計算機アーキテクチャ 142−30
ハイパフォーマンス
コンピューティング 85−30
（2001. 3. 9）

各種プラットフォームにおける DSM クラスタの性能と拡張性に関する評価

福澤 毅　下野靖史　手塚忠則
大西淑雅　Bernady O. Apduhan　有田五次郎
九州工業大学

概要

本稿では，多様なクラスタシステム上で分散共有メモリモデルを提供する，分散スーパーコンピューティング環境 DSE の性能調査を行った．汎用のネットワークで接続された PC クラスタと，専用のネットワークと汎用のネットワークの両方で接続された WS クラスタ上において，DSE の性能調査を行った．プリミティブに対する予備的な調査からは，プロセッサアーキテクチャや，計算速度，通信速度による影響が観察された．また，並列アプリケーションによる実行速度からは，台数効果による処理速度向上がはっきり現れた．

Performance and Scalability Evaluation of a DSM Cluster on Various Platforms

Takesi Fukuzawa　Yasushi Shimono　Tadanori Tezuka
Yoshimasa Ohnishi　Bernady O. Apduhan　Itsujiro Arita
Kyushu Institute of Technology

Abstract

In this paper, we evaluate the performance and scalability of a DSM cluster, called DSE, on different platforms. We evaluated DSE on a cluster of PCs interconnected by a commodity network, and on a cluster of workstations interconnected by both a high-speed dedicated interconnection network and a commodity network. Experimental results with primitive operations demonstrate the influence of processor's architecture, processing capacity, and network speed. Likewise, the results with parallel applications demonstrate a promising scalability performance.

1 はじめに

近年、大規模な計算を複数のプロセッサに分割して処理を行う並列処理システムとしては，一般的な計算機を多数相互接続したクラスタシステムが，使われるようになってきている．これは，クラスタシステムが安価な計算機をプロセッサとして利用可能であるので，大規模な並列アプリケーション向けのシステムが，従来の並列計算機に比べて容易に構築できるからである．このようなクラスタシステムの構成法としては，専用のネットワークを用いて相互接続する場合と，汎用のネットワークに接続された計算機を利用する場合があり，それぞれ専用の並列処理支援用ソフトウェアが，ユーザとのインターフェイスを担当している．我々の研究室では，分散共有メモリモデルに基づくクラスタシステムを提供するため，分散スーパーコンピューティング環境 DSE の研究開発を行ってきた．[1][2]

DSE は移植性を考慮して，UNIX のユーザ環境上で標準の TCP ソケットを用いた実装を行っている．そのため，安価な LAN で接続された PC クラスタから専用のネットワークを用いた並列計算機まで，同一のソースコードを用いた並列処理が可能となっている．

ここで，多種多様なハードウェアから構成されるクラスタシステムでは，プロセッサの持つ

計算能力やプロセッサ間の通信能力は，実行するアプリケーションプログラムの計算量，通信量と共に，並列処理の効率に大きな影響を与える．そこで，高い移植性を持つDSEでは，並列アプリケーションの計算量，通信量とクラスタシステムの計算能力，通信能力が，並列処理の効率に与える影響を予測したプログラミングが必要となる．

本稿では，2種類のプロセッサ及び2種類のネットワークの組合わせからなるDSEクラスタシステムにおいて，通信時間とアプリケーションの実行時間に関する評価を行った．

2 実験環境

2.1 DSEの構成

DSEでは並列処理の最小単位はタスクと呼ばれ，FIFOキューによって管理されている．また，分散共有メモリモデルによるプログラミングに必要な共有メモリアクセスプリミティブや同期用のプリミティブが提供されている．これらの機能はDSEカーネルプロセスに実装されており，実際に並列アプリケーションの処理を担当する並列アプリケーションプロセスとは，プロセス間通信によって結ばれている．(図1)

今回利用した実装では，DSEカーネル間の相互接続にTCPコネクションを利用しているため，最大ノード数は59台に制限されている．このノード数の制限を解消するには，UDPを基礎とする通信機構の実装が必要である．

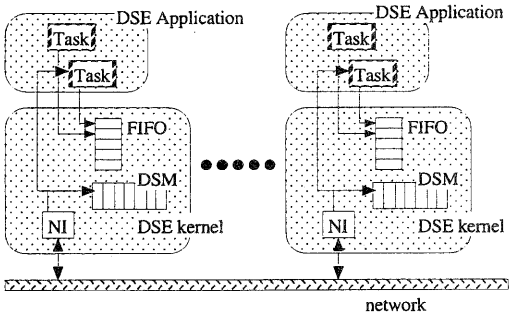

図 1: DSEのシステム構成

2.2 PCクラスタシステムの構成

本学情報科学センターでは，教育用システムとして222台のLinux端末からなるシステムを導入している．各端末は，100MbpsEthernetによって相互に接続されている．各端末の仕様は，表1に示す通りである．

