

## 拡張記憶を利用した 仮想記憶制御方式ならびに性能評価

片田 久 細内昌明 新井利明 吉澤康文  
日立製作所システム開発研究所

汎用大型計算機では、大容量で安価な拡張記憶(ES;Extended Storage)が利用可能となっている。システムのコスト性能比の向上を目的に、ESをページング装置として利用する仮想記憶制御方式を開発した。本方式はページング処理のCPUオーバーヘッドを削減する同期ページング、不活性なページをESから外部記憶へ追い出すESマイグレーションなどの機能からなる。TSS、バッチ環境において本方式を評価した。外部記憶へのページングに比べ、CPUオーバーヘッドは25%以下になり、応答時間も50%削減される。また、解析により、ESが主記憶よりもコスト性能比に優れていることを確認した。

## Paging Algorithm using Extended Storage — Its Design and Evaluation

Hisashi Katada Masaaki Hosouchi Toshiaki Arai Yasufumi Yoshizawa  
Systems Development Laboratory, Hitachi, Ltd.  
1099 Ohzenji Asao-ku Kawasaki-shi, 215, Japan

Extended Storage(ES) is available in mainframe computers. In order to improve cost-performance of system, we have developed ES paging methodology which manages storage hierarchy including ES. It consists of synchronous paging which reduces CPU time for paging and ES migration which migrates inactive pages to auxiliary storage. We have shown that CPU overhead is reduced to 25% and TSS response time is improved by 50% in TSS and batch environments. By analysis, we have also shown that the cost-performance is improved by ES more than by Main Storage(MS) extention.

## 1. はじめに

半導体技術およびマルチプロセッサ技術の進展により、計算機システムのCPU内部の能力は向上している。しかし、磁気ディスクなどの外部記憶のアクセス時間は、従来よりさほど進歩しておらず、応答時間の削減が困難な状況にある。また、外部記憶へのアクセスはCPUオーバーヘッドが大きく、スループットの向上を妨げる要因になっている。システムの性能を向上させるには、外部記憶へのアクセス頻度を削減する必要がある。

これを目的として、我々はこれまでにプログラムライブラリやデータベースを仮想記憶上に展開するXPL (eXtended Program Loading) およびHAF (High performance Access Facility)を開発してきた[1][2]。これらの機能は、アプリケーションプログラムを修正することなく、外部記憶装置へのアクセス要求をメモリへのアクセスに変換するものである。これらにより、応答時間が大幅に短縮されることと、スループットが向上することを確認している。しかし、仮想記憶にはこれまで以上に大きな負荷が発生する。十分な性能を達成するには大容量のMS (Main Storage)が必要となり、システムのコストが上昇するという問題が生ずる。

この問題を解決するため、我々は拡張記憶(ES; Extended Storage)をページング装置として利用する仮想記憶制御方式を開発した。この方式を以降ESページング方式と呼ぶことにする。ESは大容量かつ安価な記憶装置であり、ESとMSとの間で高速なデータ転送(10マイクロ秒オーダー)を行える。

本報告では、ESページング方式の設計方針と制御方式について述べるとともに、主としてTSSとバッチ環境における実測と解析結果から、ESページング方式の効果について報告する。

## 2. ESページング方式

### 2.1 ESのハードウェア

ESのハードウェア構成を図1に示す。ES上のデータは、プログラムから直接アクセスすることはできないが、MSとの間で10マイクロ秒オーダーの高速なデータ転送が可能である。転送の方法には、同期転送(転送命令の実行中CPUは次の命令を実行せずに待つ)と非同期転送(データ転送はチャネルが行いCPUは他の命令を実行可能)がある。非同期転送は大容量データを転送するときに、CPUオーバーヘッドを軽減するのに有効である。また、ESから外部記憶へのページ移動用に3媒

