社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2004－ＱＡＩ－11 （５）
2004／5／26

広帯域ネットワークにおけるIPストリーム伝送性能測定ツールの開発

川上 貴宏† 岸田 崇志† 河野 英太郎‡ 前田 香織‡

†広島市立大学大学院 情報科学研究科，‡広島市立大学 情報処理センター
〒 731-3194 広島市安佐南区大塚東 3-4-1

E-mail: †{kawakami,takashi}@v6.ipc.hiroshima-cu.ac.jp, ‡{kouno,kaori}@ipc.hiroshima-cu.ac.jp

あらまし 広帯域ネットワークの普及に伴い，動画像や音声等のストリームデータの伝送を行う機会が増えてきている．この時，ネットワークの帯域幅だけではなく，ストリーム伝送に影響を及ぼす原因を把握する必要性が高まってくると考えられる．本研究では，擬似的なIPストリームを伝送することによって，ネットワークの性能を測定するツールSperfの開発を行った．Sperfではパケットのロス率，ロスの傾向，順序エラー，ジッタ，遅延時間等が測定でき，また結果を視覚的に表示できる．本稿では，Sperfの構成について述べ，評価を行うと共に，Sperfを用いて実際に数種類の広帯域ネットワークを測定し，各ネットワークのストリーム伝送性能について評価や考察を行った．

キーワード ストリーム伝送，ネットワーク計測，インターネット，ブロードバンド

Development of a Network Performance Measurement Tool to Transmit IP streams on Broadband Networks

Takahiro KAWAKAMI† Takashi KISHIDA† Eitaro KOHNO‡ and Kaori MAEDA‡

† Graduate School of Information Sciences, Hiroshima City University
‡ Information Processing Center, Hiroshima City University
3-4-1 Ozuka-Higashi, Asaminami-ku, Hiroshima, 731-3194. Japan

E-mail: {kawakami,takashi}@v6.ipc.hiroshima-cu.ac.jp, {kouno,kaori}@ipc.hiroshima-cu.ac.jp

Abstract As a spreading broadband networks, it is popular to transmit stream data such as moving pictures or voice on the Internet. Sometimes, it is required to know not only available bandwidth of networks but also other network performance for streaming. Then we developed a network measurement tool called "Sperf". In Sperf, simulated IP streams depending on stream applications are transmitted and network performance parameters can be measured such as available bandwidth, packet loss ratio, sequence errors, jitter, and delay. In this paper, we show the detail of Sperf and its evaluation. Also, we mention the effectiveness of Sperf by measuring some actual broadband networks.

Keyword Stream transmission, Measurement of network, the Internet, Broadband network

1. はじめに

近年の広帯域ネットワークの急速な普及に伴い，動画像や音声等のストリームデータをインターネット上でリアルタイムに伝送する機会が増えてきている．しかし，DVTS[1]，Robst[2]等ストリーム伝送アプリケーションの多くはRTPを代表としたUDPベースのプロトコルを採用しているため，ベストエフォート型のネットワークであるインターネット上で通信を行う際にはパケット損失やジッタに大きな影響を受けてしまう．このため，実際にストリーム伝送を行った際に帯域幅は十分あるにも関わらず，動画像の乱れや音声の途切れ等の問題に遭遇することは珍しいことではない．そこで，問題の原因の特定に対する要求が高まっている．しかし，この問題はストリーム伝送時のようにネットワークが高負荷状態になって始めて表面化することが多い．

ネットワークの性能の測定には，Netperf[3]やIperf[4]等既存のネットワーク測定ツールを用いることが多いが，多くのツールはストリーム伝送を想定していないため得られる情報が不十分であり，問題の原因を特定することが難しい．そこで，本研究では擬似的なIPストリームを伝送することにより，想定するストリーム伝送時のネットワークの性能を測定するツール" Sperf" の開発を行った．また，数種のネットワークの測定を行い，それらのストリーム伝送性能について評価や考察を行った．

