

高安定PCを用いたネットワーク計測

北口 善明^{†,††,†††} 町澤 朗彦^{††} 箱崎 勝也^{†††} 中川 晋一^{††}

† 通信・放送機構 小金井分室
†† 独立行政法人 通信総合研究所
††† 電気通信大学

〒184-8795 東京都小金井市貫井北町4-2-1
通信・放送機構 小金井分室
E-mail: kita@netarc.tao.go.jp

あらまし 近年のネットワークはその需要に伴い、回線容量が大幅に増加している。アクセス網での帯域がADSL等により数Mbpsとなっている現在では、バックボーンネットワークが10Gbpsを越えるようになっている。このような高速なネットワークになってくると、そのネットワーク計測の際に計測機器の時刻精度が重要になってくる。現在の一般的に利用されているPC(AT互換機)では、その時刻精度は内部に利用している水晶発振子の性能上毎秒10マイクロ秒程度のゆれを持っている。この精度だと上述した高速ネットワークでの帯域等の計測を行うためには十分とは言えない。本研究では、内部の水晶発振子の代わりに外部からの10MHz信号を利用して駆動可能にした高安定PCを用い、ネットワーク計測を実施した。計測には現在広く利用されているツール等を利用し、その有用性を示す。

キーワード ネットワーク計測, 高速ネットワーク, 高精度時刻

Network measurement with a high accurate clock installed PC

Yoshiaki KITAGUCHI^{†,††,†††}, Akihiko MACHIZAWA^{††}, Katsuya HAKOZAKI^{†††}, and Shin-ichi
NAKAGAWA^{††}

† Telecommunications Advancement Organization of Japan
†† Communications Research Laboratory
††† University of Electro-Communications

Communications Research Laboratory,
Nukui-kitamachi 4-2-1, Koganei-shi, Tokyo, 184-8795 Japan
E-mail: kita@netarc.tao.go.jp

Abstract With its increasing demand, today's network is facing to deal with a large amount of circuit capacity. Since bandwidth of access networks becomes a few Mbps now with ADSL networks and others, that of backbone networks exceeds 10 Gbps. For that kind of high-speed networks, the time granularity of measuring instruments matters while measuring the networks. The time granularity of today's generally used PCs (AT compatible machine) has its jitter at about 10 microseconds depending on performance of a crystal oscillator inside the PC. This degree of accuracy would not be enough to measure the bandwidth on high-speed networks mentioned above. We used high-accurate servers that could be enabled on an external signal at 10MHz instead of an internal crystal oscillator to measure the network. The tools that are presently widespread are used to measure the network and prove their usefulness in the paper.

Key words Network measurement, broadband network, high accurate time

1. はじめに

インターネットにおいて利用されるアプリケーションは近年多様化が進み、動画像などのストリーミングによる広帯域通信によるネットワークの大容量化が進んでいる。これに伴い、エンドユーザへのアクセス網がADSL等による数Mbpsから光ファイバー接続による100Mbpsまでになっており、現在ではバックボーン回線の容量も10Gbpsを越える必要があるようになった。

また、インターネットが社会経済でのインフラストラクチャとして利用されるようになり、ネットワーク利用時のサービス保証のためにQoSの要求も強まっている。QoSでは現在利用可能な資源の情報を正確に把握し、経路の選択やフローコントロールを実施しなければならない。その時必要となるパラメータとして、ネットワークの帯域や遅延のゆれ（ジッタ）の情報が挙げられ、それらを正確に計測することは重要である。そのため、今後は高速ネットワークの計測は不可欠なものになると予測され、その計測を行うためにはマイクロ秒以下のオーダーで時刻の精度が必要とされる。

一方、現在一般的に利用されているPC（AT互換機）では、自身の時間保持のために水晶発振子による内部時計を持っている。PCも利用ソフトウェアの高度化などに伴い性能を格段に進歩させてきており、その性能は数年前のスーパーコンピュータに匹敵するものとなっているが、PCの内部時計に利用している水晶発振子の精度はそれほど変化しておらず、毎秒10マイクロ秒程度のゆれを持っている。この時刻精度では上述したような高速ネットワークにおいてネットワーク計測を行う場合に問題となる。たとえばガガビットイーサネットでの500バイトのパケットの転送時間は4マイクロ秒と非常に小さく、時刻精度以下になってしまい、現状のPCを計測器として使うには十分でないと言える。さらに今後現在以上の高速ネットワークが登場した場合も考えた計測の方法や必要な時刻精度の調査は重要である。

