

## 高速並列処理ワークステーション (TOP-1) 性能予測

7N-5

大庭 信之

日本アイ・ビー・エム株式会社 東京基礎研究所

### 1. はじめに

TOP-1のような共有メモリ・共有バス方式のマルチプロセッサシステムでは、従来のソフトウェアとの親和性は良好だが、バスがシステム性能のボトルネックになりやすい。そのためTOP-1ではマルチキャッシュを搭載し、バストラヒックの軽減を実現している。TOP-1を研究・開発するにあたり、プロセッサの個数・バス構成・キャッシュなどを十分検討する必要があった。本稿では、待ち行列モデルを用いたTOP-1の性能評価に関して、その方法、ツール、モデリング及び評価結果について述べる。

### 2. シミュレーションの方法

共有バス・共有メモリから成るマルチプロセッサは、システム全体の性能がハードウェア資源利用の競合によって大きく左右される。このような資源競合環境をシミュレートするためには、待ち行列モデルを用いるのが適していると考えられる。

待ち行列モデルを構築し評価するためのツールとしてIBMワトソン研究所で開発されたソフトウェアパッケージRESQ (RESEARCH Queueing package) [1]を使用した。RESQは待ち行列モデルの構成要素の記述が従来のシミュレーション言語(例えばGPSS)に比較して、より抽象化された形で記述でき、そのためのテンプレートも豊富に用意されているため、マルチプロセッサシステムの性能評価に有効である。さらに、キャッシュ制御・スヌープ制御はPL/Iで記述し、RESQとリンクさせてシミュレーションを行っており、マルチプロセッサシステムの細かな制御に対しても十分応用可能である。

### 3. シミュレーションモデル

図1にTOP-1の性能評価に用いた待ち行列モデルを示す。モデル全体はMPE (Micro Processing Element)・ARB (Bus Arbiter)・BUSの3つの機能サーバから構成され、複数台のCPUから発生するメモリアクセス要求は系内を流れるトークンで表現される。トークンはキャッシュがリードヒットして要求がシステムバス上に送出されない場合はそのままCPUへ戻る。もし、キャッシュがミスした場合などでバスを要求する場合はARBブロックを通過しバスへの待ち行列に入る。続いて、トークンはバスが使用可能になった時点でメモリサイクルで与えられるサービ

ス時間をバスで消費し、CPUへ戻る。また、MPE並びにBUSはスヌープ機能をシミュレートするため、お互いに情報を交換できるようになっている(図2、図3)。

次に、シミュレーションに用いたハードウェア・ソフトウェアパラメータを示す。

#### 【ハードウェアパラメータ】

|                      |       |
|----------------------|-------|
| Processor Cycle Time | 100ns |
| Cache Cycle Time     | 100ns |
| Arbitration Time     | 50ns  |
| Memory Cycle Time    | 200ns |

#### 【ソフトウェアパラメータ】

|                                 |                |
|---------------------------------|----------------|
| Processor Performance (no wait) | 5MIPS          |
| Instruction Fetch Rate          | 0.38word/cycle |
| Data Read Rate                  | 0.23word/cycle |
| Data Write Rate                 | 0.12word/cycle |
| Data Shared Rate                | 0.05-0.90      |
| CPU Bus Utilization (no-wait)   | 0.73           |

### 4. シミュレーション結果

図4にヒット率とシステム性能の関係をバスの本数をパラメータとして求めた結果を示す。キャッシュヒット率が低い領域でバスを2本にする効果が特に大きい。

図5にプロセッサの台数とシステム性能の関係をキャッシュヒット率をパラメータとして評価した結果を示す。キャッシュヒット率が0.95以上得られればプロセッサの台数を増やすことによってシステム性能をほぼ直線的に向上することができるが、ヒット率が0.80を下回るとプロセッサ数が10台程度まで性能向上が得られることがわかる。

図6はヒット率とシステム性能の関係をデータのシェア率をパラメータとして求めたものである。ヒット率が0.60以下の低い領域ではバスの転送能力が飽和状態になり、システム性能はデータのシェア率が高くなってもあまり変わらない。一方、ヒット率が高い領域ではデータシェア率によってシステム性能がかなり変化することがわかる。これはデータがシェアされるとデータ書込み時にコピーを持つプロセッサのデータを同時に更新しなければならなくなり、そのためにバス使用を要求するためである。しかしながら、データのシェア率が90%と非常に高い値に達してもシステム性能はシェア率10%時に比較して4割程度の減少にとどまっている。

5. むすび

以上のシミュレーション結果より、システム全体の性能はキャッシュのヒット率、データのシェア率に大きく依存する事が分る。

本稿では主にヒット率をパラメータとして行った性能評価について報告したが、シングルプロセッサ環境で収集したCPUアクセスト्रेसデータを用いたシミュレーションも行い、高いヒット率が得られることと高いシステム性能が得られることを確認した。また、TOP-1のキャッシュコントローラチップには、ヒット率・データシェア率などマルチプロセッサ環境での重要な統計情報を収集するためのハードウェアが組みこんであり、これらを用いたOS・アプリケーションを含む実際の動作環境での総合的な性能評価を今後行う予定である。

