クラスタコンピューティングのための通信路最適化
～*Maestro*ネットワークの開発と性能評価～

山際　伸一[*]　　福田　宗弘[**]　　和田　耕一[**]

[*]筑波大学工学研究科

[**]筑波大学電子・情報工学系

Optimization of Communication Path for Cluster Computing
- Development and Performance Evaluation of *Maestro* Network-

Sinichi Yamagiwa　　Munehiro Fukuda　　Koichi Wada

Institute of Information Sciences and Electronics,

University of Tsukuba

Technical Report No. ISE-TR-00-166

概要

マイクロプロセッサの劇的な性能の向上に伴い，価格/性能比が優れた並列計算システムとしてPCクラスタを利用する傾向が高まっている．しかしながら，従来の汎用ネットワークハードウェアと通信プロトコルで構成される PC クラスタは，通信オーバヘッドが大きく，内在する性能を引き出すことが困難である．独立したシステムとしての PC クラスタの特性を考慮すると，通信の最適化が可能である．

本論文では，クラスタコンピューティング向けの通信路最適化として，ネットワークハードウェアのリンク層におけるバースト転送と通信単位の縮小を提案している．これらの高速化技法を実現するリンク制御プロトコル SMB/FA(Splitting Message with Burst /Fairness Arbitration)を提案し，リンクレイヤコントローラ MLwB(Maestro Link with Burst)への実装について述べている．さらに，MLwB を用いて構築した Maestro ネットワークの通信実験により，本最適化技法が通信性能の向上に有効であることを示している．

Abstruct

The emergence of high-performance microprocessors has made it attractive to use a PC cluster as a parallel computing system with an excellent cost/performance ratio. Most existing PC clusters have used WAN or LAN-oriented hardware products and protocols due to their market availability. This however loses possibilities of improving the intra-cluster communication, which can be further optimized for the use within a geographically small area. We have achieved such performance optimization from the network hardware point of view.

This paper presents two optimization techniques for cluster computing: the burst transfer and the minimization of transfer unit, each at hardware link layer. We propose a link control protocol realizing these two techniques, called SMB/FA (Splitting Message with Burst /Fairness Arbitration). This paper describes its implementation in a link control hardware, referred to as MLwB(Maestro Link with Burst) and demonstrates the efficiency of our proposed optimization techniques through the experiments on the Maestro network constructed with MLwB.

1. はじめに

PC の価格/性能比の向上に伴い，汎用ネットワークで複数の PC を接続した PC クラスタによる，並列・分散コンピューティングに注目が集まっている[3)4)]．

多くの場合，PC クラスタは，広域ネットワーク向けのネットワークハードウェアと通信プロトコルを用いて構成されている．例えば，ネットワークハードウェアについては Ethernet が，通信プロトコルについては TCP/IP が主流となっている．しかし，これらは広域の通信を前提としているため，PC クラスタに適用した場合，通信におけるオーバヘッドが大きく，内在する性能を十分に引き出すことは難しい[8)]．

性能向上を妨げる要因としては，(1)通信プロトコルソフトウェア，(2)デバイスハンドラ，および，(3)ネットワークハードウェアが挙げられる．高性能を実現するには，PC クラスタの構成上の特性，すなわち，地理的に限定された環境に構築される点を考慮して，これらの最適化を図る必要がある．

上記(1)の通信プロトコルソフトウェアに関しては，現在までに，FM[11)]，PM[16)]，BIP[14)]等の研究で性能向上策が提案されている．本研究では，(3)のネットワークハードウェアに特に注目して，クラスタコンピューティング向けのリンク制御プロトコル Splitting Message with Burst /Fairness Arbitration(SMB/FA)を設計し，本プロトコルで制御される Maestro ネットワークを構築した．本論文では，SMB/FA プロトコルの設計，および，Maestro ネットワークの構築について論ずる．さらに，Maestro ネットワークの評価結果を示し，SMB/FA プロトコルに取り入れた高速化技法である，リンク層における(1)バースト転送と(2)送信単位の小粒度化が，クラスタコンピューティング向けの通信に対して有効であることを示す．

以下，本論文では 2 章において，PC クラスタにおける通信形態の特徴を述べ，その問題点を指摘する．3 章では，それらの問題点の解決方法を提案し，Maestro ネットワークの構築について述べる．4 章では，提案する方法の有効性について，実験を用いて議論する．最後に，本論文のまとめと今後の課題について述べる．

2. 従来の通信

2.1 クラスタコンピューティングにおける通信

PC クラスタでの通信形態は,並列処理における情報交換の形態を反映したものとなる．このような通信の特徴として，同期に必要な少量データの頻繁な交換と，行列計算等における大量のデータ転送，が挙げられる．

プロセッサ間の同期は,少量データの送信・受信の組で行われる．このとき，通信の最小単位が同期に要するデータ量より大きい場合，通信レイテンシの増大を招く．すなわち，PC クラスタにおける通信機能として，同期等で要求される小粒度の通信を低レイテンシで実現できることが求められる．

一方，大量のデータ転送に関しては，広域ネットワークのための通信プロトコルでは転送データが複数個の単位に分けられ,複数回に渡って送受信される．この複数回の送受信操作に伴うオーバヘッドは，スループット低下の一因となっている．並列計算でしばしば必要となる大量のデータ転送に対応して，適切な粒度でのバースト転送が効率良く行えることも，クラスタ向けネットワークに必要とされる機能である．

以上に加えて，ルーティングにおいても，クラスタは広域ネットワークと異なる特徴を持つ．例えば，クラスタでは固定的な計算機を対象とし，地理的に限定された範囲でルーティングを行うため，広域ネットワークのルーティングにおける IP(Internet Protocol)アドレスから MAC(Media Access Control)アドレスを導出する過程は不要である．すなわち，ルーティングに対しても，冗長な処理を排して最適化されたクラスタ向けルーティングが必要である．

2.2 従来の通信機能と遅延要因

本節では， PC クラスタにおける通信オーバヘッドの個々の要因について述べる．

(a) 通信プロトコルソフトウェア

通信プロトコルソフトウェアの問題は，アプリケーションプログラムに，PC が受信するメッセージを渡すまでのデータのコピー回数である．コピー回数の増加により，通信レイテンシが増大する．さらに，コピー操作を OS カーネル内で行うと，OS の実行モードをユーザモードからカーネルモードへ移行する必要がある．従って，粒度の小さい通信を多量に行うと，この実行モード移行

