ISSN 1349-1113
JAXA-RR-14-003

宇宙航空研究開発機構研究開発報告

JAXA Research and Development Report

事業所間仮想プライベートネットワークの高速化

大川　博文，奥居　毅彦，伊藤　利佳，藤田　直行

2014年12月

宇宙航空研究開発機構

Japan Aerospace Expoloration Agency

目　次

事業所間仮想プライベートネットワークの高速化

大川　博文[*1]，奥居　毅彦[*1]，伊藤　利佳[*1]，藤田　直行[*1]

Reconstructing of “JSSnet” for network acceleration

Hirofumi OHKAWA[*1], Takehiko OKUI[*1], Rika ITO[*1], and Naoyuki FUJITA[*1]

概要

本資料では, JAXA の 4 事業所から JAXA スーパーコンピュータシステム（以下，JSS と呼ぶ）を高速に利用できるようにすることを目的に構築したネットワーク（以下，JSSnet と呼ぶ）を対象に，仮想プライベートネットワークで構築した場合（以下，VPN と呼ぶ）と，SINET が提供する L2VPN サービスを利用して構築した場合（以下，SINET-L2VPN 網と呼ぶ）で，それぞれ性能測定を実施，比較した．これにより，ネットワーク環境のボトルネックの特定とネットワーク構成の変更による高速化の可否について分析を行い，JSSnet の再構築を行った。

Abstract

“JSSnet” is a Virtual Private Network, which connects four remote branches of JAXA to use "JAXA Supercomputer System" at higher speed. In this report, we first measure and compare the performance of two network configurations, a configuration with VPN router and a one with the L2VPN service provided by SINET, in order to accelerate JSSnet. Then, we reconstruct JSSnet on the basis of an analysis result and a performance bottleneck identification.

*　平成 26 年 10 月 14 日受付　(Received 14 October, 2014)

*1　航空本部数値解析技術研究グループ
(Numerical Simulation Research Group, Institute of Aeronautical Technology)

1. 背景・目的

JAXA のように事業所が地理的に分散している組織の場合，ネットワーク帯域が十分に広く回線品質に問題がなくとも，ユーザが想定するネットワーク性能とはかけ離れてしまうことが多い．また，事業所間をインターネット経由の VPN ルータにより相互接続することがあるが，専用線で接続する場合に比べて回線費用が安価に抑えられる一方で，回線品質に課題を抱える場合がある．

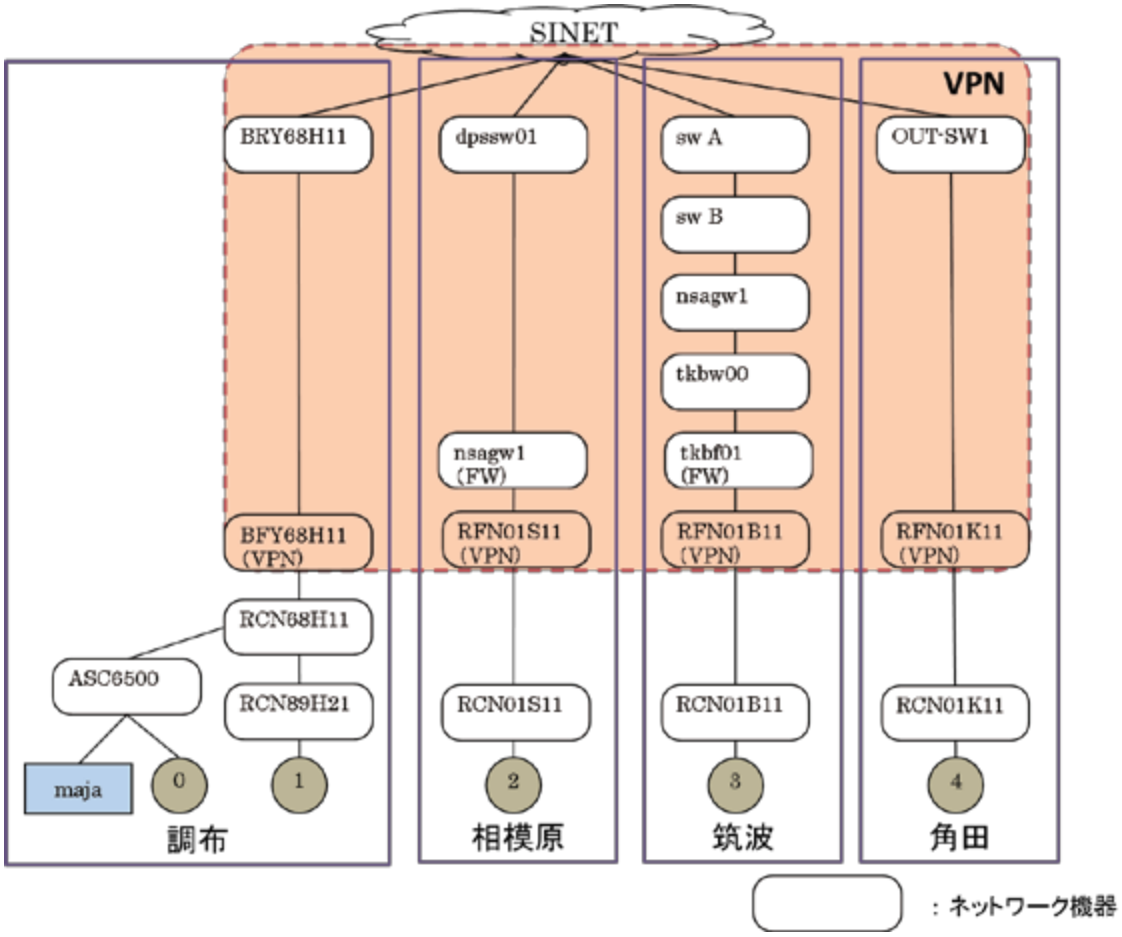
JAXA でも JSS を ALL JAXA の資源として使用するために，VPN ルータを用いて調布・相模原・筑波・角田の４事業所の事業所間を相互接続してきたが，遠隔地からのファイル転送性能についてはユーザから改善要望が出ていた．

そこで，セキュリティ確保と回線品質の向上のために，事業所間接続を SINET-L2VPN 網に変更することを検討した．また，距離の問題に対応するため，サーバの TCP パラメータの調整方針についても検討を行った．

本稿では，JSS に対する調布・相模原・筑波・角田の各事業所からのネットワーク性能とファイル転送性能の測定を実施することで，ネットワーク構成による性能の違いを分析した結果を記述する．

2. ネットワーク構成と測定点

図 2-1 に VPN で構築した JSSnet の構成と，本実験の測定点を示す.

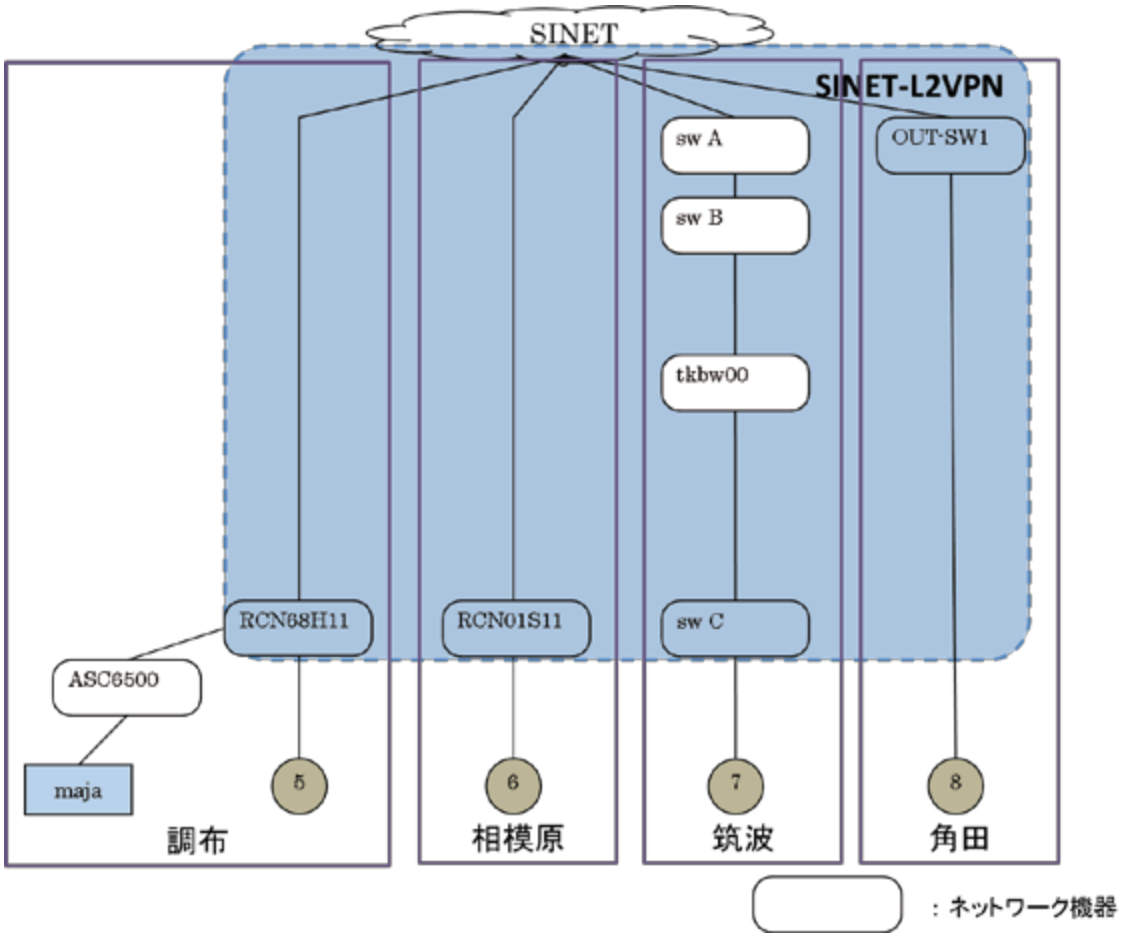


測定点	測定点名
0	調布スパコン SW
1	調布 VPN
2	相模原 VPN
3	筑波 VPN
4	角田 VPN

図 2-1　JSSnet（VPN）の物理構成と測定点

図 2-2 に，SINET-L2VPN 網を利用したネットワーク構成と，実験の測定点を示す.

