グリッド環境における MPI_Scatter/Gather 通信アルゴリズムの最適化

千葉立寛[†,††] 遠藤敏夫[†] 松岡聡[†,†††]

ネットワークの状態に応じて最適なネットワークトポロジを構築してグリッド環境上での MPI 集団通信を高速化させるための様々な手法が提案されてきた．それらの手法では WAN のバンド幅は狭く集団通信を実行する上でボトルネックリンクとなるという前提が置かれていた．しかし近年のネットワーク技術の発展により，WAN のバンド幅は向上し，またサイト内のネットワークも高速化しており，従来の仮定では適応しなくなっているため，グリッドを構成する WAN と LAN のネットワーク帯域十分に扱えるように集団通信アルゴリズムを適応させる必要がある．本稿では，このようなネットワーク環境に適応させたマルチレーン Scatter/Gather 通信アルゴリズムを提案する．下位の通信レイヤに TCP/IP を用いた MPI 実装を想定し，エミュレートした複数サイトにまたがるグリッド環境において実験・評価を行い，性能を確認した．

Optimization of MPI_Scatter/Gather Algorithm for Grid Environment

Tatsuhiro Chiba,[†] Toshio Endo[†,††] and Satoshi Matsuoka[†,†††]

Many Collective algorithms have been proposed for grid environments, that enable us to construct optimized network topologies and to perform fast collective communications, but they are optimized under the condition that WAN is low and bottleneck bandwidth. However, recent WAN has become much wider and many nodes in LAN are connected with high-speed netoworks, so the previous assumption isn't suitable now. In this paper, we proposed multilane MPI_Scatter/Gather Algorithms to effectively utlize the available WAN and LAN bandwidth. We assumed MPI systems use TCP/IP in low-level communications, and experimentations on an emulated network environment show that proposed multilane collective algorithms achieve higher performance than traditional methods.

1. はじめに

近年，これまで単独のスーパーコンピュータやクラスタで行われてきた大規模な科学技術計算の実行環境として，より多くの計算資源を確保して高性能に処理可能となるグリッド環境の利用が高まってきている．これらのアプリケーションの多くが並列計算ライブラリである MPI を用いて記述されるのが一般的であり，GridMPI[1] や MPICH-G2[2] などの様々なグリッド向け MPI 実装が提案されている．

MPI アプリケーションの実行性能を大きく左右する要因の一つとして MPI 集団通信が挙げられる．グリッド環境では，通信遅延やサイト間を結ぶバンド幅の影響を十分に考慮した通信・トポロジ最適化を行われない場合，MPI アプリケーションの性能が著しく低下するため，これらを考慮した集団通信アルゴリズムが提案されてきたが，それらのアルゴリズムは，LAN に比べて高遅延・低バンド幅な WAN でグリッド環境が構成されていることを暗に想定していた．

しかしながら，近年のネットワーク環境の高性能化により WAN の広帯域化は著しく，TeraGrid や SuperSINET のような数 10Gbps 以上のバンド幅を利用可能な WAN も登場してきた．また，グリッドを構成する各クラスタでは，各計算ノードが Infiniband や 10GbE などの高速なネットワークで結合されていることも珍しくなくなってきている．近年のネットワーク環境と，以前に想定されていたネットワーク環境では大きく異なってきており，従来のアルゴリズムでは，その性能を十分に発揮することができない．

そこで我々は，上記のようなネットワーク環境を有効に利用する MPI 集団通信アルゴリズムの研究を行っている．[3] 本稿では集団通信の中で一対多の通信を実現する Scatter/Gather 通信についてのアルゴリズムを提案する．提案するアルゴリズムは，広帯域な WAN と高速な LAN で構成されるグリッド環境において，ユーザが利用できるバンド幅を最大限利用することが可能となるよう設計されている．ネットワークエミュ

† 東京工業大学
Tokyo Institute of Technology
†† 日本学術振興会特別研究員 DC
Research Fellow of the Japan Society for the Promotion of Science
††† 国立情報学研究所
National Institute of Infomatics