本稿では，2章でSperfの特徴，仕様，実装について述べ，3章でSperfの利用例とその考察について述べる．

さらに，4 章で Sperf の評価について述べ，5 章では，まとめと今後の課題について述べる．

2. IP ストリーム伝送性能測定ツール Sperf

2.1. Sperf の特徴

Sperf の特徴を以下にまとめる．

a) IP ストリーム伝送を想定した測定

IP ストリーム伝送時のネットワークはパケットが絶えず狭い間隔で流れてくる状況にある．そこで，Sperf では擬似 IP ストリームを伝送しその時のネットワークを測定する．これにより，IP ストリーム伝送時のネットワークの状況で各パラメータの測定を行うことができる．

ストリーム伝送を行うアプリケーションには様々な種類があり，使用するパケットの形態も多様である．そこで，Sperf では想定するアプリケーションのストリームを再現するために送信帯域，PPS(packet per second)の指定が可能である．また，Sperf はネットワークのストリーム伝送性能を測定するために有効帯域，パケット損失，バースト損失，順序エラー，ジッタ，RTT(Round Trip Time)の 6 項目について測定可能である．各項目の定義については 2.2.2 小節で述べる．

b) 長時間の測定が可能

ストリーム伝送は一般に数分以上の時間継続して行われる場合が多いと考えられる．よって，ネットワークのストリーム伝送性能を把握するためには数秒から数分の短時間の測定によりその瞬間の性能を調べるだけではなく，長時間測定を行うことで，そのネットワークの性能の平均値として信頼できる測定値を得ることや，突発的に発生するパラメータの大きな変動を観測することが重要である．そこで，Sperf では 24 時間以上の長時間の測定にも対応している．

c) 測定結果の視覚化が可能

Sperf では 1 秒毎のパケット損失数，ジッタおよび，測定中に何個連続の損失が何回発生したのかをグラフ化が容易な形でログに保存する．これらをグラフ化することで，パーセント表示や平均値表示だけでは得ることのできない情報を得ることができる．

2.2. Sperf の仕様

Sperf は，測定用の擬似 IP ストリームを送信する機能と擬似 IP ストリームを受信し測定値を求める機能に分かれており，それぞれが別の PC で実行されることを前提としている．以降では，前者を sender，後者を receiver と呼ぶこととする．

2.2.1. Sperf の動作概要

Sperf の動作の流れを図 1 を用いて説明する．

①sender の起動．情報パケットの待ち受けを開始する．

②receiver の起動．このときデータパケットの送信帯域と送信時間を指定する．

③receiver が sender へ測定に必要な各種パラメータを格納した情報パケットを送信する．情報パケットの送信には TCP が使用される．

④sender が情報パケット内のパラメータに従って擬似 IP ストリームを構成するデータパケットを送信する．データパケットの送信には UDP が使用される．

⑤receiver は受信したデータパケットに格納された情報を元に 1 秒毎の区間結果を計算し出力する．このとき，同時にログに区間結果を保存する．また，receiver は RTT の測定のために ICMP パケットを送信する．

⑥データパケットの送信終了．sender は終了フラグを立てたデータパケットを送信する．このパケットは損失に備えて数回送信される．

⑦receiver は受信したデータパケットの終了フラグが立っていたら受信を終了して最終結果を出力する．このとき同時にログに最終結果を保存する．

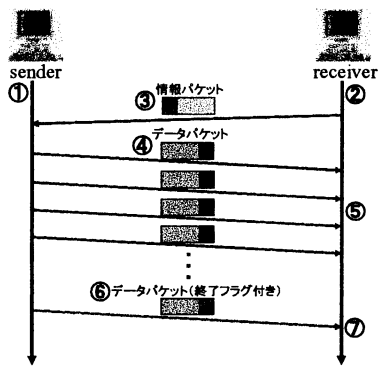

図 1．Sperf の動作の流れ

2.2.2. 測定項目の定義

2.1 節で挙げた Sperf の測定項目の定義について述べる．詳細な導出法は，文献[5]の通りで，文献から変更のあったパケット損失のみ導出法の詳細を記す．

① 有効帯域

Sperf における有効帯域は，総受信データパケットサイズと総受信時間の商と定義する．

② パケット損失

Sperf におけるパケット損失の導出法を図 2 を用いて説明する．receiver は，各データパケット用に受信バッファを用意し，受信したデータパケットのシーケンス番号を一致する番号のバッファに格納する．受信バッファには初期化時に 0 が格納されているため，0 のままのバッファを調べることで損失したパケットが分かる．この方式は文献[5]の方式と比較して演算負荷が小さく，リアルタイム処理に適している．

