Diffserv AF 環境において 動的な契約帯域制御を行う大規模データ転送方式

野呂正明[†1] 馬場健一[†2] 下條真司[†2]

広域ネットワークを介した大量のデータ転送に，性能保証を必要とするアプリケーションのための，データ転送技術の研究を行っている．この種のアプリケーションでは，グリッドにおけるパイプライン処理や計算機の遠隔バックアップ等，処理の効率的な実行や，終了時間の保証のために，データ転送の所要時間見積りと，見積り時間内にデータ転送が終了するよう性能保証を行うことが有効である．しかし，性能を保証するために，広域網との間で帯域を契約する場合，契約帯域に応じた経費が必要になることが多い．性能保証に必要な契約すべき帯域を削減することができれば，アプリケーションを実行するシステムの運用コストを低減させることが可能となる．本研究では，各データ転送がスループット保証に必要とする契約帯域を削減する方式を提案する．提案方式では，Diffserv の AF サービス，複数の TCP コネクションと TCP に対してウィンドウサイズの制御を行うデータ転送手法および契約帯域を性能保証に必要な最小値となるよう定期的に制御する手法を組み合わせる．さらに，シミュレーションによる評価を行い，同一負荷時に必要となる契約帯域を減少させることができること，データ転送が指定された性能を確保できることを確認した．

Volume Data Transfer Method with Dynamic Reservation Bandwidth Control over Diffserv AF Networks

Masaaki Noro,[†1] Ken-ichi Baba[†2] and Shinji Shimojo[†2]

We are working on large scale data transfer technology over wide area network. Some kinds of application need guarantee of throughput over wide area networks. Pipeline processing of large scale Grid computing and remote data backup over wide area network are typical example. Pipeline processing needs dispatches jobs to CPUs which are distributed to many places, and transfers data between CPUs over wide area network. Data transfer of remote backups transfers large amount of data during night. Data transfer of these applications should be done in time to follow the application schedule. Reservation of bandwidth is effective to transfer data in time. Usually, cost of service level agreement increases in proportional to reservation bandwidth. The method should reduce cost of these applications that decreases necessary bandwidth for guaranteeing throughput of data transfer. We propose the method that decreases reservation bandwidth in Diffserv AF network environments. Our method consists of dynamic QoS control method and data transfer protocol using two kinds of TCP connections. In this paper, we denote proposal method and its evaluation result.

1. はじめに

近年，ネットワークの高速化，低価格化にともない，従来 LAN 上で行われてきたサービスやアプリケーションが，広域ネットワーク上で実施可能となっている．本研究では，それらのサービスやアプリケーションのうち，広域ネットワークの品質制御技術を利用して，大量データを転送するアプリケーションを対象とする．このようなアプリケーションには，グリッドおけるパイプライン処理や大規模なリモートバックアップ等があげられる．

グリッドにおける大規模なパイプライン処理（**図 1**）では，センサ等から定期的に出力されるデータやストレージに蓄えられたデータを，複数拠点の計算機で処理し，結果をユーザに提示する．この処理では，各拠点の処理，および，拠点間のデータ転送に必要な時間を事前に見積もり，パイプラインが途切れることなくデータが供給されることが望ましい．これには，データ転送を事前の見積りに基づいて実行するため，デー

†1 独立行政法人情報通信研究機構
National Institute of Information and Communication Technology
†2 大阪大学サイバーメディアセンター
Cybermedia Center, Osaka University

タ転送の性能保証が有効である．

IP ネットワークにおいて，広域ネットワークを経由するデータ転送に対して，性能保証を行うには，広域網の運用者との SLA（Service Level Agreement）が必要となる．通信速度を保証する SLA を，IP レベルで実現する技術として Diffserv [1] がある．Diffserv は多くのルータやホスト OS に実装されており，さまざまな環境で容易に適用できる．

本研究の対象アプリケーションは，個々のデータ転送の終了時間を保証することが求められる．また，転送するデータはサイズが既知なファイルであり，ある帯域を契約した場合の実効スループットの最悪値が計算可能であれば，データ転送ごとに帯域を契約することで，終了時間の保証が可能となる．また，ファイル転送はストリーミングと異なり，一定のレートで転送する必要はない．このような目的には Diffserv における AF（Assured Forwarding）[2] が適している．さらに，Diffserv AF の環境では，輻輳していない場合，契約した帯域以上のレートでのデータ転送が可能である．この場合，各データ転送の終了時間を保証するために必要な契約帯域は時間の経過とともに減少する．

Diffserv AF を用いて SLA を締結する場合，必要なコストは契約する帯域に比例する場合が多いと想定される．そのため，各データ転送が性能を確保するために必要とする契約帯域を削減できれば，システム全体の SLA 対象となる契約帯域の減少によるコスト低減や，同一 SLA 環境での処理量の増加といったメリットが期待できる．

また，現在データ転送には TCP [3)–6)] が多く用いられるが，このプロトコルはネットワークが極端に輻輳している環境において，契約している帯域と実際に得られるスループットの差が大きくなる．そのため，契約帯域内でより多くの転送性能を得ることが重要である．逆にいうと，必要な性能を得るために契約する帯域をできるだけ小さくおさえることのできる転送プロトコルを用いることにより，ネットワーク全体の帯域を有効に活用することが可能となる．

そこで本研究では，Diffserv AF 環境において，データ転送の終了時間を保証するために必要な契約帯域を動的に削減することにより，ネットワークの利用効率を高め，システム全体としてより多くのデータ転送を可能にする大規模データ転送方式を提案する．具体的には，2 種類のフローを組み合わせたデータ転送手法に，契約帯域を動的に制御する手法を組み合わせて，大規模データを転送する．従来手法と提案手法の基本性能，および，実アプリケーションを想定した場合の性能を，シミュレーションにより評価する．

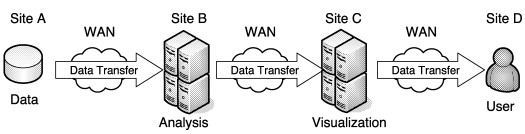

図 1 グリッドコンピューティングにおけるパイプライン処理
Fig. 1 Pipeline processing in Grid computing.

