RL-001

FPGA を用いた広帯域高遅延ネットワーク向けの利用可能帯域推定
FPGA-based Available Bandwidth Measurement for Long Fat-pipe Network

小泉 賢一† 稲葉 真理† 平木 敬†
Kenichi Koizumi Mary Inaba Kei Hiraki

1. はじめに

1.1 利用可能帯域推定

コンピュータネットワーク上には多くのトラフィックが流れている。トラフィック量がネットワークの帯域幅を超えると輻輳が発生し、パケットの喪失が発生してしまう。

ネットワーク上でユーザができるだけ高速に通信を行うためには、ユーザが利用可能な帯域を全て使って通信することが理想である。元々ネットワーク上を流れておりそのユーザと少しでもネットワークを共有するようなトラフィックのことを背景トラフィックと呼ぶ。ここで、ユーザが背景トラフィックとの輻輳を発生させることなく通信することができる最大の転送レートのことを利用可能帯域と呼ぶことにする。

近年ネットワークは広帯域化し転送するデータ量が増大している。また通信する地点間が長距離の場合、パケットが送信側によってネットワーク上に送出されてから、受信側によって受け取られるまでにかかる遅延時間が大きい。このような場合、輻輳によるパケット喪失のペナルティが大きくなる。そのため、コンピュータネットワークにおいて自分の利用可能な帯域を推定することが非常に重要である。

あるネットワークパス P において、送信者 Sender と受信者 Receiver が存在するとする。Sender から Receiver までに存在するスイッチやルータなどの数を N とすると、これらを結ぶリンク L_i (i = 1, 2, …, N+1)はそれぞれ静的な帯域幅 C_i を持っている。このとき、Cj = min{C_1, C_2, …, C_{N+1}}のことをボトルネック帯域と呼び、L_j をボトルネックリンクと呼ぶ。また、これらのリンクにそれぞれ B_i (i = 1, 2, …, N+1)の背景トラフィックが流れているとすると、そのときに各リンクにおける利用可能な帯域は

$$A_i = C_i - B_i$$

と表される。このとき、A_k = min{A_1, A_2, …, A_{N+1}}をネットワーク P において利用可能な帯域と定義し、リンク L_k を利用可能帯域最小リンクとする。さらに L_k の両隣のスイッチのうち Sender 側のスイッチやルータのことを飽和スイッチと呼ぶことにする。本論文では C_i の値は既知であるものとして、そこから A_k を推定する手法を対象とする。

1.2 利用可能帯域推定手法

これまで様々な利用可能帯域を推定する手法が提案されてきた。多くの手法は、プローブパケットと呼ばれる複数のパケットをネットワーク上に流し、受信側で受け取ったプローブパケットの振る舞いから利用可能帯域の推定を行っている。飽和スイッチ中のバッファによるパケットのキューイングによりスイッチに到達する前のパケットとスイッチから放出されたパケットのギャップは異なる。このパケットギャップの値から利用可能帯域の推定値を計算する。実際のプローブパケットはごく微量であるため、スイッチは一時的に飽和状態になるが、輻輳は発生しない。一般的に、隣り合う二つのプローブパケット（これはパケットペアと呼ばれている）のうち、先行のパケットの末尾から後続のパケットの先頭までの差をインターパケットギャップと呼ぶ。本論文ではこれと区別して、パケットペアの先頭同士の差分のことをパケットギャップと呼ぶことにする。推定手法の中には連続した三個以上のパケットのギャップを計測するものがあり、ここではこれらのことをパケット列と呼ぶことにする。1 回の計測のために、異なるパケット長やパケットギャップを持つ複数のパケット列を使用する方法が用いられている。

Spruce [1]では、飽和スイッチがネットワークパス上に一つしかなく、プローブパケットがスイッチを通過する間はスイッチのキューは常に空ではないことを仮定している。背景トラフィックの転送レートが R_B であるとする。Sender 側からパケットギャップ Δ_{in} でパケット長 L のパケットペアが飽和スイッチに到達するとする。このときプローブパケット 1 個がスイッチに到達する間に背景トラフィックは $B \times \Delta_{in}$ だけ飽和スイッチに到達する。スイッチは単純な FIFO のため、スイッチの出力側におけるプローブパケットギャップは $B \times \Delta_{in} + L$ となる。ゆえに飽和スイッチの直後で計測されたパケットギャップを Δ_{out} とおくと、

$$C \times \Delta_{out} = B \times \Delta_{in} + L$$

ゆえに利用可能帯域 A は

$$A = C - B = C \times \left(\frac{\Delta_{in} - \Delta_{out} + \frac{L}{C}}{\Delta_{in}} \right)$$