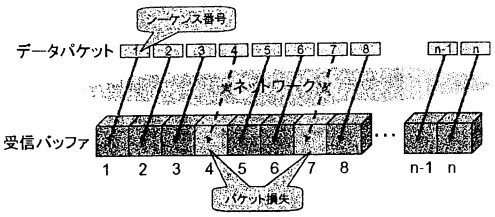

図 2. パケット損失の導出

③ バースト損失

Sperf では 5 個以上の連続したパケット損失をバースト損失と定義する.

④ 順序エラー

Sperf では受信したデータパケットのシーケンス番号が直前に受信したパケットの番号より小さかった場合を，順序エラーと定義する.

⑤ ジッタ

Sperf では平均遅延時間からの揺らぎの絶対値の平均値をジッタと定義する.この算出方法を採用することにより,瞬間的に発生する測定値の大きな変動を観測することができるようになっている.

⑥ RTT

Sperf では，測定時に receiver からデータパケットと同サイズの ICMP パケットを送信することにより，RTT を測定し，遅延時間の目安としている.

2.2.3. パケットフォーマット

Sperf が測定で用いる 2 種類のパケットのパケットフォーマットを図 3 に，各フィールドの説明を表 1，2 に示す.

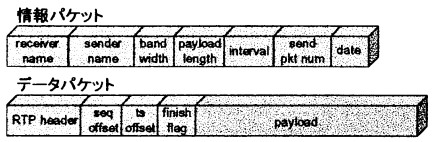

図 3. Sperf のパケットフォーマット

表 1. 情報パケットの各フィールドの説明

フィールド名	size(byte)	説明
receiver name	100	受信ホストのIP アドレスを格納
sender name	100	送信ホストのIPアドレスを格納
bandwidth	4	IPストリームの送信帯域を格納
message length	4	データパケットのサイズを格納
interval	4	データパケットの送信間隔を格納
send pkt num	8	データパケットの送信数を格納
date	50	測定日時を格納

表 2. データパケットの各フィールドの説明

フィールド名	size(byte)	説明
RTP header	16	RTPヘッダ中のseqフィールドとtsフィールドを使用
seq offset	2	シーケンス番号の周回数を格納
ts offset	2	タイムスタンプの周回数を格納
finish flag	2	測定終了を示すフラグを格納
payload	42～1478	ダミーデータを格納. 設定されたデータパケットサイズにより42byte～1478byteの間で可変の値を取る.

2.2.4. Sperf の測定用パラメータ群

Sperf が測定に用いる各パラメータとその説明を以下に示す.

a) 使用ポート

デフォルトでは，TCP に 8967 番，UDP に 9876 番を使用する．オプションで変更可能.

b) 送信帯域

1Mbps～96Mbps の範囲で，1Mbps 単位で指定可能．デフォルト値は 6Mbps.

c) メッセージサイズ

64byte～1500byte の範囲で，1byte 単位で指定可能．デフォルト値は 1280byte.

d) データパケットの送出間隔

122 μ sec の整数倍で指定可能．それ以外の値が指定された場合は自動的に最も近い倍数に近似される．デフォルト値は 1586 μ sec．このパラメータにより PPS を指定することができる.

e) 測定時間

10 秒～24 時間の測定に対応している．デフォルト値は 60 秒.

f) オプション

上記のパラメータ郡を指定するオプションを表 3 にまとめる.

表 3. Sperf のオプション

オプション	機能	備考
-s	senderモードで起動	-sと-rオプションは排他的に利用可能
-r	receiverモードで起動	
-b	送信帯域を指定	-bオプションまたは，-lオプションと-iオプションの組み合わせを排他的に利用可能
-l	データパケットサイズを指定	
-i	データパケットの送出間隔を指定	
-t	測定時間を秒単位で指定	-tと-hオプションは排他的に利用可能
-h	測定時間を時間単位で指定	
-p	UDPのポート番号を指定	TCPのポート番号は指定した値+1

2.2.5. Sperf の使用法

Sperf の動作は大きく分けて sender モードと receiver モードに分かれる.

a) sender モード

以下のように-s オプションを付けることで sender モードが起動する．使用法を以下に示す.