本稿では、PCにて高速ネットワークでの十分な計測を行えるように、PCのマザーボード上の水晶発振子を置き換え、外部のセシウム原子時計からの10MHz信号を入力源として利用可能にした高安定PCの開発を行い、この高安定PCを用いてネットワークの計測を行った結果について報告する。

2. 高安定PCの開発

我々は、PCの時刻安定度の向上のための研究を進めた結果、内部時計の精度による影響が一番大きいと結論し、その改善を実施した。PC内部時計の14.318MHzの基準信号発信源として用いられている水晶発振子は、現在そのほとんどが ± 20 ppmの精度しかなく、これは一日に約1.8秒もの誤差を生じることの意味し、短期安定度、長期安定度ともに性能が低いものである。そこでこの水晶発振子をより精度の良いものに置き換える方法を検討し、その結果、外部からの10MHz信号を水晶発振子の代わりに利用する方法を取り、PCで利用できるデバイスの開発を行った[1]。このデバイスは、外部からの10MHz信号

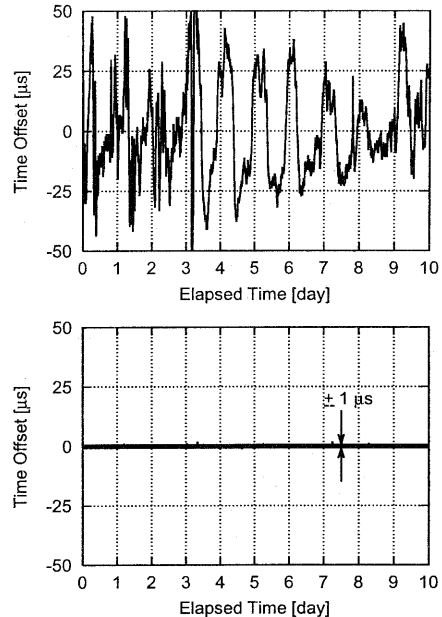


図1 NTP Stratum 1サーバでの時刻精度
上：一般的な内部水晶発振子を用いた場合
下：セシウム原子時計信号を利用した場合

を水晶発振子と同じ14.318MHzに変換して入力できるもので、10MHz基準信号を発生するいかなるものでも利用できる特長を持っている。

このデバイスを用い、外部信号としてセシウム原子時計を用いたPCを高安定PCとし、その時刻安定度を、従来のPCの場合と比較した。その比較方法として、高安定PCと従来のPCとを併にNTPサーバのStratum1として運用し、その時間のゆれを求める方法を取った。図1にそれぞれの時間のゆれの様子を計測した結果を示す。この値はレファレンスとしているGPS時刻と約30秒毎に時刻補正を行った値の変化を表している。この図より補正のたびに従来のPCにて数十マイクロ秒もの時刻の変動が観測されているのに対し、セシウム原子時計の信号を利用した高精度PCではその変動は1マイクロ秒以下で安定している。この変化の値を累積値として表現したものが図2となり、従来のPCでは数ミリ秒もの時刻変化があることがわかる。さらに利用したLinux OSの時刻精度の向上を図り高精度PCの時刻安定度を計測したところ、二日間の標準偏差として時刻のゆれが200ナノ秒と得ることができている[2]。このように開発した高安定PCでは1マイクロ秒以下のジッタにて伝送遅延が計測可能であり[3]、この高安定PCを用い実際に計測を行いその有効性を検証することは重要である。

今回の計測ではネットワークの計測対象として帯域を取り上げた。帯域はQoS制御においても重要なパラメータの1つであり、精度良く計測されることが望まれている。次章では、ネッ