と求められる。

別に手法に Pathload [2]がある。Pathload では、Sender と Receiver でのプローブパケットの転送レート R_S と R_R を計測する。$R_S \geqq R_R$ のとき $R_S \geqq A$ であり、$R_S < R_R$ のとき $R_S < A$ であることを用いて A を求める。また、Pathload を改良した Yaz [3]という手法が提案されている。Spruce と

† 東京大学大学院情報理工学系研究科 Graduate school of Information Science and Technology, University of Tokyo

Pathload は、飽和スイッチの前後でパケットギャップが変化することを利用している点で共通している。

また実際にストリームをネットワークに流してみてそのストリーム自体の転送レートを計測する手法（Iperf など）が存在するが、本論文ではネットワークにできるだけ影響を及ぼさない推定手法を対象としているため、Iperf については対象外とし、パケット生成ツールとしてのみ捉える。

多くの利用可能帯域推定手法は、飽和スイッチがネットワーク上に一つしか存在しないことを仮定しているため、実際のネットワークに適用すると推定値に誤差が生じてしまう。また、プローブパケット自体がネットワークに与える影響が誤差の一因となっており、未だ十分な誤差性能を持つ利用可能帯域推定手法は発見されていない。

1.3 背景トラフィックのバースト性

現在、広く TCP 通信が使用されている。TCP では輻輳ウィンドウを用意し、受信側からの確認パケット(ACK)がくるまで後続のパケットの送出を待ち、ACK パケットが来てからパケットを送出する。高遅延ネットワークにおいてはパケットの往復遅延時間(RTT: Round Trip Time)が大きいため、ACK パケットが送信側に帰ってくるたびに輻輳ウィンドウサイズのデータを一度に送出する状況が発生する。そのため高遅延ネットワークを流れる TCP ストリームは図 1 のようにバースト性を持つ。図 1 は RTT が 500 ミリ秒のネットワークを流れる TCP ストリームの様子であり、500 ミリ秒ごとにワイヤレートでパケットが流れていることがわかる。

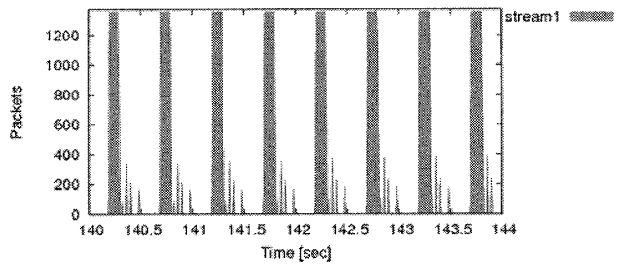

図 1　高遅延ネットワーク上の
TCP ストリームのバースト性

Spruce は背景トラフィックが流体モデルであると仮定している。しかし実際の高遅延ネットワークにおいてはバーストトラフィックが発生しており、推定値がずれてしまう。プローブパケットに長いパケット長を使用したりパケット列を長くしたりすることによってずれが緩和されることがあるが、完全な解決には至っていない [4, 5]。

そこで我々は、背景トラフィックが(1)バーストのあるトラフィックと、(2)バーストのないトラフィックの 2 つの重ね合わせで構成されるとして、それぞれに対し独立に既存手法である Spruce を適用したときの振る舞いを調べた。

本論文では第 2 章で予備実験の結果について述べ既存手法の適用における問題点を指摘する。そして第 3 章で我々を使用するパケットギャップの計測手法について述べ、第 4 章で実験環境、第 5 章で既存手法を適用した時の結果を示し、第 6 章で結論を述べる。

2. 予備実験

2.1 ネットワークスイッチの複雑性

従来の利用可能帯域測定手法では、ネットワーク上のスイッチやルータは FIFO としてモデル化されている。FIFO モデルでは、複数の入力ポートからスイッチに到達したパケットは、それらの行き先が同じである場合は同じキューに蓄えられ、特に優先度が設定されていない場合は、キューに蓄えられた時刻の早いパケットから順番に出力ポートから放出されていく。あるパケットがスイッチに到達してから放出されるまでの時間を T とする。このようなモデルにおいては、対象となるスイッチに背景トラフィックが流れているときのパケットの遅延時間 T は背景トラフィックが流れていない時よりも長くなるはずである。一般に、飽和スイッチにおいては図 2 のように背景トラフィックを含む複数のストリームが入力されている。

