ハイパフォーマンス
コンピューティング 93－7
（２００３．３．１１）

高レイテンシ環境下におけるデータレゼボワールの性能評価

中村 誠* 来栖 竜太郎† 坂元 眞和† 古川 裕希† 生田 祐吉† 下國 治‡
下見 淳一郎‡ 陣崎 明‡ 玉造 潤史§ 稲葉 真理¶ 平木 敬¶

データレゼボワールシステムは超高速大域ネットワーク上で TCP/IP を用いてストレージ階層のマルチストリームデータ転送を実現する分散データファシリティーである．本稿では日米回線を用いた 12,000km, APAN OC-12/POS がボトルネックとなるネットワーク網を用いた実験を中心に，本システムの遠距離データ転送性能を評価し，考察を行う．ネットワーク帯域を占有的に利用した場合，帯域の 91% にあたる 550Mbps 以上で持続的なデータ転送を行った．

Data Reservoir: The Performance for Long Fat Pipe

Makoto Nakamura* Ryutaro Kurusu† Masakazu Sakamoto† Yuki Furukawa†
Yukichi Ikuta† Osamu Shimokuni‡ Junichiro Shitami‡ Akira Zinzaki‡
Junji Tamatsukuri§ Mary Inaba¶ Kei Hiraki¶

Data Reservoir realizes very fast data transfer on a long fat pipe. It adapts iSCSI storage protocol on TCP/IP and hierachical striping technique. In this paper, we have evaluated the performance of bulk file transfer between Japan and U.S., 7,500 miles. We attained continuous data transfer with 550Mbps, i.e., 91% of available bandwidth, provided that we can use exclusively the bottleneck line.

1 はじめに

近年ネットワークの高速化は目覚しく，計算機のネットワークインターフェイスとしてギガビット・イーサネット (GbE) が主流となってきた．また日本でも SuperSINET に代表されるように 10 ギガビット・イーサネット (10GbE) が大域ネットワークのバックボーンに採用され，各地の研究機関や実験・観測施設が超高速インターネットを介して接続されつつあり，数年後には 40Gbps や 100Gbps の帯域が利用可能になると予想される．

反面，ネットワークの高速化のスピードが計算機やストレージの高速化のそれを上まわるため，既存のシステムで超高速ネットワークの性能を十分に活用する事が困難となってきているため，計算機やストレージを並列化し，複数のストリームにデータを分割し並列に転送するシステムが提案・開発されている [11, 12]．しかしながら，多くのシステムではアプリケーションレベルでの拡張により並列ストリームを生成するため，ファイルシステム及び OS の管理オーバヘッドが避けられず，またソフトウェアの改変が不可避であることが多い．

我々は，多量の巨大データを扱う実験・観測科学研究プロジェクトが超高速ネットワークの持つ能力を十分に活用してデータを遠隔研究施設間で共用することを目標とし，(1) 遠距離通信と近距離通信を分離し，(2) 近距離通信には通常のファイルアクセス・インターフェイスを提供し (3) 遠距離通信能力はネットワークバンド幅とディスク容量に対しスケーラブルであるネットワーク利用基盤を構築することを目的とし，データレゼボワール・システムを提案し，TCP/IP 上にプロトタイプモデルを実装，性能評価を行ってきた [1, 2, 3, 4, 5].

本稿では約 12,000km の日米間の回線を利用して実施した高レイテンシ・超高速ネットワーク通信実験，及び実験室での疑似超高速遠距離通信環境における通信実験について述べ，本システムの遠距離通信でのデータ転送に関する適応性及び性能評価を行なう．

2 データレゼボワール・システムの概要

我々は計算機システムの並列化における分散共有メモリの手法にならいファイルへのアクセスをローカルアクセスとリモートアクセスに分離しディス

* 東京大学情報基盤センター. Information Technology Center, the University of Tokyo
† (株) 富士通プログラム技研. Fujitsu Program Laboratories
‡ (株) 富士通研究所. Fujitsu Laboratories
§ 東京大学大学院理学系研究科. Graduate School of Science, the University of Tokyo
¶ 東京大学大学院情報理工学系研究科. Graduate School of Information Science and Technology, the University of Tokyo