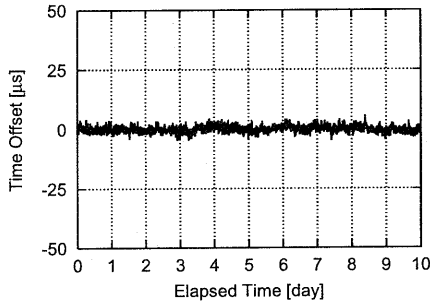
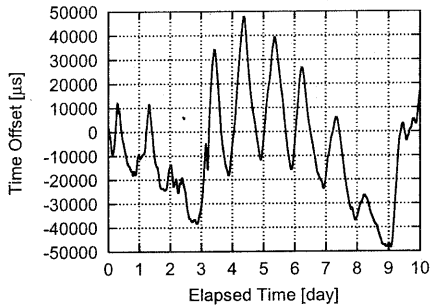


図2 NTP Stratum 1サーバでの時刻累積値
上：一般的な内部水晶発振器を用いた場合
下：セシウム原子時計信号を利用した場合

ネットワークの帯域計測に用いられる計測手法について、利点と問題点についてまとめる。

3. ネットワーク帯域計測手法

ネットワークの帯域を計測する手段としてこれまでに多くの研究がなされ、それに伴うアプリケーションも数多く開発され広く利用されている。中でも One Packet 計測手法と Packet Pair 計測手法が有名で、以下にそれぞれの手法についてまとめる。

3.1 One Packet 計測手法

Bellovin [4] や Jacobson [5] による手法で、サイズを様々な変化した単一の計測パケットを用い、その転送遅延時間 (RTT) を計測し、各パケットサイズでの最小 RTT を求めその回帰直線から通過したリンクの帯域を推定する。計測した RTT にはパケットがネットワークに送出される時間 (1)、パケットがネットワークを通過する時間 (2)、計測対象にて返答処理が行われる時間 (3)、返信パケットがネットワークを通過する時間 (4)、そして返信パケットを受信する時間 (5) が含まれる。計測時に変化するのは送信パケットのサイズ変化による (1) と (2) の時間であり、他は一定の値 (ネットワークの反応速度：レイテンシなど) として計算できるが、実際にはシステムの実装により (3) は一定と考えることができない場合もある (付録参照)。実際に図3のように、各パケットサイズでの最小 RTT を結ぶ直線が式 1 のような一次式で求められ、その直線の傾き

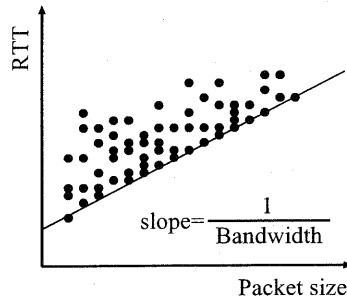


図3 One Packet 計測手法での帯域推定方法

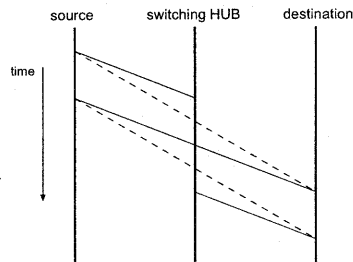


図4 One Packet 手法での問題点

の逆数が計測対象リンクの帯域として得られる。

$$RTT = \frac{PacketSize}{Bandwidth} + latency + RTT^{ICMP} \quad (1)$$

このモデルを利用した計測ツールとしては、pathchar [6]、pchar [7]、clink [8] などが広く利用されている。これらのツールの利点としては、計測方法として UDP パケットを送信し ICMP エラー通知を利用していることによる計測対象地点に特別なソフトウェアを用意する必要がない点であり、経路の各リンクをそれぞれ計測可能である。またこのモデルでは、RTT がパケットサイズに関して線形であることが求められる。それ故、経路内にスイッチングハブのようなブリッジノードなどの IP 的に見えないノードがある場合などでは、データ転送中にデータコピーが多数発生することになり、リンクの帯域計算を難しくするという問題を持っている。たとえば経路内にスイッチングハブが 1 つ存在した場合、図4のように実線のスルーポイントが見かけ上点線のように半分値として求められることになる [9]。

