

科学研究費助成事業 研究成果報告書

平成 26 年 6 月 3 日現在

機関番号：12102

研究種目：挑戦的萌芽研究

研究期間：2011～2013

課題番号：23650011

研究課題名(和文) プロセスへのマイクロ経済学的資源割り当て

研究課題名(英文) Microeconomic resource allocation for processes

研究代表者

前田 敦司 (MAEDA, Atusi)

筑波大学・システム情報系・准教授

研究者番号：50293139

交付決定額(研究期間全体)：(直接経費) 2,800,000円、(間接経費) 840,000円

研究成果の概要(和文)：実行時にメモリを多く与えることによって速度が向上するプログラムを複数実行するとき、性能指標(効用)をプロセスごとに定義し監視する手段と、計算資源をある交換比率(価格)で取引できる機構を導入することで、資源配分を動的に最適化し、システム全体の性能を向上させる経済学的な枠組みを得ることを目標とし、Webサーバで用いられる複数のプログラムを対象に実現法について調査し、評価を行った。

研究成果の概要(英文)：Some programs run faster if given more memories. When running more than one such programs, throughput of the programs can be optimized by allocating the resource appropriately. Our goal was to give a microeconomic framework to improve system's overall performance by introducing methods to define and monitor per-process performance measure (utility) and mechanisms to enable processes to trade resources in some exchange rate (price). We have examined and evaluated implementation methodology with multiple programs used in web servers.

研究分野：情報学

科研費の分科・細目：情報学・ソフトウェア

キーワード：オペレーティングシステム 資源割り当て

1. 研究開始当初の背景

OS がプロセスにメモリや CPU 時間などの計算資源を割り当てる手法については膨大な研究がなされてきた。また、近年ではマルチコア CPU のキャッシュにおいて個々のコアに割り当てるキャッシュ量を自動的に増減して、トータルの IPC を最大化する手法が活発に研究されている。しかし、プログラムのスループットは IPC と異なり OS 等が外部から観察することは一般にできない。たとえば、ガーベッジコレクション(GC)を用いるシステムでは、メモリ量を削減すると GC に費やす時間が増大して本来のワークロードの処理速度が低下するが、これを IPC の変化として検出することはできない。同様に、計算結果をソフトウェア的なキャッシュに保存して再利用するプログラムにおいては、キャッシュの量を削減すると計算に費やす時間が増大し、処理速度が低下するが、これも外部から観察することは困難である。

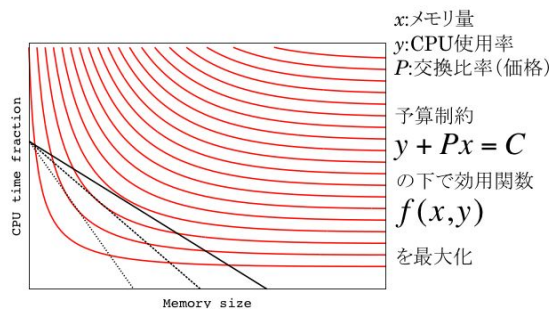


図1：交換比率が変化するときの、予算制約直線のもとでの効用最大化

この問題は、プログラム個々の性能指標を経済学の効用関数と解釈し、予算制約のもとで効用関数を最大化する経済主体の行動をシミュレートすることでシステム全体の資源配分を最適化できるのではないかと考えられる(図1)。

2. 研究の目的

本課題では、ミクロ経済学の知見を利用して、単一の計算機内での資源配分を最適化する枠組みを作成し、その効果を検証することを目的とした。この枠組みは、個々のモジュールが、独自の性能指標を定義する機構、資源量の変化に応じた性能指標の変化を測定し、そこから資源量1単位と等価なCPU時間率(資源の価格)を計算する機構、資源価格を提示してモジュール間で資源とCPU時間を取引する機構等から成り、当初は単一のプログラム内でのコンポーネント間の資源最適化、次に単一の計算機内でのプロセス間の資源配分の最適化に対して適用する。

3. 研究の方法

研究の最初の段階では、Packrat Parsing や

正規表現の処理など、計算結果をキャッシュすることにより速度が向上するいくつかのプログラムに関して実際に本研究の提案手法を適用し、プログラム内のモジュール間におけるメモリ量を、プログラムのスループットを最大化するように自動的に配分する機構を実現することを目指した。

