社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

サービスオーバーレイネットワークのための インラインネットワーク計測に関する一検討

CaoLe Thanh Man† 長谷川　剛†† 村田　正幸††

† 大阪大学情報科学研究科
〒 560-8531 大阪府豊中市待兼山町 1-3
†† 大阪大学サイバーメディアセンター
〒 560-0043 大阪府豊中市待兼山町 1-32
E-mail: †mlt-cao@ist.osaka-u.ac.jp, ††{hasegawa,murata}@cmc.osaka-u.ac.jp

あらまし　サービスオーバレイネットワークにおいては、下位層ネットワークに相当する IP ネットワークの状態 (利用可能帯域、ネットワーク輻輳レベル、ルーティング等) に関するできるだけ最新かつ詳細な情報を随時把握することで、サービス品質の向上が可能となる。しかし、既存のネットワーク計測アルゴリズムは、より正確な情報を得ることを目的とするために、計測に時間がかかる、多くの計測用のパケットを用いるため通常トラヒックに与える影響が大きいなどの問題点を持つ。そこで本稿では、アクティブな TCP コネクションのデータ転送時に得られる情報に基づいて、エンドホスト間で利用可能な帯域幅 (Available Bandwdith) を計測する新たな計測方式を提案する。提案方式は外部トラヒックに与える影響が小さく、計測の初期段階から結果を求めることができるという特徴を持つ。
キーワード　計測、利用可能帯域、TCP (Transmission Control Protocol)、インライン計測

A study on Inline Network Measurement Mechanism for Service Overlay Networks

Cao LE THANH MAN†, Go HASEGWA††, and Masayuki MURATA††

† Graduate School of Information Science and Technology, Osaka University
1-3, Machikaneyama, Toyonaka, Osaka 560-8531, Japan
†† Cybermedia Center, Osaka University
1-32, Machikaneyama, Toyonaka, Osaka 560-0043, Japan
E-mail: †mlt-cao@ist.osaka-u.ac.jp, ††{hasegawa,murata}@cmc.osaka-u.ac.jp

Abstract　In this paper, we introduce a new network measurement technique to measure available bandwidth for a path between two endhosts, which is necessary for service overlay network to improve its service quality. Although our method is based on the active measurement approach, we do not use extra packets for the measurement. Instead, the measurement is performed by correcting the network characteristics information obtained from the packets which an active TCP connection is transmitting for providing a service. We call this kind of measurement approach “Inline Network Measurement”. Furthermore, our approach gives estimated measurement results even at the beginning stage of the measurement task.
Key words　Measurement, Available Bandwidth, TCP (Transmission Control Protocol), Inline Network Measurement

1. はじめに

ネットワーク計測に関する研究はこれまでに活発に行われてきており、数多くの計測ツールが開発されてきた [1-5]。これらのツールにより、リンクの物理的な帯域、エンドホスト間で現在利用可能な帯域、伝送遅延時間、パケットロス率、ネットワークトポロジーなど、様々なネットワーク特性を計測することができる。ネットワーク特性の計測結果は、ネットワーク内の故障箇所の特定や解決、およびネットワーク設計等に用いられる。ネットワーク特性の計測方法は通常、受動的計測法 (Passive Measurement) と能動的計測法 (Active Measurement) の 2 種類に分類される。受動的計測法では、ネットワーク内のある計測地点を通過するトラヒックを監視し、その情報を収集することでネットワーク特性の導出を行う。一方、能動的計測法では、あるホストからネットワーク内に計測用のパケットを送

出し、その結果 (転送遅延、パケット到着間隔等) からネットワーク特性を推測する。後者は計測のために余分なパケットをネットワーク内に送出する必要があるが、前者に比べより詳細なネットワーク特性情報を得ることができる。

一方、近年のネットワークサービスの多様化に伴い、サービスオリエンテッドなネットワーク (サービスオーバーレイネットワーク) が拡がりつつある。例えば、ピア同士の直接的な通信を実現する P2P ネットワーク、ネットワーク上での分散計算環境を提供するグリッドネットワーク、コンテンツ配信を目的とした Contents Delivery Network (CDN)、IP ネットワーク上に仮想網を構築する IP-VPN などである。これらのネットワークは、IP ネットワークを下位層ネットワークとして、特定のサービスを提供する上位層ネットワークととらえることができる。したがって、これらのネットワークにおいてサービス品質を向上させるためには、下位層ネットワークである IP ネットワークを与条件として、サービス提供のためのコネクション設定要求が発生した時に、利用可能な下位層ネットワーク資源量を適切に把握することが重要である。

例えば、Akamai [6] や Exodus [7] 等の CDN サービスを考える。CDN サービスにおいては、複数台の実 Web サーバと、プロキシ (キャッシュ) サーバがネットワーク内に配置される。Web クライアントからプロキシサーバに対してドキュメントの転送要求が発生した場合、プロキシサーバは、自身のドキュメントキャッシュに要求されたドキュメントが存在すればそれを転送し、存在しなければ対応する実 Web サーバからそのドキュメントを取得し、Web クライアントへ転送する (この時発生するトラヒックをここではフォアグラウンドトラヒックと呼ぶ)。また、プロキシサーバを複数台設置し、プロキシサーバ間でドキュメントのやりとりを行うことで、実 Web サーバへのトラヒックを減少させるものもある [8]。プロキシサーバは、ミスキャッシュ転送に加えて、Web クライアントが近い将来に要求すると考えられるドキュメントをあらかじめ実 Web サーバから取得 (プリフェッチ) し、キャッシュに保存する (プリフェッチ動作によって発生するトラヒックをバックグラウンドトラヒックと呼ぶ)。フォアグラウンドトラヒックはキャッシュミスによって発生するため、その転送はできるだけ早く完了する必要がある。そのため、バックグラウンドトラヒックによってフォアグラウンドトラヒックが影響を受け、転送速度が低下することを避ける必要がある。このような問題を解決する方法として、例えば下位層の IP ネットワークに DiffServ アーキテクチャを前提とすることが考えられる。しかし、そのためにはトラヒックの DiffServ ネットワークの AF あるいは EF クラスへのマッピング、必要帯域量の把握などが必要となるため、トラヒック予測が困難なアプリケーションを対象とする場合には、極めて非現実的なものとなる。