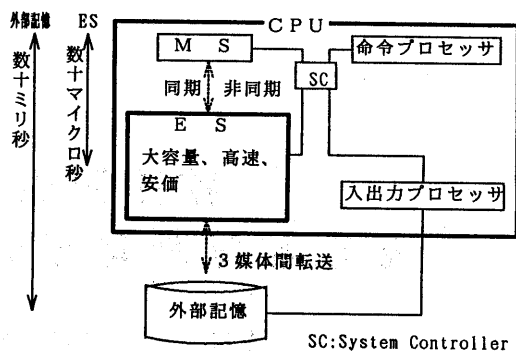


図1. ESのハードウェア構成

体間転送機能がある。これは、1回のI/Oで多量のページをESから外部記憶へ転送する機能である。

ESの単位容量当りのコストは、MSの数の1と安価であり、また、実装容量もTB (Tera Byte)以上とすることができる。

### 2.2 ESページング方式の設計方針

ESページング方式の概要を図2に示した。以下に、本方式の設計方針について述べる。

#### (1) 高速アクセスの保証

データがMSにない場合にも、ESにデータがあれば数十マイクロ秒でアクセスできる。しかし、仮想記憶の容量がMSとESの容量を超えるとデータが外部記憶に配置される可能性がある。XPLやHAFでは外部記憶へのアクセスを削減することを目的としているため、これらのデータはMSまたはESに保証する必要がある。そこで、高速性が要求されるプログラムがESを優先的に利用できる機能を設ける。

#### (2) スループット向上

応答時間が削減できてもCPUオーバーヘッドを削減できなければ、スループットは向上しない。そこで、同期転送命令を用いてESへのページングオーバーヘッドを削減する。また、少量のページ転送には同期転送を用い、大量のページ転送には非同期転送を用いる。さらに、マルチプロセッサ構成時には、同期転送中に仮想記憶管理のロックを外し並列度を向上させる。

#### (3) ESの有効利用

ESは大容量であるが、仮想記憶の容量よりも小さい。このためESを有効に利用する機能が必要になる。この機能として、以下の3つを設けることにした。これらを三階層記憶制御と呼ぶ。

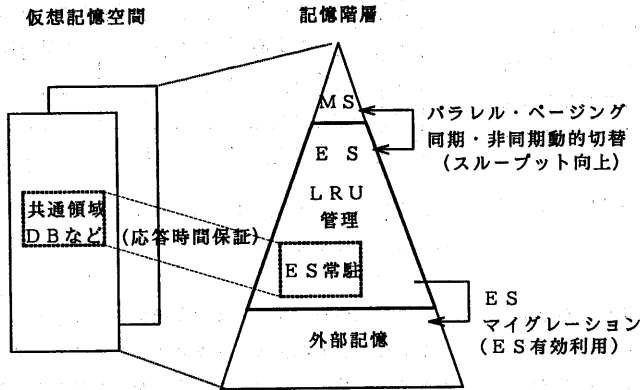


図2. ESページング方式の概要

(a) ESマイグレーション機能

一般に、プログラムやデータの参照特性にはローカリティが存在するので、未参照時間の長いページほど参照される確率は低くなる。そこで、このようなページを検出し外部記憶に追い出す。

(b) ESページアウト制御機能

MSから追い出されたばかりのページは、ES上の他のページよりも参照される確率が高いという考えに基づき、MSからページアウトしたページは必ずESに配置する。ただし、ローカリティのないデータに関しては、ESマイグレーションの発生頻度が高い場合に、外部記憶へ直接ページアウトする。

(c) 重複ページ消去機能

ESからMSへページインしたページも再びMSから追い出されるまで、長時間参照されることはない。そこで、ページイン時にESのページ枠を解放する。これにより、ESの実効容量を拡大する。

2.3 実現方式の概要

(1) 初期設定方式

ESの領域はすべてページング用に利用するわけではなく、サブシステムや一般ユーザのシーケンシャルおよびランダムファイルを配置する領域としても使用する。これらの領域をESファイルと呼びES管理部が一元的に管理する。

仮想記憶管理は、IPL時にシステム初期設定パラメータに指定されるページング用のES容量を読み込み、この値をES管理部に渡すことにより、ESファイルを確保する。このESファイル内の領域をESスロット(4KB)単位に管理するため、MS上に管理テーブルを作

成する。このテーブルには、ESスロットの所有者やLRU制御を行うための情報を記録する。

(2) 同期転送とパラレルページング方式

非同期転送方式では、チャンネルがデータを転送するため、CPUはチャンネルプログラムの作成、チャンネルの起動、および入出力完了割り込みを行う必要がある。また、プロセスの切替えも発生するため、CPUオーバーヘッドはさらに大きくなる。そこで、ESがある場合には、ページング事象が発生した時点で、同期転送によりページ転送を