SINET-L2VPN 網



測定点	測定点名
5	調布 L2
6	相模原 L2
7	筑波 L2
8	角田 L2

図 2-2 SINET-L2VPN 網の物理構成と測定点

3. 実験計画

3.1. 測定機器構成

(1) クライアント

図 2-1 および図 2-2 の各測定点にクライアント端末を接続した. クライアント端末の性能を表 3-1 に，TCP パラメータを表 3-2 に示す．クライアント端末の TCP パラメータは，以下のコマンドで取得した.

```
# more /proc/sys/net/ipv4/tcp_?mem
# more /proc/sys/net/core/?mem_max
```

表 3-1 クライアント端末スペック

ホスト名	db-ssd
OS	Linux 2.6.31.12-174.2.3.fc12.i686
CPU	Intel C2D U9400 1.40GHz
メモリ	2892 [MB]
ディスク	TOSHIBA THNS128G

表 3-2 クライアント端末 TCP パラメータ

パラメータ名	説明	サイズ(min,default,max)
tcp_rmem	受信ウィンドウサイズ	4096, 87380, 3391488 [B]
tcp_wmem	送信ウィンドウサイズ	4096, 16384, 3391488 [B]

(2) サーバ

図 2-1 および図 2-2 の maja と記した位置にサーバが設置されている．サーバの性能を表 3-3 に，サーバの TCP パラメータを表 3-4 に示す．サーバの TCP パラメータは，以下のコマンドで取得した.

```
$ ndd -get /dev/tcp パラメータ名
```

表 3-3 サーバスペック

ホスト名	maja0.jss.in-jaxa
OS	SunOS 5.10
CPU	SPARC 64 VII
メモリ	1 [TB]
ディスク	/tmp

表 3-4 サーバ TCP パラメータ

パラメータ名	説明	サイズ
tcp_recv_hiwat	デフォルト受信ウィンドウサイズ	49152 [B]
tcp_max_buf	最大バッファサイズ	1048576 [B]
tcp_cwnd_max	最大輻輳ウィンドウサイズ	1048576 [B]

(3) L システム

測定点 0 から測定点 2～4 方向への TCP 性能測定の測定を行うために，測定点 2～4 相当の位置に設置されている各事業所の L システムをサーバとし，測定点 0 に接続したクライアント端末から測定を行った．L システムの性能を表 3-5 に，L システムの TCP パラメータを表 3-6 に示す．サーバの TCP パラメータは，以下のコマンドで取得した．

$ ndd -get /dev/tcp パラメータ名

表 3-5 L システムスペック

ホスト名	kakuta.jss.in-jaxa sagami.jss.in-jaxa tsukuba.jss.in-jaxa
OS	SunOS 5.10
CPU	SPARC 64 VII
メモリ	256 [GB]
ディスク	/tmp

表 3-6 L システム TCP パラメータ

パラメータ名	説明	サイズ
tcp_recv_hiwat	デフォルト受信ウィンドウサイズ	131072 [B]

※ 1 TCP 受信ウインドウについて[1)]

TCP は信頼性のあるデータリンクを提供するため，リモートの TCP との間で受信確認やエラー検出，フロー制御などの一種のハンドシェイク通信を行う．

しかし，WAN のような長距離ネットワークでは，セグメントごとの受信確認はパフォーマンスを低下させるため，スライディングウインドウという処理を行う．図 3-1 に，スライディングウインドウによるセグメント送信の概念を示す．ここではウインドウサイズとしてセグメント 4 個分の大きさが設定されているとする．

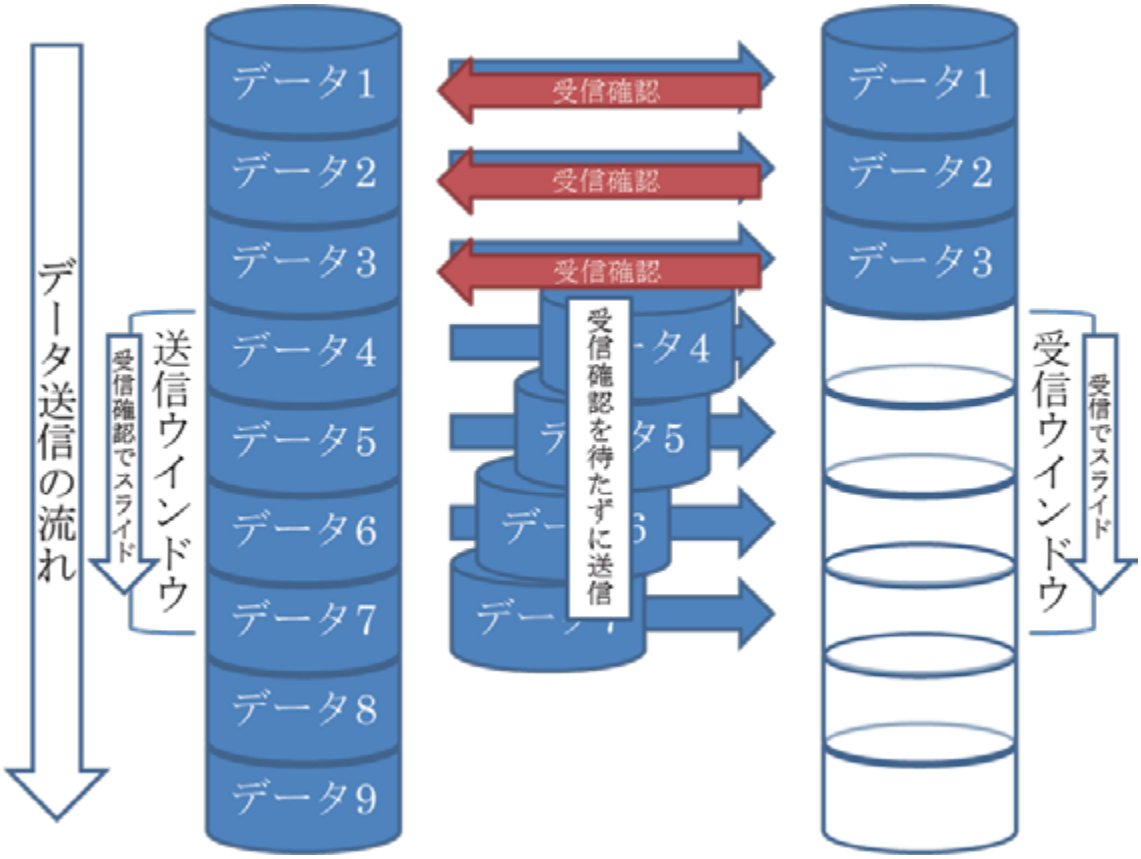


図 3-1 TCP ウインドウの解説図

ウィンドウは受信側からの受信確認があるごとにスライドしていく．セグメントには順序を示す番号が振られており，受信側ではセグメントが連続で送られてくると期待して特定の番号を持つセグメントを待つ．そして，確認応答を返す前に次のセグメントを受け取った場合は，それまでのセグメントの受信確認は省略して最後のセグメントの受信確認応答だけを返す．これにより，送信側で途中のセグメントの確認応答が返ってこなくとも，後のセグメントの応答が返ってくれば，それ以前のセグメントがすべて到着したとみなす．このようにして確認応答のオーバーヘッドを減少させることができる．しかし，一定期間内に受信応答が無い場合は，確認待ちのセグメント(現在のウインドウ内に入っているセグメント)をすべて再送するので，再送時の効率はよくない．

受信ウインドウの値は，このスライディングウインドウのサイズを指定するものであり，受信側が受け取ったセグメントをバッファリングできるサイズである．

3.2. 測定ツール

測定に使用したアプリケーションとそのバージョンを以下に示す.

(1) ネットワーク性能測定ツール

サーバ，クライアントとも

iperf version 2.0.4 (7 Apr 2008) pthreads

(2) ftp デーモン，ftp クライアント

サーバ側

ftpd version wu-2.6.2+Sun

クライアント側

lftp-4.0.0-1.fc12.i686 (ftp-0.17-51.fc12.i686)

(3) SSH 環境（scp，sftp）

サーバ側

Sun_SSH_1.1，SSH プロトコル 1.5/2.0，OpenSSL 0x0090704f

クライアント側

OpenSSH_5.2p1, OpenSSL 1.0.0-fips-beta3

3.3. 測定項目

本実験の測定項目と測定方法を以下に示す.

(1) 往復遅延時間

ping コマンドにより RTT を測定する．これを不定期に 5 回実施する.

クライアントより下記を実行

```
# ping –c 100 maja0.jss.in-jaxa
```

(2) TCP 性能測定

iperf により TCP 通信でのネットワーク性能を測定する．iperf は通信性能を測定するツールであるが，通信を並列に行うことによって回線の最大転送性能を測定することができる．回線圧迫により他の通信への影響が大きくなるため，測定は 1 回の実施とする.

サーバ側で iperf をサーバモードで起動

```
# iperf –s
```

クライアント側で下記を実行する際に，並列数 –P を変更
iperf –c 202.26.66.28 –P *並列数*

(3) UDP 性能測定

回線性能測定装置 Nextstream 100G を使用して，UDP 通信でのネットワーク性能を測定する．回線帯域の 10%から 100%までの 10 段階で UDP のバースト通信をそれぞれ 3 秒間実施して，そのフレームロス率を測定する．フレームロス率が 5%を超えた時点で測定を終了する．UDP のフレームサイズは 64 [B] と 1518 [B] の 2 種類を測定する．TCP 性能測定と同様に 1 回の実施とする．

(4) ファイル転送性能測定

ファイル転送プロトコルとして，ftp，sftp，scp を用いる．ゼロデータで生成した 100 [MB] のファイルを対象に，クライアント端末からみて put 方向/get 方向のファイル転送の性能測定を連続して 3 回行った平均値を，それぞれ取得する．この一連の測定を実施するスクリプトを作成し，不定期に 5 回実行する．

- **ftp**

バイナリモードで put/get を行う．ftp はバッチモードで使用し，put/get コマンドの結果表示より，転送速度を記録する．

- **sftp**

sftp はバッチモードで使用し，time コマンドによる時間測定を実施する．転送性能はファイルサイズを実時間（real）で割ったもので求める．
time sftp –b バッチファイル ユーザ名@maja0.jss.in-jaxa

- **scp**

下記コマンドにより，put 方向/get 方向の転送を行い，time コマンドによる時間測定を実施する．転送性能はファイルサイズを実時間（real）で割ったもので求める．

put 方向
time scp 100Mfile ユーザ名@maja0.jss.in-jaxa:/tmp/100M_実行日時_実行回数_put
get 方向
time scp ユーザ名@maja0.jss.in-jaxa:/tmp/100M_実行日時_実行回数 100M_実行日時_実行回数

4. 測定結果

4.1. JSSnet（VPN）での測定（測定点 1～4）

（1） 往復遅延時間

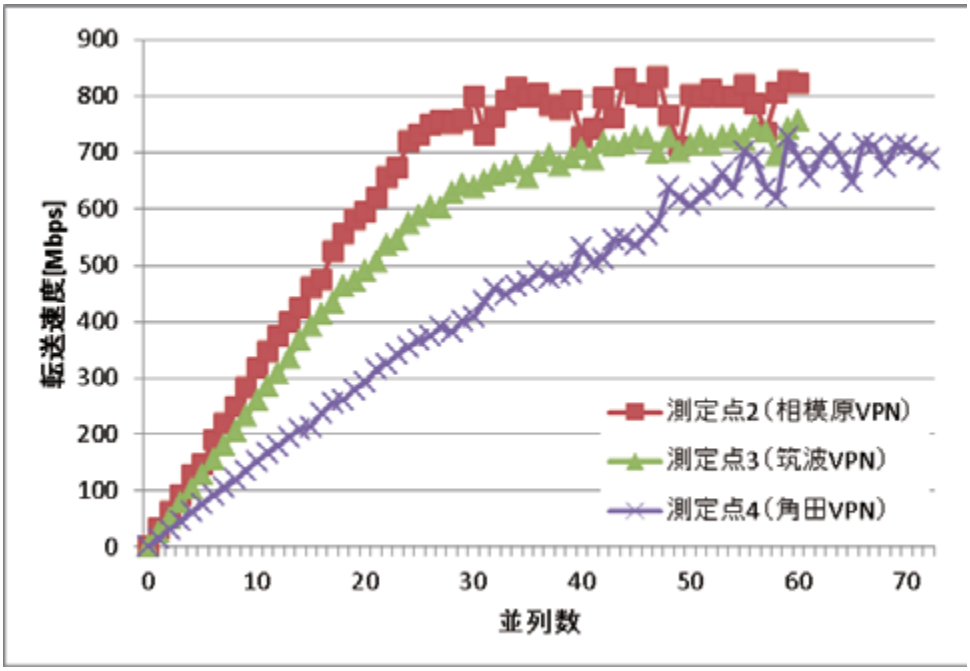
各測定点から maja.jss.in-jaxa までの往復遅延時間を，表 4-1 に示す.

