分散メモリ型並列計算機のための可変インターリーブ・ファイルシステムの構成

7H－5

甲村康人　大上靖弘　松本健志　大西一正　清水雅久
三洋電機(株) 東京情報通信研究所
(E-mail: koumura@tk.ic.rd.sanyo.co.jp)

1　はじめに

並列計算機システムにおいては、演算性能のスケーラビリティを確立するのと同様に、演算速度に見合った入出力処理性能を提供するために入出力性能のスケーラビリティを確立することが重要な課題である。本稿では、ネットワークによって複数のプロセッサが相互に結合され、個々のプロセッサが独立な2次記憶へのアクセス手段をもつような形態の分散メモリ型の並列計算機において、ファイルをこれらの2次記憶にインターリーブして配置し、複数のプロセッサから個々のファイルへの並列アクセスを可能にしたファイルシステムの構成について述べる。

2　並列計算機及び並列ファイルシステムのモデル

本稿で想定する並列計算機のモデルは以下のようなものである(図1)。

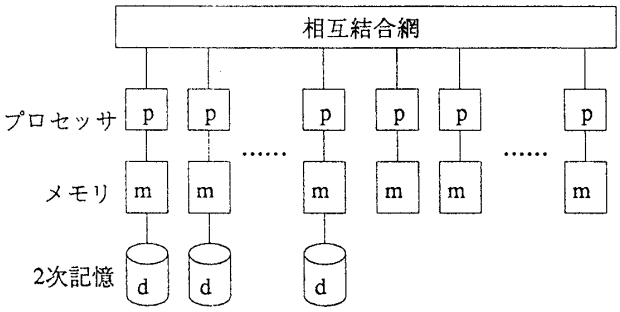

図 1: 並列計算機のモデル

- 複数のプロセッサがネットワークによって相互に結合されている
- 各プロセッサは局所メモリを持つ
- 一部あるいは全てのプロセッサが2次記憶に対するアクセス手段を持ち、局所メモリ空間が仮想化されている(局所仮想空間)
- 各プロセッサは他のプロセッサのメモリ空間に対するアクセスをネットワーク経由で行なうことができる

ここで、全てのプロセッサからアクセスできる論理的な連続領域としての大域空間を考える。大域空間における各ワードはいずれかの局所仮想空間中のワードと1対1に対応づけられる。

ここで大域空間から局所仮想空間への写像を定義する。以下、簡単のためアドレスはワードアドレスとする。局所仮想空間を持つプロセッサの数を M、局所仮想空間の大きさを L とする。大域空間の大きさは ML となる。ここで $0 \leq ga < ML$ を大域空間のアドレス、$0 \leq pid < M$ を局所仮想空間を持つプロセッサのプロセッサ番号、$0 \leq la < L$ をプロセッサ内の局所仮想空間のアドレスとする時、$ga \mapsto \langle pid, la \rangle$ なる写像を

$$pid = ga \bmod M \quad (1)$$

$$la = \lfloor ga/M \rfloor \quad (2)$$

と定義することで、大域空間の連続領域がワード毎に各局所仮想空間にインターリーブして配置される。これによって、大域空間へのアクセスが各局所空間に均等に分散されることが期待できる(図2)。

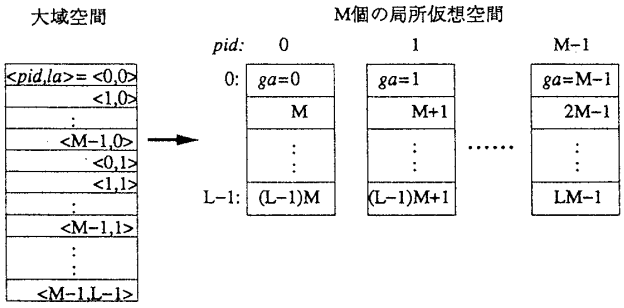

図 2: 大域空間から局所仮想空間への写像

本稿では、オープンされたファイルを、上で定義した大域空間の適当な連続領域にマップすることでアクセス可能にするファイルシステムを考える。ファイルをオープンする操作は、そのファイルを大域空間の適当な未使用領域にマップし、その先頭アドレスを返す手続きと定義される。また、ファイルの内容の読み出し及び書き込み操作は、大域空間へのリモートメモリアクセスによって実現される。2次記憶装置とのデータ転送は、局所仮想空間の仮想記憶システムによって局所仮想空間毎に独立に管理される。

上記で定義した大域空間から局所仮想空間への写像を見れば明らかなように、ファイルは複数の2次記憶にインターリーブして配置されることになる[1]。

3　可変インターリーブ構造

並列計算機におけるファイルへの典型的なアクセスパターンの1つは、逐次計算機における連続アクセスに相

Distributed Memory-mapped Filesystem with Variable Interleaving Factor for Highly Parallel Computers.

Yasuhito KOUMURA, Yasuhiro OUE, Kenshi MATSUMOTO, Kazumasa OHNISHI, Masahisa SHIMIZU.
SANYO Electric Co., Ltd.