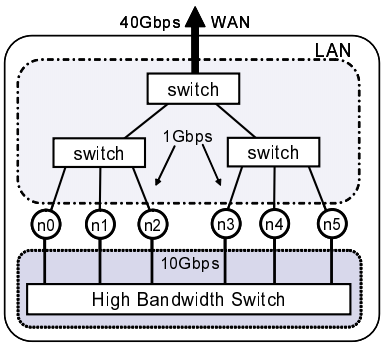

図 1 想定するネットワーク環境

レータを用いて構築した擬似的な 2 サイト環境にて提案アルゴリズムと従来のアルゴリズムとの性能比較を行った．

2. 関 連 研 究

グリッド環境上での集団通信アルゴリズムの最適化に関するこれまでの研究の多くでは，通信遅延が大きく低バンド幅で集団通信性能のボトルネックリンクとなる WAN 間通信の影響をいかに最小限にするかが考えられてきた．MagPIe[4] や MPICH-G2[2] では，WAN 間同士の通信に対しては代表ノード同時がまとめて通信しあい，また，LAN 内では環境に合わせた最良のツリートポロジを構築して，ボトルネックリンクで発生する性能低下を抑えるアルゴリズムが提案されている．

一方，近年の高遅延・広帯域な WAN 環境での実行に最適化した集団通信アルゴリズムもいくつか提案されている．GridMPI[1] では，高バンド幅な WAN を備えるグリッド環境に対して，複数のリンクで WAN 間通信を行う Bcast 通信と AlltoAll 通信アルゴリズムを提案している．また論文[3] では，パイプライン通信によって高速な転送が可能になる Bcast 通信用のマルチレーンツリーアルゴリズムを提案している．

今後ますます WAN のバンド幅性能が向上し，MPI アプリケーションで扱うメッセージサイズも増大することが予測されるため，WAN のバンド幅を十分に利用可能な集団通信アルゴリズムを考えることは，MPI アプリケーションの性能を向上させる上で重要な課題の 1 つである．

3. ネットワークモデル

3.1 ネットワーク環境

本稿で提案するアルゴリズムは，サイト間通信において十分なバンド幅を利用できるときに効率よく動作するよう設計されている．そのため，以下のようなネットワーク環境を前提として考える．

図 1 は想定するグリッド環境におけるサイト内のネットワークを表したものである．各ノードは LAN 用の高速なネットワーク (10GbE, InfiniBand, etc) で結合されている一方，サイト外との通信用に一般的なイーサネット環境も備えており，通信相手に応じてネットワークを切り替えて通信を行う．サイト間を結ぶネットワークは広帯域なネットワークで結合されており，サイト間をまたぐ通信において WAN 用にチューニングされた TCP/IP を用いた場合，そのバンド幅はノードの通信性能に依存し，図 1 においては理論的には最大で 1Gbps のバンド幅で通信が可能であるとする．

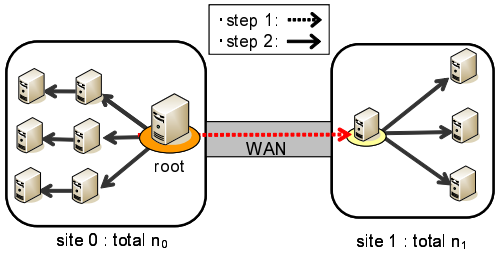

図 2 2 サイト環境での一般的なアルゴリズムによる scatter 通信

LAN 内のノード間通信のバンド幅を b_{LAN}，WAN をまたいだノード間の通信バンド幅を b_{WAN}，WAN で利用可能な合計バンド幅を B_{WAN} と記述する．このとき，$b_{LAN} > b_{WAN}$，$B_{WAN} > b_{WAN}$ の関係が成り立ち，WAN をまたぐ通信を行うペアが p 組存在するとき，$B_{WAN} \geq p \cdot b_{WAN}$ を満たすならば，WAN を共有するそれぞれの通信は，互いに干渉せず b_{WAN} の性能を維持して独立に通信が可能であるとする．

3.2 一般的なアルゴリズムと課題

Scatter/Gather 通信は，通信の向きが異なるだけで，通信のタイプとしては 1 対多通信に分類され，同じタイプの通信であると言える．以後スペースの都合上，Scatter 通信に対しての説明だけを行い，Gather 通信に対しては送受信の向きを逆にしたものととらえて頂きたい．以下では，従来手法による Scatter 通信について説明する．

サイト内での Scatter 通信

ノード A からノード B へメッセージ M を送信することを考える．ノード間のバンド幅を b，ノード間の通信遅延を α とすると，1 度の通信にかかるコストは，$\alpha + M/b$ としてモデル化することができる．

