社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

輻輳ウィンドウ及びパケット解析を用いた iSCSI 遠隔ストレージアクセスの評価

比嘉　玲華† 　松原　幸助†† 　岡廻　隆生†† 　山口　実靖††† 　小口　正人††††

† お茶の水女子大学 人間文化創成科学研究科　〒 112–8610 東京都文京区大塚 2–1–1
†† ソフトバンクテレコム株式会社　〒 105–7316 東京都港区東新橋 1–9–1
††† 工学院大学　〒 163-8677 東京都新宿区西新宿 1–24—2
†††† お茶の水女子大学 人間文化創成科学研究科　〒 112–8610 東京都文京区大塚 2–1–1
E-mail: †reika@ogl.is.ocha.ac.jp, †††sane@cc.kogakuin.ac.jp, ††††oguchi@computer.org

あらまし　iSCSI は複雑な階層構造を持つため，性能を向上させるためには複数の層にまたがる最適化を行う必要がある．既存研究において複数の層にまたがる最適化を行ったところ，ある程度の性能向上は達成できたものの，高遅延環境においてはまだなお大きな性能低下が確認された．そこで本研究においては，パケット解析・輻輳ウィンドウ解析を行い，その結果に基づいて，iSCSI 遠隔ストレージにおける性能劣化の原因を検討していく．
キーワード　iSCSI，IP-SAN，ストレージ

Evaluation of iSCSI remote storage access in long latency with analysis of congestion window and packets

Reika HIGA†, Kosuke MATSUBARA††, Takao OKAMAWARI††,
Saneyasu YAMAGUCHI†††, and Masato OGUCHI††††

† Ochanomizu University　2–1–1, Otsuka, Bunkyo-ku, Tokyo, 112–8610 Japan
†† SOFTBANK TELECOM Corp.　1–9–1, Higashishinbashi, Minato-ku, Tokyo, 105–7316 Japan
††† Kogakuin University　1–24–2 Nishishinjuku, Shinju-ku, Tokyo, 163–8677 Japan
†††† Ochanomizu University　2–1–1, Otsuka, Bunkyo-ku, Tokyo, 112–8610 Japan
E-mail: †reika@ogl.is.ocha.ac.jp, †††sane@cc.kogakuin.ac.jp, ††††oguchi@computer.org

Abstract　iSCSI has a complex hierarchical structure , SCSI over TCP/IP over Ethernet. As a result, for the purpose of getting the better performance of iSCSI, multiple layers should be controlled. In our existing work, iSCSI remote storage access through multiple layers has been optimized. However, in the case of long latency, drastic performance deterioration has been observed. Thus, in this paper, we have monitored packets and TCP congestion window. Based on the results, we have analyzed the factor of iSCSI performance deterioration.
Key words　iSCSI, IP-SAN, Storage

1. はじめに

コンピュータシステムにおける処理データ量の増大に伴い，効率的にストレージを管理したいという要望が高まっている．そこで SAN (Storage Area Network) が登場し，広く用いられるようになった．SAN の中でも次世代 SAN として期待されているのが IP ネットワークを用いた IP-SAN である．iSCSI はその IP-SAN の代表的なプロトコルであり，SCSI コマンドを TCP/IP パケットでカプセル化する規格である [1] [2]．iSCSI を用いることにより広域環境における IP-SAN を低コストで構築でき，遠隔地のデータセンタなどにデータをバックアップすることが容易となるため，ストレージのアウトソーシングといったサービスへの利用が期待されている．

しかし現状において iSCSI は，複雑な階層構成のプロトコルスタックで処理されており，バースト的なデータ転送も多いことから，通常のソケット通信と比較して，特に高遅延環境においては性能の劣化が著しい [3]．さらに下位基盤の TCP/IP 層が提供できる限界性能を超えることはできない．従って iSCSI

を用いたストレージアクセスにおいては，iSCSI 層だけではなく複数の層にまたがる制御を施すことによる性能向上が期待される．既存研究において，複数の層にまたがる最適化を行った結果，RTT32ms においてデフォルト時よりも約 4 倍の性能向上が得られた．しかし，高遅延環境においては，なお性能低下が著しい．そこで本研究では，パケット解析，輻輳ウィンドウ解析を行うことで，iSCSI 遠隔ストレージにおける性能劣化の原因を検討していく．