ただちに完了させ、プロセスを切り替えることなくページング事象発生元に制御を戻すことにする。

また、ページング処理はグローバルな資源を使用することから、マルチプロセッサ環境下ではグローバルなロックを保持する。しかし、ページ転送中にはグローバルな資源を使用することはなく、また、同期転送命令は通常の命令の数百倍の遅さであるため、同期転送中はグローバルなロックを外すこととした。これにより、複数のプロセッサがページフォルトやページスチール処理を並列に実行可能となり、マルチプロセッサ時の性能を向上できる。

(3) 転送方法の動的切替方式

同期転送により少ないCPUオーバーヘッドでMSとES間のページング処理を実行できる。しかし、大量のページを転送する場合には、非同期的にMSとESのデータ転送を行った方がCPUオーバーヘッドを少なくできる。

そこで、予めしきい値を決めておき、その値を超えてページを転送する場合には、非同期転送を選択する。

(4) ESマイグレーション方式

仮想記憶に展開されたプログラムやデータの容量が、MSとESの容量を超える場合には、不活性なページを選択し外部記憶に配置する必要がある。ESに配置されるページには、ワーキングセットから外れたページ、スワップアウトされたアドレス空間、一時的ファイルを配置するVIO(Virtual Input Output)、プログラムライブラリ、および、データベースなどが配置される。これらのうち、VIO以外のページには、ローカリティがあるものとして、グローバルLRU法により一括して管理する。

一時的ファイルは、一般にシーケンシャルにアクセス

される。このため、一時的ファイルを配置するVIOページはLIFO(Last In First Out)法により管理する。また、ESマイグレーションが頻繁に発生するような事態では、VIOページをESに配置する効果は期待できない。そこで、このような状況では、VIOページをMSから直接外部記憶に配置する。

複数の不活性ページは、3媒体間転送機能により一括してESから外部記憶へ転送する。

#### (5) ES常駐およびES優先配置方式

仮想記憶に配置されるデータには、高速な応答時間を保証すべきものがある。例えば、システム共通領域や、サブシステムのアドレス空間である。また、HAFによって仮想記憶へ常駐したデータベースも、トランザクションの応答時間を保証するため、外部記憶へ追い出されるような事態は避けるべきである。そこで、仮想記憶の領域をMSまたはESに保証するES常駐機能を設けた。また、特定のアドレス空間をESに優先的に配置するES優先配置機能も設けてある。表1にこれら機能の概要を示す。

表1. ES常駐およびES優先配置機能

#	機能名	指定方法	対象範囲	制限
1	ES常駐	システムパラメタ (EPLP, ECSA)	システム共通領域	使用量がES容量を超えると以降対象外
2		OSマクロ	マクロに指定した範囲	トホのES常駐ページ < ES容量
3	ES優先配置	システムパラメタ (認可属性)	指定ジョブ全体	ES常駐ページ数が多いとディスクへ配置

### 3. ESページングの性能評価

#### 3.1 基本性能

##### (1) ページフォルト解決時間

M-880Hにおいてページフォルト解決時間を測定した。ESからページインする場合には数十マイクロ秒で解決できる。これは、磁気ディスクに比べ数百倍以上の速さである。

##### (2) CPUオーバヘッド

ページングのCPUオーバヘッドを命令トレーサを使用して測定した。表2に、ESと外部記憶に対するページング事象ごとの処理ステップ数の比を示す。同期転送命令は、通常の命令よりも実行時間がかかるので、重みをつけてある。

ページフォルトとページスチール処理の合計は外部記憶の場合の4分の1のオーバヘッドである。一方、スワッピング処理のCPUオーバヘッドはさほど変わらない。

表2. ESと外部記憶のページング処理ステップ数比

#	ページング事象	ES	外部記憶
1	ページフォルト	0.2	1
2	ページスチール	0.25	1
3	スワッピング	0.8	1

これは、ESに対するスワッピングを非同期転送方式としているためである。

#### 3.2 磁気ディスクおよび半導体ディスクとの比較

ESの性能を磁気ディスクおよび半導体ディスクと比較する。性能の尺度として、応答時間とOSのCPU利用率を用いる。システムのスループットは、OSのCPU利用率の逆数から求めることができる。