サイト内でのノード間の通信遅延は非常に小さく，一般的な GbEthernet を備えるクラスタ環境であれば，その値は数 100μs 単位である．このため，集団通信の全実行時間に対して相対的に通信遅延よりもメッセージ転送時間の占める割合が多くなるので，メッセージサイズの大きさによって通信方式を変えるのが一般的となっている．

メッセージサイズが小さい場合，1 通信あたりの転送コストが非常に小さいため，binary tree や binomial tree などのツリー構造を用い，通信遅延や送受信バッ

ファ確保を含むオーバーヘッドを最小にすることを目指す．

一方メッセージサイズが大きい場合，1 通信あたりの転送コストが大きいため，フラットなツリー構造を用いて *root* から直接全ノードに対して通信を行う．

複数サイトでの Scatter 通信

次に，複数サイトで Scatter 通信を行う場合を考える．地理的に分散している複数サイトの計算資源を用いる場合，WAN をまたいだ通信が必要となる．LAN に比べて大きな通信遅延が発生したり，細い帯域しか確保出来ないなど，ネットワーク性能が異なっているため，その違い考慮してサイト間のメッセージ転送量を最小とするよう各サイトの代表ノード間で 1 度だけサイト間通信を行い，サイト内では，上記で述べた手法を用いて Scatter 通信を行うのが一般的である．このようなポリシーに基づいたグリッド環境上での集団通信アルゴリズムは MagPIe などで提案されている．[4] このときの通信アルゴリズムの流れを図 2 に示す．このアルゴリズムでの集団通信コストはモデル式を用いると，

$$T = L + \frac{n_1 \cdot M}{b_{WAN}} + (n_{alt} - 1) \cdot (l + \frac{M}{b_{LAN}})$$
$$n_{alt} = max(n_0, n_1)$$

と表すことができる．

このようなアルゴリズムが最適であるのは，$B_{WAN} < b_{WAN}$ の関係が成立する場合，つまり WAN の合計バンド幅が集団通信全体のボトルネックとなる場合である．このとき，WAN をまたいだノード間のバンド幅性能はノードの通信性能 b_{WAN} に関係なく B_{WAN} に制限される．

次章で述べる提案アルゴリズムでは，広帯域な WAN を有効利用するためサイト間に複数の通信ペアを用意して WAN 全体の利用効率を上げて転送時間を削減している．上記の $B_{WAN} < b_{WAN}$ となる状況と $b_{LAN} = b_{WAN}$ となる状況では，提案アルゴリズムは上手く機能しないことが予想される．この要因として以下の 2 つが考えられる．

1 点目は，通信のコンテンションによって WAN の利用効率が改善しない点である．例えば，2 コネクションで WAN の通信を行うと仮定すると，1 コネクションあたりの WAN 間通信のバンド幅は $B_{WAN}/2$ となり，結局合計で B_{WAN} 以上にはならずに 1 コネクションでの総利用帯域と同じなため，WAN の利用効率改善には繋がらない．2 点目は，Scatter 通信で送信されるべきメッセージがそれぞれ異なっている点である．提案アルゴリズムでは複数のコネクションで WAN 間を通信するため，あらかじめ root ノードが他のノードにメッセージの転送を行う．$b_{LAN} = b_{WAN}$ であるとすると，root ノードが送信したいメッセージの総数 $n_0 + n_1 - 1$ が変わらないため，他のノードに転送すればするほどネットワークを流れるメッセージの総量は増加し，その結果 Scatter 通信の実行時間は増え，WAN の利用効率を上げたとしても通信時間の改善には繋がらない．

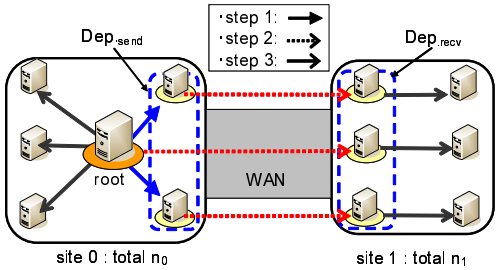

図 3 2 サイト環境での提案アルゴリズムによる scatter 通信

しかしながら，図 1 のようなネットワーク環境の場合，状況は異なり集団通信の性能向上が期待できる．1 点目に関しては，$B_{WAN} > b_{WAN}$ となるので，あるコネクション数 p に対して，$B_{WAN} \geq p \cdot b_{WAN}$ を満たす限り，WAN を利用するトータルバンド幅は増加するので，集団通信時の WAN の利用効率は改善される．2 点目に関しては，$b_{LAN} > b_{WAN}$ となるので，root ノードが同一サイト内の他のノードにメッセージ転送を行って，ネットワークを流れる総メッセージサイズが増加した場合でも，転送に要するコスト増加はそれほど多くなく，またその後に WAN の帯域を十分に活用可能なため，集団通信全体の全実行時間に対しては軽微なものとなる．

