

インターネットにおける一方向遅延時間の 不均衡に関する特性解析

串田 高幸 柴田 義孝

日本アイ・ビー・エム株式会社東京基礎研究所
岩手県立大学ソフトウェア情報学部

概要

一般にインターネットは、ベストエフォート型サービスを提供しているネットワークであり、ネットワークの内部においてQoS(Quality of Services)制御の機能を持っていない。そのため、このネットワークにおいて、End-to-Endパスのパフォーマンスが保証されないことにある。今まで、多くのEnd-to-Endパスのパフォーマンス特性を測定解析することが行なわれてきた。一般に、インターネットのパスにおいては、パスによる指向性があると想定できる。この論文では、End-to-Endパスの一方向の遅延特性に関して、定義、測定方法及び測定結果の解析を行なった結果について報告する。

Characteristics of one-way delay of packets on end-to-end paths

Takayuki Kushida and Yoshitaka Shibata

IBM Research, Tokyo Research Laboratory

Iwate Prefectural University, Faculty of Software and Information Science

Abstract

The Internet provides a best-effort service. It is that there is no control mechanism of Quality of Service (QoS) for end users. Since the end-to-end performance isn't guaranteed on the Internet, the network measurement and analysis of the end-to-end performance is important task for an application that is using the Internet. This paper describes the definition of one-way end-to-end performance on the path, its measurement method and results of measured data.

1. はじめに

インターネットに代表されるパケット交換ネットワークにおいて、ルータのなかでネットワーク層において、パケットを始点ホストからヘッダーの目的アドレスを使って、次々と転送していき、最終的な目的ホストまでにデータを到達させる。このパケットを相互に折り合うことで、双方向のデータ転送を行なうことを基本している。このときの基本単位は、データの断片であるパケットになる。ただし、途中のルータにおいて、パケットを転送するとき、経路情報だけを見て、目的アドレスに向かって、次のルータにパケット転送するだけであり、それ以外のネットワークQoS(Quality of Service)に関するサポートあるいは保証する機能をもっていない。このことは、インターネットの設置及び管理コストを低

減し、またネットワークに対して拡張性を持つことになるが、一方では、QoSが必要となるアプリケーションでは、例えば、キャッシュやプリフェッチのように、なんらかの対処がネットワーク層より上の層において常時、必要となってくる。

このように、インターネットにおいて、一般にEnd-to-Endのパフォーマンスに関して保証がないために、ネットワークのリアルタイムでの測定解析がこの分野の研究において重要なテーマになっている。現状のネットワーク内部におけるパフォーマンスパラメータを取得するため、IETF(Internet Engineering Task Force)では、IPPM(IP Performance Metrics Working Group)というグループがあり、インターネットのパフォーマンスとして"metric"の定義とその測定方法を規定している [1].

遅延時間の測定において、周回遅延ではなく、一方方向遅延の測定が必要な理由を次の項目に上げる。

- 今日のインターネットにおいては、始点ホストから終点ホストのパスと終点ホストから始点ホストまでのパスがしばしば異なっていることがある。このような状態を非対称パスと定義する。この場合、パスの方向性によってネットワークの特性が異なる。当然、このときの周回遅延時間は、パフォーマンスの両方の和を測定していることになる。非対称パスになっている場合、それぞれのパスの特性を正確に調べたい場合には、測定データを2つの異なったISPで別個に転送して測定するか、あるいは、異なったタイプのネットワークにわけて、測定データを転送して、これら2つのパスの間のパフォーマンスを、非対称のそれぞれのパスにおいて、独立して測定する必要がある。
- さらに、パスが対称であったときでさえ、途中のルータ内部では、一般に非対称のキューの長さになると考えられるために、パフォーマンスは、根本的に異なった特徴を持つと考えてよい。
- また、アプリケーションのパフォーマンスは、どちらか一方でのパフォーマンスにほとんど依存していると考えられることができる。例えば、TCPを使ったファイル転送において、ACKパケットが転送されている方向よりも、どちらかというデータパケットが転送されている方向のパフォーマンスがアプリケーションパフォーマンスの大半が依存していると考えてよい。

