

トレース駆動シミュレーションを用いた
計算機性能評価技術の検証

3F-1

茂木 久

(株)東芝 情報・通信システム技術研究所

1 はじめに

既存の計算機で収集したアドレステレースをワークロードとして、測定不可能な開発途上の計算機の動作をシミュレートすることによりその性能を予測する技術(トレース駆動シミュレーション技術)を開発した。[1][2]

今回この技術の正当性を検証するために、既存の計算機に対してトレース駆動シミュレーションを適用し、この結果を実測結果と比較した。本稿ではその検証結果について述べる。

2 性能予測技術の概要

本性能予測技術による評価の手順を図1に示す。その概要は以下の通りである。

(1) アドレステレースを収集する

評価対象のアプリケーションプログラムに参照した論理アドレスをファイルに出力するコードを埋め込み、これを実行することによりアドレステレースを収集する。トレース収集ツールを改造し、ユーザモードだけでなくカーネルモードのアドレステレースも収集している。カーネルモードについては、収集ツールの制約により一部トレースを収集できていない部分がある。[2]

(2) シミュレーション

アドレステレースをワークロードとして、評価対象の計算機のキャッシュとソフトウェア(プロセスのディスパッチ、アドレス変換など)をモデル化したキャッシュシミュレータと、バスとメインメモリをモデル化したバスシミュレータに与え、性能予測を行う。[1]

3 検証方法

図2で示す手順で検証を行なった。既存の計算機で事務処理アプリケーションを実行し、アドレステレース

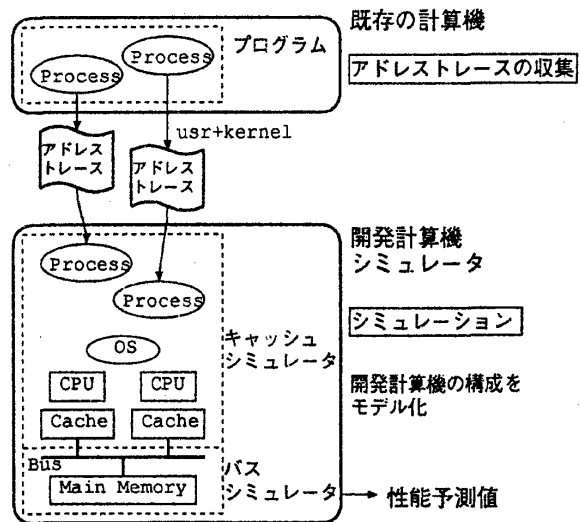


図1: 性能予測技術

を収集するとともに性能データを実測する。そして既存の計算機をモデル化したシミュレータを作成し、上記のアドレステレースを用いて得られた性能予測値と、上記の実測値と比較することによりトレース駆動シミュレーション技術の検証を行なった。なお、バスシミュレータのモデルは比較的単純であるので、今回はキャッシュシミュレータについてのみ検証を行なった。

アドレステレースの検証方法

改造したツールで収集したカーネルモードを含めたアドレステレースの正当性を検証するために、事務処理アプリケーションの処理単位あたりのアドレステレースの実行命令数を実測値と比較する。その理由は、実行命令数が合えば、アドレステレースは実際のプログラムの振舞いをおおよそ反映していると判断できるためである。

キャッシュシミュレータの検証方法

キャッシュシミュレータの検証は、キャッシュのミス率を予測値と実測値で比較することで行う。ミス率は、キャッシュの振舞い、メモリアクセス順序を反映するものであり、シミュレータの検証の指標としてこれを選んだ。

Validation of Performance Evaluation Technique
Based on Trace Driven Simulation
Hisashi MOGI
Information & Communication Systems Laboratory,
TOSHIBA Corporation