この段階における実装の形態としては、関数を自動的にメモ化し、さらにメモ化に用いるキャッシュ量を自動的に調整する高水準なライブラリとして開発を行う。ヒット率および計算時間の統計から、メモリ量を変化させた時のスループットの変化を推定し、メモリ量を自動的に増加あるいは減少させるが、この時のキャッシュ表のデータ構造およびキャッシュからあふれるデータの置換ポリシー、メモリを増減させる際の変化量などを実験により決定する必要があると考えられた。

この段階で、課題の基本的なアイデアの効果を検証するとともに、収束の速度・解の精度・オーバーヘッドなどに関する基礎的な知見を得ることを目指した。ここで収束の速度等の性能が実用に耐える範囲にないと思われる場合には、メモ化に限定せず、問題の粒度等から提案手法が実用的な性能を発揮する可能性のある問題を幅広く探索する計画であった。

次の段階では、プログラム間においてメモリとCPU時間を取引することでシステム全体のスループットを最大化する資源配分を自動的に達成するような協調資源配分機構を実装する計画であった。この段階では、プロセスが実行を開始するときに与えられる資源の初期配分(メモリを買う取引に利用できるCPU時間の配分)は、平等な場合に限定し、個々のコンポーネント間のインターフェースと、ユーザプログラムの資源利用量の精密な制御にフォーカスした研究を行うこととした。

その後は、Webサーバとアプリケーションサーバ、データベースサーバなど、協調する複数の実用プログラムにおいて、提案する協調資源配分機構を用いて効率的な資源配分を達成することを目指し、全てのプロセスを統合的に評価したシステム全体の性能指標を与えることによって初期資源配分を自動的に変化させ、システム全体の性能が最適となるような資源配分を自動的に達成する機構を実現する計画で研究を進めた。

4. 研究成果

メモリとCPU時間という2種類の資源の、単一のシステム内における配分問題を取り扱うフレームワークを提案し、プロトタイプの

実装を行なった。このプロトタイプは、プロセス間でメモリとCPU時間を取引することで、個々のプロセスのスループットを向上させることを目指すものである。プロトタイプを、WebサーバとKey-Value Storeに適用して実験を行い、スループット向上の効果について確認した。その成果を以下の通り学会で発表した。

- 岡本太一, 前田敦司, 山口喜教, 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, 第53回プログラミング・シンポジウムプログラム, (Jan, 2012)
- 岡本太一, 前田敦司, 山口喜教, 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, 先進的計算基盤システムシンポジウムSACSIS2011ポスター発表, (May, 2011)

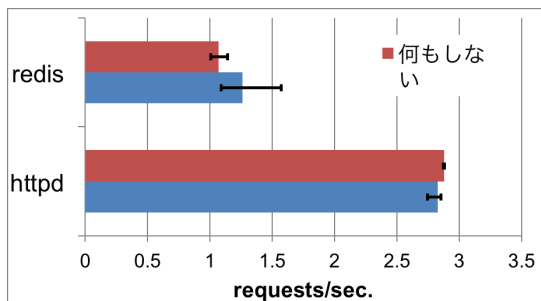


図2: 資源交換の前後におけるスループットの変化

この時点では、等価交換比率の測定誤差や、資源配分を変更する際のオーバーヘッドによって、かえってスループットが低下する場合があります。この知見が得られた(図2)。

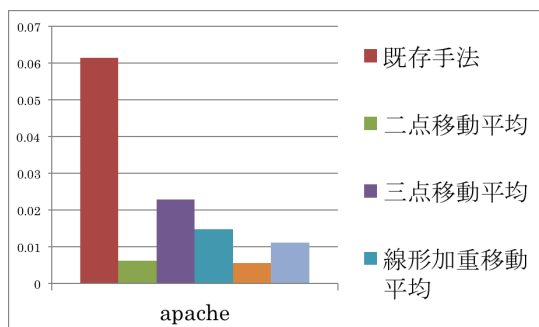


図3: さまざまな推定手法における分散

メモリとCPU時間という2種類の資源の、単一のシステム内における配分問題を取り扱うフレームワークのプロトタイプ実装について、取引に用いる等価交換比率のばらつきを減らし、精度を改善することで、誤ってスループットが低下する交換を行う可能性を減らし、より有効なスループット向上を図るものである。そのため、サンプリングした測定値をそのまま等価交換比率とする従来の手法に代えて、移動平均や移動メジアンな

どの手法を用いて測定値からより精度よく真の交換比率の推定を行った結果、交換比率のばらつきを削減し、安定した性能が得られることを実験により確かめ(図3)、以下の論文にまとめた。