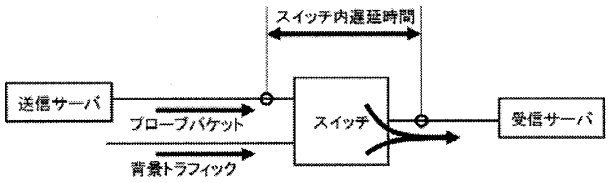

図 2　予備実験ネットワーク図

図 2 のようなネットワークにおいて、パケットの遅延時間を計測した結果が図 3 である。図 3 において、最初の 30 秒は背景トラフィックが流れておらず、30 秒から 50 秒は背景トラフィックとして 3.0Gbps の UDP ストリームが流

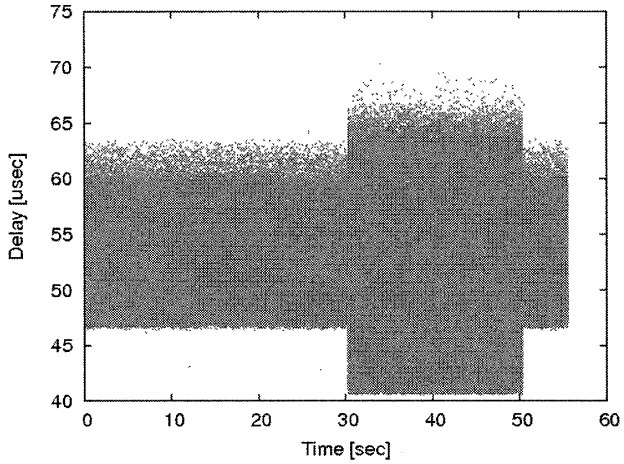

図 3　背景トラフィックの有無による
スイッチ内遅延時間の変化

れている。背景トラフィックが流れていない時の遅延時間は 46 マイクロ秒から 63 マイクロ秒であり、背景トラフ

ィックが流れているときの遅延時間は 41 マイクロ秒から 69 マイクロ秒である。このことからスイッチによるパケットスケジューリングが単純な FIFO モデルではないことがわかる。このようなスイッチ内部のスケジューリングの複雑性が推定値の誤差の原因の一つになっている。そのため、遅延時間変化の粒度に適切な計測手法が必要である。

2.2 プローブパケットのパケットギャップ精度

従来の利用可能帯域推定手法では、サーバ上で推定アプリケーションを実行し、アプリケーションや OS のカーネル内でプローブパケットを生成する。

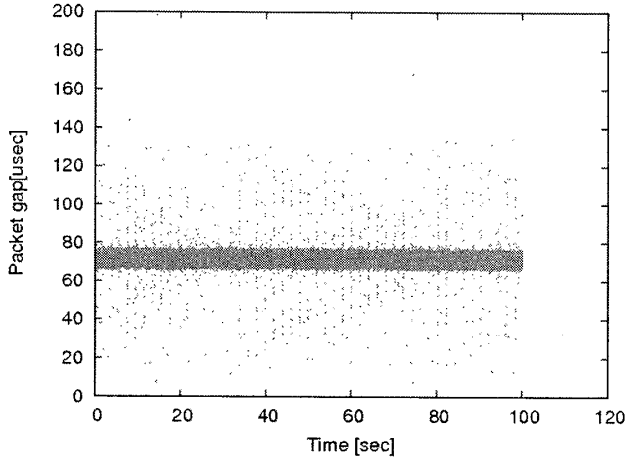

図 4　Iperf UDP ストリームのパケットギャップ

図 4 はネットワーク帯域測定ツール Iperf において、1.0Gbps の UDP ストリームを生成したときのパケットギャップを測定した結果である。パケットサイズ 9142Bytes より、パケットギャップは平均 73 マイクロ秒になるはずであるが、実際にはばらつきがあることがわかる。これに対し、ハードウェアレベルでパケットギャップを調整したときの結果を図 5 に示す。

