

## CPU 使用率とメモリのプロセス間交換に もとづく動的なスループット向上手法

岡本 太一<sup>†</sup> 前田 敦司<sup>†</sup> 山口 喜教<sup>†</sup>

### 1. はじめに

Web サーバでは Apache HTTP Server をはじめ MySQL, memcached などの複数のアプリケーションが動作しているのが一般的である。こうした状況では、どのプロセスにどれだけの CPU 使用率やメモリ使用量を割り振ればいいのか。

本研究では、計算資源として CPU 使用率とメモリ使用量の適切な比較尺度や等価となる交換比率について考察する。またその等価交換比率をもとに、プロセスが最高のスループットを出せる CPU 使用率とメモリ使用量の割り当ての組を自動的に探索する手法について議論する。この手法を用いることで、各プロセスが限りある資源を最大限有効に利用して最高のスループットを出せるようにするメソッドの確立を目指す。また、プロセス間で資源を交換する一般的な資源管理のインターフェースについて展望する。

### 2. 提案の概要

#### 2.1 時間と空間にトレードオフのあるプログラム

時間と空間にトレードオフのあるプログラムは、実行時にメモリを多く割り当てることで、もしくは CPU 使用率を多く割り当てることで同じ出力を得るまでの実行時間が短くなる（スループットが向上する）。

Apache HTTP Server のような Web サーバは、多くのメモリを利用して接続を処理する子プロセスを大量に生成することで、単位時間に処理できる HTTP 要求数が増大する。

このようなプログラムにおいて、CPU 使用率の増大によるスループット向上は線形と考えられる。それに対し、メモリ使用量の増大によるスループット向上は一般に飽和する（上に凸な右上がりの曲線になる）と考えられる。本研究で対象とするプログラムは、このような特性をもつと仮定する。

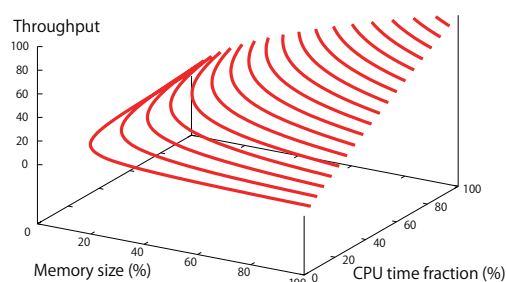


図 1 資源とスループットの関係 (3 次元)

このような特徴を持つプログラムは、微小な範囲で CPU 使用率をメモリ使用量で代替できると考えられる。逆も同様に代替できると考えられる。

#### 2.2 CPU 使用率とメモリ使用量の交換

CPU 使用率およびメモリ使用量の増加とスループット向上の関係より、この 2 つの資源とスループットの関係が図 1 のように表せる。

図 1 のグラフ上の曲線は、同じスループットを出せる資源配分（CPU 使用率とメモリ使用量の組み合わせ）を結んでいる。また、ある点における曲線の接線の傾きは、そのスループットにおける CPU 使用率とメモリ使用量のスループット上等価となる交換比率を表している。つまり、このグラフを用いることで、CPU 使用率とメモリ使用量のスループットへの寄与が等価となる量がわかる。

#### 2.3 スループットが向上する理由

プロセス A および B のスループット関数  $T_A, T_B$  は一般に異なる。したがって、それぞれのプロセスの資源配分の座標における接線の傾き（資源の等価交換比率）も異なるといえる。これは、プロセスが行っている処理はそれぞれ異なったり、プロセスの内部状態もそれぞれ異なったりするためである。

もし、あるプロセスにおいて、プロセスにおける資源の等価交換比率と異なる比率で資源が交換できたとする。そのときは資源を交換することによって、より高いスループットの点へ遷移することが可能である。プロセス A および B の傾きの中間の比率で資源を

<sup>†</sup> 筑波大学大学院 システム情報工学研究科  
University of Tsukuba, Graduate School of Systems  
and Information Engineering