3.2 Packet Pair 計測手法

Paxson [10] や Jacobson [5]、Dovrolis [11] による手法で、ある計測区間のボトルネック帯域の計測を行うために、二つの近接したパケット (パケットペア) を用い、到着したパケットペアの時間間隔からボトルネック帯域の推定を行う。図5にパケットペア計測手法の概念を示す。計測元からサイズ S のパケットペアを計測対象に向け送信する。経路の帯域が小さくなる場合は図のようにパケットがキューから送出される時間が延びることになる。この時のパケット時間間隔はボトルネック部の帯域 B とパケットサイズから S/B となる。この影響はその後の

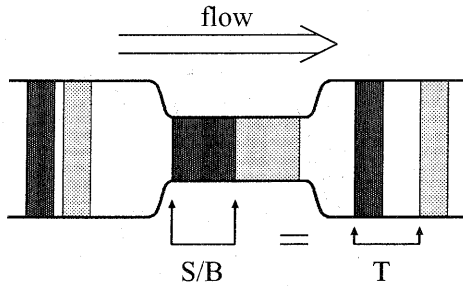


図5 Packet Pair計測手法の概念図

表1 計測に使用した高安定PCの仕様

OS	RedHat Linux 7.2
Motherboard	MSI MS-6351 (Ver5)
CPU	Pentium III 996.68MHz
Memory	512MB
NIC	Intel Pro/1000F Server Adapter Intel 82562EM (On Board)

リンクを通過する際にパケットペアの間隔として残り、パケットペアの到着時間間隔 T と等しくなる。よって計測対象におけるパケットペアの到着時間間隔を計測することで、ボトルネック部分の帯域 B が求まる。このモデルを利用した計測ツールにはBprobe[12]やnettimer[13]、pathrate[14]が挙げられる。

このモデルの特長としては、計測に利用するパケットの量が非常に少なくすむ点である。One Packet計測手法ではサイズを変化させたパケットをそれぞれのサイズで数回送信し、最小RTTを求める必要があったのに対し、Packet Pair計測手法では最良の場合でパケット2つで計測が可能となる。しかし、この計測ではパケットペアの間に別のパケットが入ることなく終点到達することが必要となるため、別のトラフィックがあるネットワークでの計測ではこの点を考慮した計測が必要となる。また、転送ノードのキューが複数ある場合にもパケットペア間に影響がでるため、計測が難しくなる。

4. 計 測

3章でまとめたような計測手法を利用し、高安定PCでのネットワーク計測を実施した。今回計測に利用したPCのスペックを表1にまとめる。

4.1 One Packet計測手法を用いた計測

まずOne Packet計測手法での計測を実施した。この計測にはclinkを用い、二台のPC間の帯域計測を実施した。利用したインターフェイスはファーストイーサネットとギガビットイーサネットの二種類で10Mbps、100Mbps、1Gbpsの各帯域の計測を行った。10Mbpsの場合のみ間にダムHUBを挟み、他はクロスケーブルで接続して計測を行った。計測時の構成図を図6に示す。計測値は10回計測した平均として求め、通常のPCでの計測と高安定PCによる計測結果をそれぞれ表2にまとめる。全ての帯域の計測において、従来PCと高精度PCでは同

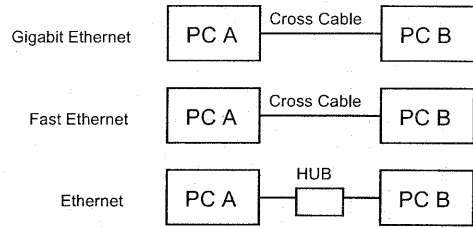


図6 ネットワーク計測時の構成

じ値の結果が得られたが、10回の計測結果の分散を求めると表2のように高精度PCのほうが分散の値がどの計測においても良いことがわかる。また、ギガビットイーサネットの計測結果が低いのは、計測対象PCのICMP処理による影響かと考えられる。