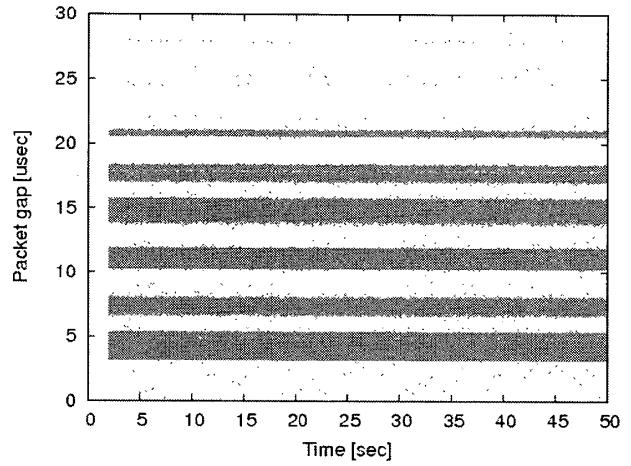

図 5　NIC のインターパケットギャップ制御による Iperf TCP ストリームのパケットギャップ

図 5 の実験では RTT が 20 ミリ秒であり、ネットワークインターフェースカード(NIC) として Chelsio S310E-SR を使用している。S310E-SR には NIC の最大転送レートを制限するための IPG 制御機能があり、図 5 ではパケットギャップの最小値が 4.9 マイクロ秒となるように設定している。パケットギャップの下限が調整されていることがわかるが、4.9 マイクロ秒を下回るパケットペアが観察されている。

サーバ上でプローブパケットを生成する際にはコンテキストスイッチによる割り込みによってプローブパケットの送出が不連続になることがある [2]。[3]において、一回の計測で 20 ストリームのプローブパケットを使用すれば、コンテキストスイッチが推定値に与える影響を緩和できると言われているが、プローブパケットが背景トラフィックに与える影響を最低限に抑え、計測時間を短縮するためにはできるだけ少ないストリーム数で計測できることが望ましい。

推定値の計算には、精密なプローブパケットを生成するためにハードウェアによる生成あるいは調整が必要不可欠である。しかし S310E-SR においては調整できるパケットギャップに制限があり、様々なパケット長で様々なパケットギャップを持つプローブパケットを柔軟に生成することが困難である。

2.3 パケットのタイムスタンプ計測精度

ネットワークパス上の異なる二地点間（例えば送信側と受信側や、飽和スイッチに到達する直前と直後）でのパケットの遅延時間を計測したり、ある一点を通過する時のパケットギャップを計測したりするため、TCP Timestamps Option [6]と GPS などの送信側と受信側の時刻同期を組み合わせた手法や、Endace DAG Card [7]や TAPEE [8]などの専用ハードウェアによるパケットキャプチャ手法が提案されている。

例えば、10 ギガビット/秒環境において 1500 バイトのパケットは 1.2 マイクロ秒で処理される。このためタイムスタンプの計測には 10〜100 ナノ秒の精度が必要である。

3. 提案

第 2 章で見たように、帯域推定においてプローブパケットを使用する際には、プローブパケットの生成・測定精度が重要である。しかし一般に行われている、送信サーバ上の OS カーネルや NIC で生成する手法では、正確なパケットギャップを持つプローブパケットを作成することが困難である。そこで我々は、プローブパケットを FPGA 基板上で作成し、さらにパケットギャップの計測を FPGA 基板を使用して行うことを提案する。

また、帯域推定には測定結果から推定値を計算するアルゴリズムのパラメータの最適化が重要である [3]。そのため再構成可能なハードウェアを用いてプローブパケットを生成することは有効である。

4. 実装

4.1 ネットワーク FPGA 基板

高精度なプローブパケットの生成や、プローブパケットのタイムスタンプを正確に計測するため、図 6 に示すネットワークテストベッド MaSTER-1 を使用した。MaSTER-1 には 10GBASE-SR/LR のポートが 5 つあり、それぞれのポートに FPGA(XC5VFX70T-1FF1136)が一つずつ接続されている。FPGA は相互に接続されており、通信を行うことが可能である。

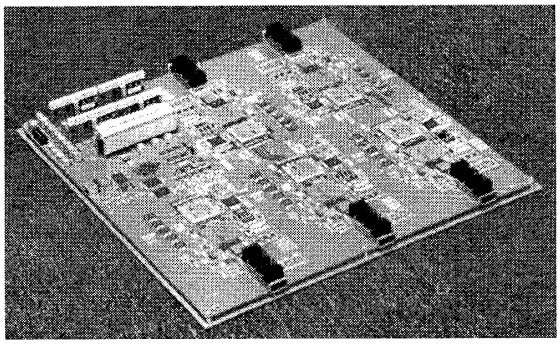

図 6　MaSTER-1

4.2 プローブパケット生成器

図 7 にプローブパケットの構成図を示す。MaSTER-1 上に存在する 10 ギガビットポート 1 つと、対応する FPGA1 つにパケット生成器用のファームウェアを実装した。パケット生成回路において生成されたパケットは、帯域推定のためにあらゆるパケットギャップ値に設定され、FPGA より MAC PHY と Serdes を経由して XFP 光モジュールより送出される。