表 4-1 往復遅延時間（JSSnet（VPN））

測定点	測定点名	RTT [ms]					
		1	2	3	4	5	平均
1	調布 VPN	0.415	0.404	0.443	0.372	0.393	**0.405**
2	相模原 VPN	9.993	10.039	10.009	9.998	10.010	**10.006**
3	筑波 VPN	13.806	12.040	12.038	12.015	12.035	**12.387**
4	角田 VPN	20.997	20.912	20.936	21.019	20.980	**20.969**

（2） TCP 性能測定

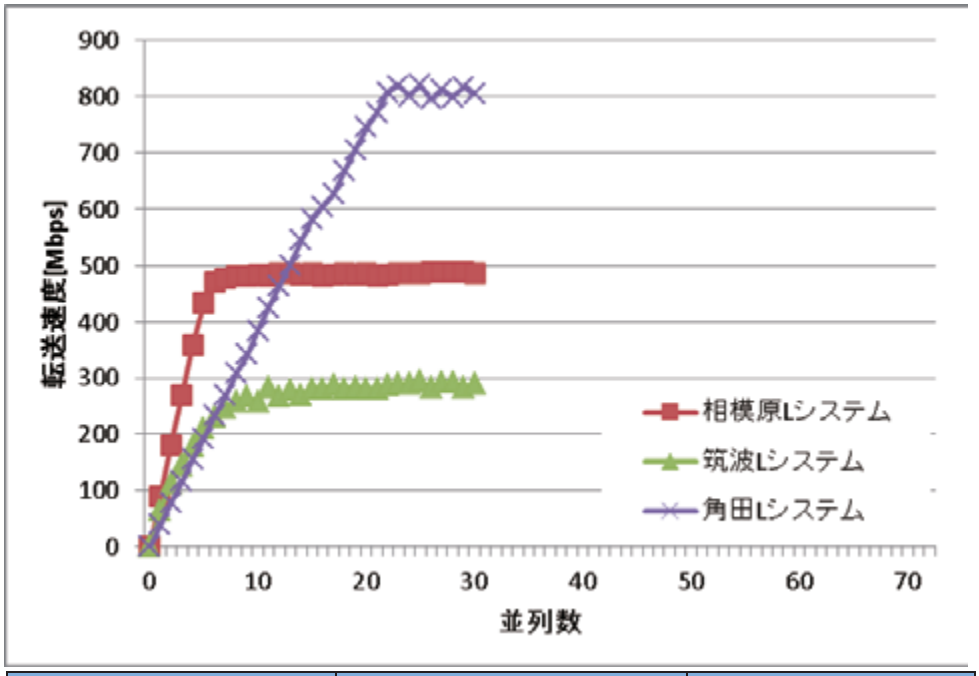
各測定点からサーバ（maja0.jss.in-jaxa）に対する TCP 性能の測定結果を，図 4-1 のグラフに示す. 調布からは, 並列数=1 の時点からピークに達するため，測定を実施していない.



測定点	測定点名	転送速度（並列数=1）	最大転送速度
2	相模原 VPN	32.1	833.0
3	筑波 VPN	25.9	756.0
4	角田 VPN	15.4	726.0

図 4-1 TCP 性能測定結果（JSSnet(VPN) maja0 への iperf）

また，逆方向の TCP 性能を測定するために，測定点 0 から測定点 2～4 相当に設置されている各事業所の L システムに対する iperf による測定を同様に実施した．結果を図 4-2 のグラフに示す．



測定先	転送速度（並列数=1）	最大転送速度
相模原 L システム	89.4	488.0
筑波 L システム	65.1	295.0
角田 L システム	38.6	820.0

図 4-2　TCP 性能測定結果（JSSnet(VPN)　各 L System への iperf）

（3） UDP 性能測定

各測定点からの UDP 性能の測定結果を，表 4-2 に示す.

表 4-2 UDP 性能測定結果（JSSnet(VPN)）

測定点	測定点名	往路（調布向け）		復路（調布から）	
		64 [B]	1518 [B]	64 [B]	1518 [B]
2	相模原 VPN	測定失敗			
3	筑波 VPN	9.7 [%]	35.5 [%]	51.6 [%]	35.5 [%]
4	角田 VPN	60.4 [%]	90.6 [%]	59.4 [%]	91.9 [%]

（4） ファイル転送性能測定

各測定点からのファイル転送性能の測定結果を，表 4-3 に示す.

表 4-3 ファイル転送性能測定結果（JSSnet(VPN)）

測定点	測定点名	方向	アプリ	転送性能 [MB/s]				
				1	2	3	4	5
1	調布 VPN	get	ftp	100.670	94.726	92.151	94.235	95.456
			scp	21.092	22.381	22.558	21.456	21.464
			sftp	20.972	20.912	21.696	21.284	18.146
		put	ftp	69.300	68.523	52.815	61.983	77.529
			scp	22.273	21.722	22.358	18.304	20.846
			sftp	18.026	22.456	21.952	21.583	17.751
2	相模原 VPN	get	ftp	3.233	2.638	2.986	3.161	3.786
			scp	4.182	3.884	3.943	3.478	4.440
			sftp	4.288	3.818	3.558	3.631	4.036
		put	ftp	3.774	3.710	3.681	3.807	3.750
			scp	3.796	3.780	3.722	3.716	3.826
			sftp	2.947	2.919	2.980	2.965	2.894
4	筑波 VPN	get	ftp	5.042	12.925	8.959	11.453	11.910
			scp	4.627	12.323	14.521	12.661	13.287
			sftp	5.644	15.491	12.251	14.814	14.318
		put	ftp	3.135	3.109	3.141	3.122	3.125
			scp	3.068	3.095	3.066	3.111	3.116
			sftp	2.146	2.195	2.240	2.187	2.165
3	角田 VPN	get	ftp	40.292	42.983	42.554	33.611	42.257
			scp	19.323	19.159	19.494	17.255	19.347
			sftp	18.555	18.544	18.060	14.466	18.088
		put	ftp	1.761	1.817	1.790	1.799	1.825
			scp	1.904	1.910	1.840	1.896	1.901
			sftp	0.912	0.901	0.912	0.913	0.926

4.2. SINET-L2VPN 網での測定（測定点 5～8）

(1) 往復遅延時間

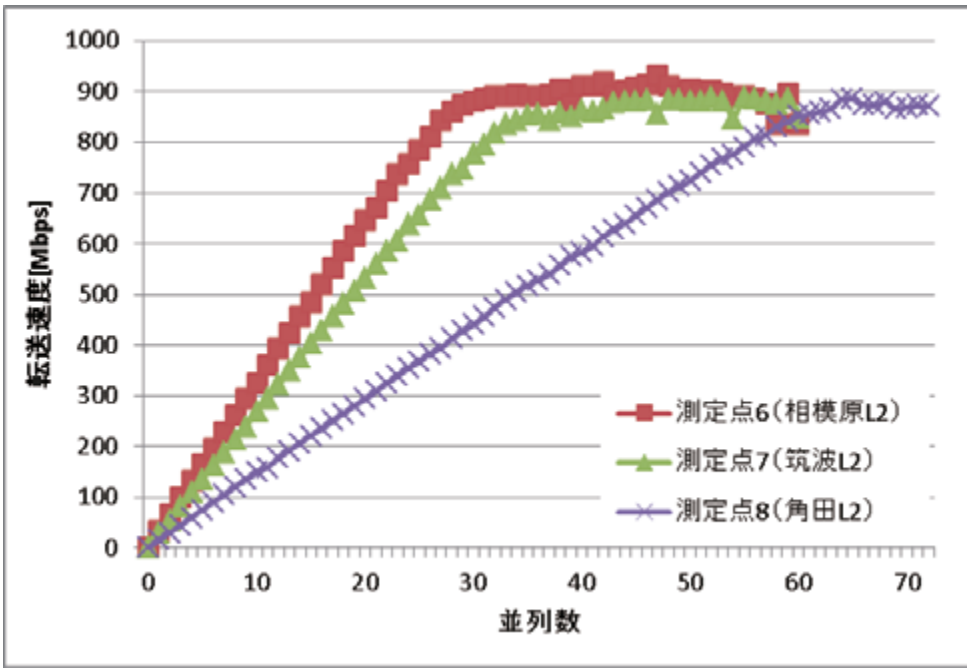
各測定点から maja.jss.in-jaxa までの往復遅延時間を，表 4-4 に示す.

表 4-4 往復遅延時間（SINET-L2VPN 網）

測定点	測定点名	RTT [ms]					
		1	2	3	4	5	平均
5	調布 L2	0.397	0.395	0.413	0.361	0.367	**0.378**
6	相模原 L2	9.335	9.256	9.268	9.240	9.228	**9.265**
7	筑波 L2	11.383	11.315	11.362	11.381	11.374	**11.363**
8	角田 L2	20.518	20.510	20.553	20.509	20.574	**20.533**

(2) TCP 性能測定

各測定点からの iperf の測定結果を，図 4-3 にグラフで示す．調布からは，並列数が 1 の時点からピークに達するため，測定を実施していない.



測定点	測定点名	転送速度（並列数=1）	最大転送速度
6	相模原 L2	33.0	931.0
7	筑波 L2	27.4	889.0
8	角田 L2	14.9	885.0

図 4-3 TCP 性能測定結果（SINET-L2VPN 網 maja0 への iperf）

(3) UDP 性能測定

各測定点からの UDP 性能の測定結果を，表 4-5 に示す.

