2N-8

データ転送オーバヘッドの削減を主眼とした並列処理アーキテクチャの提案

田中 輝雄[1]　濱中 直樹[1]　村松 晃[2]

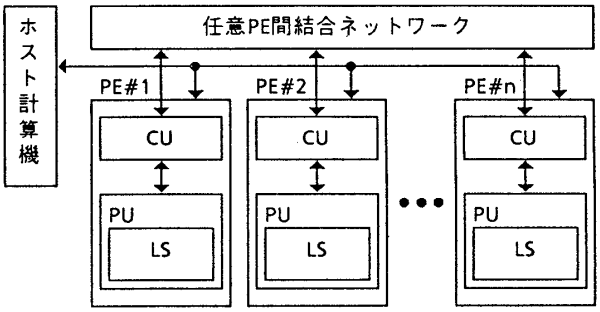
[1](株)日立製作所 中央研究所　[2]同 システム開発研究所

1. はじめに

ベクトル処理を基本とするスーパコンピュータの登場により，科学技術計算分野での計算機パワーは飛躍的に高められた．しかし，計算機パワーに対するユーザのあくなき要求はとどまるところを知らず，スーパコンピュータを越える超高速計算機が求められている．計算機の能力Pを定式化すると，

$$P = \eta P_{PE} N$$

となる．ここで，ηは並列化効率，P_{PE}は要素プロセッサ(PE)の性能，NはPE台数をあらわす．この要素プロセッサの性能向上率はすでにトレンド上にあり，飛躍的な性能向上をはかるためにはNを大幅に増やす必要がある．本報告では，ηを落とさずに，数百，数千のPEの接続に耐えうる並列処理アーキテクチャの一方式を提案する．



PE：要素プロセッサ
PU：命令処理部
CU：通信処理部
LS：ローカルメモリ

図1 並列計算機の基本構成

2. 並列処理アーキテクチャの基本構成

並列処理アーキテクチャには種々の基本構成が考えられる．本報告では，この基本構成を，

(1)PE間ネットワーク：任意PE間結合
(2)メモリ構成：分散メモリ
(3)並列実行方式：MIMD
(4)PE間通信方式：メッセージ通信
(5)PEアーキテクチャ：ノイマン型計算機

と仮定する(図1参照)．

3. 並列化効率ηの低下要因

並列化効率ηを低下させる要因としては，

(1)PE間データ転送オーバヘッド
(2) PE間 (PE内で複数のプロセスを並行に実行する場合はプロセス間)同期オーバヘッド
(3)プロセス切り換えオーバヘッド
(4)PE間の負荷の不均一

などが考えられる．

本報告では，これらのオーバヘッドのうちもっとも重要な要因である(1)の削減に注目し，次節に述べるPE間データ転送方式を検討した．また，この方式は(2)の削減も可能である．

(3)に関しては，PEにはできるだけ少数のプロセス(可能ならば，1PEに1プロセス)を割り付けることを基本とした．

(4)に関しては，静的負荷分散で均一な負荷分散が可能であると考えた．この理由は，大規模な科学技術計算分野においては，処理手順が実行以前に決定されている場合が多く，また，対象とする問題の持つ並列処理構造を利用することが可能であるためである．静的負荷分散のみを利用することにより，PE間のデータ転送関係のプログラム記述が実行開始以前に可能となる．

4. データ転送オーバヘッドの削減を可能とするPE間データ転送方式

MIMD方式の並列計算機では，データ送信側のPEが送るべきデータを確定する時刻(データ送信可能時刻)と，データ受信側のPEがそのデータを参照する時刻(データ受信可能時刻)は一般に異なる．したがって，可能なかぎり早くデータ送信処理を行い，

A Parallel Processing Architecture for Reducing Data Transfer Overhead,
Teruo Tanaka[1], Naoki Hamanaka[1], Akira Muramatsu[2],
[1]Central Research Laboratory,Hitachi Ltd., [2]Systems Development Laboratory,Hitachi Ltd.