##### (1) 測定環境

測定の対象としたハードウェアおよびソフトウェアは、それぞれM-680H/EとVOS3/ASである。負荷として、数百台のTSS端末が接続されるTSS環境とクイックソートジョブが多重に実行されるバッチ環境を想定した。TSS環境は、仮想端末を実現する多端末シミュレータを使用した。

##### (2) 性能測定ツール

ESの使用状況やページング頻度を測定するツールを表3に示す。以下の測定ではこれらのツールを使用した。

表3. ESページング性能測定ツール

#	ツール名	測定内容
1	SAR	システム全体のESに対するページング、スワッピング、ESマイグレーション頻度、および、ES利用状況
2	SAR/D/ES	上記の内容をジョブごとに表示
3	IMP3/E	ESページング時刻、非同期転送時間・ページ数

##### (3) 応答時間の比較

MSの容量および入出力構成が等しい2つのシステムにおいて、一方にMSと同容量のESを接続して、両者の性能の差を測定した。図3は、TSS環境において、TSS端末台数を増やした時のスワッピング率とTSSコマンドの応答時間を比較したものである。

端末の増加に伴い、MSから溢れるTSSユーザ空間が発生すると、ESがないシステムでは、磁気ディスクへのスワッピングが発生する。これにより、応答時間が2倍以上遅くなる。

図4は、ページングデバイスにES、磁気ディスク、および半導体ディスクを適用したときの、ページング頻度とOSによるCPU利用率の関係を示したものである。

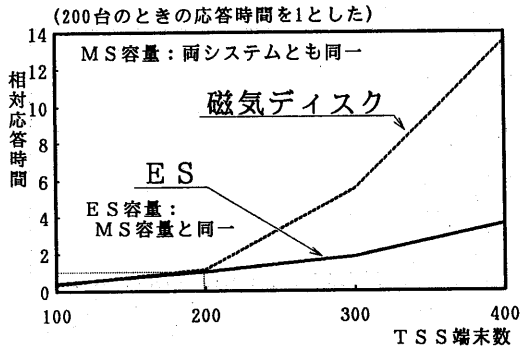


図3. TSS端末増加時の応答時間比較

負荷としてクイックソートジョブを使用した。このジョブは入出力を行わないように改造してあるため、OSのCPU利用率をページングによるCPU利用率とみなすことができる。

ESがある場合には、ページングが多発した場合にもさほどページングオーバーヘッドは大きくならない。これに対して、半導体ディスクをページングデバイスとした場合には、ESの約4倍のCPUオーバーヘッドとなる。これは、ページスチールとページインの処理ステップ数の和の比にほぼ一致する。一方、磁気ディスクのみをページングデバイスとして使用すると半導体ディスクよりもさらにCPUオーバーヘッドが大きくなる。これは、ページングの遅延によりMSの空き領域が常に不足する状態となり、MSを管理するオーバーヘッドが増えること、および磁気ディスクへのIO要求をキューイングする処理が増えたことが原因であると推定する。

このように、外部記憶のみからなるシステムでは、メモリ負荷が高くなると、スループットおよび応答時間に大きな影響が生ずる。これに対し、ESのあるシステムでは、安定したスループットと応答時間を保証できる。

### 3.3 MSとの比較

3.2節ではMSの容量を一定とし、ESのあるシステムとないシステムを比較した。この比較は、CPU内のメモリコストを無視しており、ESのあるシステムの方が有利な結果となる。そこで、より公平な評価を行うため、同一コストの条件のもとで、MSとESの容量を変化させて比較することにする。

#### (1) 解析モデル

プログラムの動作は多様であり、また、同一コストにおけるMSとES容量の組み合わせも無数にある。このため、実測によりESの効果を把握するには限界がある。そこで、負荷の特性が定義できMSとESの容量を可変