表 4-5 UDP 性能測定結果（SINET-L2VPN 網）

測定点	測定点名	往路（調布向け）		復路（調布から）	
		64 [B]	1518 [B]	64 [B]	1518 [B]
6	相模原 L2	100 [%]	100 [%]	100 [%]	100 [%]
7	筑波 L2	94.0 [%]	99.3 [%]	91.5 [%]	93.8 [%]
8	角田 L2	95.4 [%]	99.7 [%]	95.4 [%]	99.7 [%]

(4) ファイル転送性能測定

各測定点からのファイル転送性能の測定結果を，表 4-6 に示す.

表 4-6 ファイル転送性能測定結果（SINET-L2VPN 網）

測定点	測定点名	方向	アプリ	転送性能 [MB/s]				
				1	2	3	4	5
5	**調布 L2**	**get**	**ftp**	**97.728**	**106.031**	**105.924**	**111.400**	**107.774**
			scp	20.592	23.042	22.613	22.632	22.127
			sftp	22.307	20.984	22.062	22.030	21.914
		put	ftp	66.967	65.637	74.229	64.585	73.294
			scp	22.275	22.136	22.127	21.728	21.650
			sftp	22.779	21.999	21.039	21.001	21.789
6	相模原 L2	get	ftp	84.525	83.345	84.528	82.882	81.333
			scp	20.994	21.308	21.322	21.400	21.496
			sftp	19.883	19.725	19.896	19.730	20.027
		put	ftp	3.896	3.876	3.886	3.927	3.886
			scp	3.925	3.901	3.896	3.913	3.916
			sftp	3.777	3.766	3.782	3.769	3.778
7	筑波 L2	get	ftp	72.300	70.094	68.821	69.453	70.281
			scp	20.150	19.767	18.082	18.058	18.306
			sftp	19.157	19.519	17.146	18.602	18.088
		put	ftp	3.188	3.178	3.198	3.195	3.195
			scp	3.233	3.214	3.190	3.209	3.202
			sftp	3.100	3.107	3.089	3.071	3.081
8	角田 L2	get	ftp	41.042	41.049	41.211	40.762	40.552
			scp	18.802	19.033	18.153	18.838	18.741
			sftp	18.095	18.171	18.013	17.686	17.660
		put	ftp	1.794	1.783	1.782	1.786	1.780
			scp	1.830	1.816	1.818	1.817	1.806
			sftp	1.767	1.764	1.760	1.764	1.756

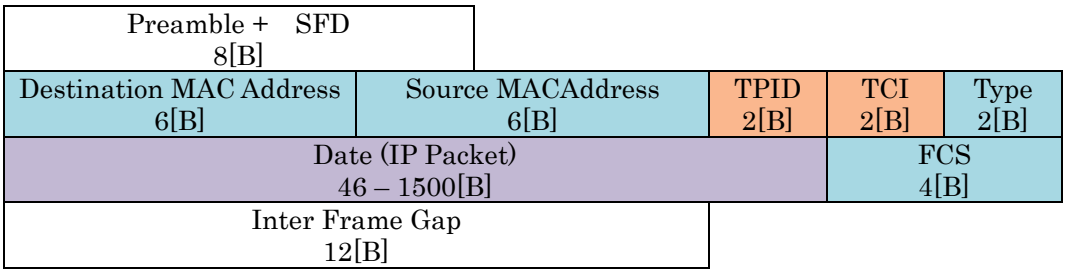
5. 結果の分析

5.1. TCP 性能

事業所間の TCP 性能測定の結果において，JSSnet(VPN)と SINET-L2VPN 網とで明確な差が見られる．これはネットワーク環境の構成要素（ネットワークスイッチ，VPN ルータ，ファイヤーウォール）の違いによるものと考えられる．

SINET-L2VPN 網では，TCP 性能測定で iperf の並列数を多くすることによって 900 [Mbps]前後の通信性能に至った．この結果はネットワーク帯域の 95 [%]を利用できており，良好なネットワーク環境であると言える．一方 JSSnet(VPN)では，700 [Mbps]～800 [Mbps]の通信性能で頭打ちとなった．この要因の一つとして，VPN ルータでの暗号化処理により IP フラグメンテーションが発生していることが挙げられる．IP フラグメンテーションが発生しないようにするには，サーバおよびクライアントの MTU を現状の 1500 [B]から 1448 [B]以下にする必要がある．以下にその詳細を説明する．

イーサネットでは 1 フレームが，ヘッダ部 14 [B]＋データ部（最大で 1500 [B]）＋チェックサム 4 [B]で構成される．さらに，VLAN を使用しているため，ヘッダ部に 4 [B]が付加されることから 1 フレームは最大 1522 [B]である．また，1 フレームを送信する間に，フレーム間ギャップ 12 [B]とフレーム前プリアンブル 8 [B]が必要なため，1 フレームを送信するのに必要なバイト数は合計で最大 1542 [B]となる（図 5-1）．



(Byte)

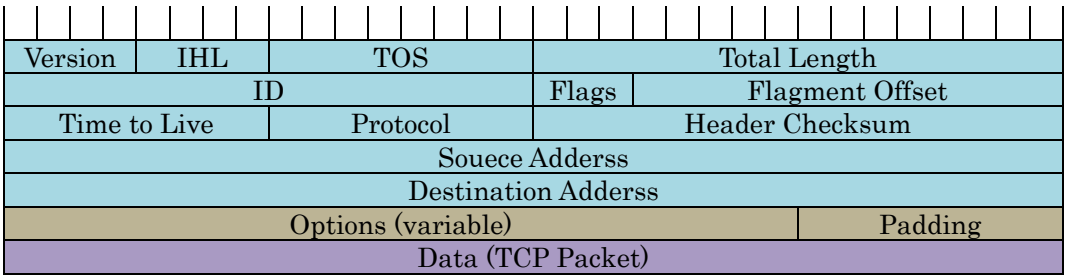
FSC:Frame Check Sequence

TPID:Tag Protocol Identifier

TCI:Tag Control Information

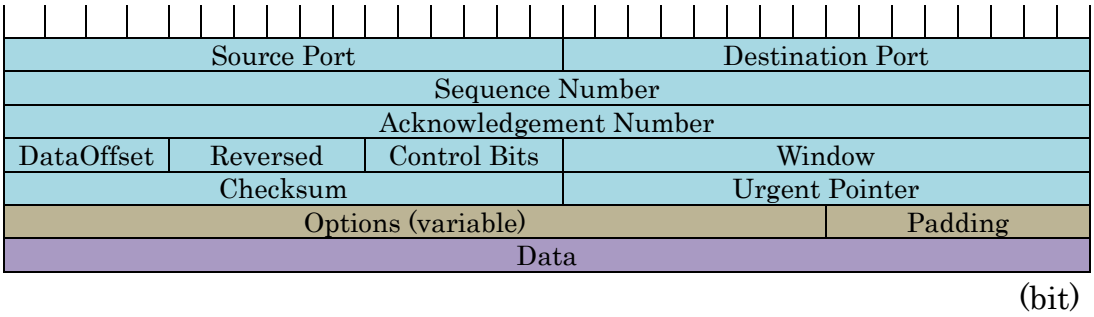
図 5-1 イーサネットフレーム（IEEE802.3 規格）

その内，データ部には IP および TCP のヘッダとして各 20 [B]が含まれるため，実際に転送できるデータ量としては 1460 [B]となる（図 5-2，5-3）．したがって，イーサネットで理論上，最大に利用できる割合は 1460 / 1542 = 0.947 であり，1 [Gbps]の帯域を有している場合での理論最大性能は 947 [Mbps]である．



IHL:Internet Header Length
TOS:Type of Service

図 5-2 IP パケット（RFC791）



Control Bits:URG , ACK , PSH , RST , SYN , FIN

図 5-3 TCP パケット（RFC793）

JSSnet(VPN)での VPN は，IPSec ESP のトンネルモードでなされており，暗号化方式として 3DES，認証方式として HMAC-SHA-1 を使用している．元となる IP パケットに，暗号化のための初期ベクトル 8[B]，パッド長 1 [B]，次ヘッダ 1 [B]を加え，これらを 8 [B]のブロックサイズに整えるためのパディング 0～7 [B]を付加する．これを暗号化した情報に対して，ESP ヘッダーとして 8 [B]，認証用データの 12 [B]が加わり，IP パケットのデータ部となる．ここに VPN 機器のアドレス等を示す IP ヘッダー20 [B]が付加されて通信が行われるため，合計で 50 [B] + パディング分だけパケットが大きくなる（図 5-4）．

元の IP パケットが 1500 [B]であった場合，パディングは 2 [B]必要であるので，増加分は 52 [B]となるが，イーサネットの規約上，データ部は最大 1500 [B]であるため，IP フラグメンテーションが発生する（図 5-5）.

IP フラグメンテーションが発生した場合の理論上の性能の割合は，1460 / (1542 + 114) = 0.882 となり，1 [Gbps]の帯域を有している場合での理論最大性能も 882 [Mbps]に下がる．実際には，パケットの分割およびフラグメント化されたパケットの再構成のためのルータの処理負荷も性能低下の原因となる.

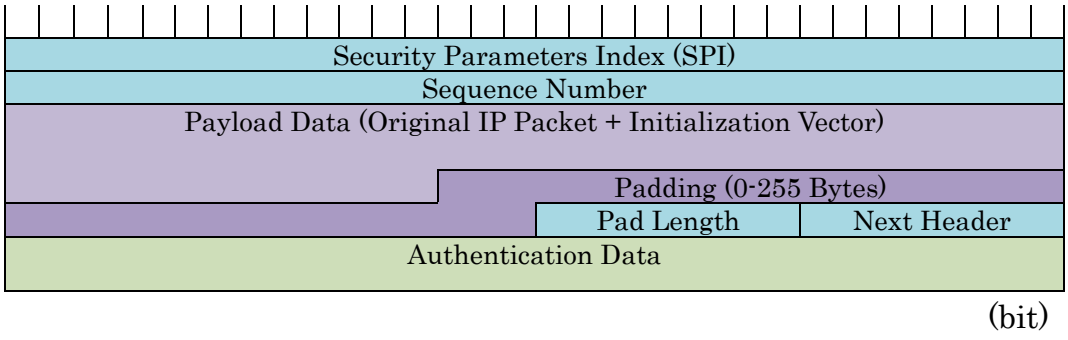


図 5-4　ESP パケット（RFC2406）

［元パケット］
IP ヘッダ＋TCP ヘッダ＋ペイロードデータ
　20 + 20 + 1460 = 1500 [B]

［ESP でカプセル化した IP パケット］
新 IP ヘッダ＋（ESP ヘッダ＋（データ部＋ESP トレイラ）＋認証データ）
　20 + (8 + (1500 + 8 + 1 + 1 + 2) + 12) = 1552 [B]

［IP フラグメンテーション］
新 IP ヘッダ＋フラグメントデータ
　20 + 1480 = 1500 [B]
　20 + 52 = 72 [B]