2. 関連技術と本研究のアプローチ

ここでは，従来のデータ転送方式の問題点，近年の新しいデータ転送手法，および本研究におけるアプローチについて述べる．

2.1 Diffserv AF

Diffserv では，1 つのポリシで品質制御を行うネットワークの範囲を Diffserv ドメインと呼ぶ．AF では，ドメイン境界上の装置（エッジデバイス）で契約内容に従い，トラフィックの流量測定を行い，ドメイン内部に流れ込むパケットの DSCP（Diffserv Code Point）フィールド [7] を書き換える．ドメイン内部のルータ（コアデバイス）では，定義された動作（PHB; Per Hop Behavier）に従い，パケットの廃棄や流量調整といった品質制御を行う（**図 2**）．

Diffserv AF は最低帯域保証サービスとも呼ばれ，ユーザとネットワークの間で帯域を契約する．ネットワークは，輻輳時は契約内の流量となるトラフィックを保護し，非輻輳時は契約以上のトラフィックも転送する．これを実現するために，ルータにおけるキュー管理に RIO（RED with IN and OUT）[8] を用いることが一般的である．RIO では契約内のトラフィックを IN，契約外のトラフィックを OUT と呼ぶ．Diffserv エッジでは，AF の Green と Red を，それぞれ RIO の IN と OUT に対応させる．RIO では IN および OUT のパケットの廃棄確率に差を設け，輻輳時に OUT のパケットから優先的に廃棄することで，IN となるパケットを保護する（**図 3**）．これにより，契約内の性能が保証されるだけでなく，帯域に余裕がある場合は契

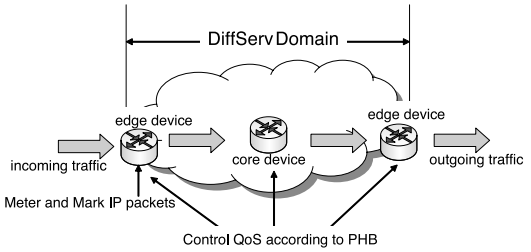

図 2 Diffserv のアーキテクチャ
Fig. 2 Architecture of Diffserv.

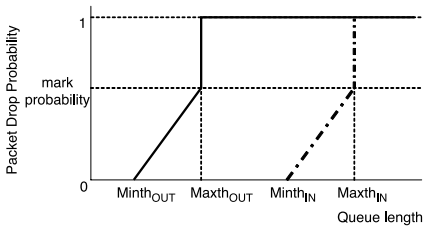

図 3 RIO におけるパケット廃棄確率
Fig. 3 Packet drop probability in RIO.

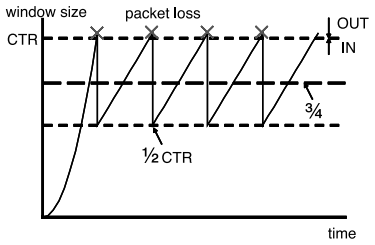

図 4 TCP Reno のウィンドウサイズの変化例
Fig. 4 Window size transition example of TCP Reno.

約を超える流量のトラフィックも配送される．

2.2 従来のデータ転送方式

現在，一般的にファイル転送には TCP Reno [3),4)] もしくは NewReno [5),6)] が用いられることが多い．これらは，受信側から Ack パケットが到着するまでの間に送信するデータ量を，輻輳ウィンドウのサイズで管理し，そのサイズを増減させることでデータの転送レートを調整する．転送開始時（スロースタートフェーズ）は，指数的に輻輳ウィンドウのサイズ（以下ウィンドウサイズと省略）を増加させ，ウィンドウサイズが一定値に達すると，輻輳回避フェーズに移行する．輻輳回避フェーズでは，パケットロス等を検知するまで，ウィンドウサイズを線形に増加させる．もし，パケットロスを検知した場合，ウィンドウサイズは直前の半分とし，再度線形増加する動作を繰り返す．

ネットワークが非常に輻輳しており，Diffserv のエッジにおいて OUT（契約外）とマークされたパケットのほとんどが，ボトルネックにおいて廃棄される状況を想定する．この場合，TCP Reno や NewReno は**図 4** のような挙動となり，実際に得られるスループットは，ネットワークとの契約帯域（CTR; Committed Target Rate）の 3/4 程度となる．

2.3 データ転送における関連研究

近年，TCP Reno や NewReno の弱点を克服するためのさまざまな新しいバージョンの TCP [9),10)] が提案されている．しかし，これらの新しい TCP の実装が利用できるオペレーティングシステムは限定される．また，UDP を利用した，新しいデータ転送プロトコルも研究されている [11)]．これらのプロトコルはオペレーティングシステムの改造は不要である．しかし，複数の組織をまたいでデータ転送を行う場合は，Firewall により利用できない場合が珍しくない．そのため，複数組織間でデータ転送を行うアプリケーションに適用するのは困難である．

また，TCP NewReno を利用した大量データ転送方式の研究も行われている．パラレル TCP [12)] は，複数の TCP コネクションで 1 つのファイルを転送し，長距離や非常に広帯域なネットワークで大量データを効率良く転送する手法である．そのため，Diffserv AF 環境ではネットワークが輻輳していない場合の，利用可能帯域の有効活用は可能である．しかし，契約外のパケットがすべて廃棄されるような極端な輻輳状況では，TCP の輻輳回避フェーズの特性のため，契約帯域とスループットの差を小さくすることは難しい．