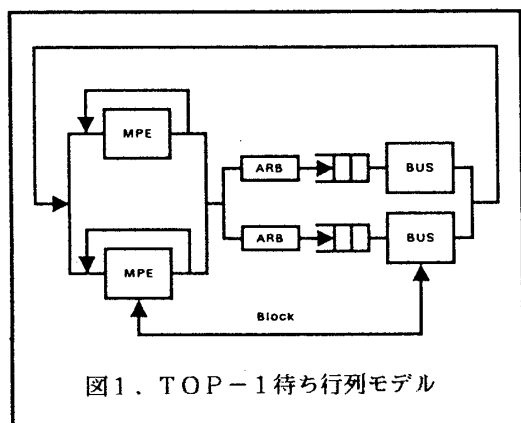


図1. TOP-1 待ち行列モデル

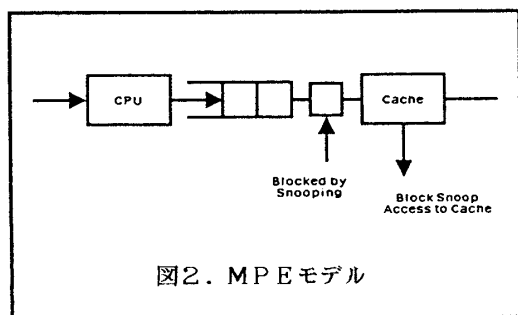


図2. MPEモデル

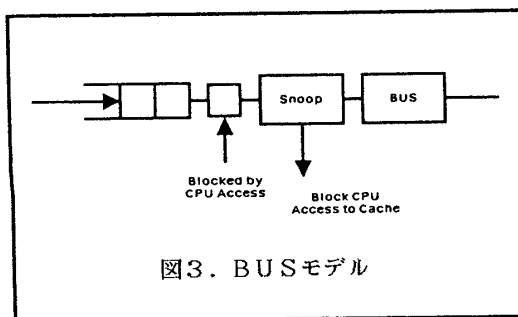


図3. BUSモデル

【参考文献】

[1]村田正幸：“待ち行列システムのモデリング言語RESQ”，オペレーションズリサーチ，1985年7月。

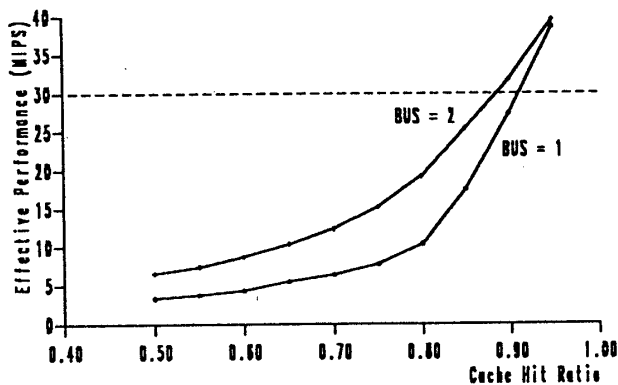


図4. ヒット率-性能

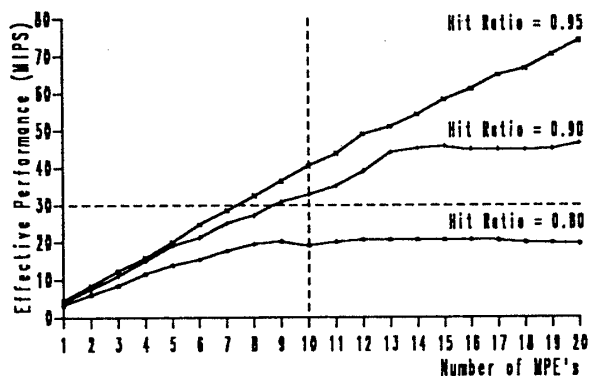


図5. プロセッサ数-性能

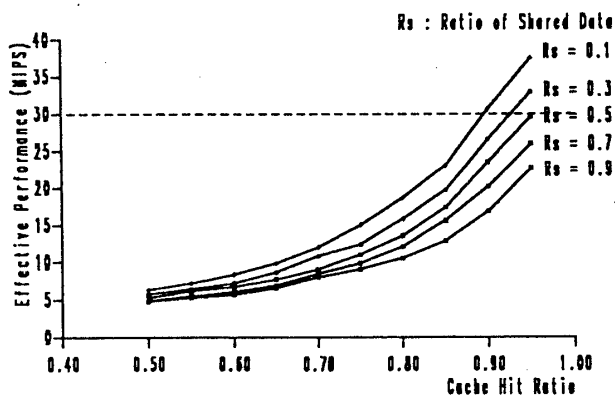


図6. ヒット率-性能