4. 提 案 手 法

4.1 マルチレーンアルゴリズム

図 3 に提案するマルチレーン Scatter/Gather アルゴリズムの概要を示す．step.1(転送準備フェーズ)，step.2(サイト間転送フェーズ)，step.3(サイト内転送フェーズ) の 3 ステップで Scatter/Gather を行う．Dep_{send} は root が送信する代わりに別のサイトにメッセージを転送する役割のノード群である．それに対して Dep_{recv} は，Dep_{send} から転送されたメッセージを受信し，その後そのサイト内で対応するノードにメッセージを転送する役割を持ったノード群である．

step.1 root は Dep_{send} に対してそれらが転送を担当するメッセージを LAN 内で転送する．

step.2 Dep_{send} は root からメッセージを受信次第，また root も Dep_{send} への転送が終わり次第，対応する Dep_{recv} に対してサイト間転送を実行する．

step.3 Dep_{recv} は担当する子ノードに対して転送を行う．また root も Dep_{send} 以外のサイト内の子ノードへの転送を行う．

4.2 通信コストモデル

前述の提案アルゴリズムによる Scatter/Gather 通信を実現するため，WAN 間転送を行う Dep_{send}，

表 1 マルチレーンアルゴリズムとシンプルアルゴリズムの通信コストモデルの比較

	シンプルアルゴリズム	マルチレーンアルゴリズム
集団通信の全実行時間：T_{total}	$T_{WAN} + T_{LAN}$	$T_{prepare} + T_{WAN} + T_{LAN}$
Dep_{send} への転送時間：$T_{prepare}$	0	$\alpha + (n_1 + P_{opt} - 1)M/b_{LAN}$
サイト間転送時間：T_{WAN}	$L + n_1 M/b_{WAN}(1)$	$L + n_1 M/P_{opt} b_{WAN}(P_{opt})$
サイト内転送時間：T_{LAN}	$\alpha + (n_1 - 1)M/b_{LAN}$	$\alpha + n_1 M/P_{opt} b_{LAN}$

Dep_{recv} に該当するノードの組を決定する必要がある．この組を求めるため，提案アルゴリズムによる通信をモデル化し，通信コストを定式化する．WAN 間の通信を行うノードの組数を P とすると，P の取りうる範囲は，$1 \leq P \leq min(n_0, n_1)$ となる．この範囲内で P を変化させるとき，後述するコストモデルによって集団通信の実行時間を見積もり，その通信コストが最小とするような P を探索することで，Dep_{send} と Dep_{recv} の組を決定し通信トポロジの最適化を行う．

WAN の遅延を L，メッセージサイズを M，P 組で同時に WAN 間通信するときのバンド幅を $b_{WAN}(P)$，LAN 内のバンド幅を b_{LAN}，P のとりうる値の上限を $P_{sup} = min(n_0, n_1)$，メッセージ転送以外のオーバーヘッド α とするとき，提案アルゴリズムを用いたときの全実行時間を表す通信コスト $T(P)$ は，前述したアルゴリズムの 3 ステップにかかる時間の合計であり，以下のように定式化することができる．

$$T(P) = L + \frac{X(P) \cdot M}{b(P)} + \frac{Y(P) \cdot M}{b_{LAN}} + \alpha$$

$X(P), Y(P)$ は転送するメッセージの個数である．また，$X(P), Y(P)$ の個数を n_0, n_1 の大小によって場合分けをすることで，さらに 2 通りの通信コストモデルとして以下のように定式化する．

$n_0 \geq n_1$ のとき

$$X(P) = n_1/P + \begin{cases} 0 & n_1 \% P = 0 \\ 1 & n_1 \% P \neq 0 \end{cases}$$
$$Y(P) = n_0 + n_1 - 1 - X(P)$$

$n_0 < n_1$ のとき

$$X(P) = n_1/P + \begin{cases} 0 & n_1 \% P = 0 \\ 1 & n_1 \% P \neq 0 \end{cases}$$
$$Y(P) = (n_1 - 1) + (P - 1)$$
$$= n_1 + P - 2$$