さらに精度の良い計測を目指し、その改善方法として時刻情報を取得している部分に注目し、その調査を実施した。一般的に時刻を呼び出すためのシステムコールとしてgettimeofday(2)が利用されるが、その呼び出しにかかる時間コストを計測した。計測はgettimeofdayを連続して実行した際に得られるそれぞれの時刻情報の時間差を求めることで行った。その結果、gettimeofday実行には計測に用いたPCでは平均9マイクロ秒のコストがかかっていることがわかった。ネットワークの計測に必要なのは絶対時間ではなく時間差であるため、gettimeofdayのように日時までわかる必要はない。そこで、CPUのクロックカウンタを利用する方法を試みた。Pentium以降のIntel CPUでは、PCの起動時からのCPUのクロック数をカウントする64ビットのカウンタ(TSC: TimeStamp Counter)を持っており、RDTSC命令で簡単に読みだすことが可能となっている。この呼び出しにかかる時間もgettimeofdayのときと同様な方法で計測してみたところ平均52クロックという結果を計測用PCにて得ることができた。このTSCを用いた場合、最終的に計測時間を求めるためには利用しているCPUの周波数を求め、クロック数を時間に直す必要がある。そこで、計測に使用した高安定PCをNTPサーバとして時刻同期を行い、周期的にTSCと時刻情報を取得しその変動を観測した。10日間における観測結果を図9に示す。これからCPU周波数を996677116Hzと求め、クロックカウンタ数から経過時間を式2のように求めている。

$$tureRTT = \frac{MeasuredRTT}{0.996677116} \quad (2)$$

これにより、RDTSC命令に用いられる時間は約52ナノ秒と求まり、gettimeofdayと比べはるかに時間コストが低いことがわかる。この時間情報の読み出しによる影響を計測するため、clinkにてTSCを利用する改造を行い、10Mbps、100Mbps、1Gbpsそれぞれの場合の計測を実施し、結果を表2に追記する。計測値は他の場合と同様に10回の平均値で求めている。このTSCを用いた計測方法では10Mbpsの計測以外で、さらに分散の値が小さく計測精度が向上していることがわかる。

4.2 Packet Pair計測手法を用いた計測

次にPacket Pair計測手法を用いた計測を実施した。計測に

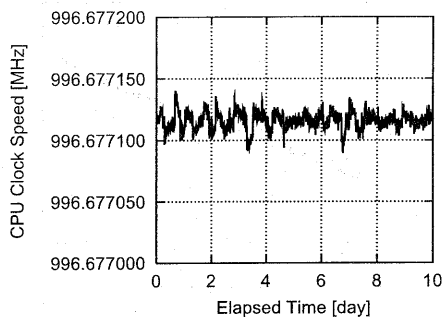


図7 高安定PCのCPU周波数の変動

表2 clinkを用いた計測結果
上：平均値 下：分散

	10Mbps	100Mbps	1Gbps
Normal PC	9.948 Mbps 1.4×10^{-5}	94.148 Mbps 0.137	359.408 Mbps 39.8
Cesium PC	9.942 Mbps 5.29×10^{-6}	93.939 Mbps 0.092	356.155 Mbps 21.48
Using TSC	9.997 Mbps 1.9×10^{-5}	94.323 Mbps 0.069	360.319 Mbps 5.594

はまず pathrate を利用し、機器の構成は One Packet 計測時と同様にして実施した (図6参照)。計測値の求め方も10回の計測で得た帯域と変異係数 (Coefficient of Variation) をそれぞれ平均として表3に示す。この結果からはファーストイーサネットとギガビットイーサネットでの比較では多少高精度PCの結果が優れていることが確認できるが、One Packet 計測手法と比べ大きな違いは見られなかった。また、ギガビットイーサネットにおいて低い計測結果となっている点について検証の必要が見受けられた。

Packet Pair 計測手法では、リンクのキューでの影響を計測するためその通過時間を大きく取る必要があり、小さいパケットでの計測は有効ではないことが知られている。しかし大きなパケットでは他の外乱パケットの影響を受ける可能性が増すと考えられる。pathrate ではリンクの最大パケットサイズを用いて計測を行っており、この影響がないか調査を行うため、送信側のPCにてUDPパケットを連続して送出し、受信側にて到着時間を計測するアプリケーションを独自に用意し、計測を行った。この計測ツールではOne Packet 計測手法でも利用したTSC情報を時間情報として用い、到着時間間隔を求めている。このツールを用い、パケットサイズによる影響を調査した。ギガビットイーサネットを用い、パケットサイズを1000バイトから9100バイトまで変化させて計測した結果を図8に示す。図には帯域の計測結果とパケット間隔の標準偏差 (単位はクロックカウント数) を描画している。この図からわかるように、3000バイト以上のサイズでは6000バイト付近まで値が大きくなるにつれ計測結果が1Gbpsに近づき、8000バイト以降では極端に計測結果が悪くなるのがわかる。