また，遠隔バックアップを行う場合，データの書き込み量と読込み量とを比較すると，圧倒的にデータの書き込み量のほうが多い．さらに，遠隔ストレージ側では標準的なシステムのみを用いることができ，カスタマイズできないことが想定される．そこで，本研究においては，iSCSI シーケンシャルライトの Initiator 側における解析を行う．

2. 研究背景

2.1 iSCSI

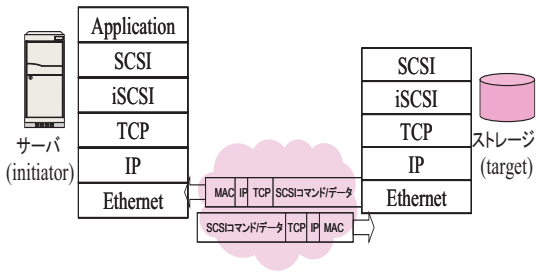

図 1　iSCSI

IP-SAN の代表的なプロトコルに iSCSI がある．iSCSI は SCSI コマンドを TCP/IP パケットでカプセル化する規格で，iSCSI により SAN を IP 機器だけで構築することが可能となる．一方で図 1 のように複雑な階層構成をとることになり，下位のプロトコルの限界性能を超えることはできない．また，iSCSI には長距離アクセスの実現が期待されているが，広帯域な回線を用いた場合には遅延帯域積の問題も存在する．そこで iSCSI 遠隔ストレージアクセスには複数の層にまたがる適切な制御が求められている．

2.2 TCP 輻輳ウィンドウ制御アルゴリズム

輻輳制御はネットワークの混雑解消の方法として TCP が実現する機能である．一般的な TCP の輻輳制御においては，通信開始時にスロースタートと呼ばれるアルゴリズムに従い指数関数的に輻輳ウィンドウが大きくなる．これによりトラフィックが急激に増加するため，ネットワークが輻輳状態になる可能性がある．これを防ぐため，スロースタート閾値という値を用意し，輻輳ウィンドウがその大きさを超えると輻輳回避と呼ばれるフェーズに入り，一次関数的な増え方となる．そしてエラーが検出されると輻輳ウィンドウは急激に低下し，通常これらを繰り返すことで輻輳ウィンドウの振舞の時間変化は一般に鋸型のグラフとなる．

また LinuxOS における TCP の状態遷移を図 2 に示す．LinuxTCP においては，通信時の状態が正常であれば ACK の受

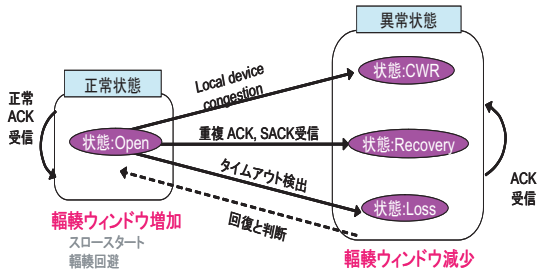

図 2　LinuxTCP の状態遷移

信ごとに輻輳ウィンドウは増加するが，エラーが検出されると異常と判断され，輻輳ウィンドウは低下する．輻輳ウィンドウが低下する原因としては，送信側デバイスドライバのバッファが一杯になったことを示す Local Congestion を検出した場合 (CWR)，重複 ACK 又は SACK を受信した場合 (Recovery)，タイムアウトを検出した場合 (Loss) の 3 つが挙げられる．さらに Linux の TCP 実装では，通信中に一度設定された輻輳ウィンドウは，そのウィンドウ値を超えるデータ量が送られない限りは変化しないという特徴を持ち，この時スループットはほぼ一定の値で安定することが確認されている．

Linux における TCP 輻輳ウィンドウ制御アルゴリズムとして，カーネル 2.6 以降では様々な種類が実装されており，コマンドを入力するだけでそのアルゴリズムを変えられるようになった．本研究においては，Reno，BIC (Binary Increase Congestion Control)，Westwood，H-TCP (Hamilton TCP) の 4 種類のアルゴリズムを扱う [4] [5]．