このコスト式を用いることで，WAN 間に何コネクションを張ったときに通信コストの最小化が出来るかを判断する．

4.3 トポロジ構築

図 3 のようなトポロジを，上述のコスト式から導かれた P の値：P_{opt} を用いて下記の手順で構築し，個々の通信の順序をスケジューリングしていく．

(1). P リンク張ったときの WAN 間の 1 リンクあたりのバンド幅：$b_{WAN}(P)$ と，サイト内の LAN バンド幅：b_{LAN} を事前に測定する．

(2). 計測した値を用いて性能モデルの評価式に値を代入し最小のコストで通信可能となる P_{opt} を求める．

(3). P_{opt} 台のノードをサイト 0，サイト 1 から選び，それぞれ Dep_{send}, Dep_{recv} として選別する．*root* はこの情報をもとに，Dep_{send}，*root* 自身で転送する Dep_{recv}，サイト内の残りのノードという順で転送するようスケジューリングする．

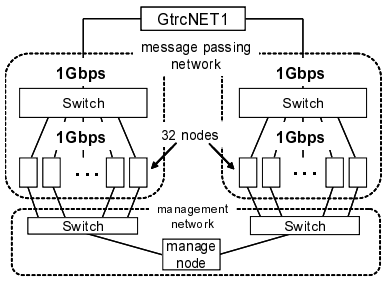

図 6 実験環境：エミュレートした 2 サイトグリッド環境

4.4 通信コストの比較

提案するマルチレーン Scatter アルゴリズムとシンプルな Scatter アルゴリズムでの実行時間を性能モデルを用いて比較する．ステップごとの通信コストで全実行時間を表した場合，表 1 のようにまとめることができる．

$T_{prepare}$ は，シンプルアルゴリズムでは Dep_{send} への転送が発生しないのでそのコストは 0 であるのに対し，マルチレーンアルゴリズムでは表の値分だけシンプルアルゴリズムよりコストがかかる．T_{WAN} では，シンプルアルゴリズムでは 1 本のリンクでサイト間転送を行うのに対し，マルチレーンアルゴリズムでは P_{opt} 分のコネクションで通信を行うので表の値分だけのコストで済む．WAN の 1 コネクションあたりのバンド幅 $b_{WAN}(P)$ とメッセージサイズ M がどの程度の性能，サイズなのかでそれぞれのコストが決定される．

5. 性能評価

5.1 実験環境

提案アルゴリズムの有効性を示すため，2 サイトからなるエミュレートした複数サイト環境において，それぞれのアルゴリズムを用いた場合の Scatter/Gather 通信の性能を比較した．図 6 は評価に用いた 2 サイトエミュレート環境である．WAN をエミュレートするため GtrcNET-1[5] を用いている．各サイト 32 ノード

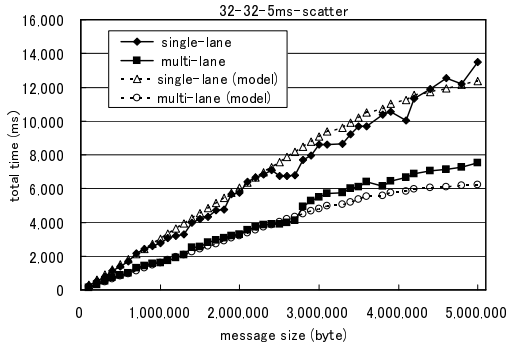

図 4 L=5ms, scatter アルゴリズムの性能

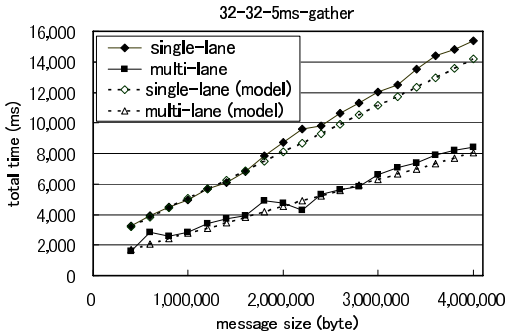

図 5 L=5ms, gather アルゴリズムの性能

で構成しており，1 ノードにつき 1 クライアントプロセスが起動している．実際にメッセージパッシングを行うクライアントノードとは別にクライアント管理用マネージャノードを用意する．マネージャノードでは，クライアントノードに対して通信トポロジや通信順序を通知し，クライアントでの通信時間完了時刻を管理する．なお，集団通信用ネットワークと管理用ネットワークに別々のネットワークを用い，管理用の通信が集団通信に影響を及ぼさないようにする．各ノードのスペックを表 2 に示す．