表 1: 本学情報科学センターの端末構成

CPU	Celeron 400MHz
Memory	256MB
OS	Linux 2.2.14
Network	Ethernet(100Mbps)

また，ディスクレス端末として設定されており，全てのファイルへのアクセスは，NFSサーバによって処理される．そのため，ネットワーク管理用のサーバとNFSサーバの設定を変更することで全ての端末の設定が更新され，容易にPCクラスタとして利用することが出来る．

2.3 RS6000/SP2の構成

本学知能情報工学科では，研究用システムとしてIBM社製並列計算機であるRS6000/SP2を設置している．[3] RS6000/SP2(以下ではSP2と呼称)は，汎用のWS(RS6000)が専用に設計された高速スイッチ(SPスイッチ[4])によって相互接続されている．(図2) SPスイッチは多段構成のOmegaネットワークを実現しており，ノード間のスループットは150Mbyte/sとなっている．また，汎用のEthernetによる接続も可能であり，今回は100MbpsのEthernetによって接続された8ノードも実験に利用している．

ここで，今回実験に利用したRS6000の仕様を表2に示す．

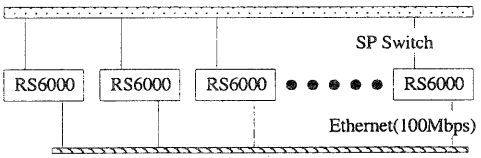

図 2: RS6000/SP2のシステム構成

3 性能評価

3.1 基礎調査

アプリケーションを並列化する目的は，実行時間の短縮である．よって，並列化に必要なプ

表 2: IBM 社製 RS6000 の構成

CPU	POWER2 66.7MHz
Memory	64MB
OS	AIX 4.2

リミティブの実行時間を測定することで，並列化が有効となる問題の規模が明らかになる．また，利用するプロセッサ及び通信媒体がこれらのプリミティブの実行時間に与える影響によって，評価に用いる各クラスタシステムにおけるDSE の通信性能を評価することが出来る．

図 1に示すように，DSE では並列アプリケーションプロセスで発行されたプリミティブは，DSE カーネルプロセスによって処理され返答を通知される．また DSE カーネルプロセスでは，プリミティブの目的となるノードに対して通信を行い，処理の実行を依頼する．よって DSE では，目的のノードからの返答を必要とするプリミティブとしないプリミティブに分類することができる．

ここで，DSE における代表的なプリミティブとして，DSE_shmem_read と DSE_shmem_write 及び DSE_shmem_writeS についての実行時間を測定する．DSE_shmem_write と DSE_shmem_writeS 間で 違いは，共有メモリに対する書き込みを行った後に，書き込み終了の返答を待つのが DSE_shmem_writeS であり，書き込み命令を発行後すぐに次の処理に移るのが DSE_shmem_write である．測定対象となるクラスタシステムは，PC クラスタ (100Mbps)，SP2 (100Mbps)，SP2 (SP スイッチ) の 3 種である．測定方法としては，指定したサイズのデータに連続したアクセスを行い，その全体の時間からアクセス一回当りに必要な実行時間を推定した．図 3に，DSE_shmem_read の実行時間を示す．図 4に，DSE_shmem_write の実行時間を示す．また，図 5に，DSE_shmem_writeS の実行時間を示す．

図 4における約 1 キロバイト以上の場合を除いて，プリミティブの実行時間は SP2 の方が短い．また，SP2 においてはネットワークによる違いは観察できない．さらに，表 1，2より，