#sperf –s [オプション(-p)]

現在利用可能なオプションはポート番号を変更する-p オプションのみである.

b) receiver モード

以下のように，-r オプションの後にその他のオプション，sender のホスト名（IP アドレス），自分自身のホスト名(IP アドレス)を入力することで，receiver モードが起動する.

#sperf –r [オプション] sender-name receiver-name

2.3. 実装

2.3.1. 実装状況

Sperf の実装は Linux 上で C 言語にて行っている．現在，2.1 節，2.2 節で述べた機能の中の 2.1 小節の c)で述べた測定結果の視覚化の自動化以外の部分の実装が完了している．

現在の Sperf の動作確認済み OS(カーネル)は以下の通りである．

- Vine Linux 2.5(kernel2.4.18,2.4.22)
- Vine Linux 2.6(kernel2.4.22)
- Redhat Linux 9(kernel2.4.20)
- Fedora CORE 1(kernel2.4.22)

2.3.2. 実行例

Sperf の実行例を図 4，5 に示す．図 4 は sender の実行例である．①では sender が receiver からの情報パケットを受け取り，測定のパラメータを出力している．図 5 は receiver の実行例である．②は，receiver モードで起動時のコマンドを表す．この例では，メッセージサイズ 1500byte，送出間隔 976 μ sec，測定時間 10 秒で測定を開始している．③は，一秒毎の区間結果の出力を示している．区間結果では，測定時刻，帯域(送信／受信)，パケット損失率(区間値／平均値)，バースト損失(区間発生回数／最大バースト損失数)，順序エラー，ジッタ(区間値，平均値)，RTT が出力される．区間結果が測定時間 10秒に対し5回しか出力されていないのは，内部の実装として送信間隔が安定するまで最初の 5 秒分のデータパケットを切り捨てているためである．④は最終結果の出力を示す．最終結果では，帯域(送信／受信)，平均パケット損失率，バースト損失(区間発生回数／最大バースト損失数)，順序エラー，平均ジッタ，RTT(平均値，最大値)が出力される．

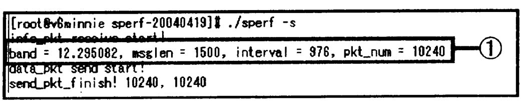

図 4. sender の実行例

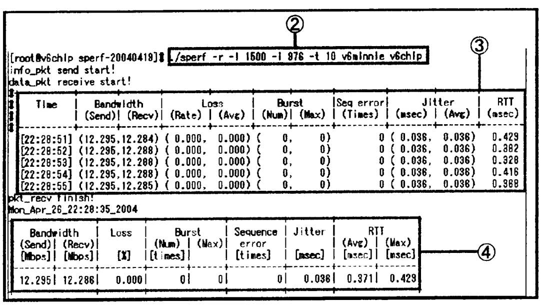

図 5. receiver の実行例

3. Sperf の実ネットワークでの利用例

3.1. 測定結果のグラフ化が有効な利用場面

Sperf の利用の一例として，B フレッツのフレッツグループで接続された 2 ホスト間の測定を行った．測定はテレビ品質の MPEG2 伝送を想定した 6Mbps，ハイビジョン品質の MPEG2 伝送を想定した 15Mbps，DV 伝送を想定した 30Mbps についてそれぞれ 1 時間行った．測定環境を図 6 と表 4 に，測定結果を表 5 に示す．また，パケット損失とジッタの区間結果をグラフ化したものをそれぞれ図 7 と 8 に示す．

既存のネットワーク測定ツールでは，測定項目が多いものでも有効帯域，パケット損失率，順序エラー，ジッタまでしか測定することができない．これだけの情報でこのネットワークを判断すると，表 5 の測定結果から帯域幅は十分に確保されており，パケット損失率やジッタも小さく，高いストリーム伝送性能を持っていると予想される．しかし，Sperf 特有の測定項目であるバースト損失の結果を見てみると，バースト損失が頻発していることが分かる．そこで，図 7 と 8 を併せて参照すると，このネットワーク中でストリーム伝送中に発生するパケット損失はバースト的に発生し，それと同時にジッタも大きく変動していることが分かる．よって，これらの情報から総合的に判断するとこのネットワークではストリーム伝送中に瞬間的な映像の乱れや音声の途切れが頻発すると予想される．