Reno は古典的なアルゴリズムであり，Reno を基にして様々なアルゴリズムが改良されてきた．Reno は輻輳をパケットロスによって検出し，パケットロス発生時の送信レートが利用可能帯域であるとみなす．具体的には，重複 ACK を三回連続して受け取ると，パケットロスが発生したとみなして，輻輳ウィンドウを半分にする．そして，再度 RTT ごとに一つずつ輻輳ウィンドウを大きくしてネットワークの様子を探る．このように，ウィンドウサイズを徐々に大きくして，輻輳を検出したら一気に落とすというアルゴリズムである．

BIC は，本実験環境における OS のデフォルトのアルゴリズムである．一般的な TCP 輻輳制御は，利用可能帯域に向かってリニアサーチを行っていると考えることができるが，BIC はバイナリサーチを行う．

Westwood は，パケットロスが頻繁に起こる環境で効果的であり，Reno を基に無線環境で使う場合を想定して最適化されたアルゴリズムである．

H-TCP は広帯域かつ高遅延の環境で推奨されているアルゴリズムである．輻輳後に素早く元の状態へ回復させることを目指した設計となっている．

3. 実験システム

3.1 プロトコルアナライザ

高遅延環境において性能が著しく劣化する原因を解明するため，本研究ではネットワーク上を飛来するパケットを調べる．

ネットワークからキャプチャしたトラフィックを直接高速アクセス可能な HDD に書き込む大容量ネットワークアナライザである ClearSight 社の Network Recorder を設置し,iSCSI アクセス時のパケットキャプチャを行った.

3.2 TCP 輻輳ウィンドウモニタツール

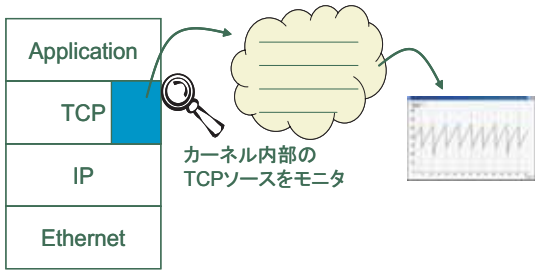

図 3 TCP 輻輳ウィンドウモニタツール

本実験では，TCP 輻輳ウィンドウをモニタするツールを構築した．図 3 に示すように，カーネル内部の TCP ソースにモニタ関数を挿入しカーネルを再コンパイルした．これによりモニタできるようになった値には，輻輳ウィンドウの他，各種エラーイベント (Local device congestion，重複 ACK，SACK 受信，タイムアウト検出) の発生タイミングなどがある．

3.3 実験環境

本研究において,Initiator と Target 間は GigabitEthernet で接続し,TCP/IP コネクションを確立した.Target のストレージには SAS ディスクを用い RAID コントローラによる RAID0 構成で接続した. 使用した実装システムと実験環境を図 4 および表 1 に示す.

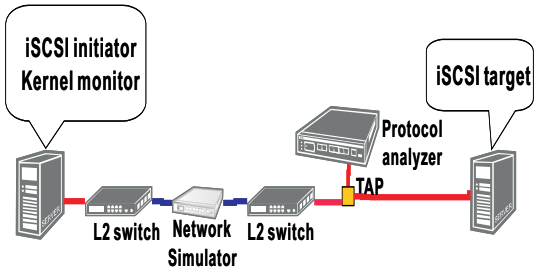

図 4 実装システム概要

表 1 実験環境

OS	Red Hat Enterprise Linux 2.618-8.e.15
CPU	Quad Core Intel Xeon 1.6GHZ
Main Memory	2GB
NIC	Intel PRO/1000PT Server Adaptor on PCI Express
HDD	73GB SAS×2(RAID0)
RAID Controller	SAS5/iR
iSCSI	Initiator : open-iscsi-2.0-865
	Target : iSCSI Enterprise Target(IET)-0.4.15
Network Analyzer	ClearSight Network Recorder
Network Simulator	ANUE

4. 既存研究

iSCSI は複雑な階層構造をとる．そこで，既存研究においては図 5 のように，複数レイヤにまたがる最適化を行った [6]. SCSI/iSCSI 層においては，iSCSI パラメータ最適化を行い，RTT32ms の場合において約 4 倍のスループットの向上が確認できた.