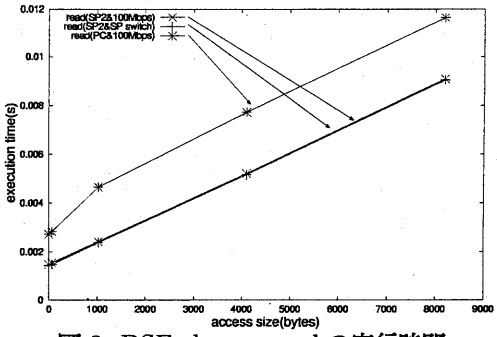

図 3: DSE_shmem_read の実行時間

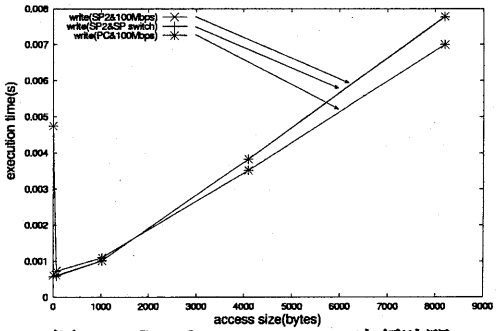

図 4: DSE_shmem_write の実行時間

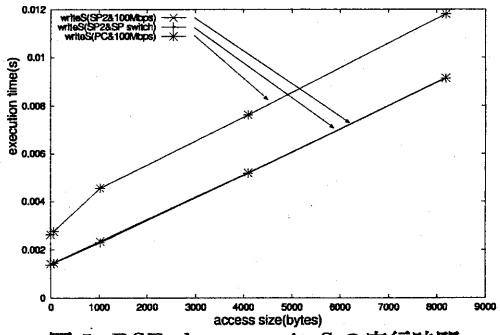

図 5: DSE_shmem_writeS の実行時間

各クラスタシステムを構成するWSやPCにおける計算処理能力は，PCの方が高いと考えられる．これは，以降の節で示すアプリケーションの実行時間(図7)からも確認できることである．このことから，PCとWS間でのプロセッサアーキテクチャの違いが，プリミティブの処理でWSに有利な結果をもたらしていると考えられる．

3.2 騎士巡歴問題

騎士巡歴問題とは，N×Nの盤面上で全てのマス目をチェスの騎士が1回だけ通るような経路を求める問題である．この問題は非常に単純な探索問題であるので，探索木は任意の部分木に分割可能である．そこで，ある深さまで逐次的に探索木を構成し，以降を部分木として並列に探索を行った．しかし、探索木全体の大きさは問題によって決まっているので，逐次的な処理の深さや利用するノード数を変更しても、計算量や通信量はほぼ一定である．

PCクラスタにおいてノード数を変えつつ6×6の盤面での探索を行い，実行時間を測定した．ただし，PC上で同一の問題を逐次処理で実現した場合には，7111.92秒かかっている．図6では，その逐次処理に対する速度向上比として示している．

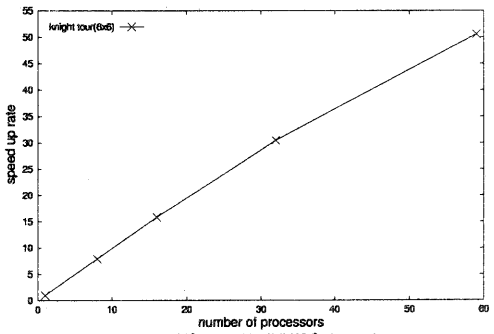

図 6: 騎士巡歴問題 (6x6)

実際の並列処理のプログラムでは，盤面の情報と探索木の状態をそれぞれノード0及びノード1に保持し，各ノード上のタスクがその共有変数にアクセスしつつ処理を行う．この場合には，実際に計算を行うタスクの数はノード数と一致し，逐次処理による探索の深さが8であるので，並列処理開始時の部分木の数は7254となる．

ここで共有変数へのアクセスは，探索木の状態を保持する場合とタスクの終了時に探索結果を集約する場合のみである．そのため，ノード数の変化に対してアクセス回数の変化はほぼ一定であると言える．また，計算量に対して通信量が非常に少なくなるので，図6で示すようにプロセッサ数に対する速度向上比は，非常に良好な値となる．

3.3 DCT変換

DCT変換とは，離散コサイン変換(Discrete Cosine Transform)を利用して画像の符号化を行う処理である．DCT変換では，M×Nの画素毎にブロック化を行い独立して処理を行うため，並列処理に向いたアプリケーションと言える．そこで，1024×1024画素の画像データを8×8画素，16×16画素，32×32画素でブロック化し，それぞれのブロック毎に並列に処理を行った．このときのブロック数は，それぞれ16384，4096，1024である．

PC及びWS上で，それぞれのブロックサイズで逐次処理を実行した場合の時間を，図7に示す．図8に，PCクラスタにおける逐次処理に対する速度向上比を示す．ただし，タスクのブロックへの割り当ては，画像の縦軸方向にグループ化して行っており，32より多いノード数ではグループが形成できないので，測定していない．