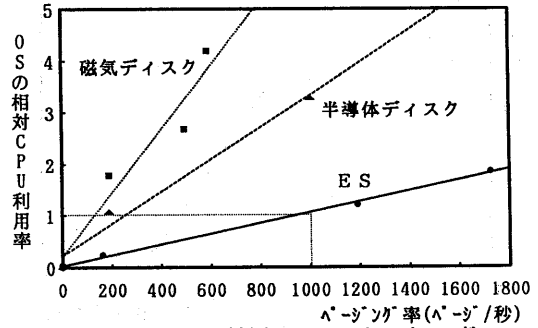


図4. ページング頻度とCPUオーバーヘッド

とする解析モデルを作成する。ここでは、最終的にユーザのCPU利用率 $\rho_{USER}$ が求められる解析モデルを定義する。ユーザCPU利用率は、スループットを求める基準となる。いま、単位時間に発生するページフォルト率を $f$ 、ページフォルト当りのOSのCPU利用率を $\rho_{PAGE}$ とすると、(1)式が成り立つ。

$$\rho_{USER} + f \cdot \rho_{PAGE} = 1 \quad (1)$$

ページフォルト率 $f$ は、プログラムの動作、メモリサイズ、およびCPU能力に影響されるはずである。 $f$ を定式化するため、まず、プログラムの動作を定義する。一般に、プログラムの動作はLRUスタックモデルにて記述することができる[3]。つまり、スタック長 $d$ と1命令実行するときのスタックサイズ $d$ 内のページ存在確率 $p(d)$ により定義する。 $1-p(d)$ は、スタック内不在確率となり、これと単位時間に実行可能な命令数 $\mu$ (CPU能力)の積は、MSサイズが $d$ のときに単位時間に発生するページフォルト率 $f$ となる((2)式)。

$$f = (1 - p(d)) \mu \rho_{USER} \quad (2)$$

(1)と(2)から、(3)式が得られる。

$$\rho_{USER} = 1 / (1 + \rho_{PAGE} \mu (1 - p(d))) \quad (3)$$

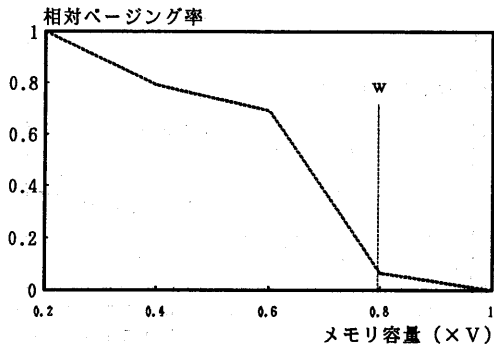
次に、ページフォルト当りのOSのCPU利用率 $\rho_{PAGE}$ を定式化する。ここで、OSはページフォルト発生時にMSから1ページ追い出し、1ページページインすると仮定する。

ESがない場合には、 $\rho_{PAGE}$ は(4)式となる。

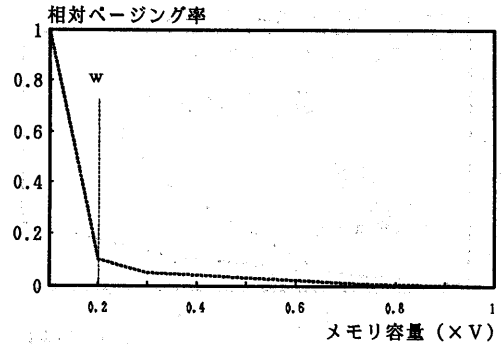
$$\rho_{PAGE} = (C_{DI} + C_{DO}) / \mu \quad (4)$$

$C_{DI}$ と $C_{DO}$ はそれぞれ外部記憶からのページイン、ページアウト処理のステップ数である。

次に、ESがあるときの $\rho_{PAGE}$ を求める。 $\rho_{PAGE}$ は、ページフォルトが発生したときに、ESにヒットするか否かにより変化する。MS容量が $m$ 、ES容量が $e$ であるときに、ESからページインする確率(ESヒット率) $h(m, e)$ は(5)式のようになる。