- 大塚優樹, 資源配分における等価交換比率の推定精度向上手法, 平成24年度情報科学類卒業研究論文, (Mar, 2013)

もう一つの改良手法は、Webサーバの負荷が100%に達しない時でも効用(スループット)の値を適切に取り扱えるよう、スループットの定義として実時間あたりの処理リクエスト数に代えて、CPU時間で正規化した処理リクエスト数を用いたものである。この結果、現在はプロトタイプ実装のオーバーヘッドが大きいと、顕著な性能向上は得られなかったものの、将来オーバーヘッドが低減されれば、適切な資源交換によって低負荷時にも性能向上を得られる可能性をたしかめる結果が得られた。この研究成果は、以下の論文にまとめた。

- 中森亮介, 計算資源配分のための性能測定・制御手法, 平成24年度情報科学類卒業研究論文, (Mar, 2013)

また、構文解析器においてメモ化を用いてCPU時間とメモリのトレードオフを改善する手法について研究を行い、その成果を以下のとおり発表した。

- 森永孝仁, 前田敦司, 山口喜教, Packrat Parserのメモ化領域の自動調整手法, 第92回情報処理学会プログラミング研究会, (Jan, 2013)

単一プログラムよりもシステム全体の方が資源の合計量および利用資源の測定方法が明確で、かえって扱いやすいことが判明し、本研究のアイデアの基本的な有効性が実証できた点で、非常に重要な成果であったと考えられる。また、資源利用状況から資源の交換比率をより高い精度で推定する研究を行い、以下の論文としてまとめた。

- 小山駿介, メモリ・CPUのトレードオフ計算資源配分における性能推定方式, 平成23年度情報科学類卒業研究論文, (Mar, 2012)

メモリとCPU時間という2種類の資源の、単一のシステム内における配分問題を取り扱うフレームワークのプロトタイプ実装について、対象としたプログラムのひとつであるkey-value store (KVS)では、メモリを削減する際に設定ファイルを書き換えてプログラムを停止・再起動する以外に手段がなく、使用メモリ量の削減に大きなオーバーヘッ

ドがかかり、資源最適化による性能向上の効果が相殺されてしまうことから、既存の KVS と互換性があり、メモリ量を削減しても性能上のペナルティの小さい KVS 実装を新たに作成した。この成果は、以下の論文にまとめた。

- 阿部聖昂「メモリ使用量を動的に変更可能な Key-Value Store の開発」(平成 25 年度情報科学類卒業研究論文)

また、資源を最適化する際の単位として、プロセス単位あるいは計算機全体を単位とするのではなく、ひとつの計算機内を論理的に区分することによって複数のサーバとして使用するコンテナ仮想化の形態を対象とし、コンテナ間での資源向環によって、コンテナをホストする計算機全体の資源の利用効率を向上させることを目的とした実装を行い、予備的な実験を行った。これに加え、さらにメモリ使用量を動的に変更可能な KVS の開発と合わせて、以下の発表を行った。

- 阿部聖昂, 中森亮介, 前田敦司 「コンテナ型仮想環境における計算資源交換に基づく性能最適化」第 16 回プログラミングおよびプログラミング言語ワークショップ (PPL2014) ポスター発表

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[学会発表](計 4 件)

岡本太一, 前田敦司, 山口喜教, 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, 第 53 回プログラミング・シンポジウムプログラム, (2012 年 1 月 6 日, 静岡県熱海市)

岡本太一, 前田敦司, 山口喜教, 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, 先進的計算基盤システムシンポジウム SACSIS2011 ポスター発表, (2011 年 5 月 25 日, 東京都中央区)

森永孝仁, 前田敦司, 山口喜教, Packrat Parser のメモ化領域の自動調整手法, 第 92 回情報処理学会プログラミング研究会, (2013 年 1 月 15 日, 鹿児島県奄美市)

阿部聖昂, 中森亮介, 前田敦司 「コンテナ型仮想環境における計算資源交換に基づく性能最適化」第 16 回プログラミングおよびプログラミング言語ワークショップ (PPL2014) ポスター発表, (2014 年 3 月 5 日, 熊本県阿蘇市)

6. 研究組織

(1) 研究代表者

前田 敦司 (MAEDA ATUSI)
筑波大学・システム情報系・准教授
研究者番号: 50293139

(2) 研究分担者

山口 喜教 (YAMAGUCHI YOSHINORI)
筑波大学・教授
研究者番号: 00312827