一方，TCP の輻輳ウィンドウサイズを制御する手法 [13)] は，TCP の送信レートを契約帯域を超えないよう設定でき，極端な輻輳状態でもスループットが契約帯域を大幅に下回ることは防止できる．しかし，この手法はネットワークが輻輳していない場合のスループットは設定値以上とならず，利用可能帯域の有効活用ができない．

2.4 本研究のアプローチ

以上のような理由から，本研究では幅広い環境に適用可能な TCP NewReno を用い，パラレル TCP と TCP のウィンドウサイズ制御手法の長所を取り入れ，さらに，個々のデータ転送がネットワークと契約する帯域を動的に制御する手法を併用する．これにより，ネットワークの非輻輳時には，利用可能帯域を有効活用しつつ，定期的に契約帯域を見直すことにより，データ転送の終了時間保証に必要な契約帯域を削減する．また，輻輳時における契約帯域とスループットの差を削減することにより，輻輳時においても，ネットワークとの契約帯域の必要量を減少させる．

3. 提 案 方 式

本研究における提案方式は，2 種類のフローを組み合わせたデータ転送手法 [14)] と動的契約帯域制御手法 [15)–17)] を組み合わせる．ここでは，個々の手法および組み合わせた場合の動作について述べる．

3.1 2 種類のフローを組み合わせたデータ転送手法

2 種類のフローを組み合わせたデータ転送手法は輻輳時における契約帯域とスループットの差の削減と，非輻輳時の利用可能帯域の有効利用を実現する．本手

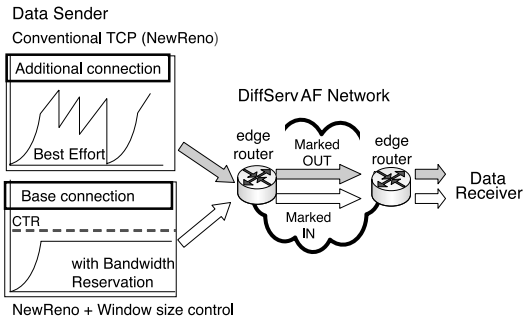

図 5 2 種類のフローを組み合わせたデータ転送手法
Fig. 5 Data transfer using two kinds of TCPs.

```
rtt = ユーザ指定値読み取り ("rtt");
ctr = ユーザ指定値読み取り ("ctr");
cwnd_max = ctr × rtt / packetsize;
ssth = 2;
cwnd = cwnd_max - 1;
TCP ソケットに対して各パラメータ設定;
データ転送開始;
```

図 6 基本コネクションのウィンドウサイズ制御
Fig. 6 Congestion window size control algorithm.

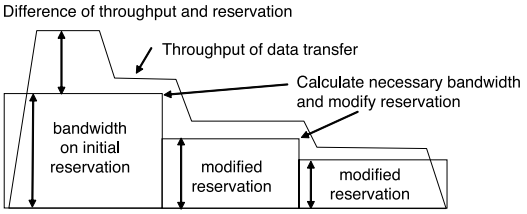

図 7 動的契約帯域制御手法における契約帯域制御例
Fig. 7 An example of reservation bandwidth in dynamic reservation bandwidth control method.

```
転送残時間 = 終了期限 - 現在時刻;
契約帯域 = データサイズ / 転送残時間
        / 契約帯域スループット比;
if (帯域契約 (契約帯域)){
    データ転送 (1 ブロック分のデータ)
    while(残りデータ量 > 0){
        転送残時間 = 終了期限 - 現在時刻;
        新契約帯域=残りデータ量/転送残時間
            / 契約帯域スループット比;
        if (新契約帯域 < 契約帯域)
            契約変更 (新契約帯域);
        データ転送 (1 ブロック分のデータ);}
} else {呼損の処理;}
```

図 8 動的契約帯域制御方式アルゴリズム
Fig. 8 Algorithm of dynamic reservation bandwidth control method.

法は，文献 12) と同様に，1 つのデータ転送に複数の TCP コネクションを利用するが，複数のコネクションを基本コネクション（Basic connection）と追加コネクション（Additional connection）に分類する．

本手法（**図 5**）では，基本コネクションに対して帯域を契約し，このコネクションのパケットが IN となるよう TCP を制御する．さらに，追加コネクションを帯域契約なしに利用することで，非輻輳時に利用可能帯域を有効活用する．また，1 つのデータ転送を 2 つのフローで転送するため，専用のアプリケーション，またはミドルウェアを想定している．

基本コネクションは契約帯域（CTR）内で，できるだけ大きなスループットを得る必要があるため，本研究では，一例として文献 13) と同様に，TCP の輻輳ウィンドウのサイズに上限を設定することで，レートが契約帯域以下となるよう制御する（**図 6**）．

これにより，本コネクションの発生するすべてのパケットが Diffserv のエッジにおいて IN とマークされ，ネットワークが非常に輻輳している場合でも，契約帯域に近いスループットを実現する．さらに，TCP のウィンドウサイズの初期値および，スロースタートフェーズと輻輳回避フェーズを切り替える閾値を設定可能な環境では，ウィンドウサイズの初期値を大きくし，閾値を小さくとることで，スループットの増加に必要な時間を短縮する．さらに，TCP のスループットは経路の RTT に反比例するため，基本コネクションのウィンドウサイズの上限値は RTT に基づいて決定する必要がある．これにはさまざまな手法が可能であるが，本研究では，実装が容易となるよう，転送開始時にウィンドウサイズの上限値を固定している．