このように項目のなかで、遅延時間の測定のうち、IETFのIPPMでの“metric”のうち、一方方向の遅延に対する測定パラメータの定義と測定方法は、“A One-way Delay Metric for IPPM”のドキュメントに記述されている [2]。

本研究では、インターネットの一方方向の遅延特性を測定解析して特徴を調べるために、測定システムを設計開発し、この測定システムを使って、一方方向遅延の測定を行ない、その結果を解析したについて報告する。この論文では、「バックグラウンド」において、この研究を行なうための背景及びこの研究と関連した研究について述べる。また、「一方方向遅延」

では、理論的な観点より一方方向遅延特性に関する定式を記述する。さらに「測定方式」では、測定するための装置及び測定の内容について述べる。「解析結果」では、測定データを解析して得られた結果について述べ、また、「まとめ」では、今回の測定解析で得られた知見について議論とまとめを行なう。

測定方式

ここでは、パケットの遅延時間を測定するための測定方式について述べ、この研究の中心となる一方方向遅延時間の測定について述べる。

図1は、周回遅延のエンド・エンドのモデルを示しており、SourceからDestinationに対してパケットが転送され、また、DestinationからSourceに測定のためのパケットが返ってくる状態を示している。途中のルータ(Router)内部では、それぞれ一定処理のための部分(Constant Processing)である d_i とキュー(Queue)である q_i^k があつて、このキューでの待ち時間は、キューに留まっているパケットの量によって時間ごとに変化がある。周回遅延時間を測定する場合、図1において、測定パケットがSourceからDestinationにルータ(Router 1, Route 2...)を経由して転送され、Destinationにおいて、戻りパケットが生成され、また、DestinationからSourceに測定パケットがルータを経由して戻ってくる。この測定パケットを送出した時間と戻ってきた送信パケットを受信した時間の差が、周回遅延時間になる。

周回遅延時間測定の場合、測定パケットの送信とパケットの受信が同じホストで行なわれるために、測定パケットの送信時間を t_s として、受信時間を t_r とすると、周回遅延時間は、 $t_{rtt} = t_r - t_s$ として定義される。このとき、 t_s と t_r を測定しているコンピュータは、同じであり、両方の時間は、同じホストであるために誤差がほとんどないと仮定することができる。そのため、この差分を測定するだけで周回遅延時間を測定することができる。

一方、図2において、一方方向の遅延時間を測定する場合、図2のSourceからDestination及びDestinationからSourceに転送されているパケットを別々に測定することと同等になる。

一方方向遅延時間の測定の場合、測定パケットの送信ホストを $HostS$ と定義して、測定パケットの受信ホスト $HostD$ と定義すると、一般に $HostS$ と $HostD$ の時刻は異なる。 $HostS$ で測定パケットを送

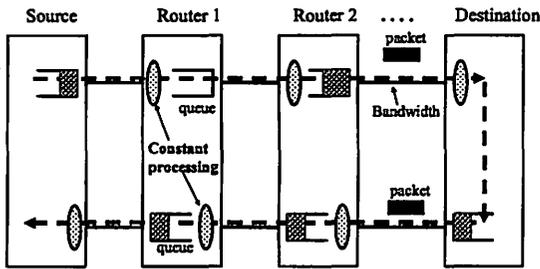


図 1: 周回遅延のモデル

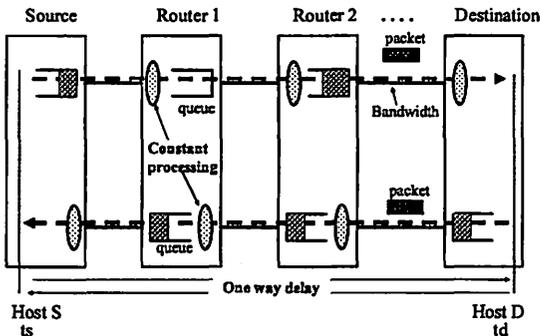


図 2: 一方向遅延のモデル

信した時間を t_s として、 $HostD$ で測定パケットを受信した時間を t_r とする。 $HostS$ 及び $HostS$ での絶対時間をそれぞれ、 $t_{s,abs}$, $t_{r,abs}$ とすると、 $t_{s,abs} - t_s = \delta_s$, $t_{r,abs} - t_r = \delta_r$ とすると、一般に $\delta_s \neq \delta_r$ である。この差は、 $HostS$ 及び $HostS$ での時間が異なることによる。そのため、一方向遅延時間測定では、 $HostS$ と $HostS$ の間で、正確な時間の同期が必要となる。この正確な同期とは、一方向測定時間に対して、時間の同期誤差が十分に小さくなるようにすることであって、次のような方法が考えられる。