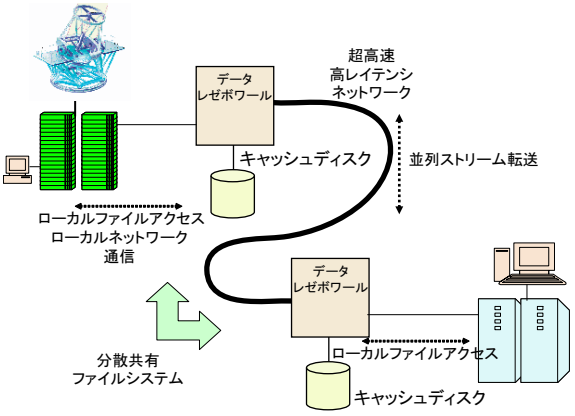

図 1: データレゼボワールの基本アーキテクチャ

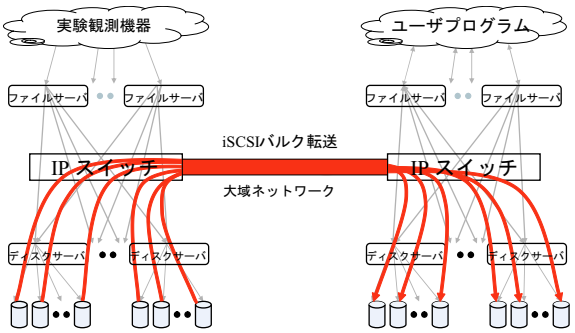

図 2: ストレージによる自立的な並列ストリーム転送

クをキャッシュ層として使用しアドレスベースでデータにアクセスする分散共有ファイル (Distributed Shared File, DSF) アーキテクチャー (図 1) を提案した．超高速かつ高レイテンシな大域ネットワークを前提とし，ネットワーク利用の並列化を図 2のように各ストレージデバイスの自立的な並列ストリーム転送で実現するデータレゼボワールシステムを構築した．

データレゼボワールシステムはファイルサーバとディスクサーバから構成され，階層的なデータのストライピングを行う．すなわちファイルサーバのデータは複数のディスクサーバに，ディスクサーバのデータは複数のローカルストレージにストライプされ分散して格納される．データアクセスのための通信には iSCSI(internet SCSI) プロトコル [6] を採用した．ローカルアクセス時は，ディスクサーバがファイルサーバのローカルディスクの役割を担い，ファイルサーバがイニシエータ，ディスクサーバがターゲットとなる転送を行なう．一方リモートアクセス時は転送元・先のディスクサーバがそれぞれイニシエータおよびターゲットとなりブロックレベルで複数ストリームによる並列転送を自立的に直接行う．