［イーサネットフレーム］
プリアンブル＋(イーサネットヘッダ＋データ部＋チェックサム)＋フレーム間ギャップ
　8 + (18 + 1500 + 4) + 12 = 1542 [B]
　8 + (18 + 72 + 4) + 12 = 114 [B]

図 5-5　ESP カプセル化からの IP フラグメンテーションの発生

図 5-2 に示した通り，通信方向を逆にした場合の TCP 性能は以下の点で異なる傾向を示した．

（1） 並列数=1 の転送速度が高い
（2） 相模原，筑波との最大転送速度が低い

（1）ついては，L システムの受信ウィンドウサイズ（tcp_recv_hiwat）が 128 [KB]と，maja の 48 [KB]に比較して大きいことが転送性能が高くなった原因である．TCP 通信の理論性能値は，受信ウィンドウサイズを RTT で割ったもので求められるため，1 並列あたりの転送性能もそれに比して高くなる（表 5-1）．

表 5-1 TCP Windowsize の比と並列数=1 の時の iperf 転送性能の比

	maja (A)	L System (B)	比率 (A/B)
受信ウィンドウサイズ [B]	49152	131072	0.375
転送速度（並列数=1）			
相模原	32.1	89.4	0.359
筑波	25.9	65.1	0.397
角田	15.4	38.6	0.399

(2)は，相模原への最大転送速度が約 500[Mbps]，筑波への最大転送速度能が約 300[Mbps]で頭打ちとなった結果を指している．これは，相模原および筑波の両事業所のインターネット口に設置されたファイヤーウォールで，攻撃検知による帯域制限が行われたこと，また，通信の輻輳を防ぐために TCP の機能の一つである輻輳制御（※2）が行われたことが原因である．

JSSnet のパケットは他のインターネット通信と同様に両事業所のファイヤーウォールを通過している．iperf での性能測定時，パケットが短時間で大量にフラグメント化された状態で到着したことをファイヤーウォールが攻撃と検知し，帯域制限を行ったことがログに記録されていた．ファイヤーウォールがパケットの一部を拒否したことにより，TCP 通信としては Timeout が返り，ウィンドウサイズが最大値である 1048576 [B]まで上昇せず小さなウィンドウサイズで通信が行われていたことをパケットキャプチャで確認した．

これに対しては，JSSnet の通信をファイヤーウォールを通過しない環境にすることが必要であるが，実運用上，ネットワーク構成を変更することは困難である．

※ 2　輻輳制御

TCP 通信では，パケット喪失が起きない場合にはネットワーク帯域に余裕があるとみなしてウィンドウサイズを増加させることで転送速度を上げ，パケット喪失が起きた場合にウィンドウサイズを減少させることで輻輳を回避するように転送量を制御する．

実装により制御アルゴリズムは異なるが，New Reno の場合では，ウィンドウサイズを最小値から指数的に増加（スロースタート・フェーズ）させた後に，ウィンドウサイズが閾値を超えた以降からは線形に増加（輻輳回避フェーズ）させる．輻輳発生によるパケット喪失時にはウィンドウサイズを半分に減少（Fast Recovery）させ，またタイムアウト時にはウィンドウサイズを最小値に減少させることにより，輻輳を回避する（図 5-6）．

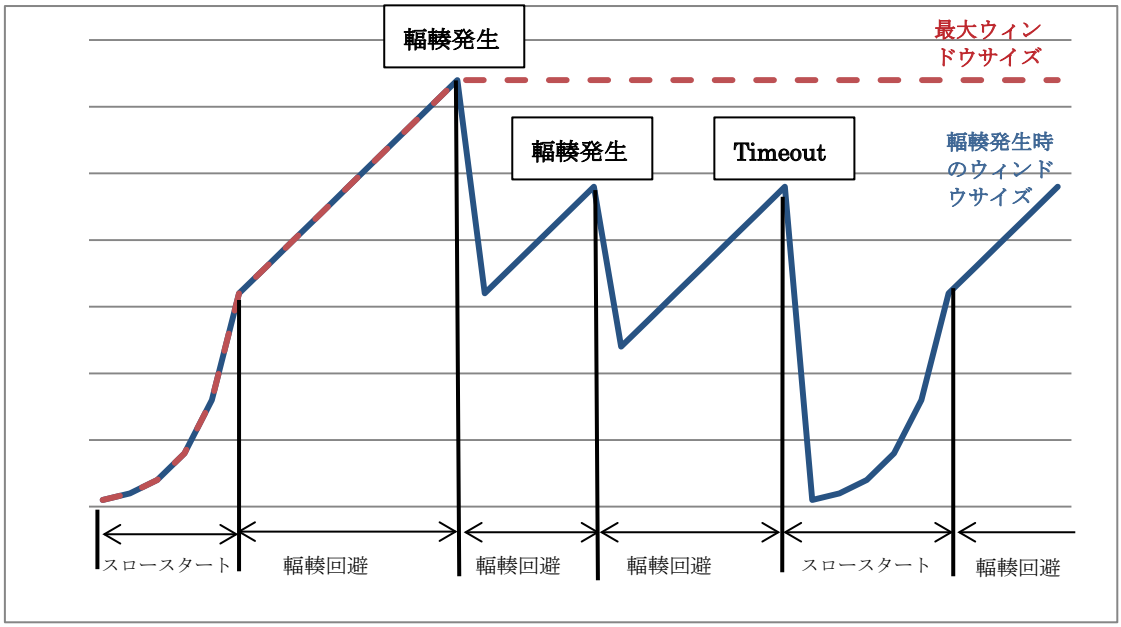


図 5-6　輻輳ウィンドウサイズの変化

5.2. UDP 性能

UDP 性能測定の結果において, SINET-L2VPN 網では, 相模原では 100 [%], 角田および筑波の往路でフレームサイズが 1518 [B]の場合に 99 [%]以上の負荷に耐えられることが判った．この場合の負荷はイーサネットのフレーム間ギャップ等を含むものである．筑波の復路で若干の通信性能の低下が見られた．物理的に同じネットワーク経路上に他のインターネット通信が通過していることの影響であると考えられる．測定時に他の通信を排除することはできないため, その通信量を推定して考慮する必要があったが，本実験では他の通信量を記録していない．フレームサイズが 64 [B]の場合には，角田および筑波で 95 [%]程度に低下している．小さいフレームサイズで負荷をかける分，フレーム数は多量になり，経路上のネットワーク機器でフレーム処理する負荷が増大した可能性が高い．ただし，カタログからは最大フレーム処理数等の情報は読み取ることができなかった.

JSSnet(VPN)の角田において，フレームサイズが 1518 [B]の場合に 90 [%]前後の性能であった．TCP 性能測定と同様に，VPN ヘッダーの付加によりフレーム分割損等の影響があったと考えられる．また，フレームサイズが 64 [B]の場合には 60 [%]前後の性能にとどまった．VPN ルータのパケット処理能力の影響である可能性が高いが，ネットワークスイッチと同様にカタログからは最大フレーム処理数の情報は読み取れなかった.

JSSnet（VPN）の筑波では，フレームサイズ 64 [B]の場合に性能が 9.7 [%]にとどまるなど，大幅に性能が下がった．TCP 性能測定と同様に，ファイヤーウォールによる帯域制限が発生した可能性が高い.

JSSnet(VPN)の相模原からの UDP 性能は，通信が確立できなかったため測定できなかった．ネットワーク構成上の問題であったのか検証が必要である.

5.3. ファイル転送性能

各事業所から maja に向けたファイル転送（ftp put 方向）では，ネットワーク構成によらず，非常に低い転送性能であった．これは maja の受信ウィンドウサイズが 49152 [B]と小さいことが原因である．maja から各事業所に向けたファイル転送（ftp get 方向）は，put 方向と比較して性能が高い．これは，クライアントの OS が Linux 2.6 系であり，受信ウィンドウサイズをパラメータで指定した最大値（3391488 [B]）まで自動的にチューニングする機能による．ただし，maja の最大輻輳ウィンドウサイズが 1048576 [B]であるため，両者の通信で実際に指定されるウィンドウサイズは 1048576 [B]に制限される．

受信ウィンドウサイズを put 方向で 49152 [B]，get 方向で 1048576 [B]であったとすると，TCP の理論性能値と ftp による実測値の比率は表 5-2 となる．（調布については媒体性能を超えることになるため，理論性能値は求めない．）

表 5-2 理論性能値と ftp 実測値の比較

方向	測定点	測定点名	RTT [ms]	理論性能 (A) [MB/s]	転送性能 (B) [MB/s]	比率 (A/B)
put	1	調布 VPN	0.290		66.030	
	2	相模原 VPN	10.010	4.683	3.774	80%
	3	筑波 VPN	12.032	3.896	3.126	80%
	4	角田 VPN	20.969	2.235	1.798	80%
	5	調布 L2	0.356		68.943	
	6	相模原 L2	9.265	5.059	3.894	77%
	7	筑波 L2	11.363	4.125	3.191	77%
	8	角田 L2	20.533	2.283	1.785	78%
get	1	調布 VPN	0.290		95.447	
	2	相模原 VPN	10.010	99.900	3.161	3%
	3	筑波 VPN	12.032	83.112	10.058	12%
	4	角田 VPN	20.969	47.689	40.339	85%
	5	調布 L2	0.356		105.772	
	6	相模原 L2	9.265	107.933	83.323	77%
	7	筑波 L2	11.363	88.005	70.190	80%
	8	角田 L2	20.533	48.702	40.923	84%

相模原および筑波からの JSSnet(VPN)での ftp get の結果以外については，理論性能値に対して 77～85 [%]程度の性能となり，ftp のプロトコルオーバーヘッドも含めると妥当な性能であると考えられる．相模原および筑波からの ftp get は，TCP 性能測定と同様，ファイヤーウォールによる帯域制限の影響である可能性が高い．

scp において，調布では 21~22[MB/s]にとどまり，ftp に比べて低い結果となった．これはデータ暗号化の処理コストが影響したと考えられる．データ暗号化の影響については次項で分析する．put 方向では maja の受信ウィンドウサイズがネックとなり，ftp とほぼ同値の，ネットワーク遅延から推測される値となっている（図 5-7）． また，get 方向では，JSSnet(VPN)の相模原および筑波で性能が低い結果となり，ftp と同じ傾向を示した（図 5-8）．