- GPS による時間校正を行なって、同じ時間を使う。
- GPS 以外の基準時間のソース (時報, 電波時計, 2つのホストで共通に使用できる基準クロックを持つもの) を用いて、時間を校正する。
- 2つのホストの間で、NTP を使って時間を一致させる [12].
- 2つの PC を測定前に、上記の方法にて、事前に時間校正しておき、その後、測定に用いる。この方式は、PC が持っているクロックの違いによって少しずつ変化するため、たとえ、最初に時間を十分合わせていたとしても変化してしまう可能性がある。

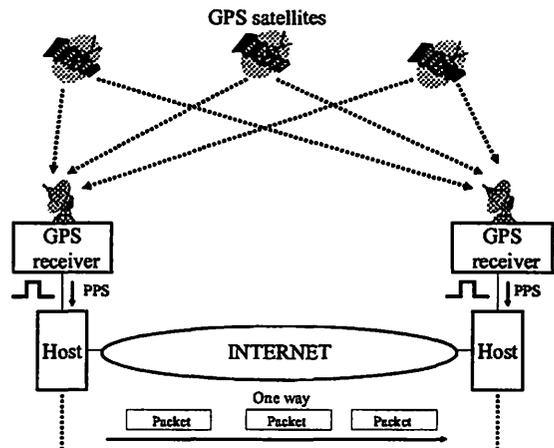


図 3: GPS 衛星を使った時間校正

このような方法があるうち、この研究では、時間校正の精度が一番よい GPS による時間校正を利用する。

GPS による時間校正: ここは、この時間を差を校正するために、GPS(Global Positioning System) を使用する。GPS は、24 個の人工衛星が地球の周回軌道にあり、電波により正確な信号を常時送っている。一方、GPS 受信機によって、この GPS からの信号を捕らえることによって、二次元及び三次元の位置を知ることができる方式である。この GPS から出されている時間情報を基準にして、GPS 受信機から高精度 (1 マイクロ秒) のパルス信号 (PPS) を測定コンピュータに入力して、時間調整を行なうことによって精度の高い時間に校正を行なうことが可能となる。この PPS においては、GPS 受信機と PC の間のインターフェイスとして、RS-232C を使って時間校正のためのパルス信号を GPS 受信機で生成し、PC 側で、このパルス信号の立ち上がりを捕らえることで 1 マイクロ秒の精度の時間を調整を行なうようにする [13]。実際には、PC 側で捕らえたパルス信号の立ち上がりを、ソフトウェアによって処理するため、1 マイクロ秒から精度は落ちてしまうが、時間校正のだけのために専用ハードウェアを用意することなく、必要とする時間精度が得られることになる。

図 3 は、周回軌道にある GPS 衛星 (GPS satellites) からの電波を GPS 受信機 (GPS receiver) で受信して、そこから基準クロックである PPS が生成され、測定コンピュータである両側の Host に取り込み、時間校正を行なっていることを示している。図のようにして 2つのホストの時間を校正する。