(a) 大WSSモデル



(b) 小WSSモデル

図5. 負荷モデル

$$h(m, e) = \{p(m+e) - p(m)\} / \{1 - p(m)\} \quad (5)$$

ESがヒットした場合には、OSはESからのページインとページアウトを行い、ミスヒットの場合にはディスクからのページインに加え、ESへのページアウトおよびESマイグレーションを行うものと仮定する。ESのページイン、ページアウトおよびESマイグレーションの処理ステップ数をそれぞれ、 $C_{E1}$ 、 $C_{E0}$ 、 $C_M$ とすれば、 $P_{PAGE}$ は(6)式のようになる。

$$P_{PAGE} = \{C_{E0} + h(m, e)C_{E1} + (1 - h(m, e))(C_{D1} + C_M)\} / \mu \quad (6)$$

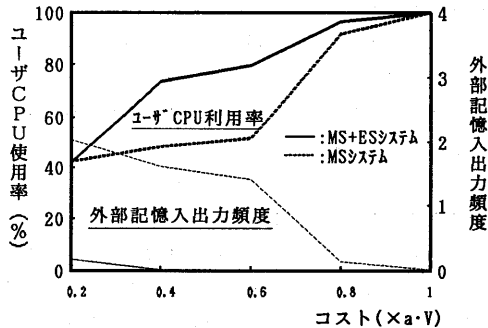
プログラムの動作モデル  $p$  と、(3)(4)(6)式から、ユーザのCPU利用率を計算できる。

### (2) 負荷モデル

ここでは、以降の解析で使用する2つのモデルを定義する(図5)。

#### (a) 大WSS (WSS: Working Set Size)

仮想記憶の最大使用量  $v$  に比べ、頻繁に参照される領域長  $w$  が大きい負荷である。クイックソートジョブの特性はこのタイプである。



(a) 大WSS

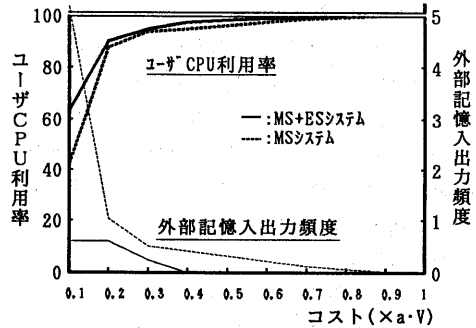
#### (b) 小WSS

仮想記憶の最大使用量  $v$  に比べ、頻繁に参照される領域長  $w$  が小さい負荷である。これは、TSS環境やファイルを仮想記憶に常駐するシステムの特長である。

### (3) コスト一定におけるMSとの比較

MSとESの単位容量当たりのコストをそれぞれ  $a$ 、 $b$  とする。メモリに投資できるコスト  $c$  を一定として ( $a m + b e = c$ )、最大のCPU利用率とそのときのMSとESの容量を求める。以下では、MSとESの単位容量当たりのコストの比 ( $b/a$ ) を  $1/3$  と仮定する。

図6には、上記(2)で定義した負荷モデルにおいて、コスト  $c$  を  $a \cdot v$  (MSの容量を  $v$  とするコスト) まで変化させたときの、各コストにおける最大のユーザCPU利用率と外部記憶への入出力頻度を示す。また同様に、MSのみのシステムについても示してある。図6から、仮定したいずれの負荷モデルにおいても、ESのあるシステムの方がユーザCPU利用率が高いこと、また、ESにより低いコストで、外部記憶へのアクセスを完全に削減できることが分かる。



(b) 小WSS

図6. 同一コストにおけるユーザCPU利用率と外部記憶入出力頻度の比較

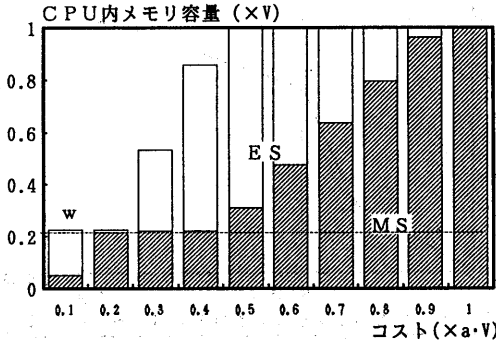


図7. 各コストにおける最適なMS・ES容量(小wss)

特に小WSSのような負荷では、少量のMSと大容量のESにより、少ないコストで外部記憶への入出力を完全に削減でき、かつ、ユーザのCPU利用率を十分大きくすることができる。

