A-11

高速通信インタフェースDIMMnet-1での並列アプリケーションによる評価

Evaluation with a parallel application for a fast communication interface that is the DIMMnet-1

三橋彰浩† 濱田芳博† 中條拓伯†
田邊昇†† 工藤知宏†††,☆

AKIHIRO MITSUHASHI ,† YOSHIHIRO HAMADA ,†
HIRONORI NAKAJO ,† NOBORU TANABE †† and TOMOHIRO KUDOH†††,☆

1. はじめに

近年，パーソナルコンピュータ（PC）やワークステーション（WS）の性能が著しく向上し，また価格も安価になってきているため，PC/WSを相互接続したネットワークで構成されるクラスタによって，スーパコンピュータに匹敵する計算性能を持った計算機を安価に導入できるようになった．このため，数十台から数百台のPC/WSクラスタによる並列分散処理が広く行われるようになってきている．

我々は現在，PCIバスでなくDIMMスロットに接続するタイプのNICであるDIMMnet-1[1]を用いたクラスタシステムを開発している．NICをDIMMスロットに接続することでバンド幅の拡大とレイテンシの大幅な削減が可能となり，ボトルネックを解消することができる．また，クラスタシステムを構築し，実際に利用するためにはMessage Passing Interface（MPI）のようなメッセージパッシングライブラリが不可欠であり，本稿ではクラスタシステムへのMPIライブラリ実装の足掛かりとして，RHINET2スイッチによって接続されたDIMMnet-1を用いたバリア同期に要する時間を計測し，DIMMnet-1の低レベルな関数を用いた簡単な並列アプリケーションによる評価を行う．

2. DIMMnet-1の概要

2.1 ユーザレベル通信

ユーザレベル通信は，OSをバイパスしユーザプロセスが直接NICにアクセスして通信を行う方法で，ユーザ空間とカーネル空間の切替えのオーバヘッドを削減することができる．ユーザレベル通信では，通信にかかわる情報がユーザプロセスから直接NICに渡された情報がどのプロセスからのもので，そのプロセスはNIC上のリソースにアクセスする権利を持っているかどうかをNIC側で判断する必要がある．NICは，内部にユーザプロセスの仮想アドレスから物理アドレスへの変換を行うためのTranslation Look-aside Buffer（TLB）を持っている．このTLBに格納される属性によりプロセス間の保護を実現している．

2.2 Atomic on-the-fly（AOTF）

AOTFは，ホストプロセッサからの書き込みによってパケットを送信する機構である．ユーザプロセスによりAOTF用の領域にデータの書き込みが行われると，あらかじめ設定しておいたヘッダシートを元にパケットが生成され，パケットとして送信される．このため，送信データがCPUレジスタ上に存在していればCPUがそのデータを所定のアドレスに書き込むという一命令だけでパケット送信が可能である．AOTFでは一度に最大8bytesのデータを送信することができ，低レイテンシな通信を実現している．

2.3 Block on-the-fly（BOTF）

BOTFはユーザプロセスがパケットを作成し，送信する機構である．ユーザプロセスは，送信をするための領域であるWindow上にヘッダを含めたパケットのすべてを書き込んだ後，Window上のキックアドレスに書き込みを行うことでDIMMnet-1のNICコントローラであるMartini[2]に送信開始を指示する．BOTFではヘッダを除いて一度に最大464bytesのデータが送信可能であり，高バンド幅な通信を実現している．

2.4 OTF受信機構

OTF受信機構とは，アドレス変換やDMAコントローラの起動をすること無しに，パケットヘッダの情報から所定の長さのデータ部を直接メモリに書き込む機構である．

DIMMnet-1ではAOTF送信に限り，リモートアドレスを物理アドレスで登録することができ，受信時のリモートにおけるアドレス変換のオーバヘッドを削除することが可能である．AOTF送信に限って立てることができるヘッダー中のフラグを受信部が判定し，アドレス部と1〜8bytesのデータ部を書込みバッファに押し込んでいく．書込みバッファはMartini上のオンチップメモリである低遅延共有メモリ（LLCM）に，書き込めるタイミングで書き込む．