時間校正のオペレーティングシステムでのサポート: PC側のハードウェアに取り込んだとき、オペレーティングシステムでは、PSS信号をサポートするためのインターフェイスとしてRFC2783にあるAPIを用意する必要がある[14][15]。通常、GPS受信機側で、RS-232CのDCD(Data Carrier Detect)ピンに対して、PPS信号を生成し、コンピュータ側では、このPPS信号の立ち上がり信号の割り込みにより高い精度でこの信号を捕らえることになる。この時間を校正するためにPC側のソフトウェアとして、NTP(Network Time Protocol)ソフトウェアに、このPPS信号をもとにして時間を校正するインターフェイスがあるので、この機能を利用する[16][12]。この機能を使って、ネットワークの両側にある測定コンピュータでの時間を校正する。GPS受信機からの信号は、1マイクロ秒程度の精度であるが、コンピュータに取り込まれるときに10マイクロ秒程度の精度になる。しかし、測定の精度は、1ミリ秒(1000マイクロ秒)以下であればよいので、十分な精度で測定が可能になっている。この研究での実験では、LINUXオペレーティングシステム(Kernel Version 2.4)を使って、上記のPPSをコンピュータのソフトウェアでサポートすることを実現している。

パスの測定: 一方向遅延時間を行なう前に、パスの特徴を調べるためにtracerouteコマンドを使って周回遅延測定により、途中経路の遅延時間の増加の程度をForward path及びReverse pathのそれぞれについて測定することを行なった。その結果を図4に示す。図4のForward Pathでは、16と17ホップ目のところに大きな遅延がある。一方、Reverse Pathでは、1ホップ目と2ホップ目に大きな遅延がある。これは、リンクのバンド幅が小さいことであり、また、式(?)でのバンド幅 b_i が小さいことになる。そのため、 s^*/b_i が大きくなり、その結果、このリンクに大きな遅延が生じている。図4では、Forward Path及びReverse Pathとも同じリンクで遅延が大きくなっているため、ボトルネックになっているリンクがあるということが出来る。

また、測定するためのパケットは、簡易な方法で返答を受け取れる形式がよいと考えられる。例えば、UDPパケットを利用することが考えられるが、もし相手側のホストにおいて、Destination PortとなるUDPのportがアプリケーションソフトウェアによって待ちうけになっていないとき、ICMP unreachable

ホップ数	順方向ホスト名	逆方向ホスト名
1	210.156.40.129	211.120.99.11
2	202.211.0.8	210.239.30.136
3	202.211.0.2	210.175.198.11
4	150.99.197.109	210.175.194.181
5	150.99.197.29	210.175.192.98
6	150.99.197.22	210.130.163.205
7	150.99.197.142	210.130.163.145
8	202.232.8.129	202.232.0.110
9	210.130.130.12	202.232.0.98
10	202.232.0.99	210.130.130.43
11	202.232.0.111	202.232.8.130
12	210.130.163.148	150.99.197.141
13	210.130.163.206	150.99.197.21
14	210.175.192.37	150.99.197.30
15	210.175.194.222	150.99.197.110
16	210.175.194.49	202.211.0.8
17	211.120.99.11	210.156.40.129
18	211.120.99.12	210.156.54.18

表1: 順方向パス
 パケットが生成され、そのパケットが戻ってくることになり、その結果、余計なパケットが生成され、測定するパスに影響することになる。そのため、この測定では、パケットが返答されるかあるいは、パケットがそのまま帰ってこないようにするため、測定パケットとしてICMP echo request及びICMP echo replyを使用している。このパケットは、pingコマンドによって使われているパケットと同じ形式である。また、パケットのデータ収集に関しては、ホストのOSに取り込むときの誤差をなくすために、Ethernet上に転送されているパケットを直接収集できるようにする。そのために、BPF(Berkeley Packet Filter)を使ったtcpdumpによって、ネットワーク上のパケットを直接モニターすることによって測定した。このように直接パケットを捕獲してモニターすることによって、収集アプリケーションプログラムに依存しないデータ捕獲の時間によってデータをとることができること、また、測定すべきパケットをフォーマットによって収集でき、またそれ以外のパケットも同時収集することが可能であるため、より多くの情報を取得することが可能になる利点を持っている。