図7は、負荷が小WSSモデルであるとき、ユーザCPU利用率が最大となるMSとESの容量をコストごとに示したものである。この図からメモリコストcと最適なMSとES容量の比率に関して以下のことがいえる。

(a)  $c/a < w$

コストが少なく、ワーキングセットサイズwを配置するだけのMS容量を確保できない場合には、MSとESの容量の和がwとなるようにMSを小さくする。

(b)  $w \leq c/a < v$

MSの容量をワーキングセットサイズw以上とできる場合には、MS容量をwとし残りのコストでESを増設し、MSとESの容量の和がvとなるようにすべきである。wの大きさは、ユーザのCPU利用率どの程度保証するかによって決まる。

(c)  $c/a \geq v$

コストに十分ゆとりがある場合には、MSの容量をvとするのがよい。

#### 4. 三階層記憶制御の評価

##### 4.1 解析モデルと最適制御

2.2に述べたように三階層記憶制御は次の3つの機能からなる。

- (a) ESページアウト制御
- (b) ESマイグレーション
- (c) 重複ページ消去

ここでは、(a)のESページアウト制御について考察する。MSからページアウトするときには、必ずESにページを配置する方式が適していると直感的に考えら

れる。しかし、これが必ずしも最適な方法ではない。例えば、ESの容量が4KBしかない場合には、MSから外部記憶へ直接ページを転送すべきであることは明らかである。そこで、ESの容量と最適なESへのページアウト比率(ESページアウト頻度/トータルページアウト頻度)の関係を求めることにする。3.3における解析モデルは、MSから必ずESに対してページアウトを行うことを前提とするものであった。ここでは、まず、ESへのページアウト比率rを制御することを考慮した解析モデルを作成する。

ESへのページアウト比率をr ( $0 \leq r \leq 1$ )とし、ディスクへのページアウト比率を $1-r$ とする。定常状態では、LRUスタックの1番目からm番目までがMSに配置され、 $m+1$ 番目から $m+e/r$ 番目までの $e/r$ 個( $\geq e$ )のうちe個のページがESにランダムに配置され、残りのページが外部記憶に配置される(図8)。

したがって、ページフォルトが発生したときにESからページイン(ESヒット)する確率 $h'(r, m, e)$ は、(7)式となる。

$$h'(r, m, e) = r \{p(m+e/r) - p(m)\} / \{1 - p(m)\} \\ = r h(m, e/r) \quad (7)$$

関数hは(5)式で定義した $r=1$ のときのESヒット率である。ここで、 $e \leq e/r \leq v-m$ であることから、rの取りうる範囲は、 $e/(v-m) \leq r \leq 1$ である。

ページフォルトあたりのOSのCPU占有率 $\rho_{PAGE}$ は、(8)式のように表すことができる。

$$\rho_{PAGE} = \{C_{D0} + C_{D1} + r(C_{E0} - C_{D0} + C_M) \\ - rh(m, e/r)(C_{D1} + C_M - C_{E1})\} / \mu \quad (8)$$

ここで、 $(C_{E0} - C_{D0} + C_M)$ は小さな値であるので無視すれば、(8)式が最小となるのは、ESヒット率 $rh(m, e/r)$ が最大のときである。また、ESヒット率が最大となることから、このときのページイン、スワップイン時間も最小になる。

ここで、 $r=e/x$ とおくと、ESヒット率は $eh(m, x)/x$ となる。

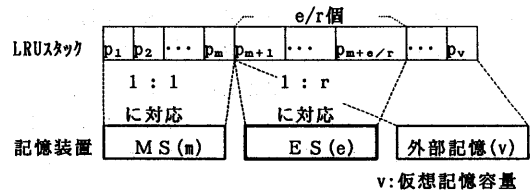


図8. LRUスタックとESに配置されるページの対応

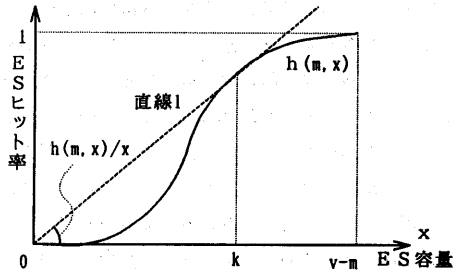


図9. ESヒット率関数 $h(m, x)$ と $k$ 点の関係

$h(m, x)/x$ は、点 $(x, h(m, x))$ と原点をとる直線 $l$ の傾きである(図9)。  $r = 1$ のときのESヒット率関数 $h(m, x)$ の増加がなだらかになる点を $k$ とすれば、 $x = k$ のとき $h(m, x)/x$ が最大となる。したがって、 $r = e/k$ とするのが最適なESページアウト比率である。ただし、 $r \leq 1$ であるから、 $e > k$ の場合には、 $r = 1$ が最適となる。