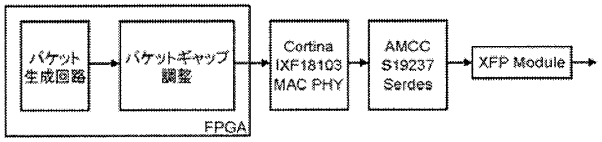

図 7　プローブパケット生成器

4.3 パケットキャプチャ

図 8 にパケットキャプチャの構成図を示す。パケットのタイムスタンプを計測する場合は、計測対象のパケットを光タップで複製し、MaSTER-1 の光モジュールの受信側に入力する。入力されたパケットは光モジュールに接続されている FPGA に転送される。その FPGA 上に実装された回路においてパケットの到着時刻を計測し、計測情報を光モジュールから送出する。送出された計測情報はサーバにて集約させる。

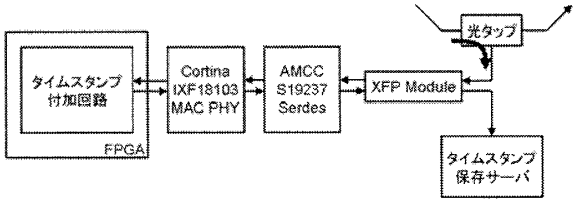

図 8　パケットキャプチャと保存用サーバ

5. 実験環境

利用可能帯域推定手法を評価するための実験環境には大きく分けて 3 通りの方法がある。一つ目は ns-2 [9]などのネットワークシミュレータを用いる方法であり、二つ目は小規模なプライベートネットワークを構成する方法であり、三つ目は実際に運用されているネットワーク回線を用いる方法である。ネットワークシミュレータを用いる手法は任意のネットワークトポロジを構成でき、実験に再現性があるが、ネットワーク機器や背景トラフィックのモデルに限界があり、現実のネットワークのシミュレーションができているとは言えない。実際のネットワーク回線を用いる手法は、背景トラフィックに再現性が無く、推定手法を評価するための環境とすることは難しい。そこで我々は、遅延エミュレータを使用して小規模な擬似高遅延ネットワークを構成し、実験を行うことにした。

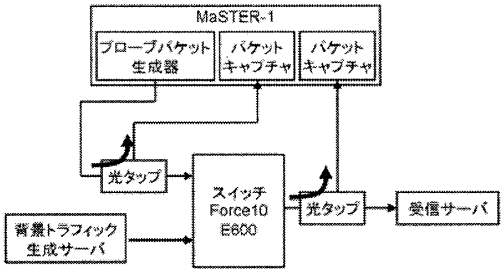

図 9　実験用ネットワーク図

表 1　Iperf のパラメータ

送信側	-c *Recv_host* –i 1 –t 100 –l 192k –w *Cwnd_size* – M 8000
受信側	-s –i 1 –w 600M –l 192k

MaSTER-1 を用いて、図 9 のような実験用ネットワークを構築した。背景トラフィック生成サーバ(Intel Xeon 5160 3.00GHz, DDR2 SDRAM 4GB, Chelsio Communications S310E-SR NIC, Linux 2.6.18 Kernel)上で Iperf を表 1 のパラメータで実行し、ネットワークスイッチ Force10 E600 [10]に入力した。プローブパケットは MaSTER-1 上の FPGA の一つで生成し(フレーム長 2960 オクテット、1000 フレーム)、光タップを通してスイッチに入力した。プローブパケットがスイッチに入る手前と、出てきた後のところに光タップを挟み、プローブパケットをキャプチャした。キャプチャされたプローブパケットは FPGA 上でタイムスタンプを記録する。

6. 実験結果

第 1 章で示したように、背景トラフィックの中にはバースト性を持つものが存在しており、推定値の誤差の原因となっている。そこで我々は背景トラフィックを、バースト性を持たないストリーム(図 10(a))とバースト性を持つストリーム(図 10(b))に分け、それぞれの背景トラフィックに対して既存の帯域推定手法である Spruce を適用し、実際の利用可能帯域との比較を行った(図 11)。バースト性を持つトラフィックは iperf の *Cwnd_size* を 10M から 150M の範囲で調整することによって生成した。赤いプロットがバースト性の無いもので、緑のプロットがバースト性のある時のものである。背景トラフィックにバースト性が無く、利用可能帯域が 5Gbps 以下である場合の Spruce による推定値は実際と非常に近い値を示しているが、5Gbps 以上の場合は正確に推定できていない。これは生成したプローブパケットのパケットギャップが現在の環境にとって不適切な値であるためである。一方、背景トラフィックにバースト性がある場合は、ほとんどの帯域で正しい値を推定できていない。