のためのオーバヘッドにより通信レイテンシが増大する．

これらの問題に対してはFM[11]，PM[16]，BIP[14]等のプロジェクトで性能改善策が提案されている．これらの通信プロトコルソフトウェアでは，アプリケーションプログラムからネットワークハードウェアに送信データが渡されるまで，コピー処理を伴わない0コピー通信を行っている．

(b) デバイスハンドラ

PCのデバイスハンドラでの問題として，それによって確保される転送データ用の領域が不連続になることがあげられる．例えば，デバイスハンドラは，送受信データのための領域として，(1)カーネル空間のセグメント（数百バイトから数Kバイト），または，(2)メモリページ（4Kバイト固定または8Kバイト固定），を複数確保する．確保されたセグメント，または，メモリページからなる領域は，仮想アドレス空間では連続であるが，物理アドレス空間で連続している保証はない．従って，数十Kバイトの長いメッセージは，物理メモリ上の不連続領域に分散配置される可能性がある．分散配置された場合，PC上のメモリ領域からネットワークハードウェアにDMA転送する際に，(1)仮想アドレスから物理アドレスを求め転送開始アドレスとし，(2)セグメント，またはページの境界までのDMAを起動する，といった一連の操作が繰り返し必要となる．この仮想-物理アドレス変換，およびDMAの起動操作のオーバヘッドが，スループットの低下を招く．

上述のような，分散領域に対する転送に伴うオーバヘッドを削減するには，大きな連続領域を確保できる機能がデバイスハンドラに必要である．

(c) リンク層

(i)フレーミングに伴うオーバヘッド

アプリケーションプログラムから送信を要求したデータを，リンク層でフレーミングする例としてEthernetを考える．Ethernetでは，最大転送データ長が1500バイトに規定され，これに36byteのヘッダとフッタが付加される[2]．したがって，送信データの大きさを*Pn* byteとすると，36/(*Pn*+36)%がリンク層で認識される情報である．プロセッサ間の同期などに使われる最小データの長さを4byteと仮定すると，36/(4+36)%= 90%がヘッダとフッタの情報となる．このように，クラスタ環境の特性を考えると広域ネットワークを前提とした付加情報は冗長で，高性能化のためには，クラスタに適したフレーミングを設計しなければならない．

(ii) 粒度の小さいメッセージに伴うオーバヘッド

多くのネットワークハードウェアでは，送受信の際，PC 間とのデータ転送に DMA が用いられる．しかし，少量のデータを転送する場合，DMA を設定・起動するオーバヘッドの転送全体に占める割合が大きくなる．この問題に対しては，ネットワークハードウェアに少量のデータ転送を高速に行える機能を持たせ，転送データ量によって，DMA と適応的に使い分けることで対応できる．

(iii) non-burst 転送によるオーバヘッド

従来のネットワークハードウェアでは，一度の送信機会に一つの送信単位（例えば、パケット）のみを送信する．このため，小さいデータを複数回送信するときには，それらの総データ量が一度の送信機会で送信できる量であっても，その回数分の送受信操作を繰り返す．通信媒体の利用率を高めるには，ネットワークハードウェアのリンク層が，一度の送信機会で複数の送信単位を一括して送出できることが重要である．

(d) 物理層

マルチキャストに伴うオーバヘッド

従来のマルチキャストでは，全ての PC がメッセージを受け取り，該当しない PC はこれを消去するか，または，該当する複数の PC に対して，その台数分の通信を繰り返す．前者は，Ethernet のような，キャリアセンシティブなメディアで使用される．しかしながら，該当者以外も受信して，それを消去するオーバヘッドが発生する．後者は，送信側が受信側の数だけ，メッセージをコピーして送らなければならない．このようなオーバヘッドを回避するには，物理層において，メッセージの選択的なコピーを行う機能が必要である．

これらの各層における遅延要因のうち，近年のクラスタコンピューティングに関する研究では(a)通信プロトコルソフトウェアを中心として行われているものが多い．(b)デバイスハンドラに関しても，Myrinet[5]，U-net[6]などにおいて積極的に改善が試みられている．我々は，(c)リンク層，(d)物理層に注目し，その高速化技法を検討した．次章で，本技法の詳細と，それを使った SMB/FA プロトコル，および，SMB/FA プロトコルを実装したリンクコントローラ MLwB について説明する．

3. クラスタ内通信の高速化と実装

3.1 クラスタ内通信の高速化技法

以下に,前節で考察を行った事柄について性能改善を実現する方法を述べる.

(a) リンク層におけるバースト転送の実現

物理層が，通信媒体へのデータ送信権を獲得するための調停について考える．調停回数が増加すると，通信媒体の利用効率が低下し，高いスループットを得ることが困難となる．そこで，我々はリンク層におけるバースト転送を実現することを考える．すなわち，物理層により取得した送信機会毎に，リンク層における最小送信単位の整数倍のデータを一括して送信する機能を実現する．本機能により，長いメッセージの送信時間を短縮し，スループットを向上させることが可能となる．以降では，このリンク層におけるバーストを伴う送信操作をネットワークバーストと呼ぶことにする．

(b) リンク層における送信単位の縮小

従来の通信インターフェースでは，リンク層での送受信はメッセージを単位として行われる．この場合，全てのデータが PC から通信インターフェースに転送されるまで次の送信処理がブロックされるので，(1)送信側 PC から通信インターフェースへの転送，(2)送信側および受信側の通信インターフェース間の通信，(3)受信側の通信インターフェースから PC への転送が逐次的になる．この様子を図 1(i)に示す．図 1での各実線上の番号は，前述の処理の番号に該当する．

