

4U-3

仮想共有メモリ方式における ページサイズの検討

濱崎陽一

(電子技術総合研究所)

はじめに 筆者らは、光による高速共有バスを持つマルチプロセッサシステムDialog.H¹⁾に、並列論理型言語Dialog.Lのインタプリタを実装しシステムの評価を進めている。そのインタプリタの高速化のためには、仮想共有メモリのページサイズを小さくする必要が有ることが解った。ここでは、その適切なページサイズについて検討し、シミュレーションした結果について述べる。

Dialogシステム Dialog.Hは共有メモリ(メモリユニット)を持つマルチプロセッサシステムで、各プロセッサユニットは、共有メモリの一部のコピーを仮想共有メモリ(VCM)に持つ方式を採用している。メモリユニット(MU)とプロセッサユニット(PU)間は、パケット通信を行なう放送型共有バスで接続されている(図1)。

仮想共有メモリは、複数のページに分かれており、アクセスしたい共有メモリの内容が仮想共有メモリ中に無い場合には、メモリユニットからそのページをスワップインするのは一般の仮想メモリと同様である。共有メモリへの書き込みは、まず仮想共有メモリの内容を書換え、ほかのプロセッサに内容を伝えたいときに(一般には意味のある情報となったときに)、共有バスを使ってすべてのプロセッサユニットとメモリユニットに放送する。放送された内容に基づいて、共有メモリの内容と他のプロセッサユニットが持つ同じ番地のページの内容が更新される。こうした共有メモリの管理は分散カーネルが行なう²⁾。

上に述べたシステムの上に、並列論理型言語Dialog.Lのインタプリタを実装した³⁾。論理型言語を並列に実行するために生じる多重バインド環境を表わすために、共有メモリ上に局所的な実行で生じた環境の変化を記録するハッシュ

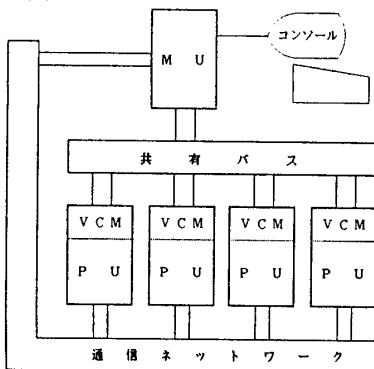


図1 Dialogシステムの構成

ウインドウと呼ばれるデータ構造をつくり、その木で多重環境の全体が表わされる方式とした。ハッシュウインドウの大きさは、Dialog.Lのプログラムの各クロズが持つ変数や関数の数によって決まり、62から100Byte程度の大きさであった。プロセスがハッシュウインドウを仮想共有メモリに書き込み、そのページを放送した後に、他のプロセッサに生成されたプロセスがそれを読む。放送はページを単位として行なわれるために、ハッシュウインドウの大きさに比べてページが大きすぎる場合には、通信量が増大して速度の低下をもたらす。現在作成されているDialog.Hのプロトタイプの場合、ページサイズが512Byteであるので、改善の余地がある。

シミュレーション 上記のような観点からシミュレーションによりDialog.Lのインタプリタに適切なページサイズを検討するために、共有バスの通信量のシミュレーションを行なった。

シミュレーションの内容は次の通りである。
① ある特定のプロセッサに注目し、メモリに1000個のデータブロック(ハッシュウインドウのように意味のあるデータの固まり)の書き込みを行なう。1つのデータブロックの書き込みが終了する度に、そのデータブロックを含むすべてのページの内容を放送する。データブロックのページへ割当は、first-fitで行なう。
② データブロックの大きさは正規分布を持つ乱数で与えた。図2にシミュレーションに用いたデータブロックの大きさのヒストグラムを示す。
③ メモリへ書き込まれたデータは、すべて2つのプロセスからアクセスされ、そのプロセスは、隣接プロセッサにランダムに割り当てられているとする。
④ データブロックをアクセスする隣接プロセッサは、そのページを持っていなければ、メモリユニットに要求し、スワップインする。

図2 ヒストグラム

図2 ヒストグラム

図2 ヒストグラム

図2 ヒストグラム

図2 ヒストグラム

図2 ヒストグラム

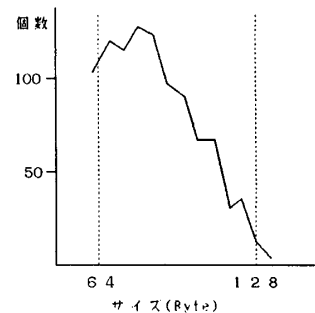


図2 ヒストグラム

但し、メモリ内の同一のページに2つ以上のデータブロックが書き込まれ、それぞれを同じ隣接プロセッサ内のプロセスがアクセスするときには、共有メモリからのスワップインは1度しか起こらない。