ただし SCSI/iSCSI 層の最適化による性能向上はウィンドウサイズにより制限される可能性があるため，次に TCP/IP 層における最適化を試みた．具体的にはスループットと関係の深い輻輳ウィドウの値を決める輻輳ウィンドウ制御アルゴリズムを変更して性能測定を行った．その結果，ソケット通信時には見られたアルゴリズムごとの違いは iSCSI 利用時には見られなかった．すなわち，本実験環境においては TCP 輻輳制御アルゴリズムの変更は，iSCSI 性能最適化には影響を与えないと言える．原因としては，TCP の輻輳ウィンドウの違いによる性能向上分が，iSCSI のブロックアクセスのシーケンスに吸収されて消えてしまっていると考えられる.

最後に Ethernet 層における最適化として NIC のパラメータを変更し iSCSI 通信を行ったところ，RTT32ms の場合において約 5%の性能向上が確認できた．このように既存研究における最適化によって約 4 倍の性能向上が得られた．この結果を，図 6 に示す．また，比較としてソケット通信，ローカルディスクアクセス性能も測定した．ソケット通信の測定には Iperf [7] を，ローカルディスク，iSCSI の測定には bonnie++ [8] を使用した．本実験においては，通信の妨げにならない程度の十分なウィンドウサイズを想定し，広告ウィンドウを 16MB に設定した.

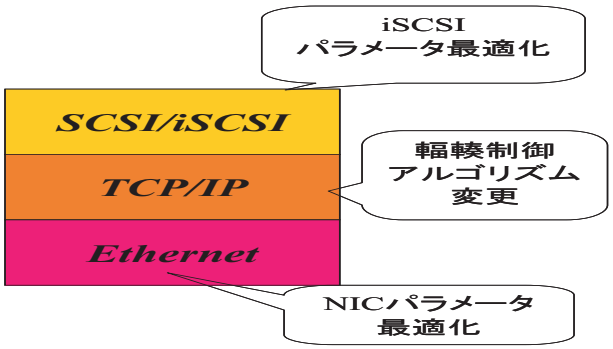

図 5 複数レイヤにまたがる最適化

図 6 からわかるように，ソケット通信の場合は高遅延環境においてもスループットを保っているのに対して，iSCSI 通信の場合は低遅延環境で保たれていたスループットが高遅延環境において性能低下が著しい．そこで本稿では，性能低下の原因を検討していく.

5. iSCSI write アクセスモデルとその解析

5.1 iSCSI write アクセスモデル

本節では高遅延環境において性能が低下する原因となるボトルネックを調べていく.

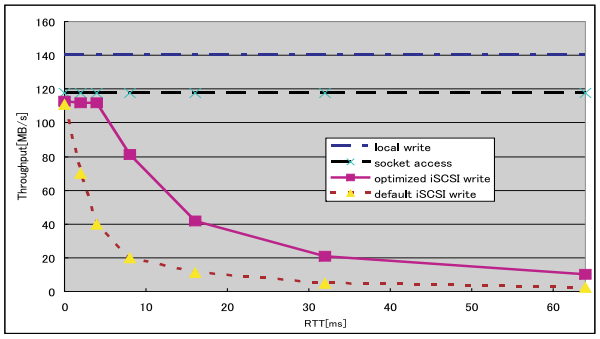

図 6　既存研究におけるスループット比較

dd コマンドを用いて実行される iSCSI ブロックアクセスのパケットをプロトコルアナライザを用いて解析したところ，様々な大きさの複数のパケットが入り混じって飛んでいたため，モデル化の検証に当たっては sg_dd コマンドを使用した．sg_dd コマンドは，dd コマンドと文法的に互換性があるが，dd コマンドとは異なり，iSCSI アクセスにおいて SCSI レベルで指定したブロックサイズによるアクセスが可能となるコマンドである [9]．カーネルを再構築したことで sg_dd コマンドでのアクセス時に最高で 4096KB のブロックサイズでのアクセスが可能となった．それに伴い，iSCSI パラメータの設定を FirstBurstLength,MaxBurstLength ともに 4194304 とした．