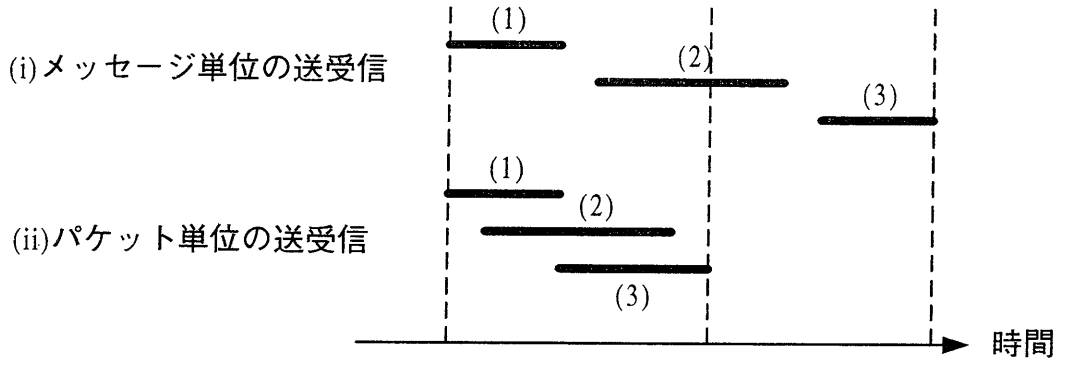

図 1　送信単位の縮小による高速化

この通信の逐次化を回避するために，我々は，リンク層での送信単位の縮小化を行う．この小さい送信単位のことをパケットとよぶ．これにより，PC から通

信インターフェースへの転送操作と，リンク層での送信操作が多重化される．図 1 (ii)は，この高速化技法を用いた場合の時間の短縮を表している．

3.2 Maestro ネットワークの実装

本節では，前述した高速化技法を適用した Maestro ネットワークの全体構成とその動作概要について述べ，SMB/FA プロトコルの詳細と，それを実現したリンクレイヤコントローラ MLwB(Maestro Link with Burst)について論ずる．

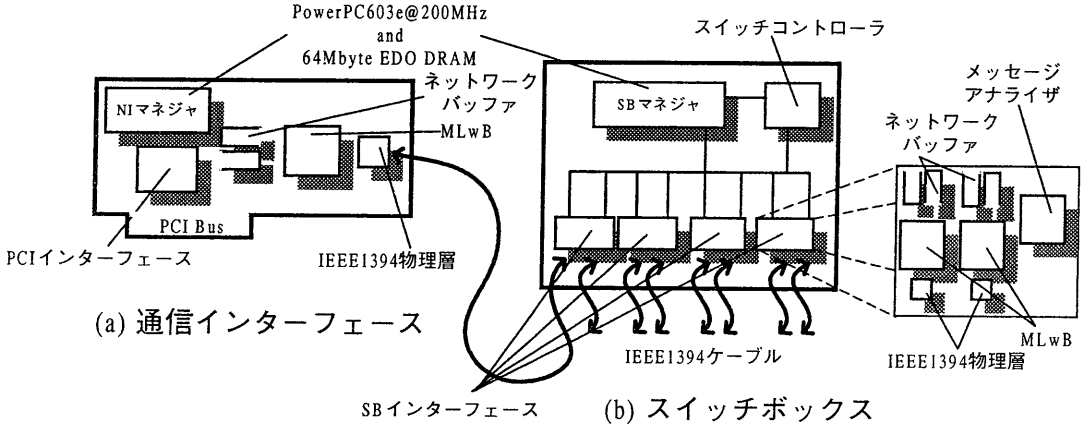

図 2 Maestro ネットワーク

3.2.1. 全体構成

図 2に示すように，Maestro ネットワークは，各 PC に PCI バス[12]を介して接続される通信インターフェース，および，IEEE1394[7]を介して各通信インターフェースからのメッセージを受信し，転送を行うスイッチボックスから構成される．以降，特に断らない限り，Maestro ネットワークの通信インターフェースとスイッチボックスを，それぞれ，NI(Network Interface)および，SB(Switch Box)と略す．NI から送信されるメッセージは，図 3 に示すフォーマットに従って，ヘッダと単一または複数のパケットに変換される．単一パケットを有するメッセージを Type0 メッセージ，複数パケットを有するメッセージを Type1 メッセージとして区別する．パケットはネットワークバーストにおける最小転送単位の集合である．メッセージヘッダはルーティングのための情報を保持する領域で，SB によって解釈される．図中の PktCounter はメッセージを構成するパケット数を保持する領域である．

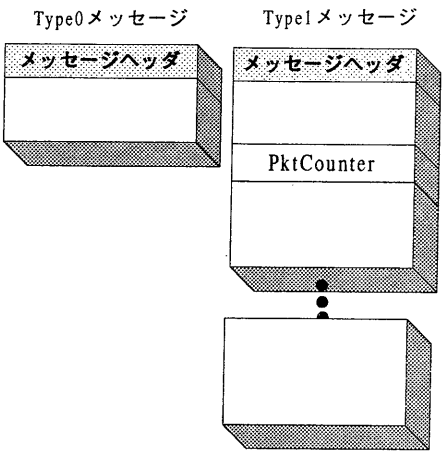

図 3　SB で認識可能なメッセージフォーマット

NI は DMA コントローラを内蔵した PCI インターフェース[13]，マイクロプロセッサ PowerPC603e(200MHz)[10]と 64Mbyte の EDO DRAM を搭載した NI マネジャ，送信，および，受信用のネットワークバッファ，SMB/FA プロトコルを実装した MLwB，および，200Mbps IEEE1394 物理層から成る．

このうち，PCI インターフェースは，NI が PC と通信するための仲立ちを担うだけでなく，内蔵の DMA コントローラを使って，PC のメモリと NI マネジャの DRAM，および，ネットワークバッファとの間のデータ転送を行う機能を持つ．

NI では，NI マネジャ内の DRAM を，PCI インターフェースを介して PC から直接アクセスできるように設定できる．これにより，PC から受け取ったデータを加工する操作を PC とは独立に行うことができる．さらに，DMA 設定のオーバヘッドが顕著に表れる程度の小さなデータを転送する場合は，NI マネジャ内の PowerPC に代行させる．

MLwB には，ネットワークのフロー制御と IEEE1394 物理層への物理的な転送手順の実現を行う SMB/FA プロトコルが実装される．SMB/FA プロトコルは，通信機会の公平化とネットワークバースト，および，メッセージより短い単位での送信操作を実現する．このプロトコルと MLwB の実装についての説明は次節で行う．