2つの一方向遅延の測定は、2002年12月から1

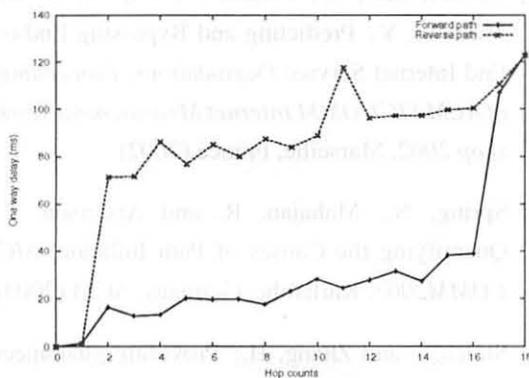


図 4: 周回遅延時間とホップ数の関係

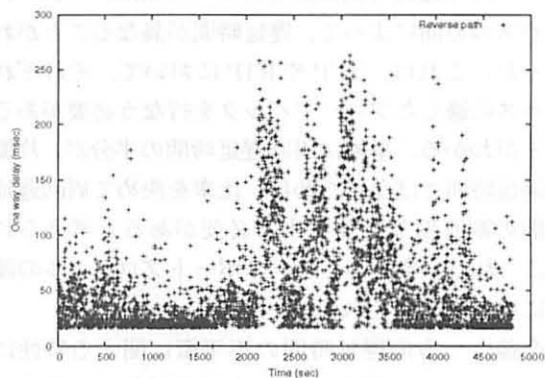


図 6: 一方向遅延時間の時系列の分布 (Reverse path)

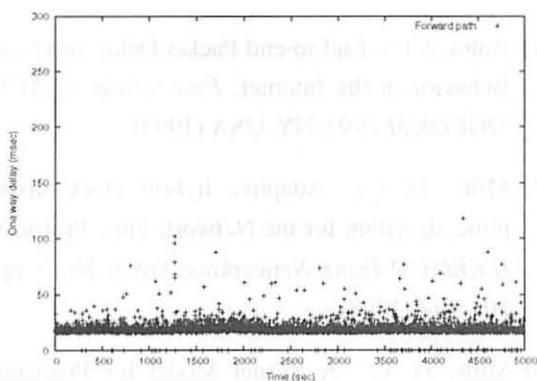


図 5: 一方向遅延時間の時系列の分布 (Forward path)

月にかけて、岩手県立大学と商用ネットワークの間をインターネットを経由して行なわれた。また、測定の際の packet サイズは、それぞれ 50, 100, 500, 1000 バイトの 4 種類を使って測定が行なわれた。測定の経路は、商用ネットワークのホストと学術ネットワークの間の 18 ホップについて測定をした。

測定結果

この章では、一方向遅延の測定結果について述べる。図 5 と図 6 は、50 バイトのデータによって、それぞれ、Forward path と Reverse path の一方向遅延時間を、同じ時間で測定した結果を示している。図 5 は、測定時間中にほぼ同じ遅延時間として測定されているが、図 6 は、2000-3500(sec)にかけて、図 5 に比べて、遅延時間が著しく増大している。この状況は、Reverse path において輻輳が起こっており、これが原因で遅延時間の増大していることを示している。

図 7 では、図 5 と図 6 の頻度分布を比較してい

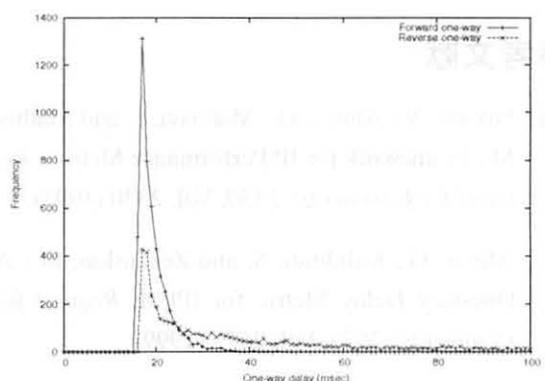


図 7: 一方向遅延時間の頻度分布

る。この図では、横軸を一方向遅延時間として、縦軸を頻度として、Forward path と Reverse path を比較している。Forward path は、頻度のピークも高く、また、急激に頻度が落ちている。これは、遅延の変動が少ないことを意味している。一方、Reverse path は、Forward path とは異なり頻度のピーク低く、頻度が長い遅延にも存在している。

これは、Forward path が測定サイトからインターネットに packet を送る経路であって、また、Reverse path が、逆にインターネットから始点サイトに packet を送る経路である。そのため、インターネットから始点サイトに packet が転送されている量が多くなっており、そのためにリンクで輻輳が生じていることを示している。