4096KB のブロックサイズで write アクセスを実行したときのプロセスは図 7 のようになる．このとき Ta とは Initiator 側における最初のパケット送出から最後のパケット送出までのデータ転送時間，Tb は Target 側で書き込みが終了し Initiator へ書き込みが終了したことを知らせるまでの時間，Tc は次の write が実行されるまでの時間である．遅延装置で設定した遅延時間ごとに Ta，Tb，Tc，RTT を測定することにより，高遅延環境下で性能が劣化する原因を解析する．2048KB，4096KB のブロックサイズで write アクセスを実行した．このときの RTT は 0ms，2ms，5ms，10ms，20ms，50ms である．

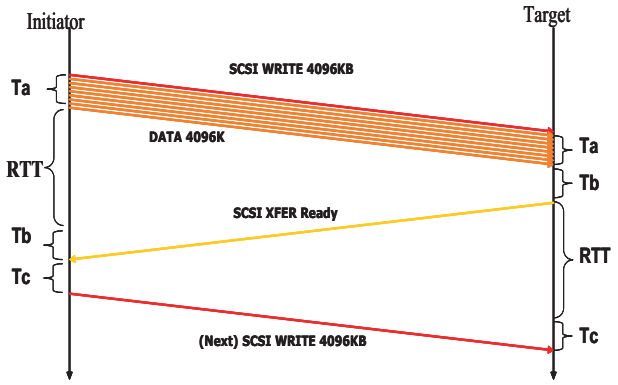

図 7　write 実行図

5.2　解析結果

Ta，Tb，Tc，RTT をアナライザを用いて測定した結果，Tb と Tc はほぼ定数であること，RTT は遅延装置で設定した値とほぼ等しいということが確認された．しかし，Ta は図 8 に示すように RTT に比例する値で，RTT の増大と共に増加していた．高遅延環境における iSCSI アクセスが理論値よりも低下する理由は，RTT によらずに一定であるはずのデータ転送時間が RTT に比例する値になっていることが原因であるとわかった．そこで次にデータ転送時間 Ta の間に何が起こっているか詳しく調べる．

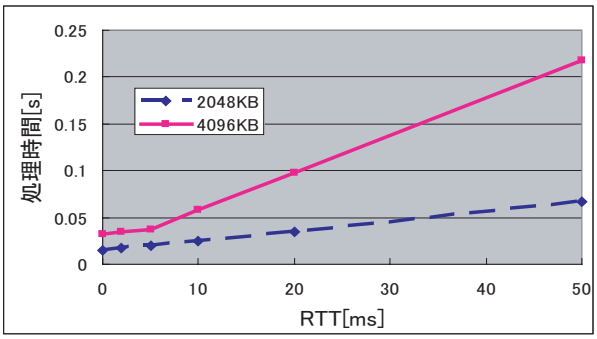

図 8　TA の測定結果

6.　tcpdump を用いたパケット解析とカーネルモニタを用いた CWND 解析

スループットと輻輳ウィンドウには密接な関係があることが知られている．そこで，Initiator 側でカーネルモニタと tcpdump を使って輻輳ウィンドウの値とパケット送出量の関係を調べた．ブロックサイズ 4MB の iSCSI アクセスを実行した．はじめに RTT50ms における測定を行い，次に検証実験として RTT20ms，80ms における測定を行った．

6.1　RTT50ms における測定

RTT50ms，ブロックサイズ 4MB の iSCSI アクセスを実行したときの輻輳ウィンドウとパケット解析の結果を図 9 に示す．4096MB を非同期に送信するには，輻輳ウィンドウは約 3000 が必要であるが，図 9 に示されたように，輻輳ウィンドウは約 1200 であり十分な大きさではないことがわかった．また，図 9 を拡大した図を図 10 に示す．この図からわかるように，短い時間に連続してパケットが送信された後 4MB を送出し終わる前に，突然パケットの送出が止まっている．一定時間の後に再びパケットの連続送信が行われており，それらの間隔は RTT に等しい約 50ms である．