一方，SB は，PowerPC603e(200MHz)と 64Mbyte を搭載した SB マネジャ，DMA コントローラを内蔵したスイッチコントローラ，異なる 2 つの NI からの通信を受理する SB インターフェース 4 対から構成される．

各 SB インターフェースは，1 つのメッセージアナライザ，二対の MLwB，

それに付随するネットワークバッファ，および，IEEE1394物理層を搭載している．このMLwBとIEEE1394物理層については，NIに搭載したものと同一のものを用いる．メッセージアナライザは，NIから転送されるメッセージがType0，Type1のどちらであるかを解析し，メッセージヘッダ部分のみをSBマネジャに転送する．

SBマネジャは，メッセージアナライザからのメッセージヘッダを解析し，メッセージの転送先を決定した後，スイッチコントローラに転送を要求する．スイッチコントローラは，転送要求を受け取り，異なるSBインターフェース上のネットワークバッファ間でDMA転送を行う．図 2(b)に示すように，4つのSBインターフェース間はバス結合されており，スイッチコントローラは，これを利用して選択的に複数のSBインターフェース内のネットワークバッファにメッセージ転送を行う．これにより，前述した従来のネットワークにおける問題点のうち，マルチキャストに伴う遅延要因を削減できる．

3.2.2. SMB/FAプロトコル

IEEE1394のような半二重通信媒体では，その双方が同時に送信を行うことはできない．そこで，我々は，半二重通信媒体にも対処できるクラスタコンピューティング向けの新しい通信プロトコルSMB/FAプロトコルを提案する．

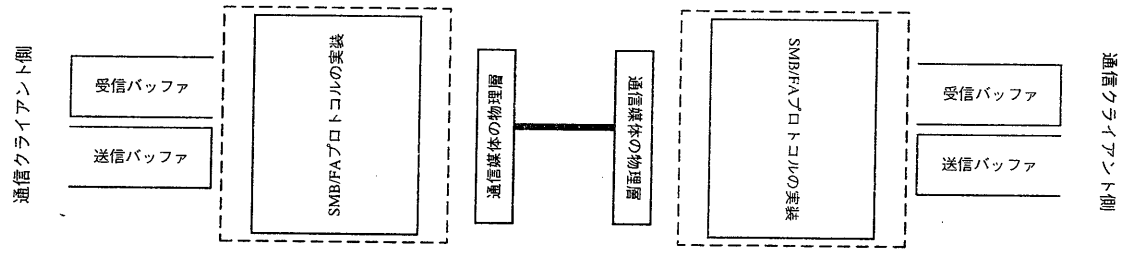

図 4 SMB/FAプロトコル周辺の構成

SMB/FAプロトコルはpoint-to-pointの通信路を規定し，図 4に示すような入出力を想定している．プロトコルを実装するモジュールは，通信媒体の物理層とネットワークバッファに接続される．ネットワークバッファは，FIFOバッファとして構成され，通信クライアント側からメッセージを単位として書き込み・読み出しが行われる．一方，通信媒体の物理層側からはメッセージより細かいパケット単位での送受信が行われる．

SMB/FAプロトコルの特徴は，(1)メッセージより細かなパケット単位での転送，(2)公平な送信権獲得制御による双方向通信，(3)ネットワークバーストを可

能にするフロー制御，(4)複数チャネルの実現，の４点である．

(1) メッセージより細かなパケット単位での転送

SMB/FA プロトコルでは，従来の通信インターフェースで行われていた，メッセージを単位とする送信操作を，より細かく分割したパケットを単位として行う．これにより，前述の高速化技法のうち，リンク層での送信単位の縮小を実現できる．

(2) 公平な送信権獲得制御による双方向通信

半二重通信媒体では，一方のリンク端に送信権が集中することがあり，送信権を得られないリンク端の送信メッセージがブロックされることがある．このように，片方のリンク端が連続的に送信権を獲得すると，もう一方からのメッセージの通信レイテンシは増大する．SMB/FA プロトコルではこのような状況を回避するために，送信権取得を完全に公平に制御する．リンク端は送信を行ったら，必ず受信を行う．しかし，この規則は，送信と受信の組が成立しないとデッドロックを引き起こす．これに対処するために，各送信機会に返信すべきメッセージが無い場合には，特別な空パケットを返信する．このパケットのことをデッドヘッドパケットと呼ぶ．一方，メッセージの一部を含むパケットのことをサービスパケットと呼ぶ．

(3) ネットワークバーストを可能にするフロー制御

SMB/FA プロトコルでは，受信側のバッファの空き容量を表す情報を送信側に知らせておくことにより，ネットワークバーストを行う．このバッファの空き容量情報を Credit とよぶ．Credit は，リンクの両端が送信するデッドヘッドパケット，および，サービスパケットに付加され交換される．

(4) 複数チャネルの実現

ネットワークバッファは，複数のチャネルで構成され，チャネル毎に送信のための優先順位を設ける．送信機会毎に高々１つのチャネルに書き込まれたメッセージが送信される．

以上の特徴において，(1)は通信レイテンシを低減し，(2)は通信媒体の送信権のリンク両端へ均等な配分を行い，(3)は通信媒体の使用効率を上げる効果がある．また，(4)については，通信媒体の物理的な機能追加をすることなく，通信の多重化を可能とする．

3.2.3. MLwB の実装

MLwB は，SMB/FA プロトコルを実装し，ネットワークバッファのフロー制

御と IEEE1394 物理層の制御を行う．

MLwB は FPGA で実装し，Altera MAX7256-71)を用いた．

以下に，実装する MLwB の仕様を示す．

- 最大転送能力 200Mbps の IEEE1394 物理層，データバス幅は 4bit
- 送受信バッファは各々 2 チャンネル(Ch0，Ch1)，各容量 2Kbyte
- サービスパケットに含まれるメッセージの最小断片 16byte
- 送受信バッファのバス幅 16bit