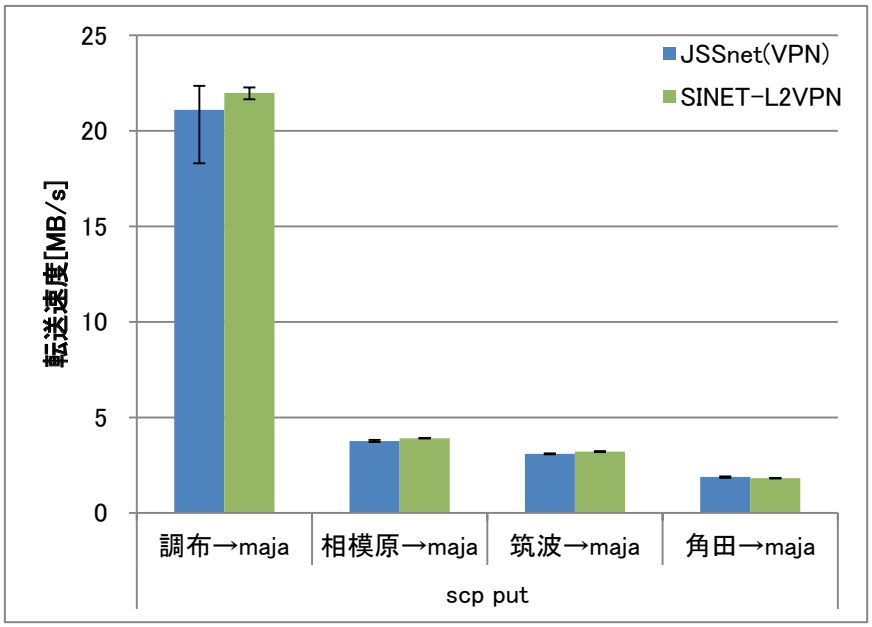


図 5-7 SCP put 性能比較

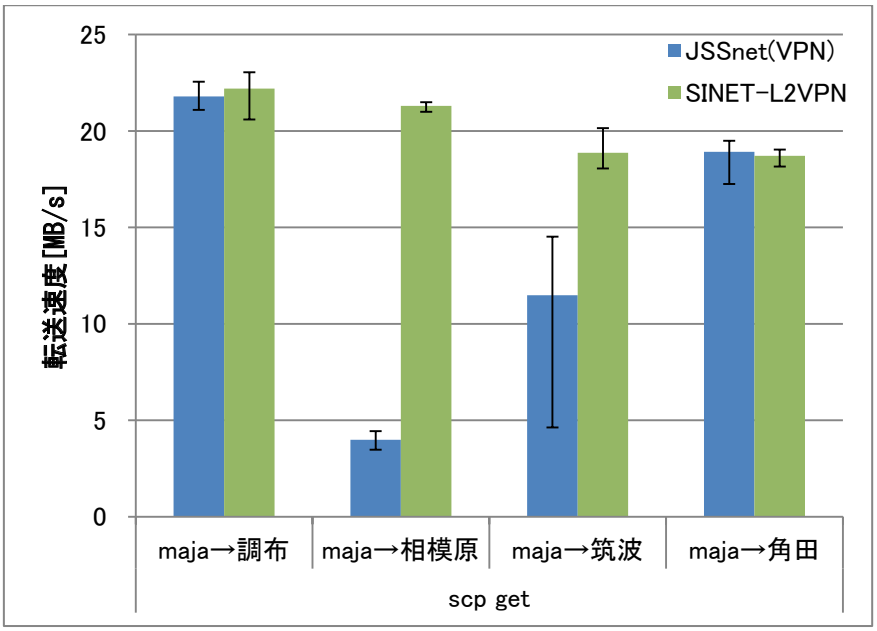


図 5-8 SCP get 性能比較

次に，sftp と scp の比較を行う．sftp はプロトコル上の通信のオーバーヘッドが scp よりも大きいため，転送速度は若干低くなる（図 5-9~5-12）．JSSnet(VPN)では SINET-L2VPN 網よりもこの傾向が顕著であるが，理由については分析できていない．JSSnet（VPN）の相模原および筑波の get 方向では，プロトコル上の通信のオーバーヘッドよりも帯域制限の影響が大きいため，この傾向が表れていない（図 5-11）．

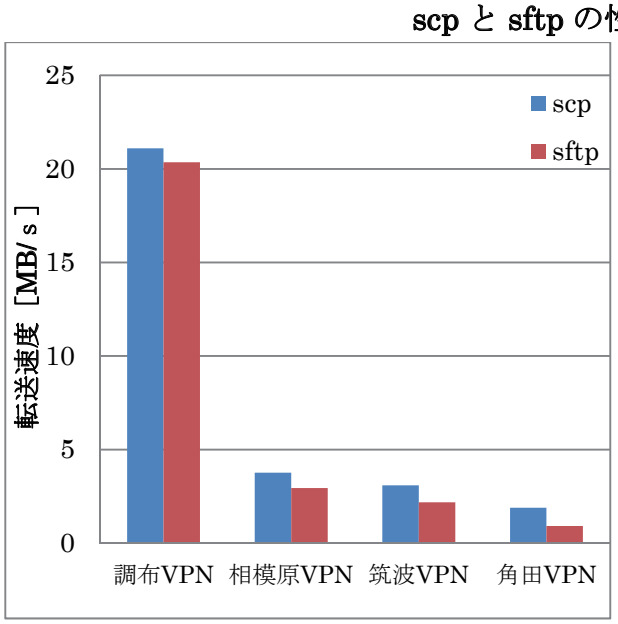


図 5-9 JSSnet(VPN) put 方向

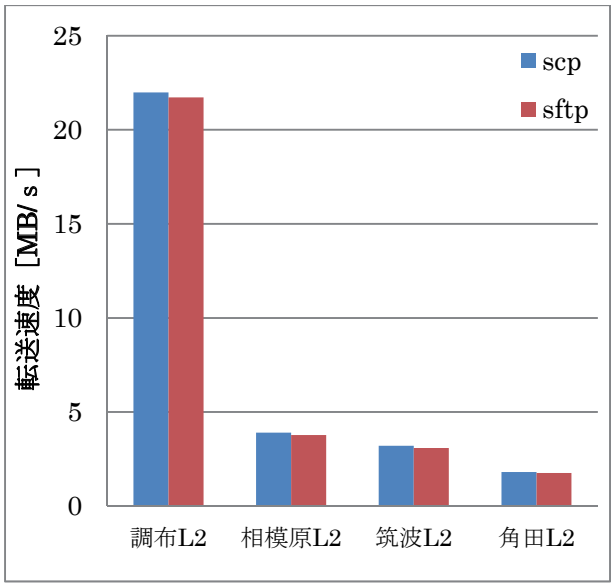


図 5-10 SINET-L2VPN put 方向

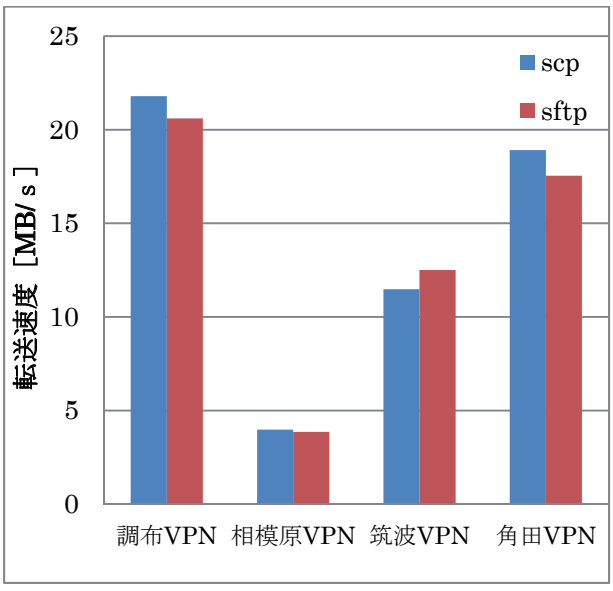


図 5-11 JSSnet(VPN) get 方向

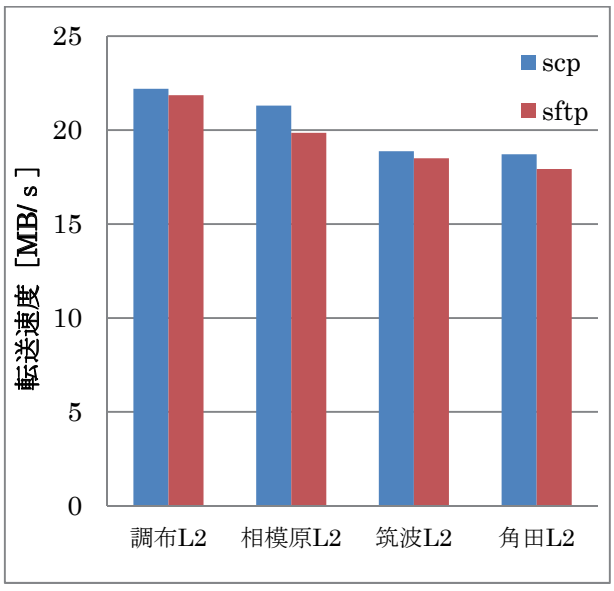


図 5-12 SINET-L2VPN get 方向

5.4. 暗号化方式の影響（scp，sftp）

scp および sftp で，ネットワーク構成によるボトルネックが無い調布において性能が頭打ちしていることから，データ暗号化の処理コストが転送性能のネックとなっていることが判る．本測定では OpenSSH のデフォルトの暗号化方式が aes128-ctr であるため aes128-ctr を用いたが，暗号化方式の処理コストの違いを検証するために，次の追加測定を行った．

100 [MB]のファイルを maja の/home から localhost（maja 自身）の/tmp に scp で転送した際のファイル転送性能を，暗号化方式を変えて測定した結果を表 5-3 に，グラフを図 5-13 に示す．

表 5-3 暗号化方式による scp ファイル転送性能の比較

暗号化方式	転送性能 [MB/s]	
	get	put
aes128-ctr	30.519	35.419
aes128-cbc	28.958	32.086
arcfour	42.313	55.453
3des-cbc	10.772	10.977
blowfish-cbc	30.644	35.088

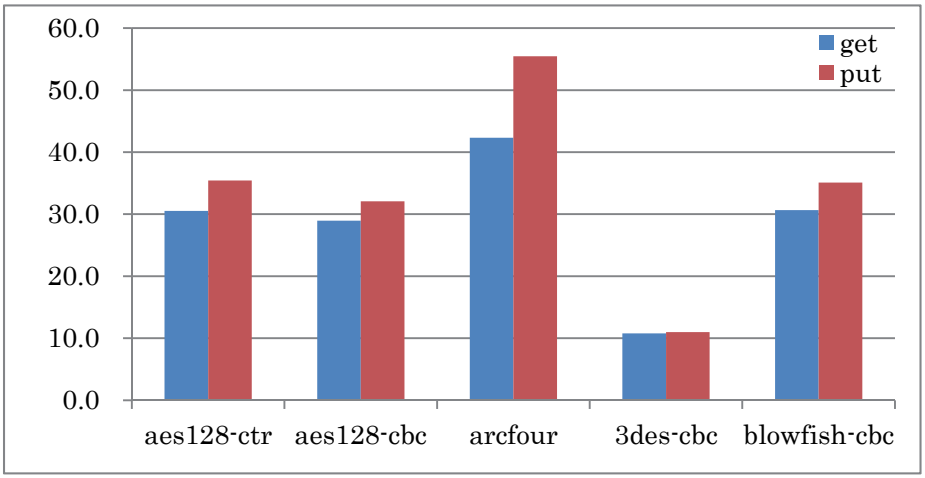


図 5-13 暗号化方式による scp ファイル転送性能の比較

測定結果より，暗号化方式によって 3～5 倍程度の違いが出ることが分かる．なお，暗号化方式はサーバ側が提供する暗号化方式であればクライアント側で選択することができるため，ユーザ自身によって性能向上が可能な範囲である．