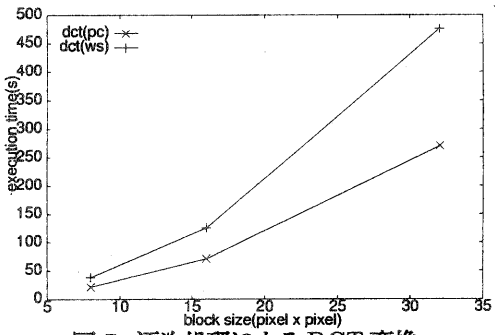

図 7: 逐次処理によるDCT変換

図8ではプロセッサ数が8と16の時，ブロックサイズが大きいほど高い速度向上比を示している．これは，ブロックサイズが小さい場合には計算量が少なくなるため，計算量に対する通信量がより多くなり，通信に係わるオーバーヘッドが速度向上を阻害しているからである．次に，同図に

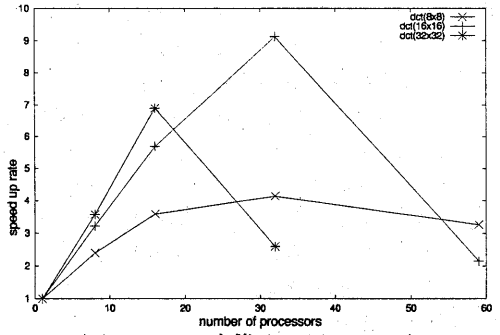

図 8: DCT 変換 (PC クラスタ)

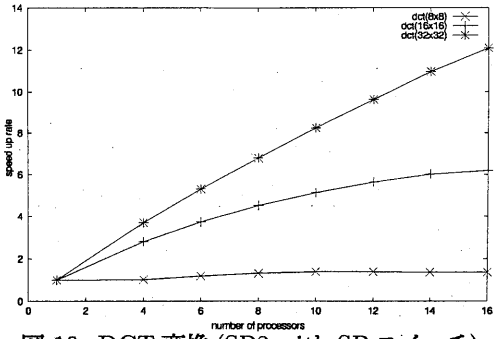

図 10: DCT 変換 (SP2 with SP スイッチ)

おいてプロセッサ数が 32 と 59 の時，ブロックサイズが大きいほど早くに速度向上比が低下するのは，ブロックへの分割数が少ないからである．ブロック数が少ないと，プロセッサ数が増大した場合に，十分なプロセッサへの割り当てが出来なくなり，割り当てられたブロックのないプロセッサが増えてくることとなる．また，ブロックサイズが大きいため，最後に割り当てられたブロックが終了するまでの時間が，アプリケーションの実行時間へより大きな影響を与えている．

また図 9,10では，100Mbps の Ethernet と SP スイッチを利用した場合の逐次処理に対する速度向上比を示している．ただし，本学の SP2 は 16 ノードで構成されるが，100Mbps の Ethernet インターフェイスが 8 ノード分しか実装されていないため，100Mbps の Ethernet を用いた測定は 8 ノードまでとなる．

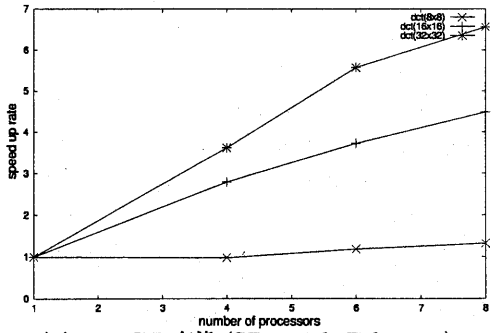

図 9: DCT 変換 (SP2 with Ethernet)

図 9,10では，先ほどの図 8と同様に，ブロックサイズが大きいほどより高い速度向上比を実現している．これは，PC クラスタの場合と同様に，通信量に対するプリミティブの実行時間が，ほぼ一定を保っているからだと考えられる．

しかし，SP2 と PC クラスタで同一のネットワーク媒体を利用した場合で比較すると，図 8, 9より，SP2 ではブロックサイズが大きい時にはより高い速度向上比を示しているが，ブロックサイズが小さい時には非常に低い速度向上比となっている．これは，図 4,5,7,7から，SP2 と PC クラスタ間で計算能力の差の方が通信能力の差よりも非常に大きくなっているからである．このことは，計算時間が実行時間に占める割合が大きいほど，並列処理による速度向上は顕著に現れ，通信時間が占める割合が大きいほど，並列処理による効果は少なくなるという，並列処理の性質を示している．