そこで、上記の問題を解決する制御の実現方法として、エンドホスト間 (CDN では実 Web サーバ/プロキシサーバ間) のパス上において利用可能な帯域を計測し、計測結果に基いてフォアグラウンドトラヒックに悪影響を与えないようにバックグラウンドトラヒックの転送速度を調整することが考えられる。例えば、計測された利用可能帯域を基に、バックグラウンドトラヒックを転送している TCP コネクションの最大ウィンドウサイズを設定することで、バックグラウンドトラヒックが不要に高いレートで転送されることを防止することができる。その他、紙面の制限上詳細は述べることができないが、

- P2P ネットワークにおいて、資源発見手続きによって複数のピアが同じ資源を有することがわかった後に、どのピアの資源を利用するかを決定する
- データグリッドにおいて、複数のサイトが同じデータを有する時に、どのサイトからデータをコピーするかを決定する

などの数多くの領域において適応型制御を実現するために計測技術が有用となる。

しかし、既存の利用可能帯域計測方式 [1-3] は、計測に長い時間がかかる、多くの計測用のパケットを用いるため外部トラヒックに与える影響が大きいなどの特徴を持つ。サービスオーバレイネットワークにおいては、常に最新の利用可能なネットワーク資源量をネットワーク内の他のトラヒックに悪影響を与えることなく取得することが重要であり、そのため既存の方式をそのまま適用することはできない。

そこで本稿では、サービスを提供しているエンドホスト間の TCP コネクションを直接用いて、データ転送中に得られる情報からエンドホスト間の利用可能帯域を随時推測するインラインネットワーク計測方式の提案を行う。この方式により、計測用のパケットをネットワーク内に送出することなく計測を行うことができるため、計測負荷を最小限に抑えることができる。そのために本稿では、まず TCP コネクションによるインライン計測の際に問題となる点を明らかにし、それを解決するために、少ない計測パケット数で計測の初期段階から利用可能帯域の計測結果を導出することのできる、利用可能帯域の計測方式を提案する。さらに、提案した利用可能帯域の計測方法を、データ転送中の TCP コネクションを用いて行うインライン計測方式に関しても検討を行う。

以下、2 章において既存の利用可能帯域計測方式の問題点、及び TCP コネクションを用いたインライン計測を行う際の問題点について述べ、3 章で少ないパケット数で、計測の初期段階から計測結果を導出する利用可能帯域計測方式を提案しする。4 章ではシミュレーションによる提案方式の評価結果を示し、提案方式の有効性を検証する。5 章では、提案方式を TCP コネクションを用いたインライン計測に適用するための指針を示す。最後に 6 章でまとめと今後の課題について述べる。

2. ネットワーク計測

本章ではまず、エンドホスト間で利用可能な帯域を計測する既存の方式を挙げ、1 章で述べた、サービスオーバレイネットワークのためのネットワーク計測にそのまま用いることができないことを指摘する。また、データ転送中の TCP コネクションを用いたインライン計測を行う際に発生する問題点に関しても議論を行う。

2.1 既存の計測方式の問題点

1 章で述べたように、ネットワーク計測方式は受動的計測法 (Passive Measurement) と能動的計測法 (Active Measurement) の 2 種類に分類される。CPAND [4]、Nettimer [5] 等の受動的計測法は、ネットワーク内やエンドホスト上に観測点を設け、そこを通過するトラヒックを監視することで、ネットワーク特性を計測する。しかし、観測できる情報が制限されており、計測したい特性のための情報が得られるとは限らないため、利用可能帯域などのエンドホスト間の特性を詳細かつ正確に計測することはできない。

一方能動的計測法は、あるホストからネットワーク内に計測用のパケットを送出し、その結果 (転送遅延、パケット到着間隔、パケット廃棄率等) からネットワーク特性を推測する。この方法を採用し、エンドホスト間の利用可能な帯域を計測する既存の方式には Cprobe [1]、TOPP [2]、PathLoad [3] 等がある。また、これらの方式はエンドホスト上で動作し、ネットワーク内部の特殊な動作を前提としないため、サービスオーバレイネットワークにおける利用可能帯域計測にも適していると考えられる。しかし、これらの方式は非常に多くの計測用のパケットを高いレートでネットワーク内に送出する。例えば TOPP は一度の計測のために、約 5000 個のパケットを送出し、計測を行うエンドホストが接続されているインターフェースの速度で送出することも必要になる。高いレートでネットワーク内に送出された計測パケットは他のトラヒックにスループット低下、パケット廃棄の発生等の悪影響を与えるため、これらの方式をそのまま用いることはできない。