提案アルゴリズムでは図 1 に示すような環境での実行を想定しているが，1Gbps の帯域幅を持つ WAN 環境しか用意できなかったため，1 リンクあたりの WAN 間通信でのバンド幅を制限して，相対的に $B_{WAN} > b_{WAN}$，$b_{LAN} > b_{WAN}$ となるようにし，図 1 と同様の状態を再現している．ソケットバッファサイズの値は linux のデフォルトの設定を用いており，WAN の RTT を大きくすることで意図的に帯域遅延積を最適化しないようにした．その結果，WAN をまたいだ 1 リンクあたりの通信バンド幅性能は抑制され，RTT の値を調整することで利用可能なバンド幅の制限を実現して実験を行った．

表 2 各サイトを構成するノードのスペック

OS	Debian/Linux(kernel 2.6.16)
CPU	Opteron242(1.6GHz) * 2
Memory	2GB DDR(PC2100)
NIC	1000Base-T * 2 系統
Network	WAN:1Gbps(GtrcNET-1), LAN:1Gbps

5.2 実験結果

Scatter 通信

図 4 は，WAN の遅延を 5ms に設定してメッセージサイズを変化させ，各アルゴリズムを用いたときに計測された実行時間，ならびにモデル式から導出された予測実行時間を示している．このとき，WAN 間通信の 1 リンクあたりで得られるスループットは 250Mbps 程度で，提案アルゴリズムでは，最大で WAN の帯域幅限界である 1Gbps で転送を行っていた．500KB 程度のメッセージサイズから徐々に性能差が現れ，十分に大きなメッセージサイズでは，提案アルゴリズムによって 1.5 から 2.0 倍程度の性能向上が確認された．

また，図 9 は，メッセージサイズを固定して WAN 間通信リンク数を変化させたときの実行時間と予測時間を示したものである．図 9 からも分かる通り，提案アルゴリズムによって通信時間を最小化する最適な WAN 間リンク数はモデル式上では 32 本であると求められているので，図 4 のマルチレーンは，WAN 間での通信ペア数が 32 となるようにトポロジを構築して集団通信を行っていた．しかしながら，実際には，16-25 リンクあたりで WAN 間通信を行うときが最も性能が良く，それ以上のリンク数になると，マルチリンク化してもトータルの実行時間が増えてしまう結果となった．

また，図 7 では各アルゴリズムでの WAN の帯域利用率を比較したものである．アルゴリズムごとの特徴を示すため，遅延を 20ms に設定して 1 リンクあたりのバンド幅を低下させて比較している．single-lane では，1 リンクのため 1Gbps の WAN 帯域幅に対して 125Mbps 程度のバンド幅だけで転送を行っている．一方，multi-lane では，Dep_{send} が転送するメッセージを受け取り次第 WAN 間通信を開始していくが，複数のリンクが順次に WAN 間通信を行い WAN の帯域を共有するため，WAN の帯域幅の利用率は single-lane のときよりも多く，250Mbps 程度の性能で通信を行っていることが分かる．

Gather 通信

図 5 では，WAN の遅延を 5ms に設定してメッセージサイズを変化させたときのそれぞれのアルゴリズムでの性能を比較している．メッセージサイズの大小に関わらず，常にマルチレーンアルゴリズムを用いることでが 2 倍程度性能が向上していることが確認される．