この仕様を満たす(1)デッドヘッドパケットと，(2)サービスパケットのフォーマットを図 5に示す．通信レイテンシは，サービスパケットのヘッダを除くパケット長に比例する．このため，通信レイテンシを縮小するためには，そのパケット長の最小値を可能な限り短縮することが理想的である．しかし，極度に小さくすると，ネットワークバッファの各チャネルのアドレッシングに使用するアドレスレジスタの幅を大きくしなければならず，ハードウェア量が増加する．本 MLwB では，我々が用いた FPGA で実装可能な 16byte をパケット長とした．

また，図 5に示すように，デッドヘッドパケット，および，サービスパケットは各チャネルの Credit を含む．IEEE1394 物理層に合わせ，並列に送ることができるデータ幅は 4bit とした．各 Credit は 4bit 幅×1bit 長で構成され，パケット数を保持する．最大 15 パケットのバーストが可能である．サービスパケットは 32×4bit(16byte)から 480×4bit(240byte)まで可変であり，メッセージの断片を 16byte の倍数で一度に送信することができる．

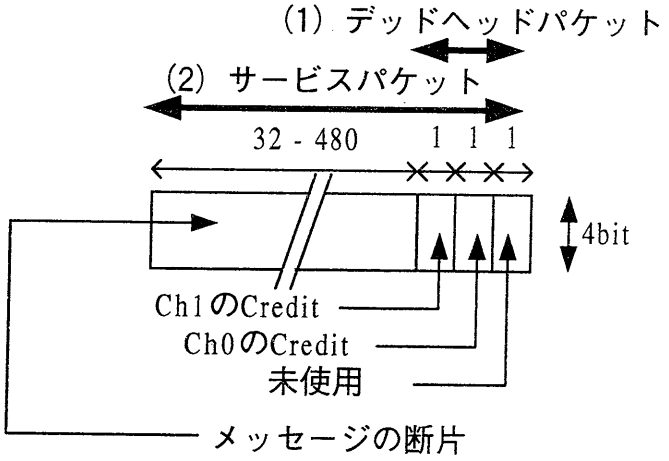

図 5　MLwB におけるパケットフォーマット

3.2.4. 通信の流れ

Maestro ネットワークにおける通信の流れを，一方の PC メモリから他方の PC メモリに，メッセージを転送する例を用いて説明する．

最初に，PC のメモリに 300byte のメッセージが用意されたとする（図 6(a)）．このメモリは，OS が起動する前に予約するメッセージ転送用の連続領域であり，予約メモリとよぶことにする．その予約領域は，デバイスドライバによってアプリケーションプログラムの一部にマップされる[15]．用意されたメッセージは，PCI インターフェース内の DMA，または，NI マネジャの PowerPC が直接ネットワークバッファに転送する（図 6(b)）．ネットワークバッファへのメッセージ書き込みが 16byte を超えると，直ちに，MLwB によりサービスパケットに変換され，SB へと転送される（図 6(c)）．このとき，MLwB におけるネットワークバーストの最大長は 240byte であるため，最少で 2 度の送信機会に分割される．この分割の間に SB 側が送信権を獲得できるので，SB 側のネットワークバッファに送信すべきメッセージがある場合，SB の MLwB が送信を行う．これにより，片方のリンク端による長時間のバスの占有，偏った送信の続行を解消できる．

SB では，NI から受信したメッセージのメッセージヘッダが SB マネジャに渡される．SB マネジャはメッセージヘッダを解析し，スイッチコントローラにメッセージ転送を要求する（図 6(d)）．このとき，マルチキャスト要求がメッセージヘッダに指示されていると，送信元から複数の送信先にスイッチコントローラが DMA を行う（図 6(d')）．この DMA は送信先の数だけ転送を起動するのではなく，1 つのネットワークバッファから，複数の送信先ネットワークバッファへ同時転送を行う．

SB のネットワークバッファにメッセージのコピーが行われると，MLwB により，NI に送信される（図 6(e)）．メッセージを受け取った NI の MLwB は，送信処理の反対の順序で処理を行い（図 6 (f)），PC の予約メモリへと書き込みを行う．このとき，MLwB のパケット単位毎の受信機能が働いて，受信側 NI のネットワークバッファから PC 上のメモリへの転送のうち，先頭 16byte 以降のメッセージは，MLwB のネットワークバッファへの書き込み処理とオーバラップされて処理できる．

以上，本 MLwB は，我々の提案する 2 つの高速化技法，すなわち，リンク層

におけるバースト転送の実現，および，送信単位の縮小を実現していることを述べた．

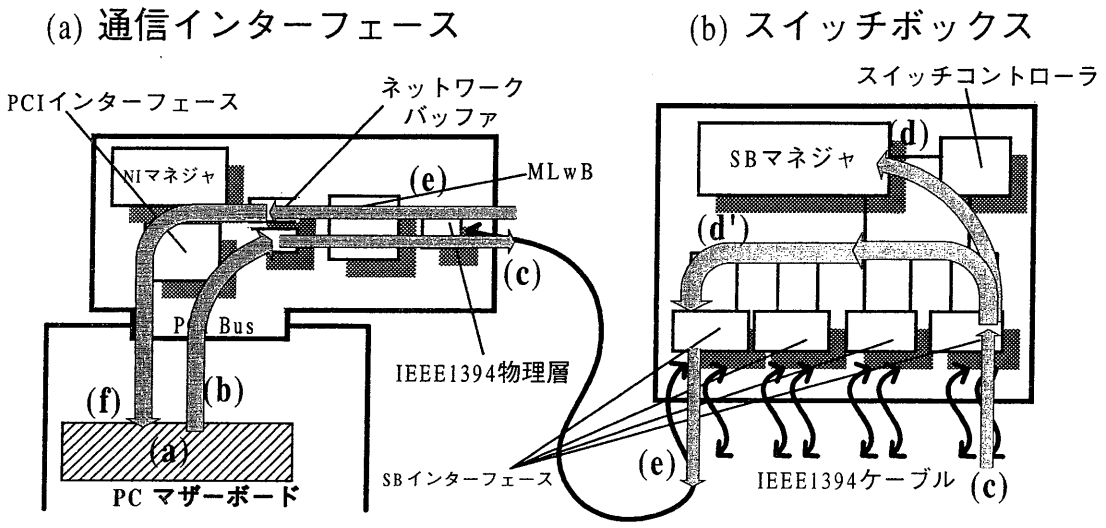

図 6　Maestro ネットワークにおける通信の流れ

4. 性能評価

本節では，Maestro ネットワークの 1)基本性能，2)リンク層におけるバースト転送の効果，および，3)リンク層における送信単位の縮小の効果，を示す．本章で扱う全ての実験には3.2.4節で説明した DMA 操作を用いる．時間計測は，NI 上の PowerPC のタイムベースレジスタをタイマとして用い，送信側が DMA によりメッセージをネットワークバッファへ転送する直前から，受信側が Acknowledge として送信側に 16byte の Type0 メッセージを返し，これを送信側が受け取るまでの時間とする．以下の実験結果では，この時間を Round Trip Time として表示している．