さらに、これらの方式は計測を完了するまでに、数 10 から数 100 ラウンドトリップ時間という長い時間を必要とする。多くのパケットを用いて長時間の計測を行うことで、より正確な利用可能帯域の計測が可能になるが、その反面、IP ネットワークのトラヒック変動に追随できないという欠点を持つ。特に、下位層ネットワークとして IP ネットワークを用いてサービスの提供を行うサービスオーバレイネットワークにおいては、できうる限り最新のネットワークトラヒックの状況を反映したデータ転送を行う必要があるため、短い時間で計測結果を導出するとともに、継続して計測を行うことで、トラヒック状況の変化を検知することが求められる。この点からも、既存の能動的計測法を用いた利用可能帯域計測方式を、そのまま適用することはできない。

そこで本稿では、既存の計測方式が持つこれらの問題点を解

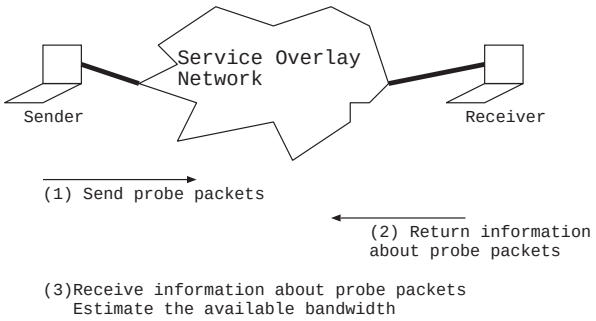


図 1　計測の流れ

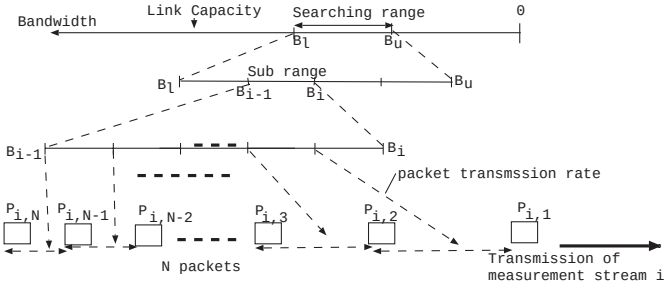


図 2　計測アルゴリズム

決する新たな利用可能帯域の計測方式を提案する。提案方式は少ない個数の計測パケットを用い、その送出レートをできるだけ低く抑えることで、他のトラヒックに与える悪影響を小さくする。また提案方式は、計測の初期段階から計測結果を継続的に導出することができる。詳細は次章で述べる。

2.2　TCP によるインライン計測における問題点

本稿で目標とする、データ転送中の TCP コネクションを用いたインライン計測を実現するためには、TCP のパケット転送方法の特性を考慮する必要がある。利用可能帯域の計測の際に問題となる TCP の特性としては、以下のようなものが挙げられる。

ウィンドウサイズ

送信側 TCP が一度に送出することのできるパケット数は、ウィンドウサイズ (輻輳ウィンドウサイズ、広告ウィンドウサイズ) によって決定されるため、計測のために一度に送出することができるパケット数がウィンドウサイズによって制限される。

さらに、ウィンドウサイズは時間とともに大きく変動するため、それにともない計測のために一度に送出することができるパケット数も変動する。また、TCP は ACK パケットの受信とともに新たなデータパケットを送信するため、その送信レートは ACK パケットの受信レートに大きく依存する。

受信側 TCP の動作

受信側端末は、送信側端末から送出されたデータパケットを受信すると、ACK パケットを生成して送信側端末へ返送する。送信側端末は返送された ACK パケットの到着間隔を見て、利用可能帯域の推測を行なうため、受信側 TCP において Delayed ACK 等の処理を行っている場合には、正しい計測を行うことができない。

生存時間

TCP コネクションは転送するデータサイズが小さい場合にはその生存時間が非常に短く、送出するデータパケット数も小さい。そのため、計測に使うことができる総パケット数が制限される。

本稿で提案する利用可能帯域計測方法は、これらの問題を解決し、TCP コネクションを用いたインライン計測に適用することを考慮したアルゴリズムを持つ。次章でその詳細を述べる。

3.　提 案 方 式

提案方式は、送信側エンドホスト、受信側エンドホスト間の現在の利用可能帯域値 A を導出する。図 1 に示すように、まず送信側エンドホストが計測パケットを送出し、受信側エンドホストは受信した計測パケットをそのまま送信側エンドホストへ返送する。送信側端末は返送されたパケットの到着間隔から利用可能帯域の推測を行う。

提案方式において利用可能帯域を計測する際には、図 2 に示すように、現在の利用可能帯域値が含まれると考えられる帯域の上限 B_u と下限 B_l を設定し、区間 $I = (B_l, B_u)$ の中から利用可能帯域を探す。この区間のことを探索区間と呼ぶ。ここで、探索区間の下限 B_l の最小値は 0、上限 B_u の最大値は、送信側エンドホストが接続しているリンク帯域である。探索区間を設定することで、不必要に高いレートで計測パケットを送出することを避けることができるため、外部トラヒックに与える影響を最小限に抑えることができる。また、TOPP のように、非常に広い帯域幅の中から利用可能帯域を探索しないため、計測の精度を保ちながら用いる計測パケット数を減少させることができる。後述するように、探索区間は過去の計測結果を基に設定するため、ネットワーク状況の変化に伴い利用可能帯域が急激に変化した時に、探索区間内に利用可能帯域が存在しない場合もある。提案するアルゴリズムでは、そのような場合においても、数回の計測で新たな利用可能帯域を発見することができる。提案する利用可能帯域計測アルゴリズムにおける、送信側エンドホストの動作概略を以下に示す。

