分間) 操作時のスレッド毎の CPU 利用率の変動を表している。アプリやその関連サービス等の多数のスレッドが動作するが、文字入力処理スレッド

や地図アプリのメインスレッドなど特定のスレッドに負荷が集中している。

さらに、個々のスレッドの負荷も1分間の期間で均等ではないため、短い期間での処理の並列度は実行キュー長より小さくなるケースがある。そのため、単純に実行キュー長だけを利用してコア数を決定した場合、実際には処理すべき負荷がほとんど存在しないコアが出現してしまい、このコアによる消費電力が無駄になる。

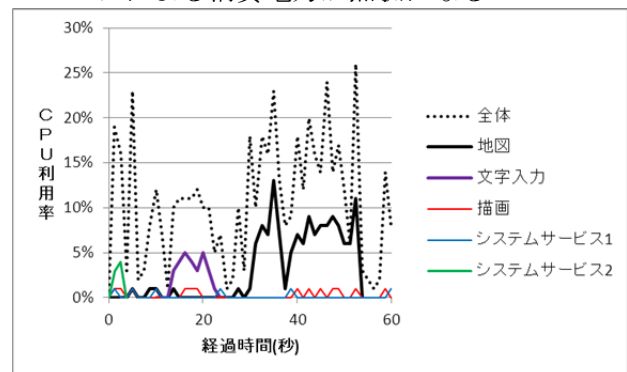


図1: 地図アプリのスレッド毎の負荷変動

このようにスマートフォンで動作しているアプリ、サービスのスレッド負荷の各コアへの分散は均等ではないため、現在のコア数制御方式では、電力の無駄が発生する。電力の無駄を抑えるためにはスレッド負荷の偏りを定期的に監視する必要があるが、コア数制御を行う間隔(50ms~250ms)でスマートフォン上で動作中のスレッド全て(500~1000)を監視する事はコストが大きく、現実的ではない。

3. 提案方式

2章で説明した課題を解決するため、本研究では図1のようなスマートフォンの負荷の偏り方の特徴を活かして、監視対象のスレッドを絞り込み、それらのスレッドの仕事量(動作周波数×CPU利用率)に基づきコア数を制御する機能を追加した。監視対象はユーザが利用中のアプリとアプリに関連するシステムサービス(例: 描画, メディア処理)に絞り込む事でスレッド観測によるコストを抑える。

図2は本方式のフローチャートである。スレッ

Evaluation of Multicore Smartphone Power Management Method Considering Variable Load Concurrency

[†]Takeo Murakami, Masaaki Noro, Toru Kamiwada, Tatsuro Matsumoto

[†]Fujitsu Laboratories LTD.

下の CPU 利用率と該当スレッドが実行されているコアの周波数から実際の仕事量を計算し、監視対象スレッドの仕事量の合計がある閾値を超えていなければ、キュー長の条件を満たしている場合のコア数の増加を抑制する。

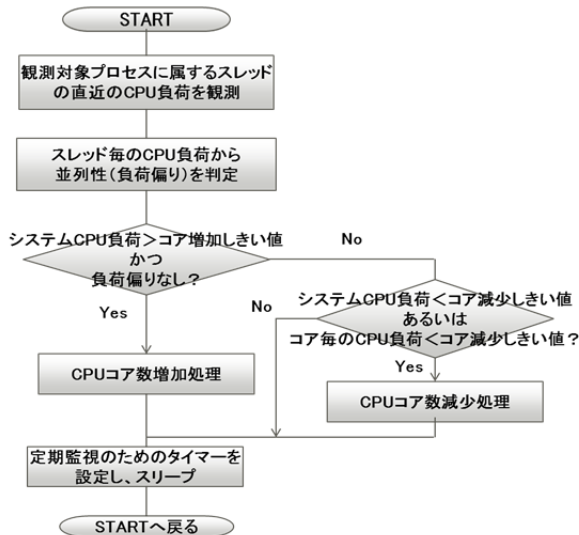


図 2: 制御方式処理フロー

4. 評価

4 コア CPU (Qualcomm APQ8064) を搭載したスマートフォンを使用し、提案制御方式の電力削減効果をシミュレーションにより評価した。まず、CPU コア数、周波数と利用率のログを取得し、それらの値から事前に計測しておいた CPU 消費電力の特性に当てはめて、各観測期間の消費電力を推定する。次に、この推定消費電力の平均値を本方式と従来方式で比較した。