この結果はネットワークを経由しない通信で得ているため，maja における scp 通信の最大性能値と言える．

6. 仮想プライベートネットワークの高速化

6.1. JSSnet 高速化

今回の測定で，VPN 通信によりフラグメンテーションが発生することで性能低下がおきていることが判った．インターネット上に仮想プライベートネットワークを構築する際に VPN ルータを利用することは一般的であるが，SINET のようにインターネット回線側で L2 接続サービスが提供されている場合は，外部からの盗聴等のリスクも少ないため，VPN ルータを利用しない構成とすることが可能である．また，相模原および筑波で想定される性能が出ない原因として，両事業所のインターネット口にあるファイヤーウォールで帯域制限されていたことが特定できた．性能向上のためにはファイヤーウォールを経由しない構成とする必要がある．

次に，TCP 性能およびファイル転送性能の測定において，maja 側で受信ウィンドウサイズなどの TCP パラメータの調整が必要であることが明らかになった．一般的に，受信ウィンドウサイズを大きくし過ぎるとコネクションごとにメモリを割り当てるために，システム全体でメモリ枯渇が問題になることがあるが，maja は 1[TB]と大きなメモリ領域を有しているため遅延をカバーするだけの十分な受信ウィンドウサイズが設定可能である．角田から 1 [Gbps] (125 [MB/s]) の帯域を十分に利用するには，受信ウィンドウサイズに，1 [Gbps] * 20 [ms] / 8 = 2.5 [MB] の大きさが必要となる．大きすぎるウィンドウサイズは，近距離では輻輳が発生してしまう要因となるが，ルート別メトリックを使用することで各事業所ごとに適切な受信ウィンドウサイズを設定することも可能である．ルート別メトリックの設定コマンドは以下である．

route change -net *a.b.c.d* -recvpipe *x*
ネットワーク *a.b.c.d* 宛のすべての接続には，デフォルト受信ウィンドウサイズ の代わりに，受信バッファーサイズ *x* を使用する．ルートが分けられていない場合は，事前にルートを追加する．

送信時の最大輻輳ウィンドウサイズ（tcp_cwnd_max）はルート別の設定ができないため，角田を基準に最大値を設定するしかない．tcp_cwnd_max は tcp_max_buf の値に制限されるため，同時に tcp_max_buf についても同値を設定する．

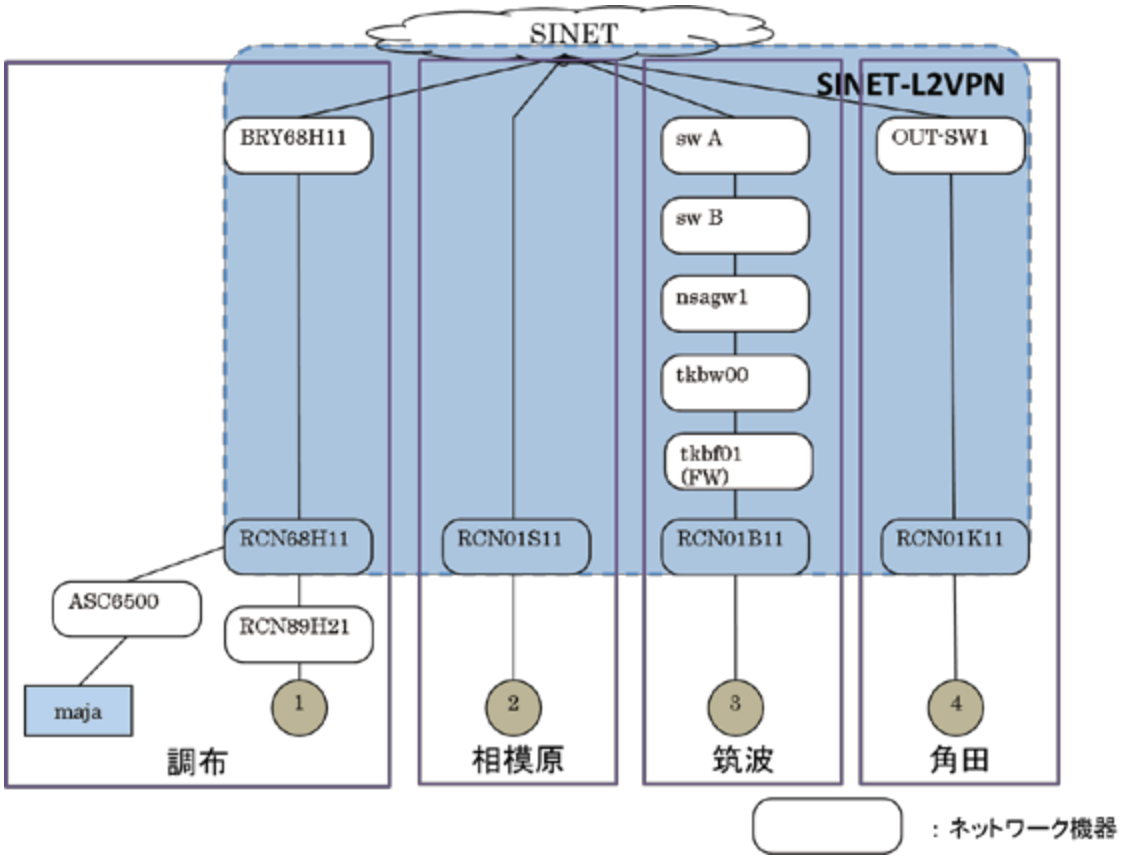
また，ファイル転送に scp や sftp を用いる場合には，暗号化の処理負荷がボトルネックとなる場合があることが判った．クライアント端末の性能を高くすればボトルネックが軽減される可能性があるが，maja 側で，aes128-ctr，aes128-cbc，arcfour，3des-cbc，blowfish が選択できるように設定されている

ので，クライアント端末側で暗号化プロトコルに arcfour を用いることでも，ファイル転送を高速化することができる．しかしながら，ネットワーク側に盗聴などのリスクが無いことが担保されているならば，ファイル転送は暗号化を行わない ftp で十分ではないかと考える．

6.2. JSSnet 構成変更後の検証

6.1 項の検討を経て， VPN ルータを切り離して SINET-L2VPN 網を用いた構成に JSSnet を移行した（図 6-1）．筑波のファイヤーウォールは JSSnet の管理所掌外であり，別途にネットワークを構築するには費用が必要であることから，経路上から切り離すことができなかった．

JSSnet（移行後）



測定点	測定点名
0	調布スパコン SW
1	調布 JSSnet
2	相模原 JSSnet
3	角田 JSSnet
4	筑波 JSSnet

図 6-1 JSSnet（移行後）の物理構成と測定点

また，maja 側の TCP パラメータを表 6-1 の通りに変更した．ルート別メトリックは運用が煩雑となるため採用しなかった．

表 6-1 サーバ TCP パラメータ

パラメータ名	説明	サイズ
tcp_recv_hiwat	デフォルト受信ウィンドウサイズ	2097152 [B]
tcp_max_buf	最大バッファサイズ	2097152 [B]
tcp_cwnd_max	最大輻輳ウィンドウサイズ	2097152 [B]

JSSnet 移行後の測定結果について，以下に示す．

(1) 往復遅延時間

各測定点から maja までの往復遅延時間を表 6-2 に示す．測定結果は 5 回の測定の平均値であり，誤差範囲は最大値と最小値を表す．また，移行前後の往復遅延時間の比較を図 6-2 に示す．経路の変化が無い調布を除いて，移行後は中継するネットワーク機器の数が減ったことにより RTT がわずかに減少した．

表 6-2 往復遅延時間（JSSnet 構成変更後）

測定点	測定点名	RTT [ms]					
		1	2	3	4	5	平均
1	調布 JSSnet	0.414	0.429	0.434	0.408	0.443	**0.426**
2	相模原 JSSnet	9.120	9.101	9.111	9.13	9.124	**9.117**
3	筑波 JSSnet	20.296	20.313	20.136	20.290	20.310	**20.269**
4	角田 JSSnet	11.457	11.458	11.411	11.369	11.383	**11.416**

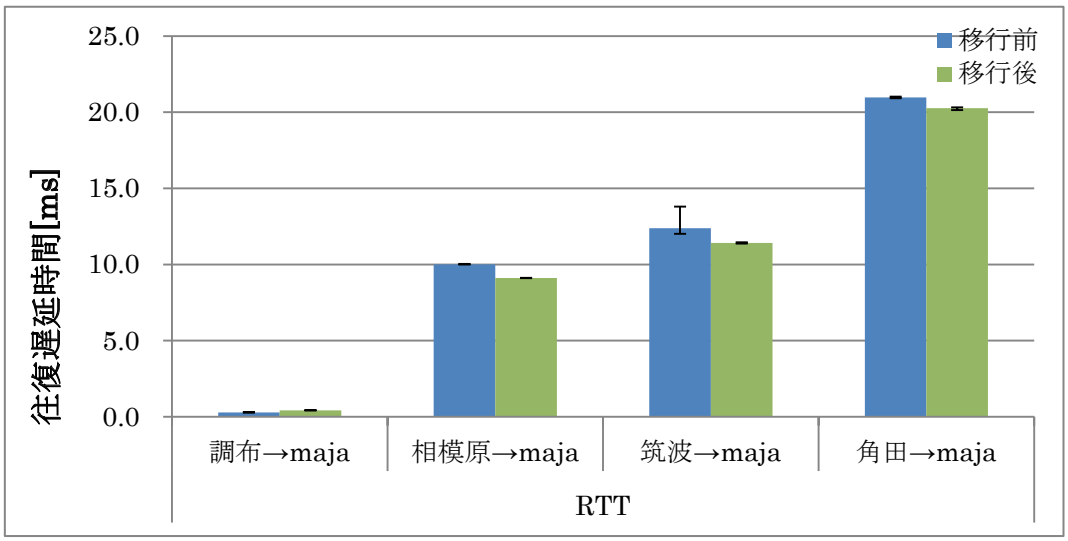


図 6-2 往復遅延時間の比較

（2） ファイル転送性能測定

各測定点からのファイル転送性能の測定結果を表 6-3 に示す．測定結果は 5 回の測定の平均値であり，誤差範囲は最大値と最小値を表す．また，移行前後の転送速度の比較を図 6-3 および図 6-4 に示す．