（1） 初期探索区間を決定する
（2） 探索区間を複数の小区間に分割する
（3） 各小区間に対応する計測ストリームを送出・受信し、送出間隔と受信間隔の比較を行い、パケット間隔が大きくなったかどうかを判断する
（4） 送出・受信結果から、利用可能帯域が含まれると考えられる小区間を選択する
（5） 選択した小区間に対応する計測ストリームの送出・受信結果から、利用可能帯域を算出する
（6） 探索区間の再計算を行い、(2) へ戻る

以下、ステップ (1)–(6) における詳細なアルゴリズムを示す。ここで、計測ストリームとは、計測のために一度にネットワークへ送出するパケット群のことを指す。また、1 つの計測ストリームで用いるパケット数を N とする。

(1) 初期探索区間の決定

まず、Cprobe のアルゴリズムに基づいて 1 つの計測ストリームを送出し、導出された利用可能帯域値 A_{cprobe} を用いて、初期探索区間を $(A_{\mathrm{cprobe}}/2, A_{\mathrm{cprobe}})$ と設定する。

(2) 探索区間の分割

探索区間を、以下のように大きさの等しい k 個の小区間 $I_i = (B_{i+1}, B_i)\ (i = 1, \ldots, k)$ に分割する (図 2)。すなわち、

$$B_i \;=\; B_u - \frac{B_u - B_l}{k}(i-1)\ (i = 1, \ldots, k+1)$$

となる。k が大きくなると、小区間の幅が小さくなるため、ステップ (4) 及び (6) においてより正確な判断が可能となるが、その反面、一回の計測に用いる計測ストリーム数が増加するため、計測にかかる時間が大きくなる。

(3) 計測ストリームの送出及びパケット間隔の比較

分割した k 個の小区間それぞれに対して、計測ストリーム i $(i = 1, \ldots, k)$ をネットワーク内に送出する。その際、1 つの計測ストリーム内のパケットの送信間隔を変化させることによって、小区間 $I_i = (B_{i+1}, B_i)$ が持つ帯域幅を全てカバーする (図 2)。つまり、計測ストリーム i 内の j 番目の計測パケット $P_{i,j}$ $(1 \leqq j \leqq N)$ の送出時刻 $S_{i,j}$ は、以下の関係を満たす。ただし、$S_{i,1}$ = 0 とし、計測パケットサイズを M とする。

$$\frac{M}{S_{i,j+1} - S_{i,j}} \;=\; B_{i+1} + \frac{B_i - B_{i+1}}{N}(j-1)$$

提案方式ではこのように、1 つの計測ストリーム内のパケット

の送出間隔を変化させて送信するため、PathLoad のように計測ストリーム内のパケットを全て同じ送出間隔で送出する方式に比べて計測誤差が大きくなる。しかしその反面、1 つの計測ストリームで様々な送信レートに対する計測結果を収集できるため、計測パケット数が少なくなり、短時間で計測結果を導出することができる。

次に、送出された計測ストリーム内のパケット $P_{i,j}$ が返送されてくると、その到着時刻を記録する。それを $R_{i,j}$ とする ($R_{i,1}$=0)。送信パケット $P_{i,j}$ $(1 \leqq j \leqq N-1)$ に対して、パケット送出間隔 $(S_{i,j+1} - S_{i,j})$ と到着間隔 $(R_{i,j+1} - R_{i,j})$ を PathLoad で用いられているアルゴリズム [3] を用いて比較し、パケット間隔に増加の傾向があるかどうかを判断する。すなわち、送出間隔よりも到着間隔の方が大きければ、その送出間隔で実現される計測パケットの送信レートが、利用可能帯域よりも大きいと考えられる。PathLoad では、計測ストリーム内の全ての計測パケットについて間隔の増加の傾向を調べることで、利用可能帯域の推測を行っている。

各計測ストリーム i の増加の傾向を示す変数 T_i を、以下のように定義する。

$$T_i = \begin{cases} 1 & \text{パケット間隔に増加の傾向がある} \\ -1 & \text{パケット間隔に増加の傾向がない} \\ 0 & \text{パケット間隔の増加の傾向を判断できない} \end{cases}$$

計測ストリームは、小区間の帯域値が大きい方から順に送出するため、i が小さい時には $T_i = 1$ になり、i が大きくなると、計測ストリームの平均送出レートが小さくなるため、ある時点で $T_i = -1$ に転じる傾向を持つ。そこで、2 つの連続する計測ストリーム m および $m+1$ で、$T_m = T_{m+1} = -1$ になれば、$(m+2)$ 番目以降の計測ストリームは送出せずに、$T_i = -1$ $(m+2 \leqq i \leqq k)$ とすることで、計測速度の向上を図っている。

提案方式ではこのように、各ストリームのパケット間隔に増加の傾向があるか否かのみを判断し、増加の度合いは用いない。これは、Cprobe や TOPP のようにパケット間隔の増加の度合いを利用可能帯域値の算出に用いると、真の利用可能帯域の値に比べて計測パケットの送出レートが高い場合に、計測される利用可能帯域の値に誤差が生じるためである [9]。