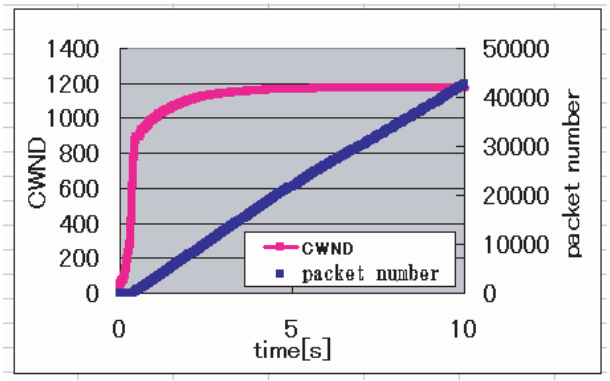

図 9　RTT50ms における解析

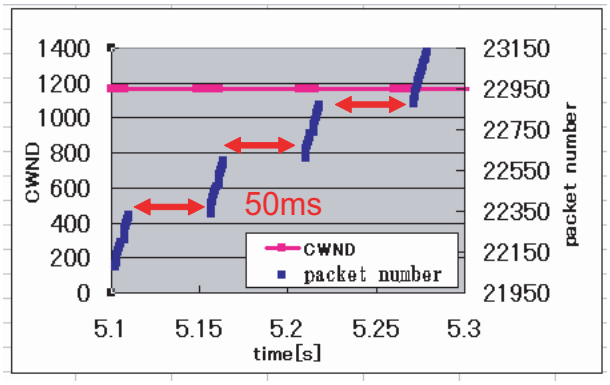

図 10　RTT50ms における解析

6.2　RTT を変えた検証測定

パケット送出の停止から再開の間隔が常に RTT に等しいのか，RTT を変えて検証していく．そこで，RTT を 20ms，80ms に設定してパケット測定を行った．その結果を図 11，図 12 に示す．

この図から，パケット送出の停止から再開の間隔は RTT であることが確認された．

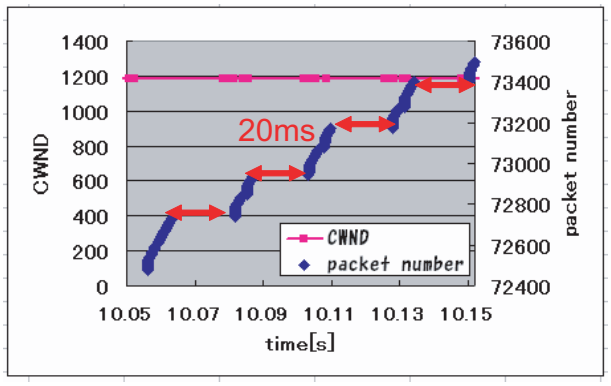

図 11　RTT20ms における解析

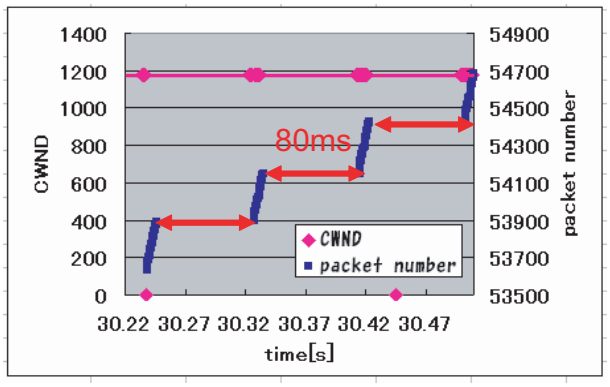

図 12　RTT80ms における解析

6.3　考　　察

パケットの送出が 4MB を書き込む前に停止してしまうこと，その後のパケット再開までの間隔は RTT に等しいことが確認された．4MB のパケットを連続して書きこむことができない理由としては，非同期で送信するためには輻輳ウィンドウは 3000 必要であるが実際には 1200 であり，十分な大きさではないことが原因と考えられる．より詳細なパケット解析を行うべく，次の章ではアナライザを用いた解析を行う．

7.　アナライザを用いたデータ送出の詳細な解析

7.1　RTT50ms におけるパケット解析

RTT50ms，ブロックサイズ 4MB の iSCSI アクセスを設定しアナライザを使用して測定を行った．結果を図 13 に示す．この図からも，短い時間に連続してパケットが送信された後突然パケットの送出が止まっていること，一定時間の後に再びパケットの連続送信が行われておりそれらの間隔は RTT に等しい約 50ms であること，パケットの送出量は約 600 であり．輻輳ウィンドウの 1200 を使い果たす値ではないことがわかる．またアナライザによりパケットを詳細に調べた結果，送信再開の前後には TCP ACK のみが受信されていることもわかった．