また、Forward path/Reverse path の平均と分散を調べたところ、それぞれ 19.54/52.07 と 35.76/2137.39 となっていた。この統計情報から両者をそれぞれ数値として比較していると、Reverse path が Forward path に比べて、より平均が大きく分散が大きいことがわかる。

一方向の遅延時間測定を行なった結果、それぞれのパスの方向によって、遅延時間が異なることがわかった。これは、TCPやRTPにおいて、それぞれのパスに適したフィードバックを行なう必要があることがわかる。単純に周回遅延時間の半分が、片側の遅延時間ではないために、比率を決めて周回遅延時間の割り振りを変更する必要があると考えられる。これは、今後のトランスポートプロトコルの課題になると考えられる。

今後、一方向遅延時間の不平衡に関する特性に対してさらに、測定を行なう必要があると考えている。

参考文献

- [1] Paxson, V., Almes, G., Mahdavi, J. and Mathis, M.: Framework for IP Performance Metrics, *Request for Comments: 2330*, Vol. 2330 (1998).
- [2] Almes, G., Kalidindi, S. and Zekauskas, M.: A One-way Delay Metric for IPPM, *Request for Comments: 2679*, Vol. 2679 (1999).
- [3] Savage, S. and A Collins, E. H.: The End-to-end Effects of Internet Path Selection, *SIGCOMM99*, Cambridge MA, USA, ACM (1999).
- [4] Lai, K. and Baker, M.: Measuring Link Bandwidths Using a Deterministic Model of Packet Delay, *Proceedings of ACM SIGCOMM 2000*, Stockholm Sweden (2000).
- [5] Jain, M. and Dovrolis, C.: End-to-End Available Bandwidth: Measurement Methodology, Dynamics, and Relation with TCP Throughput, *SIGCOMM2002*, Pittsburgh PA, USA, ACM (2002).
- [6] Zhang, Y., Breslau, L., Paxson, V. and Shenker, S.: On the Characteristics and Origins of Internet Flow Rates, *SIGCOMM2002*, Pittsburgh PA, USA, ACM (2002).
- [7] Pasztor, A. and Veitch, D.: Active Probing using Packet Quartets, *Proceedings of ACM SIGCOMM Internet Measurement Workshop 2002*, Marseille, France (2002).
- [8] Bremler-Barr, A., Cohen, E., Kaplan, H. and Mansour, Y.: Predicting and Bypassing End-to-End Internet Service Degradations, *Proceedings of ACM SIGCOMM Internet Measurement Workshop 2002*, Marseille, France (2002).
- [9] Spring, N., Mahajan, R. and Anderson, T.: Quantifying the Causes of Path Inflation, *SIGCOMM2003*, Karlsruhe, Germany, ACM (2003).
- [10] Stoica, I. and Zhang, H.: Providing guaranteed services without per flow management, *Proceedings of ACM SIGCOMM 1999*, Cambridge MA, USA (1999).
- [11] Bolot, J. C.: End-to-end Packet Delay and Loss Behavior in the Internet, *Proceedings of ACM SIGCOMM 1993*, NY, USA (1993).
- [12] Mills, D. L.: Adaptive hybrid clock discipline algorithm for the Network Time Protocol, *IEEE/ACM Trans. Networking*, Vol. 6, No. 5, pp. 505-514 (1998).
- [13] Mills, D. L.: A Kernel Model for Precision Timekeeping, *Technical Memorandum, U. of Delaware*.
- [14] Mills, D.: A Kernel Model for Precision Timekeeping, *Request for Comments: 1589*, Vol. 1589 (1994).
- [15] Mogul, J., Mills, D., Brittonson, J., Stone, J. and Windl, U.: Pulse-Per-Second API for UNIX-like Operating Systems, Version 1.0, *Request for Comments: 2783*, Vol. 2783 (2000).
- [16] Mills, D. L.: Improved algorithms for synchronizing computer network clocks, *IEEE/ACM Trans. Networking*, Vol. 3, No. 3 (1995).