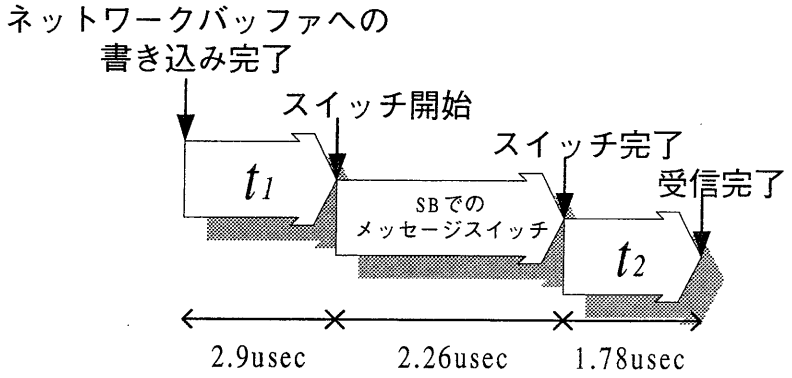

図 7　MLwB 間のレイテンシ

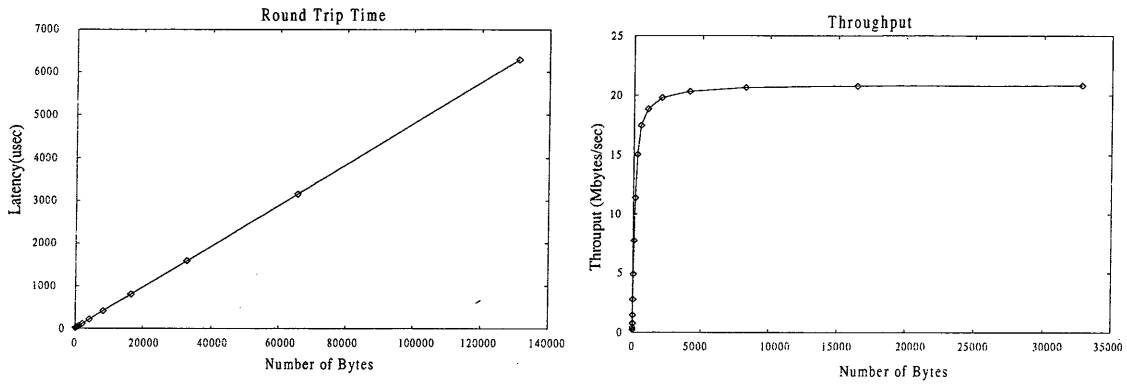

図 8 Maestro ネットワークの基本性能

4.1 Maestro ネットワークの基本性能

MLwB 間の転送レイテンシは 100MHz のロジックアナライザで計測した．16byte 構成のパケット 1 個を転送したときの各区間でのタイミングチャートを図 7に示す．送信側の時間 t_1 は，MLwB による送信権獲得に必要な最大時間を示している．受信側はすぐに Type0 メッセージを返すことができるので，受信側の時間 t_2 は，MLwB による送信権獲得のための最小時間とすることができる．送信権を獲得する時間は，送信権を放棄した直後に，送信バッファに 16byte 以上のメッセージの一部が書き込まれた時，最大となる．送信権獲得時間には 1.12usec の差があるが，この時間を考慮に入れても約 8usec (t_1 + SB でのメッセージスイッチ + t_2)で受信側 MLwB に到達できる．これは，IEEE1394 標準リンク層における転送遅延の最悪値である 125usec に比べ，約 1/15 である．

図 8に PC メモリ間の転送についてのレイテンシとスループットのグラフを示す．最大スループットは約 20Mbytes/sec であった．これは，IEEE1394 の 200Mbps 物理層でのピーク性能の 80%を達成しており，十分な効果をあげているのがわかる．

4.2 リンク層におけるバースト転送の効果

図 9は，ネットワークバーストを用いない場合について，パケット長を 16byte, 32byte，64byte に固定した場合と，我々が行ったネットワークバーストによる高速化との比較を表している．

バーストを行わず，パケット長を増加させるとスループットは向上する．最大スループットは，1 パケットを 16byte から 64byte にすることにより 1.6 倍以上改善されるが，逆にレイテンシは 3 倍以上増加する．

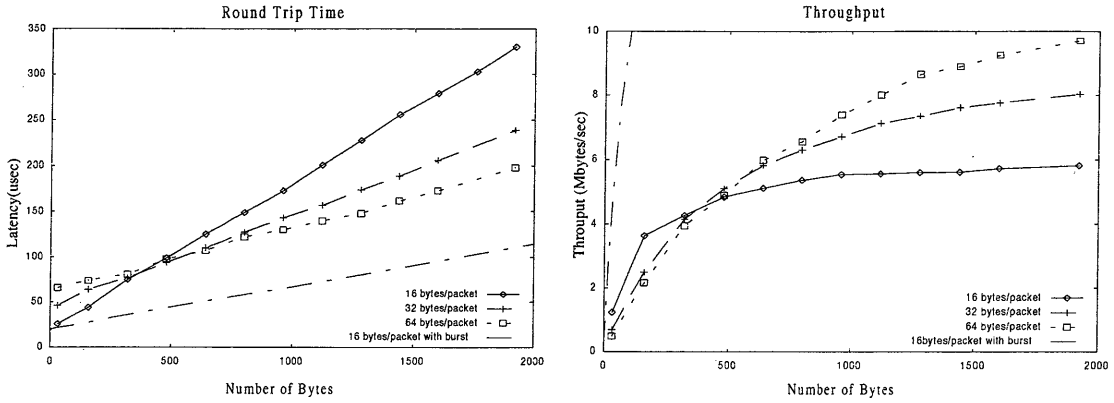

図 9　リンク層におけるバースト転送の効果

4.3 送信単位の縮小による効果

ネットワークバッファに書き込んだメッセージが，メッセージを単位として物理層に送信される場合と，パケットを単位として行われる場合の比較を図 10 に示す．この図では，メッセージを単位とした場合を Message-based Transfer，パケットを単位とした場合を Packet-based Transfer として表している．