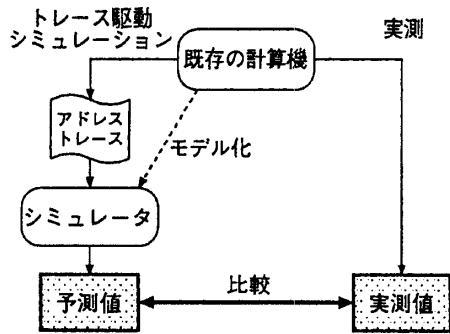


図2: 検証方法

4 検証結果

4.1 アドレストレースの検証

アドレストレースと実測値による実行命令数をカーネルモードとユーザモードに分け表1に示す。

表1: アドレストレースと実測値の実行命令数

	実行命令数 / 処理単位 [補正值] (×100万)		
	トータル	ユーザ	カーネル
アドレストレース	3.7 [4.1]	3.2	0.5 [0.9]
実測値	3.7	2.8	0.9

この表からカーネルモードにおける実行命令数の差が大きいことがわかる。その理由は、カーネル内部でトレースを収集するためのコードを埋め込むことができなかったモジュール(レジスタの使用に関して制約のある部分など)の実行される割合が、今回収集した事務処理アプリケーションでは高かったためだと思われる。該当するモジュールの呼出回数とその平均実行命令数から実行命令数を補正すると(補正値は表中の[]内の値)、表に示すように実測値と一致することがわかった。この結果より、アドレストレースはほぼ正しく収集できたと考えられる。

一方、ユーザモードにおける実行命令数にも少し差がある。アドレストレースを収集した事務処理アプリケーションは実行命令数が大幅に異なる複数種類の処理単位から構成される。実測した時間に対してアドレストレースを収集した時間が短かったために、これらの処理単位の種類ごとの比率が両者で異なり、これがユーザの実行命令数の違いとなって現れたのだと思われる。実際の評価では、処理単位の種類別にアドレストレースを収集し、目的の比率に配分したワークロードを作成するなどの工夫が必要である。

4.2 キャッシュシミュレータの検証

シミュレータと実測によるキャッシュミス率(実行命令数に対するキャッシュミス回数の割合)を表2に示す。

表2: キャッシュミス率の予測値と実測値

	キャッシュミス率 (%)	
	命令キャッシュ	データキャッシュ
シミュレーション	7.1	2.7
実測	6.0	4.1
(ユーザのみ)	6.8	4.0
(カーネルのみ)	3.2	4.4

この表から、シミュレーションによるキャッシュミス率が実測値のそれに対して、命令キャッシュでは大きく、データキャッシュでは小さいことがわかる。この原因は以下のように考えることができる。

今回のワークロードには、カーネルの一部のモジュールのトレースが含まれていない。このモジュールはデータ移動に関するもので、小さなループで構成され、メモリアクセスの頻度が高いという特徴を持つ。このため本来のアドレストレースに対し、以下のような違いとその影響がある。

- データ参照数が少ない。
→ データキャッシュミス率を下げる。
- 命令のループ部分が少ない。
→ 命令キャッシュミス率を上げる。

以上を考慮するとキャッシュシミュレータにおける動作は概ね正しいと思われる。

実測値より、命令キャッシュのミス率がユーザとカーネルで傾向が異なることがわかった。従って、より精度の高い性能予測を行うためにはカーネルモードを含むアドレストレースが重要であると考えられる。

5 おわりに

実測との比較を通して、トレース駆動シミュレーション技術の正当性について検証した。補正したアドレストレースの処理単位あたりの実行命令数はほぼ一致しており、正しさを確認できた。また、キャッシュミス率からシミュレータの動作がおおよそ実機と一致していることを検証できた。

今後、バストランザクション等の他の項目に関しても予測値と実測値の比較を行い検証を続ける予定である。

参考文献

- [1] 武内, アドレストレース駆動シミュレータの開発, 情報処理学会第51回全国大会, Sep.1995
- [2] 堀内, トレース駆動シミュレーションによるOSの性能評価, 情報処理学会第52回全国大会, Mar.1996