最大輻輳ウィンドウサイズ（tcp_cwnd_max）の見直しにより，get 方向の通信では調布を除いて転送速度が大きく向上した（図 6-3）．調布では輻輳の発生が多くなると考えられ，結果として転送速度が少し下がっている．また，デフォルト受信ウィンドウサイズ（tcp_recv_hiwat）の見直しにより，put 方向の通信では ftp の転送速度が大きく向上した（図 6-4）．しかし，put 方向の通信でも scp および sftp では，転送性能に大きな変化が見られなかった（図 6-4）．その後の調査により，SSH デーモンが scp および sftp で用いる受信ウィンドウサイズを固定し，その設定値である 48[KB]のウィンドウサイズで通信を行っていたことが判明した．今後，設定値を 2[MB]にした SSH デーモンを別ポートで起動し，サービス提供の試行を行う予定である．

表 6-3 ファイル転送性能測定結果（JSSnet 構成変更後）

測定点	測定点名	方向	アプリ	転送性能 [MB/s]				
				1	2	3	4	5
1	調布 JSSnet	get	ftp	88.849	100.888	68.026	93.458	103.771
			scp	19.254	19.195	19.177	18.768	17.552
			sftp	20.013	20.591	20.444	19.259	14.096
		put	ftp	87.470	95.611	89.387	89.737	101.255
			scp	19.690	20.122	19.728	18.772	15.329
			sftp	16.353	14.854	14.078	10.622	13.580
2	相模原 JSSnet	get	ftp	68.500	79.716	80.085	38.562	76.449
			scp	17.898	17.732	17.947	17.835	17.923
			sftp	17.265	17.963	17.807	17.737	16.483
		put	ftp	66.664	66.376	62.403	61.633	63.704
			scp	3.936	3.906	3.923	3.923	3.909
			sftp	3.764	3.746	3.752	3.759	3.750
3	角田 JSSnet	get	ftp	65.080	64.408	66.997	65.803	65.394
			scp	17.860	17.385	17.025	16.736	17.195
			sftp	17.992	17.890	17.263	16.616	17.270
		put	ftp	60.861	48.271	57.414	61.172	66.430
			scp	1.827	1.825	1.824	1.822	1.823
			sftp	1.780	1.777	1.777	1.776	1.775
4	筑波 JSSnet	get	ftp	37.782	35.094	53.789	38.991	30.794
			scp	17.058	16.648	18.969	19.361	18.102
			sftp	17.720	17.251	20.236	19.735	18.690
		put	ftp	68.388	63.319	58.352	52.007	64.095
			scp	3.176	3.183	3.198	3.207	3.186
			sftp	3.067	3.058	3.081	3.092	3.067

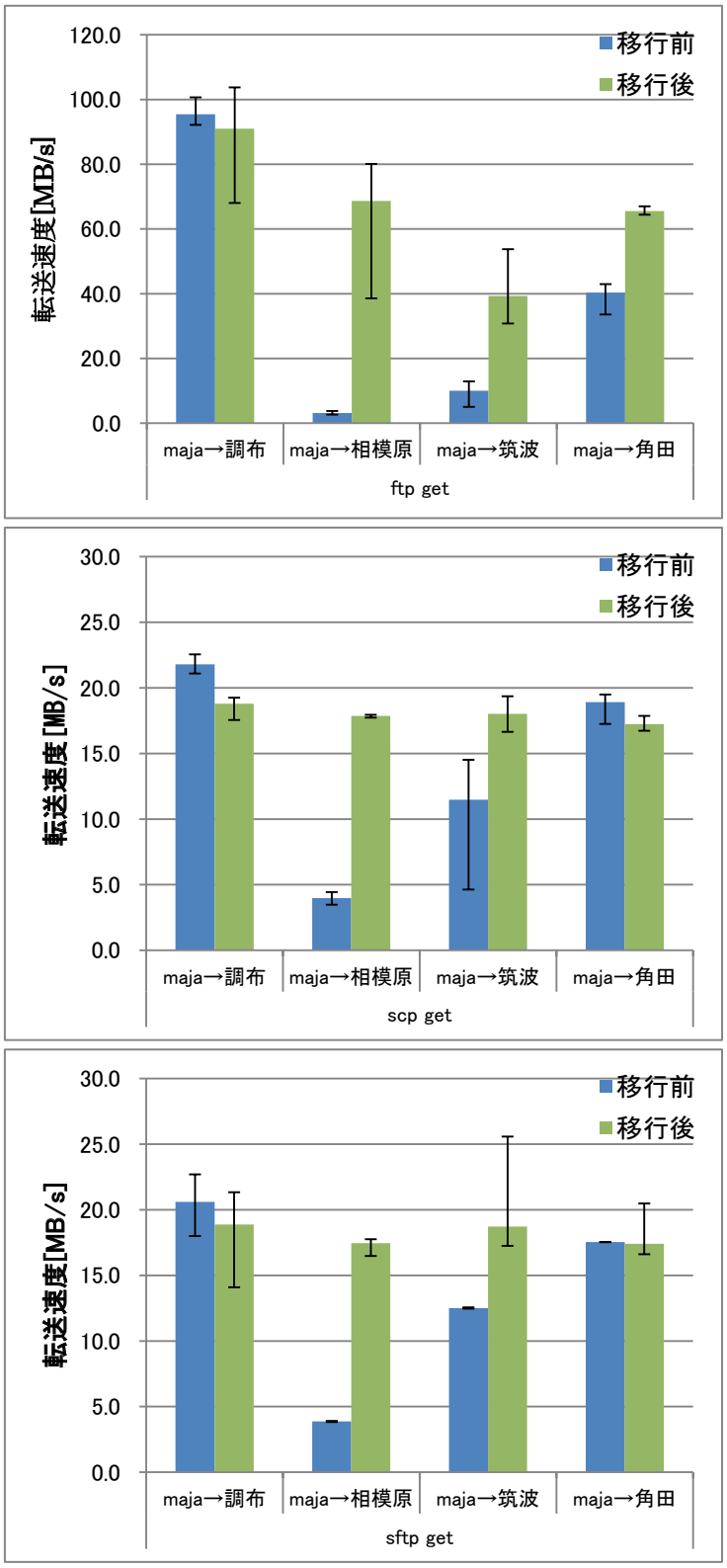


図 6-3 ファイル転送性能の比較（get 方向）

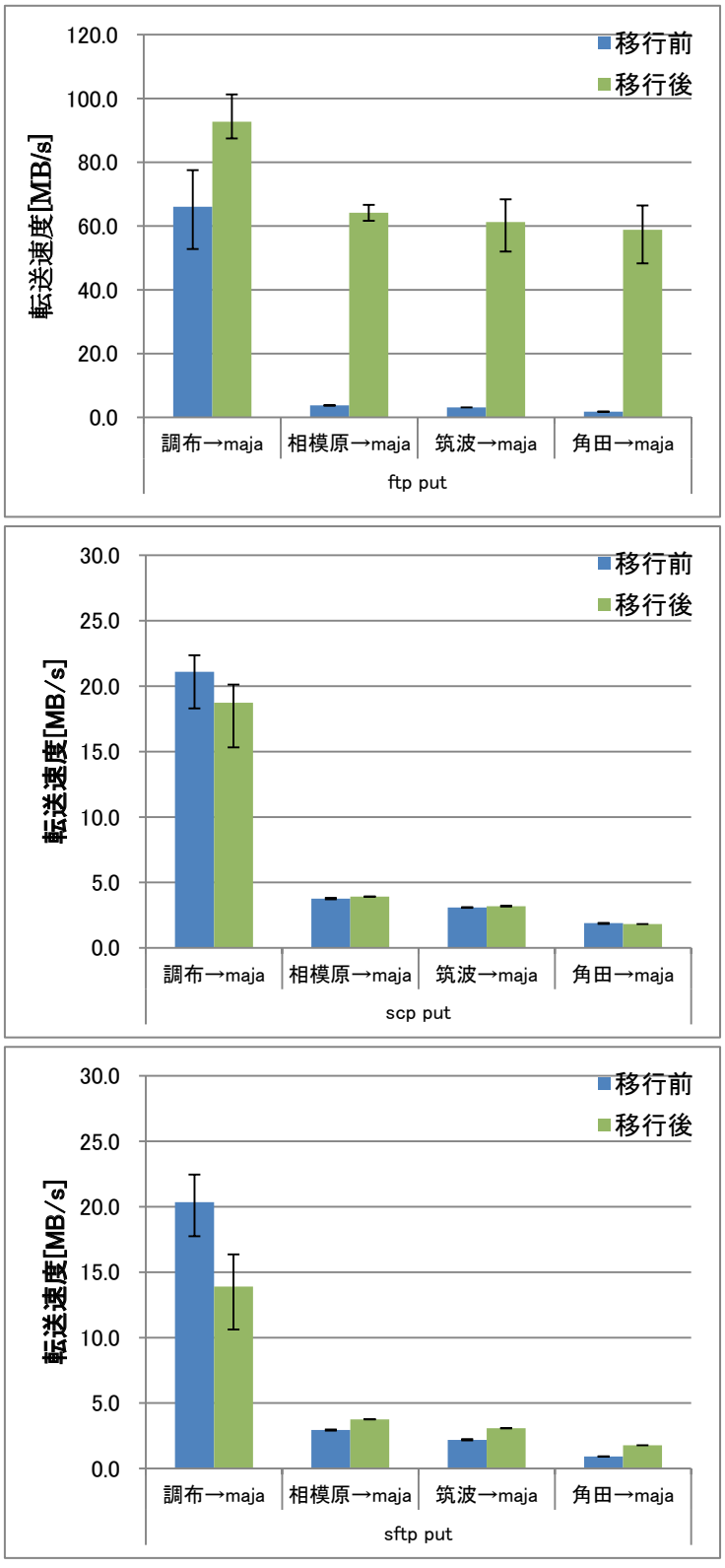


図 6-4　ファイル転送性能の比較（put 方向）

7. 結論

今回の測定によって，JSSnet のボトルネックの特定と，その検証と考察を行うことができた．また，SINET-L2VPN 網を経由した測定との比較により，これらのボトルネックの一部がネットワーク構成の変更により，改善する可能性があることを示した．一方，ネットワーク構成の問題以外にも，システム側の TCP パラメータに調整が必要であることが判った．TCP パラメータの調整により，ボトルネックの解消されたネットワークを十分に利用することができるようになった．

一方で，考察に不足する部分として，インターネット上の他の通信の通信量を推定して考察に加える必要があったが， JAXAnet のネットワーク管理者等との測定時の調整不足により情報が不足してしまった．また， VPN ルータやネットワーク機器の処理性能が原因と考えられる性能低下もあったが，機器の処理性能についても明確にできなかった．これらの点については今後，再測定の機会があれば考察に加える必要があるが，同様の環境を整えることは困難である．

今後の運用においてネットワーク構成の変更，サーバの TCP パラメータの調整を行う場合は，本資料での提案と同様の測定を実施することで，通信性能の基礎値を把握して実運用に供することとしたい．

参考文献

1) 中西 隆，はじめての TCP/IP，技術評論社，(1997)，p.48
2) J.B. Postel, “Transmission control protocol”, *Request for Comments 793,* (1981)