図 11, 12, 13 では，SP2 におけるブロックサイズ毎のネットワーク媒体と処理速度向上の関係を示している．

ブロックサイズが 8×8 のときように，通信によるオーバーヘッドが大きく働く状況でも，100Mbps の Ethernet と 150MByte/s のピークバンド幅を持つ SP スイッチ間で，システムの処理速度がほぼ等しいことが観察できた．よって，DSE における速度向上比はネットワーク媒体の通信速度にほとんど影響を受けないことが分かった．

4 おわりに

近年，クラスタシステムは大規模な並列処理を行う際の主要なプラットフォームとしての地位を確保しつつある．このクラスタシステム上で，移

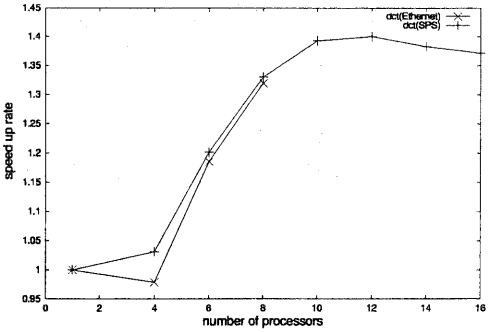

図 11: DCT 変換 (8x8)

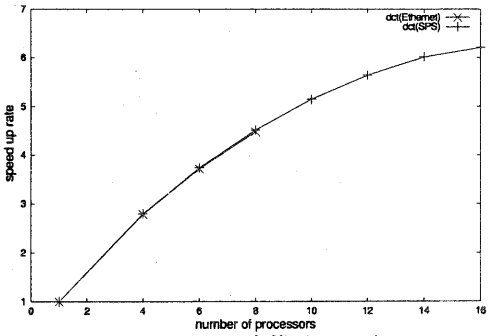

図 12: DCT 変換 (16x16)

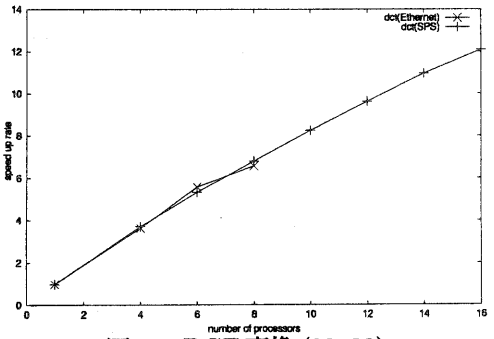

図 13: DCT 変換 (32x32)

植性の良い分散共有メモリプログラミング環境を提供する DSE について，処理能力の測定及び計算能力と通信能力の関係からの評価を行った．

今回の実験では，プリミティブの実行時間に対してはクラスタシステムを構成するノードのアーキテクチャの違いが大きな影響を与えていた．しかし，並列アプリケーションの実行時間を比較するとアーキテクチャの違いよりも単純なクロック速度の違いが大きな影響を与えている．これは，特定の命令が高速に実行できるプロセッサを使ったクラスタよりも，全ての命令が平均して高速に実行できるプロセッサを使ったクラスタの方が，最終的に高速な処理が実現できることを示している．

また，プリミティブの実行時間に対しては，クラスタで使用するネットワーク媒体の速度は影響していない．これは，並列処理により逐次処理より実行時間が短縮されるような問題領域では，実行時間中で通信に係わる時間が非常に小さくなるからである．

プロセッサの違いとネットワーク媒体の違いが，計算速度と通信速度，計算時間と通信時間に与える影響と，並列処理に対する影響をより詳しく観察する必要がある．

参考文献

[1] 手塚 忠則, “分散システムを利用した並列処理環境の構築に関する研究”, 1992 年度 九州工業大学大学院 情報工学研究科情報科学専攻 修士論文, 1993.

[2] Tatsuya Asazu, Bernady O. Apduhan, Itsujiro Arita, “Towards a Portable Cluster Computing Environment Supporting Single System Image”, In Proc. ICPP'99-MMNS Workshop, Sept. 1999.

[3] “IBM Parallel System Support Programs for AIX : Command and Technical References”, rel 2.4, document GC23-3900-05, IBM Corporation, 1998.

[4] C.B. Stunkel, et al, “The SP2 High-Performance Switch”, IBM Systems Journal, 34(2), pp. 185-204, 1995.