(4) 小区間の選択

全ての計測ストリームのパケット間隔の増加傾向 T_i $(1 \leqq i \leqq k)$ から、現在の利用可能帯域値 A が含まれると考えられる小区間を以下のように決定する。まず、$\sum_{j=1}^{a} T_j - \sum_{j=a+1}^{k} T_j$ が最大となる a $(0 \leqq a \leqq k+1)$ を求め、a が $1 \leqq a \leqq k$ を満たす場合には小区間 I_a を選択する。すなわち、I_a は、パケット間隔に増加の傾向がある部分とない部分の境界部分に存在する小区間となる。これは、パケット間隔の増加傾向が変化する境界部分に対応する計測ストリームの平均送出レートは、利用可能帯域の値に近いと考えられるためである。

また、$a = 0$ あるいは $a = k+1$ の場合は、探索区間 (B_l, B_u) 内に利用可能帯域が存在せず、$a = 0$ の場合にはもっと大きな利用可能帯域値を持つ $(B_u < A)$、$a = k+1$ の場合にはもっと小さな利用可能帯域を持つ $(A < B_l)$ ものとそれぞれ判断する。

(5) 利用可能帯域を算出

ステップ (4) において探索区間 (B_l, B_u) 内のある小区間 I_a 内に利用可能帯域が存在すると判断された場合は、以下のようにして利用可能帯域を導出する。まず、小区間 I_a に対応する計測ストリーム a 内の計測パケット $P_{a,j}$ $(i = 1, \ldots, N)$ に対して、パケット送出レートおよびパケット到着レートをそれぞれ $\frac{M}{S_{i,j+1} - S_{i,j}}$ および $\frac{M}{R_{i,j+1} - R_{i,j}}$ と定義する。次に、図 3 に示すように、送出レートと到着レートの関係を線形回帰法によって 2 本の直線で近似する。ステップ (4) において述べたパケット到着間隔の増加の傾向から、2 本の直線 (i)、(ii) のうち、送出レートが低い (送出間隔が大きい) 点が含まれる直線 (i) は傾きが 1 に近く (送出レートと到着レートがほぼ等しい)、送出レートが高い (送出間隔が小さい) 点が含まれる直線 (ii) は傾きが 1 よりも小さい (送出レートよりも到着レートの方が大きい) という傾向を持つことがわかる。したがって、2 本直線の交点付近に利用可能帯域が存在すると考えられるため、直線 (i) のうち、最も送出レートが高いパケットの送出レートを、現在の利用可能帯域 A として導出する (図 3)。

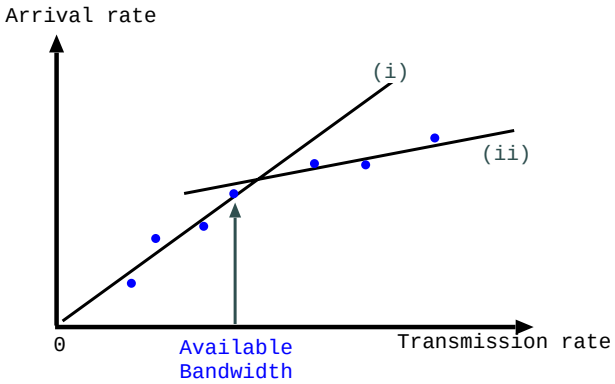


図 3　小区間内の利用可能帯域の導出方法

一方、ステップ (4) において探索区間 (B_l, B_u) 内に利用可能帯域が存在しないと判断された場合は、ネットワークの状況に変化があり、利用可能帯域が大きく変化していると考えられる。したがって、利用可能帯域の変化を陽に示すために、現在の利用可能帯域 A として暫定的に以下の値を導出する。

$$A = \begin{cases} B_l & a = 0 \\ B_u & a = k+1 \end{cases}$$

(6) 探索区間の再計算

ステップ (5) において小区間 I_a 内に利用可能帯域 A が発見され、かつその値がこれまでに蓄積されている利用可能帯域の計測値の集合から導出された 95% の信頼区間 $(B_{95,l}, B_{95,u})$ 内に含まれていれば、今回計測された利用可能帯域 A を新たに蓄積データとして保存し、新たに 95% の信頼区間を求め、その幅を次回の計測時の探索区間の幅とする。また、今回の計測結果 A を探索区間の中心とする。すなわち、次回の計測時の探索区間 (B'_l, B'_u) は以下のようになる。ここで $\bar{A}$ および S はそれぞれ統計データの平均および分散、q は蓄積データの個数である。

$$(B_{95,l}, B_{95,u}) = \left(\bar{A} - 1.96\frac{S}{\sqrt{q}}, \bar{A} + 1.96\frac{S}{\sqrt{q}} \right)$$

$$(B'_l, B'_u) = \left(A - \max\left(1.96\frac{S}{\sqrt{q}}, \frac{B_m}{2} \right), A + \max\left(1.96\frac{S}{\sqrt{q}}, \frac{B_m}{2} \right) \right)$$

ここで B_m は、探索空間が小さくなりすぎることを防止するために設定される探索区間の幅の最小値である。また、初回の計測時、および蓄積データが破棄された直後の計測時で、蓄積データが存在しない場合には、次回計測時の探索区間として、今回の探索区間を用いるものとする。

一方、ステップ (4) において探索区間 (B_l, B_u) 内に利用可能帯域が存在しないと判断された場合には、ネットワーク状況が変化した可能性があると判断し、これまでの利用可能帯域 A の蓄積を破棄する。そして、次回の探索空間 (B'_l, B'_u) を以下のように設定する。

$$B'_l = \begin{cases} B_l & a = 0 \\ B_l - \frac{B_u - B_l}{2} & a = k+1 \end{cases}$$

$$B'_u = \begin{cases} B_u + \frac{B_u - B_l}{2} & a = 0 \\ B_u & a = k+1 \end{cases}$$