一方，追加コネクションは帯域契約を実行せず，すべて契約外，すなわち OUT とマークしてデータ転送を実施する．これにより，ネットワークが非輻輳状態における帯域の有効利用を実現する．

3.2 動的契約帯域制御手法

Diffserv AF におけるデータ転送では，契約帯域以上のスループットを得る可能性がある．本研究の対象アプリケーションで転送するデータはファイルであるため，データ転送開始時に定めた終了時間に比べ早期に終了する．そこで，本手法では，転送時間を短縮する代わりに，転送すべき残りのデータ量と終了時間から定期的に契約帯域を見直し，契約帯域を削減する．つまり，**図 7** に示すように実際のスループットが帯域契約時に期待されるスループットの最小値より大きい場合に契約を変更する．これにより，ネットワークの効率利用を図る．

図 8 はこの制御手法の概要である．ここでは，目標

スループットもしくは，データ転送が終了すべき時間（終了期限）をユーザに指定させる．目標スループットが指定された場合は終了期限を計算しておき，一定量のデータ（ブロック）の転送ごとに，終了期限までの残り時間と残りデータ転送量から，契約帯域の削減が可能かどうか判定する．この際，現在の契約帯域より新たな契約すべき帯域が小さい場合のみ契約を変更する．このブロックサイズが小さいほど見直し回数が増えるが，RTT と想定される最大のスループットの積より小さいブロックサイズを用いた場合，データ転送そのものの効率が低下する．また，実際の環境では，契約見直しの計算時間や契約変更に必要な時間，ルータの負荷等があるため，あまりに小さいデータサイズを用いることは危険である．

また，必要な契約帯域を計算する際に，契約帯域と得られるスループットの最小値の比（図 8 における契約帯域スループット比）が必要となるが，この値は，利用するプロトコルやネットワーク環境によって変化する．たとえば，1 つの TCP NewReno を用いてデータ転送を行う場合，契約帯域に対する実行スループットの最小値の比は 0.75 となる．また，2 種類のフローを組み合わせたデータ転送手法では，経路の RTT によって変動する．よって，契約帯域の動的制御手法では，環境に応じて契約帯域と得られるスループットの比を決定し，定数としてユーザが指定する．

```
基本コネクションデータ転送手順:
転送残時間 = 終了期限 - 現在時刻;
契約帯域 = データサイズ / 転送残時間
        / 契約帯域スループット比;
if (!帯域契約 (契約帯域)) 呼損処理;
データ転送 (1 ブロック分のデータ)
while(残りデータ量 > 0){
    転送残時間 = 終了期限 - 現在時刻;
    新契約帯域 = 残りデータ量 / 転送残時間
          / 契約帯域スループット比;
    if (新契約帯域 < 契約帯域)
        契約変更 (新契約帯域);
    データ転送 (1 ブロック分のデータ);}
if (追加コネクション実行中フラグ == true ){
    転送残時間 = 終了期限 - 現在時刻
    新契約帯域 = 最終ブロックのデータ量
              / 転送残時間
              / 契約帯域スループット比;
    契約変更 (新契約帯域);
    実行中の追加コネクションを終了;
    データ転送 (最終ブロック);}
追加コネクションデータ転送手順:
追加コネクション実行中フラグ = true;
データ転送 (1 ブロック分のデータ);
while(残りデータ量 > 0){
    データ転送 (1 ブロック分のデータ);}
追加コネクション実行中フラグ = false;
```

図 9 提案方式の動作
Fig. 9 Algorithm of proposal method.

3.3 2 つの手法の同時利用

一般的にネットワークが非常に輻輳していた場合，OUT とマークされたパケットの廃棄率が極端に高くなり，追加コネクションを用いたデータ転送は非常に低速になるか，最悪の場合終了しない可能性がある．そのため，必要な契約帯域の計算は基本コネクションのみでデータ転送を行う場合を想定して実行する．

図 9 は処理手順の概要である．この手順では，基本コネクションが規定の 1 ブロック分のデータを終了するたびに，基本コネクションのみで残りデータすべてを転送する場合に必要な契約帯域を計算し，契約変更を行う．さらに，追加コネクションのスループットが期待できない場合を想定し，最終ブロックのデータを転送する際には，実行中の追加コネクションを終了させたうえで，基本コネクションで再送を行う．

4. シミュレーションによる評価

提案方式の性能を評価するため，NS-2（Network Simulator 2）[18] を利用した．**図 10** は本評価におけるネットワークモデルである．経路の RTT を 20 ms とし，ネットワークと各データ転送を行う端末間で帯域を契約する．また契約可能な帯域の最大値はボトルネックとなるリンクの帯域の半分とし，バックグラウンドトラフィックを流すことにより，輻輳状況を変化させる．またルータにおけるキュー管理は RIO を適用した．他のパラメータは各評価において，適宜決定しており別途説明する．

はじめに 2 種類のフローを組み合わせたデータ転送手法について基本的な特性を評価する．その後，実際のアプリケーションを想定したシナリオで，データ転送が目標時間内に終了していることを確認し，提案方式の契約帯域削減効果を評価する．最後に，契約帯域

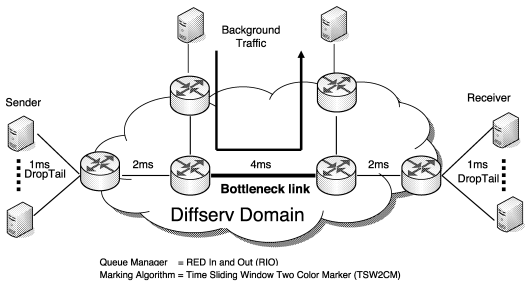

図 10 シミュレーションにおけるネットワークモデル
Fig. 10 Network model of the simulation.