このように，Sperf では既存のネットワーク測定ツールでは見逃してしまうストリーム伝送時におけるネットワークの問題を，ストリーム伝送に特化した測定項目，ログの視覚化により発見することが可能である．

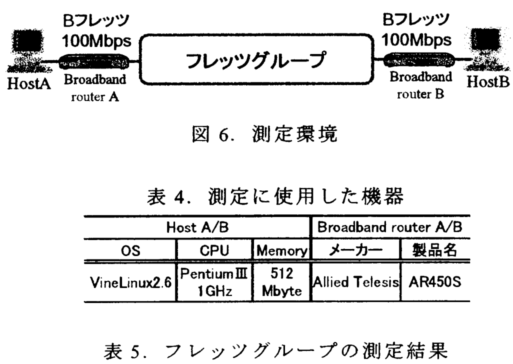

図 6. 測定環境

表 4. 測定に使用した機器

Host A/B			Broadband router A/B	
OS	CPU	Memory	メーカー	製品名
VineLinux2.6	PentiumⅢ 1GHz	512 Mbyte	Allied Telesis	AR450S

表 5. フレッツグループの測定結果

Bandwidth		Loss	Burst		Seqence	Jitter	RTT	
(Send)	(Recv)		(Num)	(Max)	error		(Avg)	(Max)
[Mbps]	[Mbps]	[%]	[times]		[times]	[msec]	[msec]	[msec]
5.997	5.991	0.012	11	25	0	0.139	27.83	30.703
14.99	14.971	0.039	24	137	0	0.129	27.777	127.9
29.989	29.963	0.047	19	261	0	0.088	28.434	31.759

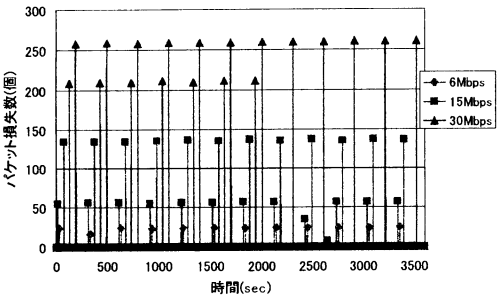

図 7. フレッツグループにおける
各帯域のパケット損失

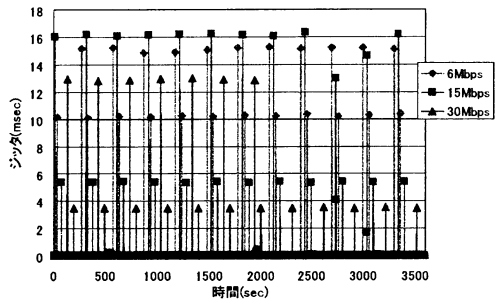

図 8. フレッツグループにおける各帯域のジッタ

3.2. PPS の指定が有用な利用場面

PPS がストリーム伝送時のネットワークの性能に与える影響を調べるため，同帯域で PPS が異なる 3 種のストリームを流し，測定を行った．測定には 4.1 節と同様に図 6 と表 4 で示された環境を用いた．測定結果を表 6 に，パケット損失，ジッタの区間結果をグラフ化したものを図 9 と 10 に示す．

これらの結果から，同帯域のストリームデータを同環境で伝送する場合においてもパケットサイズと PPS の関係によってパケット損失，ジッタ，RTT 等のストリーム伝送性能が変化することが分かる．このように，想定するストリーム伝送アプリケーションのパケットサイズ，PPS に合わせて測定を行うことで，より信頼性の高い測定結果が得られると考えられる．