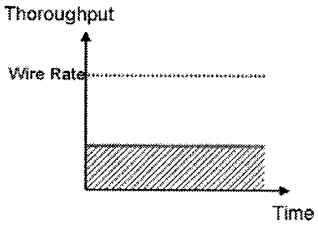

図 10(a)　バースト性を持たないストリーム

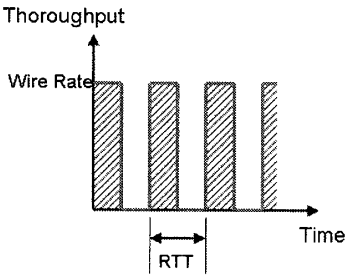

図 10(b)　バースト性を持つストリーム

7. 結論

利用可能帯域推定手法にはネットワーク環境や計測精度などに多くの困難な点が存在する。従来手法では一般的に、ネットワークスイッチが単純な FIFO モデルであったり、背景トラフィックが流体モデルであったりといった仮定を行っていた。本論文において、我々は既存の利用可能帯域推定手法の問題点と広帯域高遅延ネットワーク特有の問題点を列挙し、正確な計測手法の提案を行った。その上で既存の帯域推定手法である Spruce を広帯域高遅延ネットワークに適用した結果、バーストのない背景トラフィックに対しては正確な値を推定することができたが、バーストのあるトラフィックに対しては正しい値を推定することができなかった。これは広帯域高遅延ネットワークにおける利用可能帯域推定手法にはバーストのある背景トラフィックに対して正確な推定値を出力できるような改良が必要であることを意味している。今後、改良を重ね、あらゆる性質の背景トラフィックに対する推定値の精度向上を目指す。

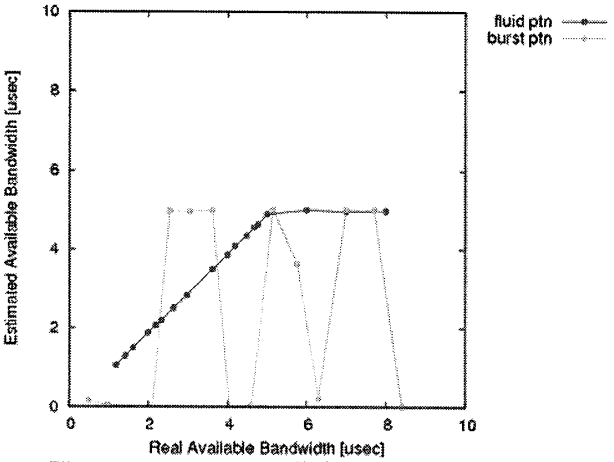

図 11　Spruce によって推定した利用可能帯域値と実際の値

参考文献

[1]Jacob Strauss et al., “A measurement study of available bandwidth estimation tools”, *the 3rd ACM SIGCOMM Conference on internet Measurement* (2003).
[2]M. Jain et al., “Pathload: A measurement tool for end-to-end available bandwidth”, Passive and Active Measurements (PAM) Workshop (2002).
[3]Joel Sommers et al., “A Proposed Framework for Calibration of Available Bandwidth Estimation Tools”, the 11th IEEE Symposium on Computers and Communications (2006).
[4]Xiliang Liu et al., “Single-hop probing asymptotics in available bandwidth estimation: sample-path analysis”, the 4th ACM SIGCOMM conference on Internet measurement (2004).
[5]Xiliang Liu et al., “What signals do packet-pair dispersions carry?”, IEEE INFOCOM (2005).
[6]V. Jacobson et al, “RFC 1323: TCP Extensions for High Performance”, (1992).
[7]Endace, “Endace DAG - 100% packet capture”, http://www.endace.com/endace-dag-high-speed-packet-capture-cards.html.
[8]Takeshi Yoshino et al, “Performance optimization of TCP/IP over 10 gigabit ethernet by precise instrumentation”, the 2008 ACM/IEEE conference on Supercomputing (2008).
[9] “The Network Simulator - ns-2”, http://www.isi.edu/nsnam/ns/.
[10] FORCE10. http://www.force10networks.com/products/.