従来のデータ転送方式では，ファイルシステム，OS を通してストレージデバイスをアクセスするため，持続的に高速データ転送を行うことが原理的に困

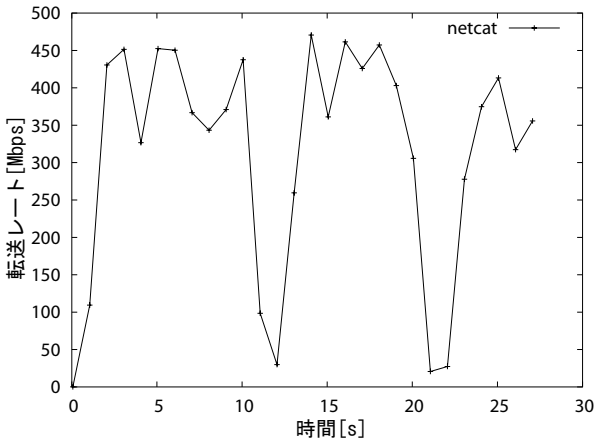

図 3: ファイルシステム，OS を通した従来のデータ転送

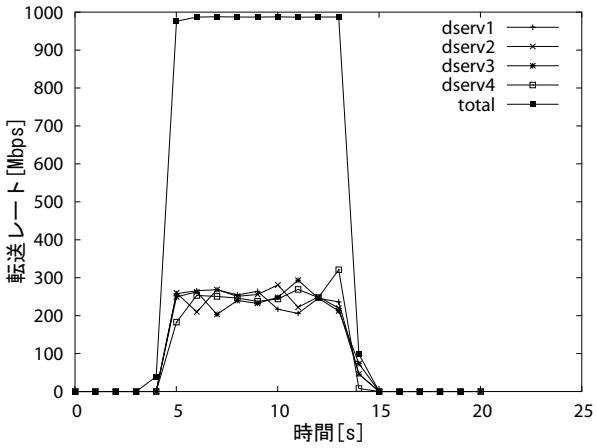

図 4: データレゼボワールによるデータ転送

難で，たとえば図 3 (GbE で接続された 2 台のホスト間を netcat で 1GB のファイルを転送．OS は Linux，ストレージは 10,000rpm Ultra160 SCSI HDD，TCP の Window Size をネットワーク帯域に対して最適化) に示すように，ピーク性能に対し平均転送バンド幅の低下は不可避であった．一方，データレゼボワールシステムは，ファイルシステムより低位の論理ストレージブロックレベルでデータ共有を実現しており，持続した転送バンド幅を実現する事が可能であり，またディスク上のデータ配置に応じたアクセス最適化によるストレージ動作の調整が可能である．図 4に，GbE で接続された 1x4x2 構成 (後述) で 1GB のファイルを転送した時の転送レートを示す．

また，データレゼボワールシステムは低位層において並列ストリーム転送の実装が行なわれているため OS，ファイルシステム，そしてユーザプログラムへの透過性を持ち，ユーザはファイルサーバが提供する通常のファイルシステム，あるいは NFS や CIFS といったネットワークファイルシステムを通して本システム上のデータを利用できる．

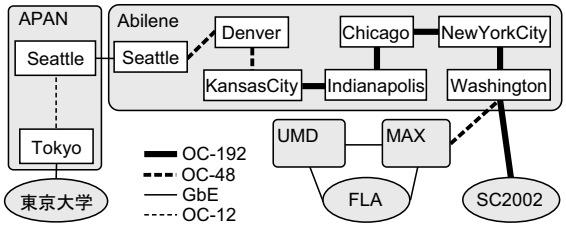

図 5: 日米回線ネットワーク構成

3 実験環境

高レイテンシ環境下 (往復遅延時間が約 200ms) で，高速なネットワークにおけるデータレゼボワール・システムのバーストデータ転送能力を評価するため，以下に示す実験を行った．

3.1 ネットワーク構成とシステム構成

高レイテンシ・高バンド幅のネットワークとして，

1. 東京大学とメリーランド大学間 約 12,000km(約 7500 マイル)
2. 東京大学とボルチモア (SC2002 展示ブース) 間 約 12,000km(約 7500 マイル)
3. 通信遅延及びパケット損失のある疑似長距離高速通信路

を介し，2 組のデータレゼボワール・システム間で，IPv4 及び IPv6 上でのバーストデータ転送による性能評価を行った．(1), (2) では図 5に示すネットワーク網の両端にシステムを設置した．システム間は，APAN[*1], Abilene[*2], MAX[*3] を経由して接続され，往復通信遅延は約 200ms，経路上の帯域ボトルネックは TransPAC[*4]回線 (東京-シアトル間) の OC-12/POS(IP パケットの実効転送レート[*5]は 596.65Mbps～298.92Mbps) ある．

