

可変ページを用いた仮想記憶管理に関する研究

5Q-5

伊藤 修一 光澤 敦

NTT 情報通信研究所

1 はじめに

近年のチップの動作周波数の高速化は著しいが、それとともに仮想物理アドレス変換の時間的コストも増大する傾向にある [1]. 性能を向上させるためには、このアドレス変換の時間的コストを削減することが重要である。

この問題に対処するため、最近のプロセッサでは TLB (transfer lookaside buffer) にスーパーページの機能が実装されていることが多い。スーパーページとは、TLB のエントリにサイズのフィールドを持つものことであり、従来より大きい領域のマッピングが可能となっている。スーパーページの機能を用いることができれば、TLB のヒット率を上げることができるが、それにはオペレーティングシステム (OS) がスーパーページの機能を意識した TLB の管理を行う必要がある。

そこで本研究では、可変ページを用いた仮想記憶管理機構を提案する。従来、仮想記憶管理ではページサイズが固定されていたが、本機構では複数サイズのページを扱うことができる。可変ページを実現するには、仮想記憶管理機構がページ表で可変サイズのページを呼出すことと保存することができる必要がある。物理記憶管理機構が複数のサイズのページフレームを管理できる必要がある。

本稿では、まず第2節で TLB のスーパーページ機能について概観し、第3節で仮想記憶管理の変更点について述べる。そして第4節で物理記憶管理で考察しなければいけない問題についてふれ、第5節で実装と評価の方針を述べる。最後に第6節で本稿をまとめる。

2 TLB のスーパーページ機能

スーパーページの機能を持つ TLB のエントリは図1の構造を持つ。すなわち、プロセス ID (PID フィールド)、マッピングするメモリ領域のサイズ (SZ フィールド)、仮想ページ番号 (VPN フィールド)、物理ページフレーム番号 (PFN フィールド) からなる。

SZ フィールドで指定できるサイズは基本となるベースページサイズの2のべき乗であり、一般には4KB程度か

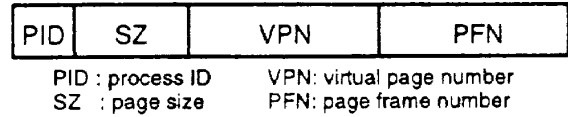


図 1: TLB のスーパーページ

ら数 MB 程度を一つのエントリでマッピングできる。現在、MIPS[2], Alpha[3] など、多くのプロセッサが TLB にスーパーページを実装しており、シミュレーションによってその効果が実証されている [4].

3 仮想記憶管理の変更

仮想記憶管理でスーパーページを扱う最も簡単な方法は、ページ表においてエントリの複製をすることである [5].

図2はページ表エントリの複製に関して説明した図である。スーパーページで指定されているメモリ領域のマッピングサイズを、ベースページサイズで割った数だけページ表エントリを複製する。図2ではスーパーページサイズが16KBでベースページサイズが4KBである。この場合、 $16KB / 4KB = 4$  個の全く同じページ表エントリがページ表に存在することになる。