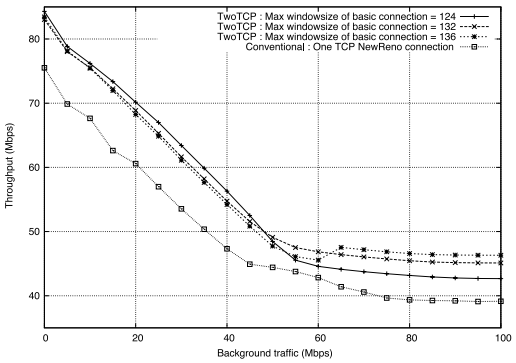

図 11 輻輳に対するスループットの変化
Fig. 11 Transition of throughput in congested network.

が削減されることにより，同一の契約帯域を用いた場合のシステム全体のスループット（単位時間あたりのデータ転送量）が向上するかを評価する．

4.1 2 種類のフローを組み合わせた手法

はじめに，2 種類のフローを組み合わせたデータ転送手法のスループットを検証する．モデルにおける各リンクの速度は 100 Mbps とし，1 つのデータ転送で 50 Mbps の帯域契約をしたうえで，長時間のデータ転送を行い，定常状態におけるデータを収集した．なお，バックグラウンドトラフィックは，一定レート（CBR; Constant Bit Rate）でパケットを発生する UDP フローを用い，エッジで OUT とマークしている．

図 11 は本データ転送手法において，基本コネクションのウィンドウサイズを変化させた場合，および，1 つの TCP NewReno を用いて，データ転送を行った場合のスループットを示している．モデルにおけるルータでの遅延が 0 と想定した場合において，基本コネクションのスループットが，契約帯域を超えない最大のウィンドウサイズが 124 である．しかし，経路の RTT はトラフィック状況により変化するため，ウィンドウサイズの最適値は変化する．よって，本評価ではウィンドウサイズを変化させ，データを収集している．

4.1.1 輻輳時の契約帯域とスループットの差

輻輳時における契約帯域とスループットの差を検証する．図 11 ではウィンドウサイズが 132 以下の場合に，スループットは緩やかに減少している．これは基本コネクションのすべてのパケットが IN にマークされており，バックグラウンドトラフィックの負荷に比例して経路の RTT が増加するためである．これに対し，ウィンドウサイズが 136 以上の場合は，バックグラウンドトラフィックの負荷が少ない場合に基本コネクションのウィンドウサイズが大きすぎ，パケットの一部が OUT とマークされ，パケットが損失する．そのため，一度低下したスループットがバックグラウンドトラフィックの増加にともない上昇している．

本研究における基本コネクションのウィンドウサイズ制御は上限を固定している．図 11 から分かるように，ウィンドウサイズの上限値の決定に，トラフィックによる経路の RTT の変動を考慮に入れない場合（ウィンドウサイズが 124）でも従来の手法より契約帯域とスループットの差を小さくすることができる．また，トラフィックによる経路の RTT に対応した値を設定することで，さらに，改善することが可能である．

しかし，実際のネットワーク環境では，トラフィック状況がつねに変化し，経路の RTT 値を定数を与える方法でウィンドウサイズを最適化するのは困難である．性能をより向上させるには，経路の RTT の変動に合わせて，ウィンドウサイズの上限値を動的に変化させる必要がある．

4.1.2 非輻輳時の利用可能帯域の有効活用

本手法では，バックグラウンドトラフィックにかかわりなく，1 つの NewReno を用いたデータ転送より大きなスループットが得られている．これは複数のコネクションを利用しているためである．また，バックグラウンドトラフィックの負荷が 45 Mbps 以下の場合には，そのスループットは契約帯域より大きくなり，非輻輳時の利用可能帯域を有効活用できている．

4.2 実際のファイル転送を想定した評価

次に，実際のアプリケーションに適用することを想定したシナリオを用い，提案方式でシステム全体の SLA 対象となる契約帯域の削減効果について評価する．

本研究における対象アプリケーションの 1 つである口腔領域における流体解析[19] では，3 次元のボリュームデータをパイプライン処理で可視化する．ここでは，広域ネットワークを利用して構成される 1 つの Virtual Oragnization（VO）で，複数ユーザがアプリケーションを同時に利用する環境を想定する．そのため，各データ転送を独立に発生させ，各データ転送が乱数を用いて目標の終了時間を指定し，さらに，各データ転送は目標達成に必要な帯域をネットワークと契約したうえでデータを転送する．このとき，ネットワークの帯域が不足する場合は，データ転送は失敗終了（呼損；call loss）させる．なお，詳細なパラメータは次のような値を用い，3 万秒のデータを収集した．

データ転送の発生 データ転送の発生時間間隔は指数分布とした．

ファイルおよび分割ブロックサイズ ファイルサイズは平均 100 Mbyte のパレート分布とした．また，転送するファイルの平均サイズが 100 Mbyte であること

と，本評価のネットワークモデルにおいてメディアのエラーによる誤りによるパケットの損失しか発生しない場合の TCP のスループットの値を考慮し，1 つのデータブロックのサイズとして 5 Mbyte を利用した．

目標データ転送終了時間 1 つの TCP NewReno で達成可能なスループットを考慮し，乱数により，データ転送の目標終了時間を決定した．

リンクの速度および契約可能帯域 ボトルネックの速度は 100 Mbps，契約対象の帯域は 50 Mbps とした．

バックグラウンドトラフィック 各方式の基本的な特性を見るためのバックグラウンドトラフィックがない場合と一定レートの UDP トラフィックおよび，実際のトラフィックに近いパケットの発生間隔がパレート分布および，ポアソン分布となる UDP トラフィックとした．すべてのバックグラウンドトラフィックの転送速度は 50 Mbps とした．