表 6. PPS 変更時の測定結果

Message length	PPS	Bandwidth (Send)	Bandwidth (Recv)	Loss	Burst (Num)	Burst (Max)	Sequence error	Jitter	RTT (Avg)	RTT (Max)
[byte]		[Mbps]	[Mbps]	[%]	[times]		[times]	[msec]	[msec]	[msec]
1280	1024	10.492	10.484	0.005	1	131	0	0.119	28.133	30.809
640	2048	10.492	10.484	0.009	23	44	0	0.092	26.006	47.743
320	4096	10.492	10.495	0.029	25	199	0	0.098	24.961	27.811

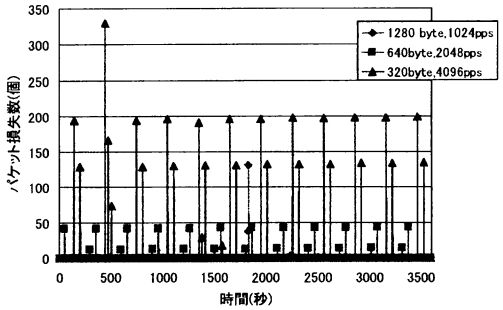

図 9. PPS によるパケット損失の変化

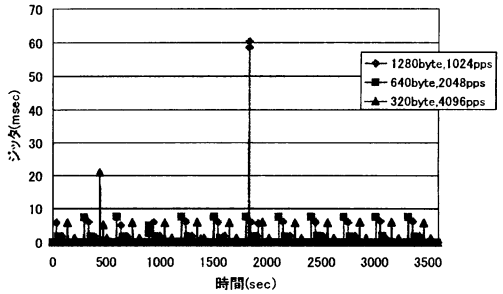

図 10. PPS によるジッタの変化

4. 評価

4.1. データパケットの送出間隔の評価

Sperf のデータパケットの送出間隔の精度は測定結果に大きく関係する．現在タイマとして RTC(Real Time Clock)を使用しているので，送出間隔はそれに依存する．RTC は，時刻を保持するためのハードウェアクロックであり，これを用いることで 1/2 秒から 1/8192 秒まで 2 のべき乗刻みで割り込みを掛けることができる．Sperf は 1/8192 秒，即ち約 122 μ sec をデータパケットの送出間隔の基準値としている．2.2.4 小節の d)で示す通り，Sperf は 122 μ sec の整数倍の送出間隔を指定できるが，何通りかの送出間隔の指定をして tcpdump[6]で測定したところ，実際にほぼ指定した間隔で送出していることを確認している．

4.2. 測定値の信頼性の評価

Sperf の測定値の信頼性を評価するために，ネットワークエミュレータ NISTNet[7]を用いて測定を行った．実験環境を図 11，表 7 に示す．測定に用いる 2 ホスト間に NISTNet を起動させた PC 設置し，パケットを中継させることで，任意の確率のパケット損失や任意の遅延時間を発生させる．NISTNet に与えるパラメータは，パケット損失率 10%，遅延 10msec，順序エラー率 3%とし，HostA→HostB 向き，HostB→HostA 向き共に同様のパラメータを指定した．また，Sperf に与えるパラメータは，送信帯域 10Mbps，測定時間 1 時間とした．測定結果を表 8 に示す．

表 8 より，パケット損失，RTT について NISTNet で指定した通りの測定値が得られていることが確認できる．順序エラーについても，測定値である 82099 回は全受信パケットの約 2.6%を占めており，設定値に近い値が測定できていると言える．以上から，これら 3 項目において，NISTNet 相当の精度が確保されていることが確認できた．

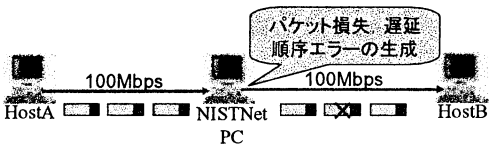

図 11. 実験環境

表 7. 実験に使用した PC

	OS	CPU	Memory
HostA	RedhatLinux9.0	Pentium4 2.6GHz	512Mbyte
HostB	VineLinux2.5	Pentium4 2.2GHz	512Mbyte
NISTNetPC	VineLinux2.5	PentiumⅢ 600MHz	384Mbyte

表 8. 測定結果

Bandwidth		Loss	Burst		Sequence	Jitter	RTT	
(Send)	(Recv)		(Num)	(Max)	error		(Avg)	(Max)
[Mbps]	[Mbps]	[%]	[times]		[times]	[msec]	[msec]	[msec]
9.992	9.224	10.08	0	5	82099	0.117	20.698	21.003