このようにDIMMnet-1では送信側のAOTFと受信側のOTFRが共同して極めて低遅延な通信を実現している．

2.5 評価環境

評価環境を表1に示す

表1 測定環境

測定環境	(a)	(b)
CPU/FSB	PentiumIII 850MHz/100MHz	
マザーボード	D6VAA	
OS	PentiumIII Linux(Kernel 2.4.2)	
	SCORE 5.0/MPICH 1.2.3	
NIC	DIMMnet-1(Martini2nd) 光モジュール	Myrinet M2M-PC132C-10450
スイッチ	RHiNET2/SW2[3] 通過遅延 240ns	Myrinet M2M-DUAL-SW8

環境(b)はMPI_Barrier測定時のみ使用．

3. バリア同期

3.1 バリア同期の実装

DIMMnet-1におけるバリア同期の実現方法[4]としては，N-1進木のツリーバリア方式を用いた．実装においては到着木，励起木共に同じ木構造を用る．到着木においてリーフノードはルートノードへ1bytesのAOTF送信にて同期フラグを転送する．ルートノードはリーフノードからの同期フラグを8bytesの変数にまとめ，全てのリーフノードに対するポーリングを1回のreadで行う．励起に関しては

† 東京農工大学
Tokyo University of Agliculture and Technology
†† （株）東芝 研究開発センター
TOSHIBA Corporate Research & Development Center
††† 新情報処理開発機構
Real World Computing Partnership
☆ 現在，独立行政法人 産業技術研究所 グリッド研究センター
Presently with National Institute of Advance Industrial Science and Technology

ルートノードは N-1 回の AOTF 送信を行い，リーフノードへ同期フラグを送信する．ここで SW2 ではマルチキャストをサポートしているため，バリア同期マルチキャスト版では，このリーフノードへの AOTF 送信を 1 回にまとめて実装を行った．

3.2 バリア同期評価

図 1 に DIMMnet-1 を用いて実装したバリア同期と，Myrinet を使用した SCore/MPI でのバリア同期の測定結果を示す．DIMMnet-1 を用いたバリア同期のレイテンシは Myrinet を用いたバリア同期に比べてほぼ 10 倍の性能を示した．また，マルチキャストを用いた DIMMnet-1 のバリア同期は，用いない場合のバリア同期に比べ，レイテンシの増加が少ないことがわかる．それはノード数増加に伴う，ツリー励起時の AOTF 送信の増加をマルチキャストで一度に送信するからである．

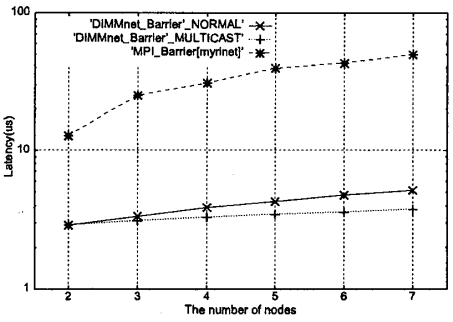

図 1 バリア同期実行時間

4. 評　価

4.1 評価に用いた並列アプリケーション

アプリケーションとして RadixSort と LU 分解を用いた．RadixSort はデータが 6,000,000 個で基数が 10 であり，LU 分解に使用した問題サイズは 1024 × 1024 の実数密行列である．ノード間の同期には，マルチキャストを備えた DIMMnet-1 バリア同期を用い，データ通信には AOTF と BOTF を用いた．第 2 版の Martini では BOTF 送信時に連続ワードを立て続けに送信すると正常に通信ができないという不具合があるため，送信間隔をあけることで回避している．第 3 版の Martini ではこの問題は改善される予定である．

4.2 演算時間・通信時間・同期待ち時間

表 2 に，各アプリケーションの 4 ノードでの実行時間の内訳，すなわち演算時間，通信時間，および同期待ち時間（バリア同期に要した時間＋受信待ち時間）をそれぞれ示す．各実行時間において，台数効果を理想値に近づけるために，RadixSort では通信時間の短縮．LU 分解では主に同期待ち時間の短縮が考えられる．