これは、ネットワーク状況が変化し、利用可能帯域に大きな変化がある場合に備えて、探索空間を利用可能帯域の変化の方向へ大きくすることを意味している。

4. 評 価 結 果

本章では、3 章で提案した利用可能帯域推測方式を、ns [10]

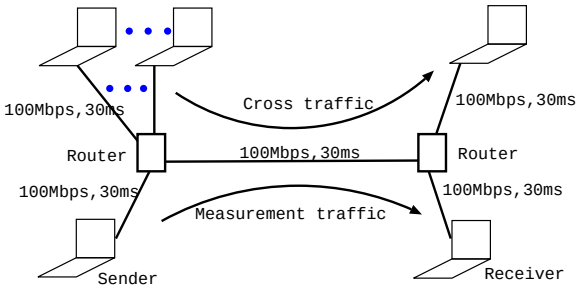


図 4　シミュレーションモデル

を用いたシミュレーションによって評価した結果を示し、提案方式の有効性を検証する。シミュレーションで用いたネットワークモデルを図 4 に示す。モデルは、帯域が 100 Mbps、伝搬遅延時間が 30 msec のボトルネックリンク上に、ルータを介して背景トラヒック (Cross traffic) を発生させる送受信エンドホスト、および利用可能帯域の計測を行う送受信エンドホスト (Sender、Receiver) が接続されている。各ホストとルータ間のリンクは全て帯域が 100 Mbps、伝搬遅延時間が 30 msec である。背景トラヒックは、指定したレートで UDP によるデータ転送を行うことで生成している。

送信側エンドホストは、3 章で述べたアルゴリズムにしたがって、受信側エンドホストとの間のパスの利用可能帯域を計測するため、このシミュレーションでは、ボトルネックリンクの利用可能帯域を計測することになる。計測アルゴリズムにおける探索区間の小区間への分割数 k は、探索区間と前回の利用可能帯域の計測結果 A_{prev} の大きさに応じて以下のように決定する。

$$k = \begin{cases} 2 & (0 \leqq \frac{B_u - B_l}{A_{\mathrm{prev}}} < 0.15) \\ 3 & (0.15 \leqq \frac{B_u - B_l}{A_{\mathrm{prev}}} < 0.2) \\ 4 & (0.2 \leqq \frac{B_u - B_l}{A_{\mathrm{prev}}}) \end{cases} \quad (1)$$

また、探索区間の幅の最小値 B_m は、計測された利用可能帯域 A の 10%とする。計測パケットの大きさは 1500 KBytes とする。

図 5 は、背景トラヒック量を時間によって変動させ、ボトルネックリンクの実際の利用可能帯域 (A-bw) が 0 sec から 50 sec までが 60 Mbps、50 sec から 100 sec までが 40 Mbps、100 sec から 150 sec までが 60 Mbps、150 sec から 200 sec までが 20 Mbps、200 sec から 300 sec までが 60 Mbps と変化した時の、利用可能帯域の計測結果 (A) およびその時の探索区間の大きさ (Searching Range) を示している。図 5(a)- 5(c) は、1 つの計測ストリーム内の計測パケット数 N をそれぞれ 3、5、8 とした時の結果である。図から、1 つの計測ストリーム内の計測パケット数が少ない場合には、計測の精度が低下し、利用可能帯域の推定がうまく行えていないことがわかる。またその反面、計測パケット数が多くなると、計測精度が向上しており、急激な利用可能帯域の変化がある場合にも、素早く対応して新たな計測結果を導出している。これは、計測パケットが少なくなると、アルゴリズムのステップ (3) におけるパケット間隔の増加傾向の判断が困難になり、ステップ (4) における適切な小区間の選択が不正確になるためである。しかし、N が 5 以上であれば、利用可能帯域をほぼ正確に計測することができている。N5 から 8 に大きくすることにより計測の精度は向上しているが、提案するインライン計測方式は、計測の精度を向上させるのではなく、計測パケット数をできるだけ少なくし、計測速度を向上させることを目的としているため、この場合においては N=5 と設定することが望ましいと考えられる。しかし、適切な計測パケット数 N は、さまざまなネットワーク状況 (利用可能帯域の大きさ、変動の大きさ、背景トラヒックの性質等) によって変化すると考えられる。適切な N の設定方法に関しては今後の課題としたい。

次に、利用可能帯域の大きさが、図 5 のように急激に変化するのではなく、0 sec から 50 sec までは 60Mbps で、その後 100 sec までに 40 Mbps に減少、150 sec までに 60Mbps に増加、210 sec までに 20Mbps に減少し、270 sec までに 60Mbps に増加し、300 sec までは 60Mbps である場合の計測結果を図 6 に示す。ここでも、1 つの計測ストリーム内の計測パケット数 N をそれぞれ 3、5、8 とした時の結果を示している。この場合においても、N が 3 の場合には正確な計測が行えていないが、N が 5 以上であれば、十分な精度で利用可能帯域を推定できていることがわかる。これらの結果から、提案した計測方式は、利用可能帯域の変化の大きさに関係なく、十分な精度で利用可能帯域を推定できることがわかる。