契約帯域とスループットの比 本提案方式では，契約帯域とスループットの最小値の比（定数）が必要である．2 種類のフローを組み合わせたデータ転送手法ではこの値は経路の RTT によって変動する．同一のレートで通信した場合，RTT が最大の場合にスループットが最少となる．そのため，評価におけるネットワークモデルの経路の遅延に経路上のルータのバッファの最大値から RTT の最大値と最少値の比（0.806）をこの値として採用した．同様に，TCP NewReno は OUT にマークされたすべてのパケットが破棄される場合のスループットを最少と考え，0.75 を利用した．

4.2.1 目標転送時間達成率

動的契約帯域制御手法では，ネットワークとの契約帯域は予想される最小の値になるよう，定期的に削減するが，予期せぬ性能低下により，大きな契約帯域が必要になっても，契約を変更しない．そのため，データ転送の終了期限に間に合わない場合が発生していないか確認する必要がある．

一般的に，動的契約帯域制御技術では，ある帯域を契約した場合における得られるスループットの最小値を用い，必要帯域を計算する．この最小値は Diffserv エッジにおいて OUT とマークされたパケットがすべて廃棄される環境を想定しているため，ネットワーク側で SLA が守られている条件下，すなわち IN とマークされたパケットの破棄がない状態では，すべてのデータ転送に対して終了期限の確保が可能となる．

本評価で実施したシミュレーションでは，バックグラウンドトラフィックの種類により違いはあるものの，契約帯域の不足により呼損したものを除き，**図 12** に示すような数のデータ転送を実行している．これらのデータ転送すべての場合においてネットワークは契約帯域内の IN にマークされたパケットの廃棄率を 0 に保つことができている．このため，すべてのデータ転送が目標時間内に終了している．

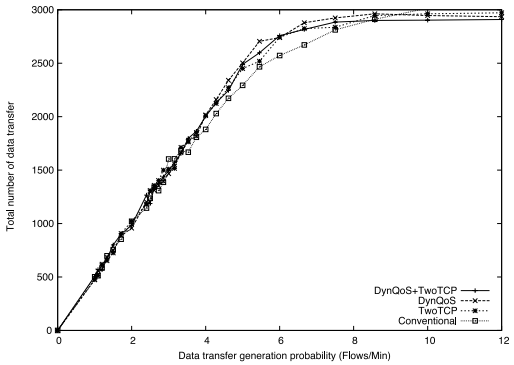

図 12 バックグラウンドトラフィックがない場合に実行した全データ転送数
Fig. 12 Total number of data transfer without background traffic.

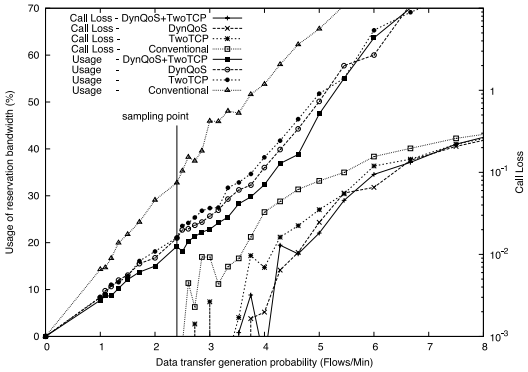

図 13 契約用帯域の利用率と呼損率
バックグラウンドトラフィックがない場合
Fig. 13 Usage of reservation bandwidth and call loss without background traffic.

4.2.2 契約帯域削減効果

ここでは，提案方式により，システム全体で必要となる契約帯域を，どの程度削減できるかを検証する．

契約帯域削減効果

図 13 にデータ転送により契約済みとなった帯域の最大値（50 Mbps）に対する比と，呼損率のグラフを重ねたものを示す．このグラフの横軸は単位時間あたりのデータ転送の発生確率である．

各グラフは，2 つのデータ転送を組み合わせたデータ転送手法単独の場合を TwoTCP，動的契約帯域制御手法を 1 つの NewReno コネクションに適用した場合を DynQoS，2 つの手法を組み合わせた場合を DynQoS+TwoTCP と表現している．

表 1 契約済み帯域の従来手法に対する比率
Table 1 Ratio of reserved bandwidth to conventional method.

方式	バックグラウンドトラフィックの種類			
	なし	Constant	Pareto	Poisson
負荷	2.40	1.50	1.71	1.50
TwoTCP	0.643	0.600	0.690	0.619
DynQoS	0.640	0.589	0.714	0.537
DynQoS +TwoTCP	0.587	0.532	0.597	0.504

契約済みとなる帯域を方式間で比較するには，呼損が存在しない負荷の範囲で行う必要がある．よって，バックグラウンドトラフィックの種類別に，すべての方式で呼損が存在しない最大負荷をデータから読み取り，その負荷における，予約済み契約帯域の平均値を算出した．このデータは，バックグラウンドトラフィックがない場合は毎分 2.4 個のデータ転送が発生する負荷（図 13 における sampling point）のデータとなる．

表 1 は 1 つの NewReno で帯域を契約した従来方式の契約済み帯域の平均値を 1 とし，提案方式の契約済み帯域をバックグラウンドトラフィック別にまとめたものである．負荷は該当バックグラウンドトラフィックの環境ですべての方式について呼損のない最大値であるが，すべての場合で最も早く呼損が発生するのは従来方式であり，従来方式における呼損がない最大負荷となる．この表から分かるように，提案方式は契約済みとなる帯域を削減することができている．