可能なかぎりデータ受信処理を遅らせ，かつデータ転送処理をPEの命令実行処理と独立に動作可能とすることにより，データ転送処理をPEの命令実行処理とオーバラップさせることができる．このときの問題点は，送信・受信処理間の転送データの識別である．このために，転送データにデータ識別子(key)を付随させる．

このデータ転送方式の手順を次に示す．

(1) データ送信PEは，送信命令により，通信処理部に受信PE番号，転送データおよびデータ識別子(key)をセットし，通信処理部を起動する．

(2) 通信処理部は，セットされた各要素をもとにパケットを生成し，ネットワークに送り出す．ネットワークは，そのパケット内の受信PE番号にしたがい，受信PEの通信処理部内受信バッファに転送データをデータ識別子(key)とともに格納する．

(3) データ受信PEでは，受信命令を用いて受信処理を実行する．受信処理は，受信命令で指定したデータ識別子を用いて受信バッファを検索し，対応するデータを命令処理部内に取り込む．データが到着していない場合は，データが到着するまで受信処理を実行する(PE内に複数のプロセスが存在している場合にはプロセススイッチがおこる)．

この手順のうち，手順(2)がPE内演算処理とオーバラップ可能な処理である．

この方式の動作例を図2に示す．図中，PE#2の送信命令(SEND)により，PE#2からPE#nに転送されたデータ(dataB)が，データ識別子(keyB)とともにPE#nの受信バッファに保持されている．さらに，PE#1の送信命令(SEND)により，PE#1からPE#nに送信されるデータ(dataA)が，データ識別子(keyA)とともにパケットを構成し，ネットワーク上を転送中である状態を示している．この状態で，PE#nが第1の受信命令(RCV)を実行する場合，識別子(keyB)を用いて受信バッファを検索し，受信データ(dataB)を取り込む事ができる．一方，PE#nが第2の受信命令(RCV)を実行する場合，受信データ(dataA)は，まだ受信バッファに届いていないので，受信処理は待たされる．

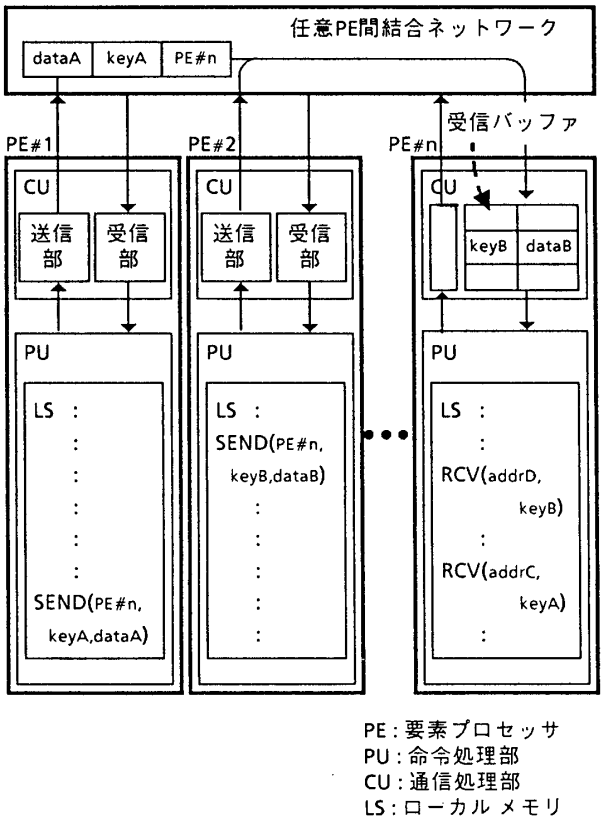


図2 提案データ転送方式の動作例

5. 提案方式による並列化効率の改善と特徴

本提案方式により，データ転送オーバヘッドの削減が可能となる[1]．さらに，次のような特徴を持つ．

(1) 転送データに識別子(key)を付加することにより，複数のデータを同時にネットワーク上で転送することが可能となる．また，任意PE間でデータ転送可能なことのみを保証すれば，ネットワークの構成によって，アーキテクチャは左右されない．

(2) データ転送時の順序保証のための同期処理(たとえば，P,Vオペレーションなど)を必要としない．つまり，データ転送によりPE間の順序制御が同時に行われている．これは，データ転送処理と順序制御処理が一体化していることを示している．

6. おわりに

本報告では，大規模科学技術計算のための並列処理アーキテクチャとして，データ転送オーバヘッドの削減を主眼とした方式を提案した．現在，基本方式の提案の段階であり，今後，実用システムに向けた方式の改善を行っていく．

参考文献

[1] 濱中，田中：データ転送オーバヘッドの削減を主眼としたテクチャの評価，第37回全国大会予稿掲載予定．