また，モデル式による予測実行時間と実測時間がほぼ同程度に推移していることが図 10 から分かる．しかしながら，モデル式では 32 リンクのときに最小の

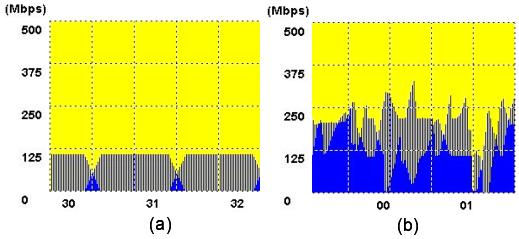

図 7 L=20ms, M=4MB, scatter 通信の WAN のバンド幅利用率 a)single-lane b)multi-lane

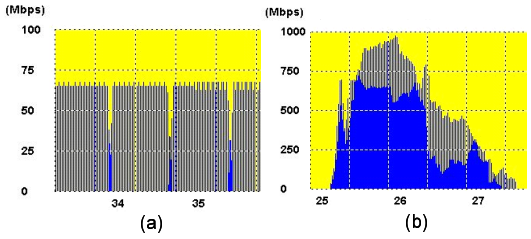

図 8 L=20ms, M=4MB, gather 通信の WAN のバンド幅利用率 a)single-lane b)multi-lane

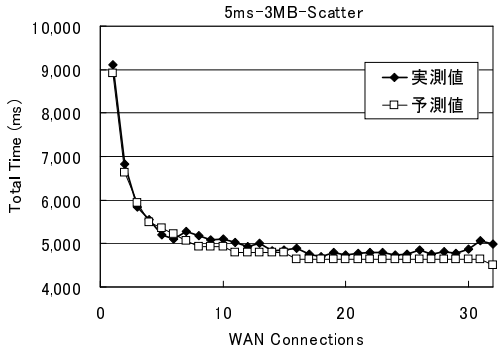

図 9 WAN 間通信リンク数による Scatter 通信の性能

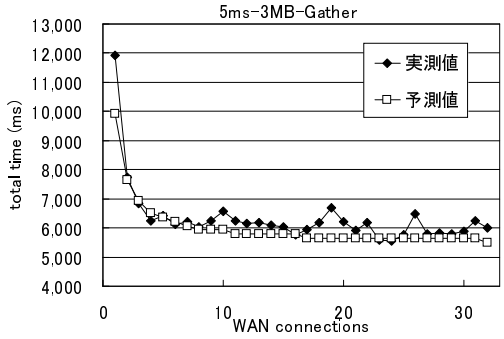

図 10 WAN 間通信リンク数による Gather 通信の性能

実行時間で通信可能であるとしているが，実際には 10 リンクを超えたあたりから性能に変化は見られず，ほぼ同程度の性能を示している．

また図 8 では，遅延を 20ms に設定したときの各アルゴリズムでの WAN の帯域利用率を比較している．1 リンクで通信を行うときは，Scatter 通信のときと同様，限られたバンド幅しか WAN の帯域を利用しておらず，65-70Mbps 程度である．複数リンクで通信を行う提案アルゴリズムでは，Scatter 通信のときとは異なり，ほぼ同時に Dep_{send} が WAN 間通信を開始するので，WAN の帯域を上限である 1Gbps 程度まで使い切り，従来手法よりも短い時間で WAN 間転送を完了していることが分かる．

6. おわりに

高速な LAN 環境を備えたサイト同士を高遅延で高バンド幅な WAN で接続したグリッド環境に適応させたマルチレーン Scatter/Gather 通信アルゴリズムを提案した．利用可能な複数のネットワークを有効に利用して，複数のリンクで WAN 間の転送を行えるマルチレーンツリートポロジを構築することで，より高速に Scatter/Gather 通信が実行可能となる提案アルゴリズムの有効性を確認した．今後の課題としては，ネットワークの輻輳が発生したときの各リンクごとの性能低下を考慮したより最適なリンク数を導出するモデル式の構築や，大規模な実環境で実アプリケーションを用いた評価を行うことを考えている．

謝辞 本研究の一部は，文部科学省科学研究費補助金 (特別研究員奨励費 20・8911, 若手研究 (B) 17700050, 特定領域研究 18049028) の支援によって行われた．

参 考 文 献

1) M.Matsuda, et al.: Efficient MPI Collective Operations for Clusters in Long-and-Fast Networks, *IEEE International Conference on Cluster Computing (cluster 2006)* (2006).
2) Karonis, N.T.: MPICH-G2: A Grid-Enabled Implementation of the Message Passing Interface, *Journal of Parallel and Distributed Computing (JPDC)*, Vol.63, No.5 (2003).
3) Chiba, T. et al.: High-Performance MPI Broadcast Algorithm for Grid Environments Utilizing Multi-lane NICs, *7th IEEE International Symposium on Cluster Computing and the Grid (CCGrid)* (2007).
4) Kielmann, T. et al.: MagPIe: MPI's collective communication operations for clustered wide area systems, *ACM SIGPLAN Notices*, Vol.34, No.8, pp.131–140 (1999).
5) Kodama, Y. et al.: GNET-1: Gigabit Ethernet Network Testbed, *IEEE International Conference on Cluster Computing*, pp.185–192 (2004).