MSとESの容量の和を仮想記憶の使用量に近い大きさとする場合には、 $e > k$ となり、必ずESへページアウトする( $r = 1$ )方式が最適であるといえる。

#### 4.2 実測結果

数百台のTSS端末が接続されるシステムにおいて、 $r = 1$ としたままESの容量を変化させたときのESヒット率曲線を図10に示す。この図から、ESヒット率が急激に上昇する点 $k$ は、(仮想記憶使用量 $v$ -主記憶容量 $m$ )の約2分の1であることが分かる。

ここで、ESの容量 $e$ が $(v-m)$ の80%であるときに、 $r = 1$ とする方式と、ESマイグレーションを停止しランダムにESへページアウトする方式( $r = e/(v-m) = 0.8$ となるはずである)について比較する。

図11に、これら2つの方式について、TSSコマンドの応答時間とESヒット率を示した。 $r = e/(v-m)$ の方式(ランダム配置)では、ESヒット率もESページアウト比率も60%程度にしかならず、応答性能が低下する。これは、ESマイグレーションを停止したことにより、不活性なページがESに滞り、ESの実効容量が低下したためである。 $r = 1$ の方式では、ESヒット率もほぼ100%になり、応答性能が向上している。これらのことから、ESマイグレーションが有効であること、および、 $e > k$ の場合には、ランダムにESへページアウトするよりも、必ずESへページアウトする方が高いESヒット率を期待できることが分かる。

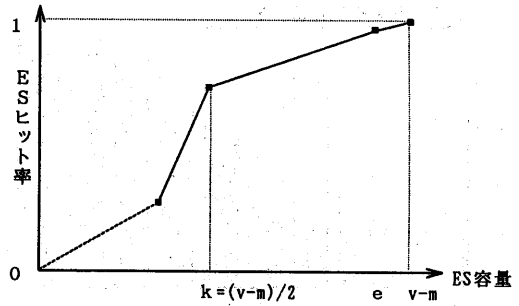


図10. TSSシステムにおけるESヒット率曲線

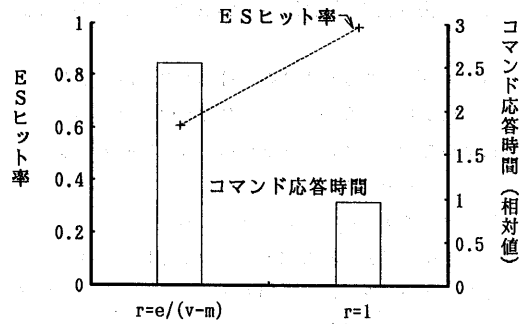


図11. ESページアウト比率による性能の変化

#### 5. まとめ

ESに対して、ページング・スワッピングを高速に行うESページング方式を開発した。これにより、ページフォルトの解決時間を数十マイクロ秒とし、また、CPUオーバーヘッドも4分の1程度とした。ESはMSに比べ単位容量当りのコストが低いため、低コストで応答時間とスループットを向上できる。

ESがあるシステムにおいて、ESへのページアウト比率は重要なパラメータである。解析および実測の結果、MSとESの容量の和を仮想記憶の使用量に近くする場合には、ESへのページアウト比率を1とする(必ずESへページアウト)のが、最適であることが分かった。

#### 6. 参考文献

- [1]新井ほか: 拡張入出力制御方式XVIOフェーズ1の開発, 情報処理学会OS研究会, 25-3, 1984. 12. 14
- [2]櫻庭ほか: 汎用入出力仮想化機能HAFの開発, 情報処理学会OS研究会, 47-3, 1990. 6. 8
- [3]Jeffrey R. Spirn: Program Behavior: Models and Measurements, Elsevier North-Holland, 1977