5. TCP コネクションによるインライン計測

TCP によるデータ転送においては、送信側端末から送信されたデータパケットを受信側端末が受信すると、ACK パケットを送信側端末へ返送する。したがって、データパケットを計測パケット、ACK パケットを返送されてくる計測パケットと見なすことで、TCP コネクションを用いた利用可能帯域の計測を行うことができると考えられる。しかし、2.2 節で述べたように、TCP によるデータ転送方式の性質が原因で、3 章で提案した計測方式をそのまま適用することはできない。そこで本章では、アクティブな TCP コネクションを用いたインライン計測を実現する際に、問題となる点を挙げ、その解決方法に関する指針を示す。

5.1 ウィンドウサイズ

2.2 節で示したように、TCP コネクションが一度に送出できるデータパケット数は、ウィンドウサイズによって制限される。受信側 TCP の受信バッファは十分あると仮定すると、1 ラウンドトリップ時間内に送信することのできるパケット数は、送信側 TCP の輻輳ウィンドウサイズ W によって制限される。したがって、$W < N$ (N は提案方式における 1 つ計測ストリーム内のパケット数) の場合には計測を行うことができない。しかし、4 章で示したシミュレーション結果から、$N = 5$ 以上であれば計測を行うことが可能であることが明らかとなったため、転送データサイズが小さく、ウィンドウサイズがあまり大きくならない場合においても、計測を行うことが可能であると言える。

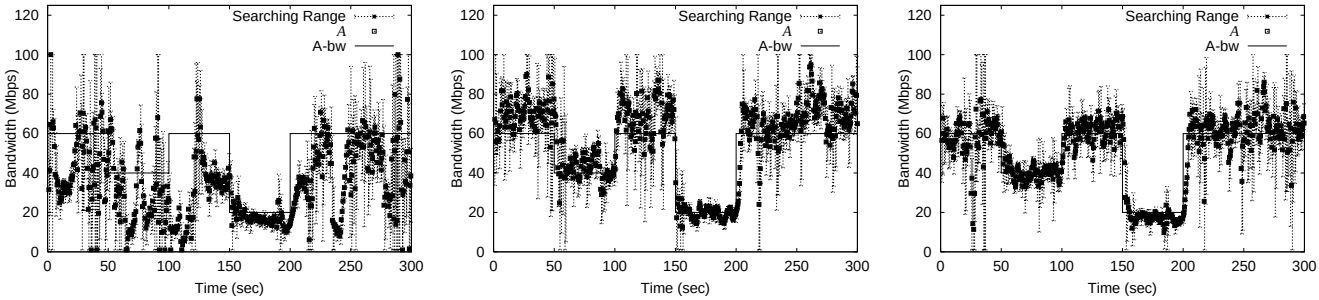
また、$N < W$ の場合、すなわち、ウィンドウサイズが 1 つの計測ストリーム内のパケット数よりも大きい場合には、(1)1 つの計測ストリーム内のパケット数を増加させる、(2) パケット数はそのままで、計測ストリーム数を増加させる、ということが考えられる。(1) は対応する小区間で用いる計測パケット数が増加するため、その小区間の計測精度が向上する。(2) は、1 ラウンドトリップ時間で送信する計測ストリーム数が増加するため、計測速度が向上する。提案するインライン計測方式においては、計測速度の向上を優先させるために、以下のようなアルゴリズムに基づき計測ストリーム数および計測パケット数を決定する。

- W 個のパケットから、それぞれ N 個のパケットから成る $\lfloor W/N \rfloor$ 個の計測ストリームを生成し、1 ラウンドトリップ時間で送出する
- 余った $(W - N \cdot \lfloor W/N \rfloor)$ 個のパケットは、生成した計測ストリームのうち、平均送出レートが最も小さい計測ストリームに加える

5.2 受信側 TCP の Delayed ACK オプション

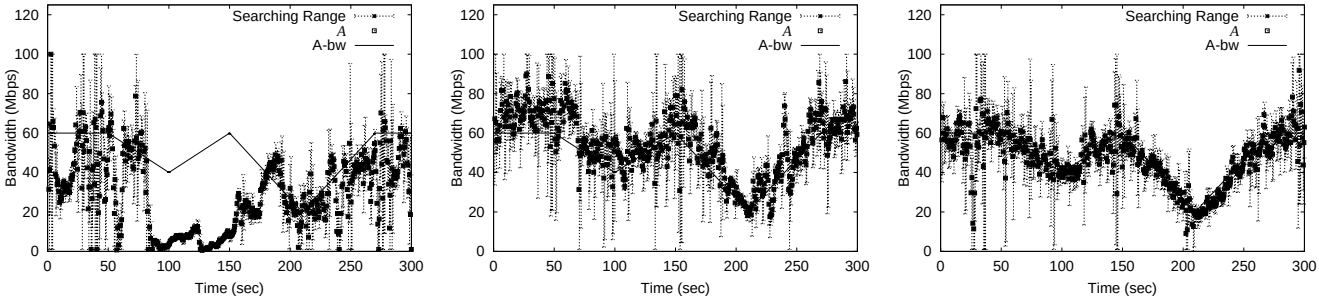
Delayed ACK オプションを用いる受信側 TCP は、1 個のデータパケットを受けとる毎に ACK パケットを送出するのではなく、2 個のデータパケットを受けとる毎に ACK パケットを 1 個送出する。3 章で示したアルゴリズムは、計測パケットが全て返送されることを前提としているため、受信側 TCP が Delayed ACK オプションを用いる場合には、そのまま適用することができない。