表3 pathrateを用いた計測結果

上：帯域平均値 下：変異係数の平均値

	10Mbps	100Mbps	1Gbps
Normal PC	9.7 Mbps 0.0000	98.0 Mbps 0.0015	500.8 Mbps 0.0057
Cesium PC	9.7 Mbps 0.0000	97.4 Mbps 0.0012	502.3 Mbps 0.0046

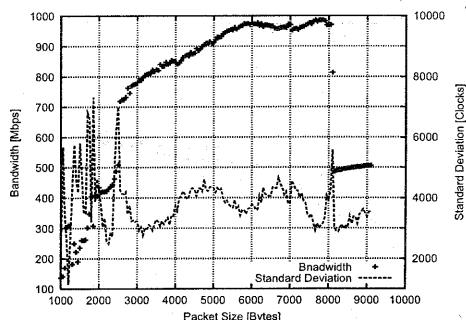


図8 パケットサイズの変化と計測結果の関係 (ギガビットイーサネット)

この結果から、7650バイトを最適パケットサイズとし、ギガビットイーサネットの帯域計測を実施し平均値で983.56 Mbps (分散: 6.276) と得ることができた。

5. 考察

One Packet 計測では、10Mbps、100Mbps、1Gbpsいずれの計測においても、従来のPCとくらべて高精度PCの計測結果の分散値を約半分の値にできており、計測値の精度を向上できたと言える。また、ギガビットイーサネットでは計測結果が360Mbpsと低い値であったのは、計測対象PCのICMP処理による影響があると考えられるが、その詳細は検証しておらず、今後調査を進めたいと考えている。

Packet Pair 計測でも、高精度PCの有効性を得ることができ、One Packet 計測では計ることができていなかったギガビットイーサネットでの帯域計測が高精度で行うことが可能であった。さらにギガビットイーサネットの計測では、パケットペアのサイズによる影響を観測することができたが、何による影響のための変化なのか検証することはできておらず、今後、最適なパケットサイズの決定要因について調査する必要がある。

6. おわりに

本稿では、高速ネットワークでのネットワーク計測に必要な精度を持つ高安定PCの開発について紹介し、実際に高安定PCによる計測を行いその有効性を評価した。高安定PCを用いることにより、従来の計測ツールでの計測値の分散を約半分にすることができ、さらにTSCの利用により高速ネットワークでの計測値の精度を向上させることが可能であった。