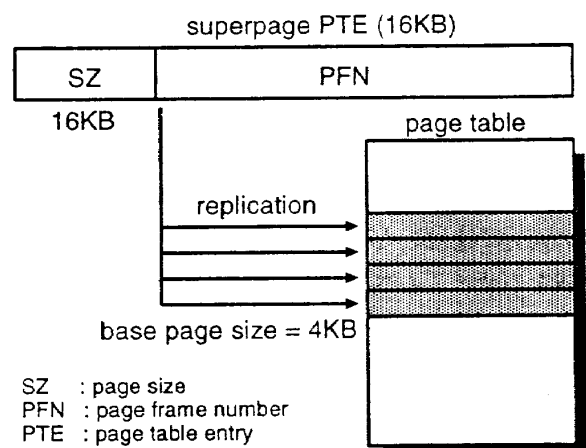


図 2: ページ表エントリの複製

#### 4 可変ページを用いる場合の物理記憶管理の問題点

可変ページのためのメモリ管理には以下の大きな三つの問題点がある。

**用意するページサイズの種類** どのサイズをページフレームのサイズとして用意すると効果が大きいのか? 多くの種類のページフレームのサイズを用意することは、物理記憶管理を複雑化し、逆効果であると我々は考える。

**ページフレームの割付けポリシー** 仮に大小二つの物理ページフレームのサイズを採用している場合を考える。今、小さなサイズの物理ページフレームしか確保できない状態であるとする。この時に小さなサイズの物理ページフレームより大きなサイズのメモリの割付け要求を受け取ったとしたら、どのように処理するべきか? 小さなページフレームの複数の割付けをするか? どうかして小さなページフレームを集めて一つの大きなページフレームにしてから割付けするか? 反対に、今、大きなサイズの物理ページフレームしか確保できない状態であるとする。この時に小さなサイズの物理ページフレームでも割付け可能なサイズのメモリの割付け要求を受け取ったとしたら、どのように処理するべきか? 大きなページフレームの割付けを許すか? 大きなページフレームを分割して小さなページフレームにしてから割付けするか? 割付けポリシーとしては、メモリの使用率によって割付け方を変えることや、割付けが要求されているデータのサイズと未使用の物理ページフレームのサイズの比によって割付け方を変えることを考えている。

**未使用なページフレームの管理方法** 従来の物理記憶管理は単一のページフレームサイズしか扱わなかった。TLBのスーパーページ機能を扱うためには、スーパーページでマッピングするメモリ領域のサイズ分、仮想空間と物理メモリの両方で連続した領域が確保される必要があり、そして、複数のサイズのページフレームの未使用を管理する必要がある。

#### 5 実装と評価

実装はMIPS R4000プロセッサ上で行う。これはMIPS R4000のTLBが割付けることのできるメモリ領域のサイズが他のプロセッサに比べて充実していることからである。具体的には4KB, 16KB, 64KB, 256KB, 1MB, 4MB, そして16MBのサイズを実装している。

OSはMach 3.0を対象とする。これはMach 3.0の仮想記憶管理のコードは4BSDと全く同じであるので、他のシステムへの移植が容易であることが理由である。

評価項目としては、アプリケーションの実行時間の他、TLBヒット率や物理ページフレームの割当て時間などを考えている。そして実装したシステムで実際のアプリケーションを長時間動作させて評価をとり、可変ページを使用した場合と使用しない場合を比較し、その優位性を考察する方針である。

#### 6 まとめ

本研究では、アドレス変換のコストが増大しているという問題を解決するために、TLBのスーパーページ機能を用いて、仮想記憶管理に可変ページを導入することを提案した。

可変ページを実現するには、ページ表で複製エントリを作り、物理記憶管理機構では複数のサイズのページフレームを管理する必要がある。しかし、物理記憶管理機構を変更するには以下の三つの問題点があった。

- 用意するページサイズの種類
- ページフレームの割付けポリシー
- 未使用なページフレームの管理方法

今回はTLBの機能のうちスーパーページのみを用いたが、将来的にはサブブロッキングやパーティション機能も使う必要があるかもしれない。これは評価の結果によって決定したい。

#### 参考文献

- [1] Mendel Rosenblum, Edouard Bugnion, Stephen A. Herrod, Emmett Witchel, and Anoop Gupta. The Impact of Architectural Trends on Operating System Performance. In *Proc.15th ACM Symposium on Operating System Principles*, December 1995.
- [2] Gerry Kane and Joe Heinrich. *MIPS RISC Architecture*. Prentice Hall, 1992. ISBN 0-13-590472-2.
- [3] Richard L. Sites. Alpha AXP Architecture. *Comm.ACM*, Vol. 36, No. 2, pp. 33-44, February 1993.
- [4] Madhusudhan Talluri and Mark D. Hill. Surpassing the TLB performance of Superpages with Less Operating System Support. In *Proc.6th International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 171-182, October 1994.
- [5] Madhusudhan Talluri, Mark D. Hill, and Yousef A. Khalidi. A New Page Table for 64-bit Address Spaces. In *Proc.15th ACM Symposium on Operating System Principles*, December 1995.