表 2 実行時間の内訳

	演算時間	通信時間	同期待ち時間
RadixSort	80.8%	18.3%	0.9%
LU 分解	84.1%	0.4%	15.5%

4.3 評価結果

図 2 に，各アプリケーションにおいて 1 ノード時の実行時間を 1 とした場合の台数効果を示す．

RadixSort，LU 分解共に，N ノードで N 倍という理想の性能向上に近い台数効果は得ることはできなかったが，台数を増やすごとに安定した台数効果を得ることができた．

4.4 考　察

4 ノードでの RadixSort の実行時間において，通信時間は 18.3%を占める．現状の BOTF 送信では送信間隔をあけてデータを送信しているため，送信バンド幅は 11MB/s，uncashed で通常命令を用いたメモリへの読み込みバンド幅は 26MB/s に抑えられている．しかし，第 3 版の Martini を用いることで 190Mbytes の通信を行うことが可能とな

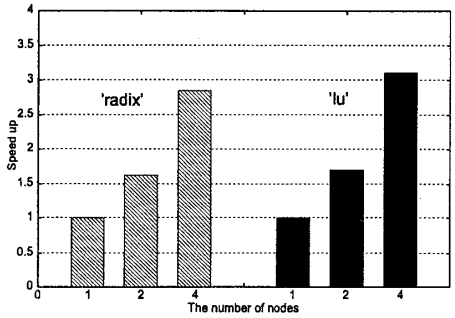

図 2 台数効果

り，読み込みバンド幅は uncashed の属性で 35Mbytes に向上する．このときの通信時間の割合は 14%になり，台数効果は 3 倍に向上する．このとき，メモリへの読み込みに SIMD 命令を用いた場合，読み込みバンド幅を 74MB/s にすることが可能であり，通信時間の割合を 6%に減少させ，台数効果は 3.4 倍に向上させることができる．また，Pentium4 において使用できるキャッシュラインをフラッシュする，読み込みバンド幅 466MB/s の CLFLUSH 命令を使用した場合，通信時間の割合は 0.01%に減少する．実行時間における通信の割合が SIMD 命令の時点で 6%まで減少しているため，台数効果は 3.4 倍と変わらないが，より大きな問題サイズを扱った場合の通信時間による性能の低下は解消することができる．

LU 分解においては，実行時間における通信時間の割合は少ない．これは BOTF 送信，通常読み込み命令という組合せでほぼ理想の通信が行われていることを示している．また，性能向上のためには同期待ち時間の減少が重要であり，アプリケーションの並列化について再考する必要がある．

5. 結　論

マルチキャストを用いた DIMMnet-1 のバリア同期に関しては，各ノードへの同期フラグの転送を AOTF 送信を用いることによって，Myrinet を使用した SCore/MPI でのバリア同期に比べ 10 倍ほど高性能なバリア同期を実装することができた．LU 分解などの計算と通信が同時に行われ，頻繁にバリア同期が用いられるようなアプリケーションの場合，DIMMnet-1 で実装したバリア同期は性能向上に貢献すると考えられる．

4 ノードシステムでの評価の結果，BOTF は LU 分解のような通信と演算を別々に行うことができ，低粒度な通信が多発するアプリケーションに対しては非常に有効であるとわかった．また，RadixSort のような大きいデータをまとめて送信するアプリケーションに対しては，現在の不完全な実装状態にある Martini を用いていたのでは、DIMM スロットや BOTF が潜在的に持っている性能を生かすことは難しいといえる．

参 考 文 献

1) 田邊，山本，工藤，“メモリスロットに搭載されるネットワークインタフェース MEMnet”，情報処理学会計算機アーキテクチャ研究会，Vol99，No67，pp.73-78(1999)
2) 山本，田邊，西，他，“高速性と柔軟性を併せ持つネットワークインタフェース用チップ Martini”，情報処理学会研究報告 2000-ARC-140, pp,19-24(2000)
3) 西，多昌，西村，山本，工藤，天野，“LASN 用 8Gbps/port 8x8 One-chip スイッチ:RHiNET-2/SW”，JSPP2000 pp173-180，(May 2000)
4) 田邊，濱田，須田，山本，今城，中條，工藤，天野，“DIMM スロット搭載型ネットワークインタフェース DIMMnet-1 の通信性能評価”，情報処理学会計算機アーキテクチャ研究会 2001-ARC-145，pp.51-56 (Dec. 2001)