⑤ ページサイズをNとしたときの共有バスの通信量は、次の通りである。これらの値は、プロトタイプで用いられているパケットの大きさである。

放送	$N+7$
スワップイン	$N+14$
ページ要求	12

⑥ シミュレーションは、隣接プロセッサ数が、2, 4, 8, 16, 32の場合のそれぞれで、ページサイズが64, 128, 256, 512のときの共有バスの通信量を計算した。

シミュレーション結果 シミュレーションの結果を図3に示す。図中、実線で示したのが隣接プロセッサ数がそれぞれ、2, 4, 8, 16, 32の場合の通信量である。また、図4にページ中にデータブロックの占める割合(メモリ利用率)を示す。

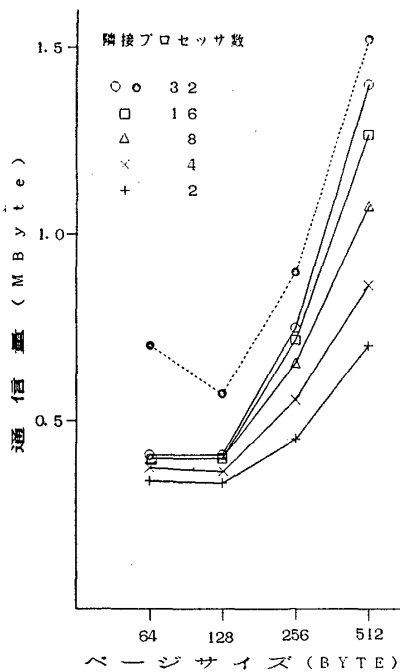


図3 シミュレーション結果

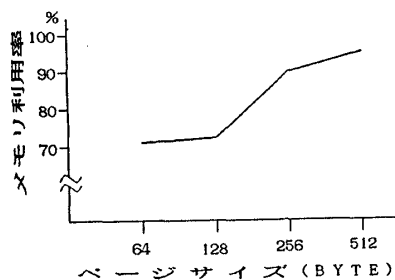


図4 メモリ利用率

での競合によるオーバーヘッドやインタフェースでのオーバーヘッドが考慮されていないが、通信1回につき25のオーバーヘッドがある場合を隣接プロセッサ数32でシミュレートしてみると、図3の点線の様になる。この場合には、ページサイズが64の場合は、128の場合よりも20%近く通信量が増える。

検討 Dialogのような方式の共有メモリにおいては、ページサイズは、平均的なデータブロックが十分に納まる大きさとするのが、通信量の点からみればよい。但し、メモリの使用効率は、もっと大きなページの方がよい。さらに小さなページサイズであっても、通信量の点からみれば、遜色が無い。しかし、メモリの使用効率や、通信回数の増加によるオーバーヘッドの増大などの問題を生じる。

Dialog.Hのボトルネックは、共有メモリや共有バスに生じることが予想され、その最大の要因は、データ共有に伴う通信である。また、メモリユニットは、これらの通信量にほぼ比例した処理を行なう必要がある。シミュレーションの結果から通信量は、ページサイズを512 Byteから128 Byteに変更することにより、3分の1程度に低減できる。このことは、システムに含まれるプロセッサユニットの数(計算パワー)の上限が約3倍になることを意味している。

実際には、共有するデータブロックへのアクセスは多様であり、その分布、時間関係も複雑で、プログラムに大きく依存している。本稿のシミュレーションでは、そうした点を反映してはいない。しかし、共有メモリに書き込まれたデータブロックは、必ず複数のプロセスからアクセスされ、その数は書き込みの直後にはそう多くないことが経験的に解っており、大局的な振舞いは、十分に表わしている。

まとめ Dialog.Lの実装に用いた方式に基づき、Dialog.Hの仮想共有メモリのページの大きさについて考察した。ページの大きさは、データブロックが十分納まる大きさのうち最小の物が、通信量の点で最良となることがシミュレーションの結果わかった。また、システムの性能向上には、ページサイズが大きく影響する。

参考文献

- 1) 濱崎他: "Dialog.Hのプロトタイプシステム", 情処学会「アーキテクチャワークショップ'84」, 1984
- 2) 濱崎他: "マルチマイクロプロセッサDialogの通信機構", 情処研報 MC43-1, 1987
- 3) 濱崎他: "並列論理型言語Dialog.Lのインタプリタ", 信学技報 CPSY85-51, 1988

このシミュレーションでは、共有バス