契約帯域削減効果の内訳

次に提案手法のどの機能が契約帯域の削減にどれだけ貢献するかについて議論する．まず，2 種類のフローを組み合わせたデータ転送手法を単独で用いた場合は，基本コネクションに対してウィンドウサイズを制御することにより，各データ転送が契約すべき帯域が削減できる効果と，追加コネクションを利用することにより，大きなスループットが得られ，結果として早期にデータ転送が終了し，契約済み帯域の平均値が減少する効果の 2 つである．

このうち，基本コネクションを制御することによる効果は，評価のパラメータにおける “契約帯域とスループットの比” から算出できる．提案方式では，目標となるデータ転送の終了時間と，転送すべきデータサイズから達成すべきスループットを計算し，このスループットから契約すべき帯域値を決定している．この際，達成すべきスループットから契約帯域を求めるために用いているのが，“契約帯域とスループットの比” であり，この値が 1 に近いほど良い．

本評価では，TCP NewReno に対して 0.75，2 種類のフローを組み合わせたデータ転送手法に対して 0.806 を用いており，この差である 0.056 が基本コネクションのウィンドウサイズを制御することによる契約帯域削減効果となり，約 5%である．しかし，表 1 の TwoTCP の行を見ると，すべての場合で 30%以上削減できている．そのため，30%と 5%の差が追加コネクションを併用することにより，データ転送が早期に終了することによる帯域削減効果である．

表 1 の TwoTCP の行を見ると，すべての場合で 0.6 から 0.7 の間の値である．これはシステム全体の契約済み帯域が，従来方式に比べて 30%から 40%程度削減できることを示している．同様に，DynQoS も 28.6%以上の帯域を削減している．しかし，DynQoS+TwoTCP を見ると，40%から 50%程度しか契約帯域を削減できていない．つまり，2 つの手法を併用した場合の契約帯域削減効果は，各データ転送手法を単独で用いた場合の削減効果の積にならない．

ネットワークが輻輳していない場合，2 つの手法を併用していると，契約帯域と基本コネクションのウィンドウサイズは徐々に減少する．これにともない，各データ転送の実行時間は動的な契約帯域制御手法を適用しない場合より増加する．この転送時間の増加は契約済み帯域の平均値を増加させる効果を持つためである．

4.2.3 同一契約帯域における転送性能

最後に，提案方式における契約帯域の削減効果により，システム全体でのスループット（単位時間あたりのデータ転送性能）が，どの程度向上するかを評価する．契約済み帯域の評価では，同一の負荷でデータを比較したが，この評価では，各方式ごとにデータ転送が帯域を契約できず，呼損する比率が一定以下となる最大の負荷を求め，その負荷時における，単位時間あたりのデータ転送量で比較する．呼損率が 1%となる負荷と，その負荷時のスループットを十分な精度で求めるため，30 万秒分のデータを収集し，データに対して最小自乗法を適用した．なお，この評価ではバックグラウンドトラフィックがない場合と，ピークレートが 50 Mbps でパケットの発生時間間隔が指数分布のトラフィックの 2 通りのみデータを収集した．

図 14 にバックグラウンドトラフィックがない場合のスループットと呼損率のグラフを重ね合わせたものを示す．このグラフから，呼損率が 1%となる各方式の負荷を読み取り，その負荷時の該当方式のスループットを求めている．この結果をまとめたのが **表 2** である．

この表において，各方式の値は単位時間あたりのデータ転送量（単位は Mbps）であり，両方式の比は

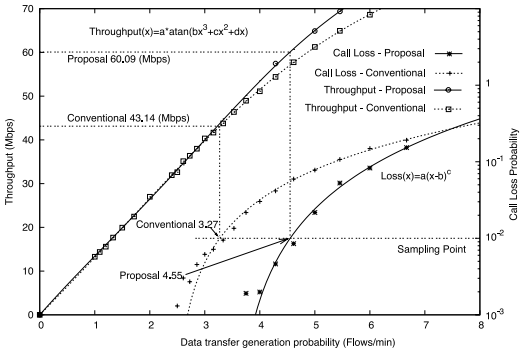

図 14 システム全体の最大スループット
Fig. 14 Maximum throughput.

表 2 同一契約帯域における転送性能
Table 2 Performance using same reservation bandwidth.

	なし	Poisson
従来方式（Conventional）	43.14	31.15
提案方式（Proposal）	60.09	39.59
両方式の比	1.39	1.27

従来方式のデータ転送量を 1 としたときの値である．提案方式（2 つの手法を同時に利用した場合）は従来方式（1 つの NewReno コネクションで帯域を契約した場合）と比較し，ネットワークの契約済み帯域が減少するため，より高い負荷でも呼損を発生させずに処理することが可能となり，単位時間あたりのデータ転送量がバックグラウンドトラフィックがない場合で 39%，バックグラウンドトラフィックがポアソン分布の場合で 27%増加している．

5. ま と め

本研究では，大量データの転送に対する終了時間の保証を必要とするアプリケーションを想定し，Diffserv AF における契約帯域を削減するデータ転送方式を提案し，そのシミュレーションによる評価を行った．シミュレーションでは，通常想定されるデータ転送の発生確率およびネットワークの輻輳状況においては，ネットワークとの契約帯域を低減効果を確認できた．

しかし，評価に用いた実装では，基本コネクションに対してウィンドウサイズの上限に定数を利用しているため，経路の RTT の変動が大きい環境では，契約帯域の帯域削減効果を十分得ることができない．また，多くの環境で利用可能とするため，追加コネクションを TCP NewReno としたが，RTT の大きなネットワークでは経路の利用可能帯域を十分活用できない．