4.3. 既存のツールとの比較

Sperf を既存のツールの中で一般に用いられかつ機能的にも近い Iperf の UDP モードと比較を行った．

4.3.1. 機能比較

ストリーム伝送を想定した測定という観点での Sperf と Iperf の機能比較を表 9 にまとめた．表 9 より，Sperf と Iperf の測定項目には類似点が多いことが分かる．しかし，3.1 節で述べたように動画像や音声の品質に大きく関係するパケット損失について損失率だけではなく，損失の仕方，特にバースト損失に関する情報がネットワークのストリーム伝送性能の評価には有効と考えられる．また，3.2 節で述べたように，想定するアプリケーションを模倣して測定を行うことでより有益な測定結果が得られると予想される．さらに Sperf では，区間結果のログを視覚化することで測定結果だけでは得られない情報を得ることができる．これらの点で Sperf は Iperf には無い利点を持っていると言える．しかし，多くのストリーム伝送アプリケーションが対応しているマルチキャストについては Sperf の現在の実装では未対応となっており，今後の対応が必要と考えている．

表 9. Sperf と Iperf の機能比較

	有効帯域	パケット損失	バースト損失	順序エラー	ジッタ	RTT	帯域指定	PPS 指定	データの後処理	IPv6	マルチキャスト
Sperf	○	○	○	○	○	○	○	○	○	△	×
Iperf	○	○	×	○	○	×	○	×	×	○	○

4.3.2. ジッタの導出法の比較

Sperf と Iperf のジッタの導出法について比較，検討を行った．Iperf は，ジッタの導出に一般的に用いられる RFC1889(RTP)で提案された方式を採用している．この方式は，遅延時間の変化量の差分をジッタとしており，演算時にノイズの影響を抑え測定値を収束させるように設計されている．このため，評価値としての信頼性は高いが，瞬間的に発生する大きな変化を平滑化してしまうため，ネットワークの特性が見えにくくなる場合がある．これに対し，Sperf のジッタでは瞬間的に発生する大きな変化をそのまま測定値に反映させるため，ストリーム伝送中に瞬間的に発生する大きなジッタもデータが採取できる．また，Sperf では 1 秒毎の区間結果をログに保存しているため，測定中のどの時点で大きなジッタが発生したのかを測定後にパケット損失との因果関係を含めて考察することができる．しかし，ジッタの変動が大きいネットワークでは最終結果で得られる平均値の信頼性に欠ける傾向がある．このため，区間結果のグラフ表示を含めてその妥当性を慎重に検討する必要がある．

5. おわりに

本稿では，ネットワークのストリーム伝送性能を調査する，ネットワーク測定ツール Sperf の開発と評価について述べた．そして，Sperf が有効な利用場面について示した．今後の課題としては実装が不完全あるいは未実装である IPv6，マルチキャストへの対応，現在手動で行っている区間結果のグラフ化の自動化の実装を進めたい．

謝辞

本研究に際し，広島市立大学の石田賢治教授に有益なご助言を頂き感謝致します．本研究の一部は広島市立大学特定研究費(平成 15 年度 3207)の支援を受けて実施されている．ここに記して感謝の意を示す．

文　献

[1] DVTS, http://www.dvts.jp/.

[2] 近堂徹，西村浩二，相原玲二，前田香織，大塚玉記，“高品質動画像伝送における FEC の性能評価，” 情報処理学会論文誌，Vol.45, No1, pp84-92, Jan. 2004

[3] Netperf, http://www.netperf.org/netperf/NetperfPage.html.

[4] Iperf, http://dast.nlanr.net/Projects/Iperf/.

[5] 川上貴宏，岸田崇志，河野英太郎，前田香織，“高品質動画像伝送を想定したネットワーク性能評価ツールの開発，” 情報処理学会研究報告，2003-DSM-29, pp1-6, 2003

[6] tcpdump, http://www.tcpdump.org/.

[7] NISTNet, http://www.itl.nist.gov/div892/itg/carson/nistnet/.