実験結果を表 2 に示す。表の (1) は従来方式でアプリ実行時にスレッド負荷の偏りが発生した場合に、コア数を増加させた事による一回 (コア数制御間隔, 50ms) 当たり無駄に消費される電力を表している (増加するコア数、周波数の違いにより異なる)。 (2) はこの偏りが 1 分間あたりに発生した回数、また、 (3) はスレッドの観測にかかるコスト (観測対象スレッド数により異なる)、 (4) は従来方式と提案方式の平均消費電力である。

地図の場合、偏り一回あたりの電力ロス (1) と偏りの発生頻度 (2) から、 (3) が 8.41mW 以下であれば、実際の消費電力が従来方式より改善可能となるはずであるが、 (4) から判るように提案方式の消費電力が従来方式より少なく、これは偏りの継続時間が長いためである。このように偏りの発生頻度が大きく、継続時間が長い利用シーンにおいては提案方式の削減効果が確認できる。

しかし、ブラウザでは地図に比べ負荷の緩急

が小さい事から、一回あたりの電力ロス (1)、偏りの発生頻度 (2) の双方が小さいため、逆に観測コストを吸収する事ができず、消費電力が増加している。

表 1: 実験結果

アプリ	(1) 電力ロス (mW)	(2) 発生頻度 (回/分)	(3) 観測コスト (mW)	(4) 消費電力(mW)	
				提案方式	従来方式
地図	0.276	30.5	15.43	510.5	513.3
ブラウザ	0.157	3.5	21.21	492.3	472.9

5. おわりに

本研究ではスマートフォンのマルチコア CPU の制御方式に対して、スレッドの動作状況を観測し、無駄なコア数の増加を抑制することで、消費電力を削減する手法を提案し、その消費電力特性を評価した。従来方式に比べ、偏り一回あたり平均 0.22mW 消費電力を削減できる可能性があり、スレッド負荷観測コストを含めても削減効果のある利用シーンがある事が確認できた。

今後の課題として、スレッドの監視コストを削減すること、および、アプリにより異なる並列度の特性に応じて監視対象のスレッドや方式で用いる様々な閾値を切り替えることがある。

参考文献

- 1) Kernel org : Linux CPUFreq, available from (<http://www.kernel.org/doc/Documentation/cpu-freq/>) (accessed 2012-12-21).
- 2) Rangan, K. and Wei, G. and Brooks, D. : Thread motion: fine-grained power management for multi-core systems, Proceedings of ISCA '09 , pp. 302-313 (2009).
- 3) Texas Instruments : Leveraging the Benefits of Symmetric Multiprocessing (SMP) in Mobile Devices, available from (<http://www.ti.com/lit/wp/swpy022/swpy022.pdf>) (accessed 2012-12-21).
- 4) GITORIOUS org : Hotplug, Linux CPUFreq, Sec2.6, available from (<http://gitorious.org/pandroid/kernel-omap/blobs/9c763866163cd9bca8ef914e8dde7502f993699a/Documentation/cpu-freq/governors.txt>) (accessed 2012-12-21).
- 5) Qualcomm : Snapdragon S4 Processors: System on Chip Solution for a New Mobile Age, available from (http://www.qualcomm.com/media/documents/sn_apdragon-s4-processors-system-chip-solutions-new-mobile-age) (accessed 2012-12-21).