今後、帯域を細かく変化させた場合の計測や他の帯域計測手法を用いた場合の有効性の検証等を進める予定としている。ま

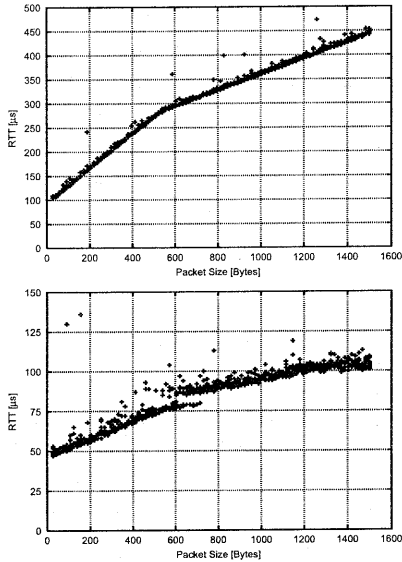


図9 パケットサイズによるRTTの変化 (Linux)
上：ファーストイーサネット 下：ギガビットイーサネット

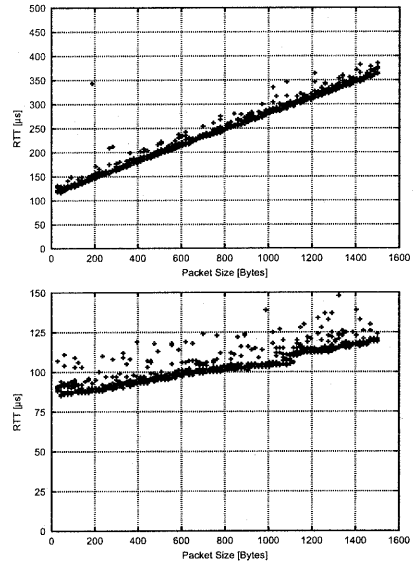


図10 パケットサイズによるRTTの変化 (FreeBSD)
上：ファーストイーサネット 下：ギガビットイーサネット

た、さらに高速なネットワークでの計測・評価を進め、高精度PCによる計測の優位性を示したいと考えている。

謝辞 本研究をまとめるにあたり、ギガビットネットワーク研究開発プロジェクトの方々へ多くの助言を頂き、特にネットワークの計測手法に関して多くのコメントを頂いた鶴正人研究員に深く感謝する。

文 献

- [1] H. Okazawa, A. Machizawa, S. Nakagawa, Y. Kitaguchi, T. Asami and A. Ito, "Advanced NTP Synchronization Device for Internet Monitoring Tools," Proc. INET2001 <http://www.isoc.org/inet2001/CD_proceedings/T42/inet2001.html>, Stockholm, June 2001.
- [2] Y. Kitaguchi, H. Okazawa, S. Shinomiya, Y. Kidawara, K. Hakozaki and S. Nakagawa, "Development of a High-Accurate Time Server for Measurements of the Internet," ICOIN16, Cheju, Korea, January 2002.
- [3] 町澤朗彦, 北口善明, 岡沢治夫, 中川晋一, "ネットワークを用いたCPU動作周波数期間安定度の精密計測," DICOMO2002, Japan, July 2002.
- [4] S. M. Bellovin, "A Best-Case Network Performance Model," <http://www.research.att.com/smb/papers/netmeas.pdf>, February 1992.
- [5] V. Jacobson, "Congestion Avoidance and Control," ACM SIGCOMM '88, Stanford CA, USA, August 1988.
- [6] V. Jacobson, "Pathchar," <http://www.caida.org/tools/utilities/others/pathchar/>, 1997.
- [7] B. A. Mah, "Pchar," <http://www.employees.org/bmah/Software/pchar/>, 1999.
- [8] A. B. Downey, "Clink," <http://rocky.wellesley.edu/downey/clink/>, 1999.
- [9] R. S. Prasad, C. Dovrolis and B. A. Mah, "The Effect of Layer-2 Switches on Pathchar-like Tools," ACM Internet

- Measurement Workshop, France, November 2002.
- [10] V. Paxson, "End-to-End Internet Packet Dynamics," ACM SIGCOMM '97, Riviera, France, 1997.
- [11] C. Dovrolis, D. Moore and P. Ramanathan, "What Do Packet Dispersion Techniques Measure?," IEEE INFOCOM 2001, Anchorage AK, US, April 2001.
- [12] B. Carter, "Bprobe and cprobe," <http://cs-people.bu.edu/carter/tools/Tools.html>, 1996.
- [13] K. Lai, "Nettimer," <http://mosquitonet.stanford.edu/laik/projects/nettimer/>, 2000.
- [14] R. S. Prasad, "BW-meter project," <http://www.pathrate.org/>, 2002.

付 録

ICMP 処理機構による問題

One Packet 計測にて、ファーストイーサネットのNICを用いた計測では図9(上)のようにパケットサイズ512バイト当たりで求める直線の傾きが変化する結果が得られた。またギガビットイーサネットでの計測でも図9(下)のように回帰直線の傾きから、512バイト以下、512バイトから1024バイトの間、1024バイト以上の3つの状態を持つように観測できた。この現象を調査する仮定で、対象PCを別のOS(今回はFreeBSD)に変更した場合に上記のような症状は発生せず、図10のように一定の傾きを得ることができた。そのため、前述したような求める直線の傾きが変化する現象は、利用したLinux OSのICMPを返すルーティンの実装によるものであると推測できる。このように、計測対象の実装によりOne Packet計測手法を用いることができない場合があることが判明した。