そのため，適用可能な環境が少なくなるものの，追加コネクションに利用するプロトコルを変更すること，および基本コネクションのウィンドウサイズの上限を RTT の変動に応じて制御することにより性能向上を目指すことが今後の課題である．

謝辞 貴重なご指導・ご支援をたまわりました JGN2 関係各位の方々，NICT 本部の皆様，ならびに，大阪大学下條研究室の皆様に心より感謝いたします．

参 考 文 献

1) Blakea, S., et al.: An Architecture for Differentiated Services, RFC 2475, IETF (1998).
2) Heinanen, J., Baker, F., Weiss, W. and Wroclawski, J.: Assured Forwarding PHB Group, RFC 2597, IETF (1999).
3) Stevens, W.: TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms, RFC 2001, IETF (1997).
4) Allman, M., Paxson, V. and Stevens, W.: TCP Congestion Control, RFC 2581, IETF (1999).
5) Floyd, S. and Henderson, T.: The NewReno modification to TCP's fast recovery algorithm, RFC 2582, IETF (1999).
6) Floyd, S., Henderson, T. and Gurtov, A.: The NewReno Modification to TCP's Fast Recovery Algorithm, RFC 3782, IETF (2004).
7) Nichols, K., Blake, S., Baker, F. and Black, D.: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, RFC 2474, IETF (1998).
8) Clark, D.D. and Fang, W.: Explicit allocation of best-effort packet delivery service, *IEEE/ACM Trans. Networking*, Vol.6, No.4, pp.362–373 (1998).
9) Gerla, M., et al.: TCP Westwood with adaptive bandwidth estimation to improve efficiency/friendliness tradeoffs, *Journal of Computer Communications*, Vol.27, No.1, pp.41–58 (2004).
10) Xu, L., Harfoush, K. and Rhee, I.: Binary Increase Congestion Control for Fast Long-Distance Network, *Proc. INFOCOM2004* (2004).
11) Gu, Y., Hong, X. and Grossman, R.: Experiences in the Design and Implementation of a High Performance Transport Protocol, *SC2004* (2004).
12) Hacker, T.J., Athey, B.D. and Noble, B.: The End-to-End Performance Effects of Parallel TCP Sockets on a Lossy Wide-Area Network, *Parallel and Distributed Processing Symposium*, Washington, DC, USA, IEEE Computer Society, pp.434–443 (2002).
13) Semke, J., Mahdavi, J. and Mathis, M.: Automatic TCP Buffer Tuning, *COMPUTER*

COMMUNICATION REVIEW, Vol.28, No.4, pp.315–323 (1998).
14) 檜山亜佑子，野呂正明，馬場健一，下條真司：転送時間を保証するための大規模データ転送手法の提案，電子情報通信学会総合大会公演論文集，pp.B–7–144 (2006).
15) 野呂正明，馬場健一，下條真司：Grid における大規模ファイル転送向け QoS 制御方式の性能評価，信学技報，IN2005-83, pp.131–136 (2005).
16) Noro, M., Baba, K. and Shimojo, S.: QoS Control Method to Reduce Resource Reservation Failure in DataGrid Applications, *Proc. PACRIM2005* (2005).
17) 野呂正明，長谷川一郎，馬場健一，下條真司：Grid における大量データ送信に適した品質保証方式，信学技報，IA2004-22, pp.21–16 (2005).
18) McCanne, S. and Floyd, S.: The Network Simulator ns-2.
19) Nozaki, K., et al.: Computational Oral and Speech Science on E-science Infrastructures, *HPC Analytics Challenge of SC06* (2006).

(平成 19 年 5 月 19 日受付)
(平成 19 年 9 月 3 日採録)

野呂　正明（正会員）

昭和 63 年名古屋大学工学部電気工学科卒業．平成 2 年同大学大学院工学研究科情報工学専攻博士前期課程修了．同年富士通（株）入社．富士通研究所にて研究開発に従事．平成 15 年大阪大学大学院情報科学研究科マルチメディア工学専攻博士後期課程入学．平成 19 年同大学院退学．平成 16 年より情報通信研究機構にてネットワークに関する研究に従事．

馬場　健一

平成 2 年 3 月大阪大学基礎工学部情報工学科卒業，平成 4 年 3 月同大学大学院基礎工学研究科物理系専攻情報工学分野博士前期課程修了，平成 4 年 4 月同大学院博士後期課程に進学し，同年 9 月同大学院退学．同年 10 月大阪大学情報処理教育センター助手として採用，平成 9 年 4 月高知工科大学工学部電子・光システム工学科講師，平成 10 年 12 月大阪大学大型計算機センター助教授，平成 12 年 4 月同大学サイバーメディアセンター助教授，平成 19 年 4 月より准教授として勤務．現在に至るまで広帯域ネットワーク，コンピュータネットワーク，フォトニックネットワークシステムの性能評価に関する研究に従事．電子情報通信学会，IEEE 各会員．

下條　真司（正会員）

昭和 61 年 3 月大阪大学大学院基礎工学研究科後期課程修了．同年 4 月大阪大学基礎工学部助手．平成元年 2 月同大学大型計算機センター講師．平成 3 年 4 月同センター助教授．平成 10 年 4 月同センター教授．平成 12 年 4 月同大学サイバーメディアセンター教授・副センター長，平成 17 年 8 月同大学サイバーメディアセンター教授・センター長，現在，同大学サイバーメディアセンター教授・副センター長．その間米国カリフォルニア大学アーバイン校客員研究員．マルチメディア応用システム・peer-to-peer コミュニケーションネットワークユビキタスネットワークシステム・グリッド技術に関する研究に従事．ACM，IEEE，ソフトウェア科学会，電子情報通信学会各会員．