当するもので、1つのファイルを並列度nに応じてn分割し、n個のアクティビティはファイルの部分領域を連続アクセスするものである。この場合、ファイルの部分領域への分割方法は、連続領域n個への分割、あるいはモジュロnによる分割等が考えられるが、いずれにせよ前節で述べたインターリーブ方式でファイルシステムのスケーラビリティは確保されると考えられる。

並列ファイルシステムでのもう1つの典型的なアクセスパターンは、ファイルの小さい部分領域に対するスパースな並列ランダムアクセスである。例えば、数ワードから数十ワードといった比較的小さな構造体の巨大な配列とみなせるようなファイルに対するランダムアクセスを含むようなアプリケーションを考える。この場合、前節で述べたファイルシステムでは、1つのkワードの構造体へのアクセスによって、最悪k個の局所仮想空間でページフォルトが生じ、さらにこれによって局所メモリ上に読み込まれた内容のほとんどが捨てられてしまうという事態が生じる。逐次ファイルシステムと比較して、実質的なページサイズが大きくなってしまっているため、無駄なアクセスが増加しているといえる。このような場合、前述の方式では並列ファイルシステムとしてのスケーラビリティは期待できない。

これはインターリーブの単位が1ワードと小さいのが原因といえる。インターリーブの単位をkにできれば、個々の構造体へのアクセスは並列化されないが、最大M個の構造体へのアクセスを並列に行なうことが可能になり、システム全体としてのスケーラビリティは確保されるものと考えられる。そこで、前述の大域空間と、局所仮想空間への写像を次のように再定義する。

インターリーブの単位ifごとに異なる大域空間を考える。ここでPを局所仮想空間のページサイズとすると、ifは$1 \leq if \leq P$を満たすような2の冪数である。ファイルをアクセスするためのアドレスを$\langle if, ga \rangle$なる2項組とし、open, read, write といった操作は、この形式のアドレスを扱う。ここで$\langle if, ga \rangle \mapsto \langle pid, la \rangle$なる写像を

$$pid = \lfloor ga/if \rfloor \bmod M \quad (3)$$

$$la = \lfloor ga/if/M \rfloor \cdot if + ga \bmod if \quad (4)$$

と定義する。すなわち、大域空間がifを単位としてM個の局所仮想空間にインターリーブして配置されることになる(図3)。

ただしこの方式ではファイルがどの大域空間に属するかはファイルの作成時に決定され、変更され得ない。このため、ファイルを作成するプログラムと、ファイルを参照するプログラムでアクセスパターンが大幅に異なるなど、最適なインターリーブ単位がファイル毎に一意に決定できないときには、スケーラビリティという観点からは依然問題がある。

4 高並列計算機 CYBERFLOW への適用と実装

CYBERFLOW は我々が開発しているデータ駆動型の高並列計算機である[2]。64プロセッサからなるシステムが稼働中であり、現在、本稿で述べたモデルに基づいたファイルシステムを構築中である。

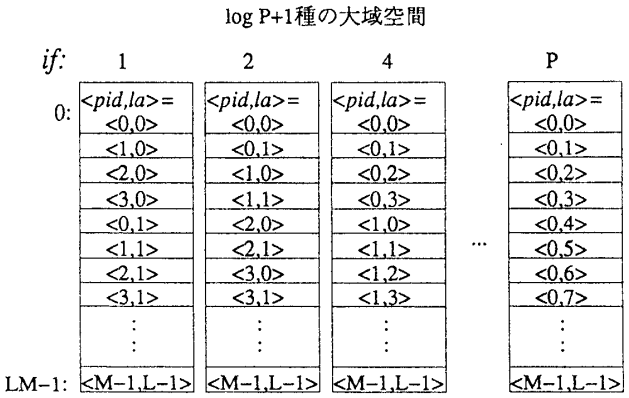

図 3: インターリーブ方式毎の大域空間と局所仮想空間への写像

CYBERFLOW のプロセッサは比較的小規模で高速なメモリのみを持ち、実記憶で稼働するように設計されている。そのため我々は仮想化された大きなメモリ空間を持ち、ファイルシステムを構築するための専用プロセッサを CYBERFLOW のネットワークに組み込み、通常のプロセッサ間通信で用いられるパケットでこの局所仮想空間をアクセスできるようなハードウエアを構築した[1]。

前節で述べたような大域空間から局所仮想空間へのアドレス変換を、各プロセッサでライブラリ関数によって行なう形態で、実際のシステムを実装している段階である。

5 おわりに

各プロセッサの局所メモリ空間が仮想化されている分散メモリ型並列計算機において、ファイルをこれら複数の局所仮想空間にインターリーブしてマップすることで全プロセッサから個々のファイルへの並列アクセスを可能にするファイルシステムの構成について述べた。

今後、CYBERFLOW を実際のプラットフォームとして、ファイルシステムの完成と、種々のアプリケーションプログラムによる並列ファイルシステムのスケーラビリティの検証、及びインターリーブ単位の変化がアプリケーションプログラムの実行効率に与える影響を検討していく計画である。また、並列ファイルシステムの構成がアプリケーションプログラムの負荷分散の最適戦略に与える影響に関する知見を得たいと考えている。

参考文献

[1] 大西, 甲村, 松本, 北村, 清水. データ駆動型計算機 CYBERFLOW における並列2次記憶. 情報処理学会第46回全国大会論文集, 6M-6, (1993).

[2] 三浦, 川口, 田中, 大橋, 清水. データ駆動計算機 EDDEN のアーキテクチャ. 情報処理学会論文誌, Vol.32, No.7, (1991), 838-848.