Maestro ネットワークでメッセージ単位の送信を計測するために，最初は MLwB をオフにしておき，PC から送信側 NI のネットワークバッファにメッセージ全体の書き込みが完了した後，MLwB をオンにする．これは，最初からオンにすると，ネットワークバッファへ 16byte 以上書き込み時点で送信操作が開始されるからである．これに対して，受信側 NI の MLwB はオンにしておき，メッセージ全体がネットワークバッファに到着したのを確認した後，PC に転送する．図 10の Round Trip Time は，送信側の PC から NI への DMA 操作の時間と，送信側 NI から受信側 PC までの時間を別に採取し，これらを足し合わせた時間とする．Maestro ネットワークにおける，NI のネットワークバッファの最大容量が 2Kbyte なので，16byte から 2Kbyte までのメッセージ長で実験した．

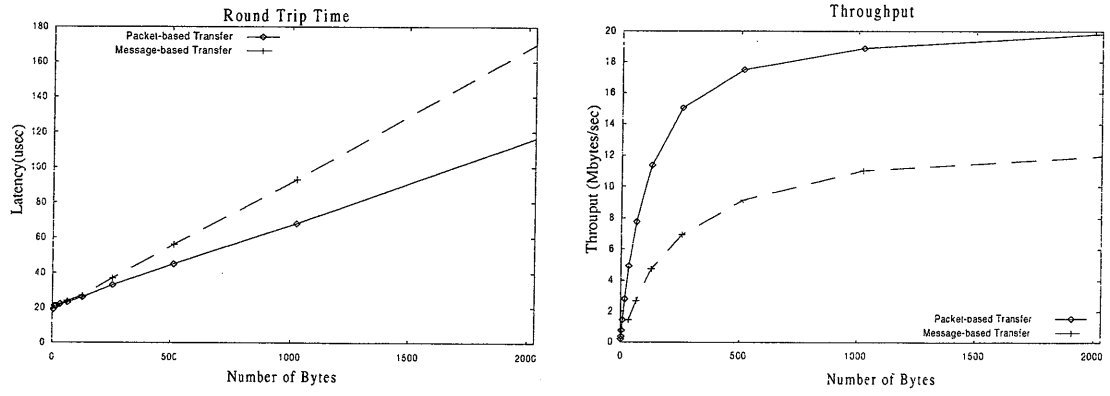

図 10　送信単位の縮小による効果

長いメッセージをパケットに分割して転送した場合，送信側 NI の DMA の完了を待たずに，受信側にメッセージヘッダが渡され，受信操作を開始できるため，メッセージ単位の転送方法に比べてスループットが高い．我々のパケット単位による方法では，メッセージ長が 16byte 付近であれば，メッセージ単位による転送との差は無いものの，メッセージ長の増大に比例して，送信と受信のオーバラップ時間が長くなる．実験から，メッセージ単位による転送に比べて，最大で 5.4Mbytes/sec スループットが向上している．

4.4 結果の考察と議論

以下に評価結果をまとめ，考察を加える．

(1) 基本性能

8usec の低レイテンシと，通信媒体のピーク性能の 80%を達成するスループットを実験により示した．半二重通信媒体である IEEE1394 を用いながら，双方向通信を実現しつつ，高いスループットが維持できていることを確認した．

(2) リンク層におけるバースト転送の効果

本実験により，ネットワークバーストが通信媒体の使用効率を向上させることができることを実証した．Maestro ネットワークでは半二重通信媒体を使用しているにも関わらず，高いネットワーク使用効率を示した．これは，バースト転送によるところが大きい．

IEEE1394 の次世代物理層（400Mbps，800Mbps）を用いると，Credit の bit 幅を大きくとれるため，より長いネットワークバーストが実現できる．従って本方式は，次世代物理層において，より有効であると考えられる．

(3) 送信単位の縮小による効果

Maestro ネットワークで行っているパケット単位の転送の場合，メッセージを単位とする場合に比べて約 5Mbytes/sec の性能向上を示した．この実験から，送信単位を縮小し，送信操作と受信操作をオーバラップさせることが，長いメッセージに対して高いスループットを維持する有効策であることがわかる．

5. 関連研究

Maestro ネットワークに適用した 高速化技法と，関連研究とを比較する．

(1) リンク層におけるバースト転送の実現

TCP 等で用いられているスライディングウィンドウは，フラグメント化された断片を非同期的に送信することによりスループットを向上させている．しかし，

フラグメント化された断片は，IP，Ethernet デバイスハンドラへと渡され，ヘッダ，フッタが付加されたメッセージとして構成されるので，メッセージ長の増大，レイテンシの増大につながっている．

また，Myrinet では，通信媒体に"B" bit を規定し，STOP and GO によるフロー制御を行い，バースト転送を可能にしている．Myrinet では 8bit 毎に送信を行うが，STOP を受け取るまで連続送信可能である．しかし，STOP and GO によるフロー制御アルゴリズムでは，バッファのチャネル毎に専用線を必要とするので，チャネル数の増加に伴って"B"bit を増加させる必要がある．

これに対し，SMB/FA プロトコルでは，ネットワークバッファに用意された1つのチャネルからメッセージがパケットに細分化され，バースト転送されている途中で，異なるチャネルのメッセージに属するパケットが転送に割り込んでも構わない．従って，通信媒体が頻繁にアイドル状態に陥ることはない．また，SMB/FA プロトコルで規定するチャネルは，通信媒体に専用線を規定しないので，ネットワークバッファ量に比例する数のチャネルを構成できる．