この場合には、提案アルゴリズムのステップ (3) で行う計測パケットの送出間隔と到着間隔の比較を、1 パケット毎に行うのではなく、2 パケット毎に行う必要がある。すなわち、計測ストリーム i 内の計測パケット $P_{i,2j'}$、$P_{i,2j'+1}$ $(j' = 1, \ldots, \lfloor N/2 \rfloor)$ に対して、その送出間隔と到着間隔をそれぞれ $(S_{i,2j'+2} - S_{i,2j'})$、$(R_{i,2j'+2} - R_{i,2j'})$ と定義し、パケット間隔の増加の傾向を調べる。しかし、これは、1 つの計測ストリーム内の計測パケット数が N から $\lfloor N/2 \rfloor$ に減少することを示しているため、計測誤差が大きくなる。したがって、N を大きくする、あるいは 1 つの小区間に対して複数の計測ストリームを用いる等の修正が必要になると考えられる。



(a) N=3　(b) N=5　(c) N=8

図 5　シミュレーション結果 (1)



(a) N=3　(b) N=5　(c) N=8

図 6　シミュレーション結果 (2)

5.3　提案方式のパラメータ設定

3 章で提案した計測方式は、探索区間を小区間に分割する際の分割数 k をパラメータとして持つ。4 章におけるシミュレーションでは、式 (1) に示すように、探索区間の大きさに応じて分割数を変化させた。しかし、TCP コネクションによるインライン計測を行う場合は、k は探索区間の大きさだけではなく、TCP コネクションの現在のウィンドウサイズを考慮して決定する必要がある。そこで、ウィンドウサイズ W が大きく、1 ラウンドトリップ時間で送出することのできる計測ストリーム数 $\lfloor W/N \rfloor$ が十分大きい場合には、$k = \lfloor W/N \rfloor$ とする。これにより、探索区間内の小区間数が大きくなるため、計測の精度が向上する。また、1 ラウンドトリップ時間で 1 回の計測が完了するため、ネットワーク状況の変化に対応しやすくなる。

また、探索区間の最小幅 B_m は、4 章のシミュレーションでは計測された利用可能帯域 A の 10%に設定した。しかし、B_m を、十分な計測精度が得られる小区間の大きさと、k の値を基に設定することにより、計測精度が高い場合にはより広い探索区間を用いることが可能となり、大きなネットワークの変動に対応することができる。また、計測精度がわかれば、1 つの計測ストリーム中のパケット数 N を適切な値に設定することも可能となる。計測精度の算出方法としては、提案アルゴリズムのステップ (3) で導出するパケット間隔の増加傾向の安定度を利用することが考えられるが、詳細については今後の課題としたい。

6. おわりに

本稿では、アクティブな TCP コネクションを用いて、データ転送中に得られる情報からエンドホスト間の利用可能帯域を推測するインラインネットワーク計測方式の提案を行った。まず、既存の利用可能帯域計測方式を修正し、少ない計測パケット数で計測を行い、計測の初期段階から継続的に結果を導出する新たな利用可能帯域計測方法の提案を行い、その有効性をシミュレーションによって確認した。また、データ転送中の TCP コネクションを用いて提案方式によるネットワーク計測を行う場合の問題点を明らかにし、それに対する解決策の提案を行った。

今後の課題としては、5 章で述べた、本稿での提案方式をインライン計測に適用する場合に発生する問題点を解決したアルゴリズムの評価を行いたい。また、提案するインライン計測方式をシミュレーションだけではなく、実験ネットワークや実ネットワークを用いて性能評価を行い、その有効性を検証する予定である。

謝　辞

本研究の一部は、総務省戦略的情報通信研究開発推進制度における特定領域重点型研究開発プロジェクト「ユビキタスインターネットのための高位レイヤスイッチング技術の研究開発」、および及び平成 13 年度文部科学省科学研究費奨励研究 (A)(13750349) によっている。ここに記して謝意を表す。

文　献

[1] R. L. Carter and M. E. Crovella, “Measuring bottleneck link speed in packet-switched networks,” Tech. Rep. TR-96-006, Boston University Computer Science Department, Mar. 1999.

[2] B. Melander, M. Bjorkman, and P. Gunningberg, “A new end-to-end probing and analysis method for estimating bandwidth bottlenecks,” in *Proceedings of IEEE GLOBECOM 2000*, Nov. 2000.

[3] M. Jain and C. Dovrolis, “End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput,” in *Proceedings of ACM SIGCOMM 2002*, Aug. 2002.

[4] S. Seshan, M. Stemm, and R. H. Katz, “SPAND: Shared passive network performance discovery,” in *Proceedings of 1st Usenix Symposium on Internet Technologies and Systems (USITS ’97)*, pp. 135–146, Dec. 1997.

[5] K. Lai and M. Baker, “Nettimer: A tool for measuring bottleneck link bandwidth,” in *Proceedings of the USENIX Symposium on Internet Technologies and Systems*, Mar. 2001.

[6] Akamai Home Page, `http://www.akamai.com/`.

[7] Exodus, Home Page, `http://www.exodus.com/`.

[8] R. Tewari, M. Dahlin, H. M. Vin, and J. S. Kay, “Beyond hierarchies: Design considerations for distributed caching on the Internet,” Tech. Rep. TR98-04, Department of Computer Science, University of Texas at Austin, Feb. 1998.

[9] P. R. Constantinos Dovrolis and D. Moore, “What do packet dispersion techniques measure?,” in *Proceedings of IEEE INFOCOM 2001*, pp. 22–26, Apr. 2001.

[10] NS Home Page, `http://www.isi.edu/nsnam/ns/`.