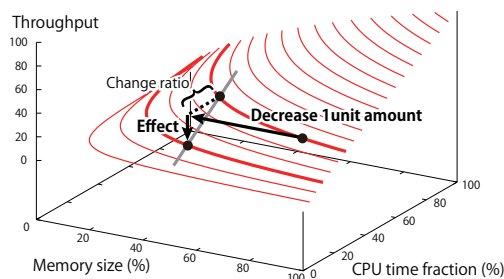


図 2 近傍探索による交換比率の推定 (3 次元)

交換することにより、以下の 2 つのことがいえる。

- 双方のプロセスのスループットが向上する (パレートの意味でスループットが改善する)
- プロセスどうしの等価交換比率の差が縮小する

### 3. 近傍探索

実際のプログラムでは処理が入力によってある程度の範囲内で変化し続けることが予想され、プロセスの内部状態は絶えず変動する。これに対して、動的に等価交換比率を推定する近傍探索の手法を提案する。本手法は現在の資源配分を微小な範囲で変更し、スループットの変化を測定することで、現時点における資源の等価交換比率を求める。

近傍探索は以下の手順で行う。

- (1) メモリ使用量を単位量減らす (図 2 に「Decrease 1 unit amount」として示した)
- (2) 結果としてスループットに変化がおきる (図 2 に「Effect」として示した)
- (3) スループットは CPU 使用率に比例するという仮定より、メモリを単位量減らした時に同じスループットを維持するのに必要な CPU 使用率がわかる
- (4) メモリ使用量を減少させる前の CPU 使用率との差から、メモリ単位量と同等にスループットに寄与する CPU 使用率がわかる (図 2 に「Change ratio」として示した)

### 4. 資源交換の枠組み

本研究において、資源の交換に必要と考えている機能は 2 つある。

ひとつめの「プローブ的役割」はプロセスの資源配分を管理・監視し、またプロセスの資源配分を制御する機能である。他にも、プロセスに対して定期的に近傍探索を行わせ、資源の等価交換比率を求める。この機能は、アプリケーションごとに作成されるエージェントとして、もしくはアプリケーション内にサブルー

チンとして実装されるべきである。

ふたつめの「マッチメーカの役割」は各プロセスのプローブから得た情報の一元管理を行うとともに、その情報をもとにプロセスどうしの資源のマッチングを行う機能である。この機能は計算機に対して 1 つあれば充分である。プローブからの決まった情報を、決まった形でマッチングするだけなので、汎用性のあるリソースデーモンという形で実装されるべきである。

## 5. 実験

Apache HTTP Server に対して、実際にスループットが測定可能かを調べる実験を行った。Apache HTTP Server における「スループット」とは、単位時間に処理することのできた HTTP 要求数であると、ここでは定義する。また、MaxClients という設定値をメモリ使用量の代理変数であるとみなした。

実験は PHP 言語で Web アプリケーションの動作を模倣するプログラムを作成し、そのプログラムに対して ApacheBench を用いて連続アクセスすることで行った。Web アプリケーションの動作の模倣には、str\_repeat 関数を用いて 5MiB のメモリ確保を行うとともに、usleep 関数によって 200ms 処理を遅延させることで行った。

実験の結果、メモリ使用量とスループットに有意な関係が得られた。

## 6. おわりに

本研究では、時間と空間にトレードオフのあるプログラムが複数存在するシステムにおける、CPU 使用率とメモリ使用量の最適な資源配分のモデルについて述べた。このモデルでは、個々のプログラムのスループットに対する寄与として等価となる、CPU 使用率とメモリ使用量の交換比率を用いて資源配分を決定するものである。また、この等価交換比率を用いて CPU 使用率とメモリ使用量を交換することにより、すべてのプロセスで資源配分の最適化を図るインターフェースを提案した。

今後の展望として資源交換を行うエージェントとリソースデーモンの実装、資源の初期配分も含めたシステム全体での最適化、CPU 使用率とメモリ使用量以外の資源への応用などが考えられる。

## 参考文献

- 1) 前田敦司: 計算資源の配分に関するプログラム間のインターフェース, 夏のプログラミング・シンポジウム報告書, pp. 59-66 (2009).