(2) リンク層における送信単位の縮小

TCP で行っているフラグメンテーションが，我々の提案する方法と似た操作を行っているが，フラグメント化された後の操作が異なる．TCP ではフラグメント化の後，それぞれをメッセージとして扱うため，送受信操作におけるオーバヘッドが大きい．また，多くのシステムでは TCP の実現は PC で行われ，通信インターフェースからの割り込み制御で送受信操作を起動する．このため，PC 内での割り込みハンドラとデバイスハンドラによって生ずるソフトウェアオーバヘッドが非常に大きくなる．

Myrinet を用いた FM2.0[9]では，送受信の開始・終了を担う関数と，メッセージの断片を送受信する関数を規定している．これらを用いて，送受信操作のパイプライン化を図っている．しかし，Myrinet は，送信時に，通信インターフェース上の SRAM にデータを一時格納しなければならないので，16byte 程度の非常に小さな断片では PC と通信インターフェースの間の DMA オーバヘッドが大きくなり，実効スループットが低下する．また，Myrinet では，メッセージの先頭にルーティング情報と，最後に tail 情報を付加する必要がある．これに加えて，通信インターフェース上から通信媒体への送信 DMA は，各メッセージの最後で区切られるので，あるメッセージの末尾部分と次のメッセージの先頭部分を一度の DMA で送信することができず，さらにスループットを低下させて

いる．

一方，我々の提案した手法では，通信インターフェース上でメッセージのフラグメンテーションと転送処理を行うため，TCP のフラグメンテーションや，FM2.0 の方法と比べて，通信媒体のアイドル時間を削減することができる．

従って，我々が提案した２つの高速化技法は，メッセージの基本的な操作面において，TCP のフラグメンテーション，Myrinet のチャネルの複数化，および，FM2.0 のパイプライン転送と関連しているものの，リンク層と物理層における通信性能の改善に一層重点を置き，システム全体の性能向上を図っていると言える．

6. おわりに

本論文では，従来の PC クラスタに使用されている通信方法の問題点を列挙し，特に，リンク層と物理層のレベルから通信ハードウェアアーキテクチャの改善を試みた．その改善方法として，リンク層においてメッセージをより短いパケットに分割し，バースト転送する方法を提案した．これら２つの高速化方法に，半二重通信用の公平な送信獲得権の配分方法を加え，SMB/FA プロトコルとして規定し，これを MLwB に実装した．MLwB を搭載した PC 用通信インタフェースとスイッチボックスからなる Maestro ネットワークを構築し，性能実験を行うことにより，我々の高速化技法が PC クラスタ向けネットワークの最適化に貢献していることを示した．

Maestro ネットワークで得られた成果は，point-to-point の通信路を構成できれば，他の通信媒体にも適応できる．その例としては，USB(Universal Serial Bus)[17] が挙げられる．また，Maestro ネットワークの IEEE1394 物理層の高速化に伴う MLwB の変更としては，データバス幅の拡大のみである．データバス幅が拡大すると 4bit の物理層から 16bit 幅のネットワークバッファのデータ幅に変換する必要がなくなるので，通信インターフェースをより小さな回路規模で構成できる．

今後の課題として，本論文で提案した高速化技法を利用するメッセージパッシングライブラリの実装，および，共有メモリライブラリの実装を考えている．

参 考 文 献

1) Altera Corporation: *1998 Data Book*, (1998).

2) Robert Breyer, Sean Riley: *Switched and Fast Ethernet, Second Edition*, Ziff Davis Press,(1996).

3) Rajkumar Buyya: *High Performance Cluster Computing: Architectures and Systems*, Prentice Hall,(1999).

4) Rajkumar Buyya: *High Performance Cluster Computing: Programming and Applications*, Prentice Hall,(1999).

5) Nannette J. Boden, Danny Cohen, Robert Felderman, Alan E. Kulawik, Charles L Sietz, Jacov N. Seizovic, and Wen-King Su: *Myrinet - A Gigabit-per-Second Local-Area Network*, IEEE Micro 15(1), (1995).

6) T. von Eicken, A. Basu, and W. Vogels: *U-Net: A User Level Network Interface for Parallel and Distributed Computing*, In Fifteenth ACM Symposium on Operating Systems Principles, pages 40-53, (1995).

7) IEEE Standard Department: *IEEE Standard for a High Performance Serial Bus Draft7.1v1*, (1994). http://www.1394ta.org

8) V. Karamcheti and A. Chien: *Software Overhead in Messaging Layers: Where Does the Time Go?*, Proceedings of International Conference on Architectural Support of Programming Languages and Operating Systems (ASPLOS-VI), (1994).

9) Mario Lauria, Scott Pakin, and Andrew A. Chien: *Efficient Layering for High Speed Communication: Fast Messages 2.x*, Proceedings of the 7th High Performance Distributed Computing (HPDC7) conference (Chicago, Illinois), (1998).

10) Motorola: *MPC603e & EC603e RISC Microprocessors Users Manual*, (1997). http://www.mot.com

11) Scott Pakin, Vijay Karamcheti, Andraw A. Chien: *Fast Messages (FM): Efficient, Portable Communication for Workstation Clusters and Massively-Parallel Processors*, IEEE Concurrency, vol. 5, no. 2, pp. 60-73 , (1997).

12) PCI Special Interest Group: *PCI Local Bus Specification. Rev. 2.1*, Hillsboro, Oregon: PCI Special Interest Group, (1995).

13) PLX Technology: *PCI9060 Data Sheet VERSION1.2*, (1995).

14) Loïc Prylli and Bernard Tourancheau, *BIP: a new protocol designed for high performance networking on myrinet*, In Workshop PC-NOW, IPPS/SPDP98,

Orlando, USA, (1998).

15) Alessandro Rubini, Andy Oram: *Linux Device Drivers*, Chapter 13, O'Reilly & Associates, (1998).

16) Hiroshi Tezuka, Atsushi Hori, Yutaka Ishikawa, and Mitsuhisa Sato: *PM: An Operating System Coordinated High Performance Communication Library*, High-Performance Computing and Networking, volume 1225 of Lecture Notes in Computer Science, pages 708-717, Springer-Verlag, (1997).

17) http://www.usb.org/