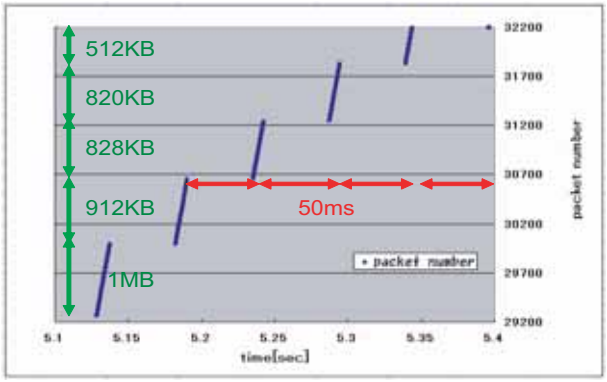

図 13　RTT50ms におけるパケット解析

7.2　考　　察

前節において RTT50ms における輻輳ウィンドウが 1200 (約 1.7MB) であることが確認された．しかし，図 13 より連続して送られるパケットの送出量は約 512KB〜1MB であり，パケットの 1 シーケンスの送出量は輻輳ウィンドウの値を使い切る量ではなかった．そして，パケットの送出は断続的であり，送信再開の前後には TCP ACK のみが受信されていることが確認された．これらのことから，iSCSI の送信において停止・開始の制御は TCP レベルで行われているが，その制御は輻輳ウィンドウの値だけによってはいないことが考えられる．

8.　まとめと今後の課題

既存研究において，iSCSI 遠隔ストレージアクセスの性能を高めるために，複数の層にまたがる最適化を行った．その結果，最適化 iSCSI は当初の iSCSI よりも 4 倍の性能を向上することが出来た．高遅延環境における性能低下の原因をより深く調べるために，iSCSI ブロックアクセスのモデル化を行うことで，高遅延環境における性能低下の原因がデータ転送時間であることが判明したため，次に輻輳ウィンドウ解析，パケット解析を行った．解析の結果，4MB を非同期で送信できるだけの輻輳ウィンドウの値よりも輻輳ウィンドウ値は小さい値であったこと，パケットの 1 シーケンスの送出量はその輻輳ウィンドウ

の値を使い切る量ではなかったこと，そして，パケットの送出は断続的であり，送信開始の前後には TCP ACK のみが受信されたことが確認された．このことから，iSCSI の送信において停止・開始の制御は TCP レベルで行われているが，その制御は輻輳ウィンドウの値だけによってはいないことが考えられる．今後の課題として，更なる解析を行っていく．具体的には，カーネルモニタを使用したログの更なる解析，アナライザを使用したパケット解析を行ってい．それを利用してシステムの改善を行う．また，現在はソフトウェア iSCSI を Target 側に用いて実験を行っているが，標準的なハードウェア iSCSI Target を用いた比較実験も行なっていきたい．

文　　献

[1] iSCSI Specification, http://www.ietf.org/rfc/rfc3720.txt?number=3270

[2] SCSI Specification, http://www.danbbs.dk/~dino/SCSI/

[3] 山口実靖，小口正人，喜連川優：”高遅延広帯域ネットワーク環境下における iSCSI プロトコルを用いたシーケンシャルストレージアクセスの性能評価ならびにその性能向上手法に関する考察”，電子情報通信学会論文誌 Vol.J87-D-I，No.2，pp.216-231，2004 年 2 月

[4] http://linuxgazette.net/135/pfeiffer.html

[5] http://acs.lbl.gov/TCP-tuning/linux.html

[6] 比嘉玲華, 松原幸助, 岡廻隆生, 山口実靖, 小口正人：”iSCSI 遠隔ストレージアクセスの複数レイヤにまたがる最適化”，インターネットコンファレンス 2008，pp.120，沖縄，2008 年 10 月

[7] http://dast.nlanr.net/Projects/Iperf/

[8] http://www.textuality.com/bonnie/intro.html

[9] http://sg.torque.net/sg/sg3_utils.html/