3.1.1 東京大学とメリーランド大学

東京大学とメリーランド大学に隣接した米国富士通研究所 (以下 FLA) とに，ファイルサーバ 1 台とディスクサーバ 4 台 (DELL Power Edge 1550, Dual Pentium-III 1GHz, 1GB メモリ, Netgear GA620 1000BASE-SX NIC, 10,000rpm Ultra160 SCSI HDD 3 台 (36GB 1 台をシステムディスク，73GB 2 台をデータディスクとして使用), Linux 2.4.18 USAGI STABLE 20020408)，Extreme Summit5i 1000BASE-SX スイッチから構成されるデータレゼボワール・システムを設置した．FLA のネットワークは，IPv6 については MAX 直結，IPv4 は MAX からメリーランド大学キャンパスネットワークを経由する経路を使用した．

3.1.2 東京大学とボルチモア

東京大学とボルチモア・コンベンションセンターで開催された SC2002 の展示ブースに，ファイルサーバ 1 台とディスクサーバ 26 台 (DELL Power Edge 1650, Dual Pentium-III 1.4GHz, 1GB メモリ, 3COM 3C996-SX 1000BASE-SX NIC, 10,000rpm Ultra160 SCSI HDD 3 台 (36GB 1 台をシステムディスク，73GB 2 台をデータディスクとして使用), Linux 2.4.18 USAGI STABLE 20020408)，Extreme Summit5i 及び 7i, Foundry BigIron8000 ギガビット・イーサネットスイッチ から構成されるシステムを設置し，SC2002 の会場ネットワーク SCinet [*6] に GbE で接続した．

3.1.3 Comet Drop による疑似遠距離高速通信実験環境

実験室で実ネットワーク環境を再現するため，Comet i-NIC[*7] を用い，疑似遠距離高速通信システム・Comet Drop を作成した．

使用した Comet i-NIC は 1000Base-SX 2 ポートを持ち，128MB のメモリを搭載しているプログラマブル NIC で，ホスト CPU に負荷をかけずに GbE 上のストリームデータの暗号化やプロトコルのオフロード処理を行うことができる．

Comet Drop は，Comet i-NIC 上のメモリをバッファとし，入力パケットをバッファ上で遅延し，ある確率でランダムにパケットを破棄する GbE L2 ブリッジとして動作することで，通信遅延及びパケット損失のある 遠距離高速ネットワーク環境をエミュレートするシステムである．動作パラメタとしては，1ms 単位で約 1 秒 (1Gbps で通信した場合) までの通信遅延，及び 1/65536 単位で任意のパケット破棄率の設定を行なえる．

今回の実験では，日米通信を再現するため 片道通信遅延を 100ms，パケット破棄確率を 0.01% に設定した．

4 性能評価

以下，性能の記述は特に断わらない限り，転送データ量を所要時間で割った平均転送レートとし，iSCSI や TCP/IP のヘッダ・オーバヘッドは含まない値を記す．また，ファイルサーバが i 台，ディスクサーバが j 台，各ディスクサーバが k 台のデータディスクを有するシステムを $i \times j \times k$ と略記する．

4.1 東京大学-FLA 間の転送実験

東京大学と FLA 間で最大 1×4×2 構成によるファイル転送の実験を行った．

[*1] Asia-Pacific Advanced Network. http://www.apan.net/

[*2] Internet2 のバックボーンネットワーク. http://abilene.internet2.edu/

[*3] Mid-Atlantic Crossroads. http://www.maxgigapop.net/

[*4] http://www.transpac.org/

[*5] MTU 1500 バイトで計算

[*6] http://scinet.supercomp.org/

[*7] Communication Enterprising Technology intellignet-NIC . http://www.comet-can.jp/

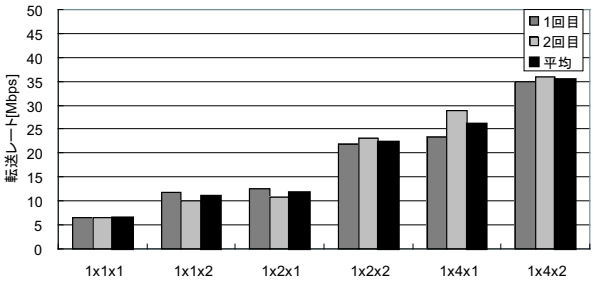

図 6: FLA IPv4 実験結果

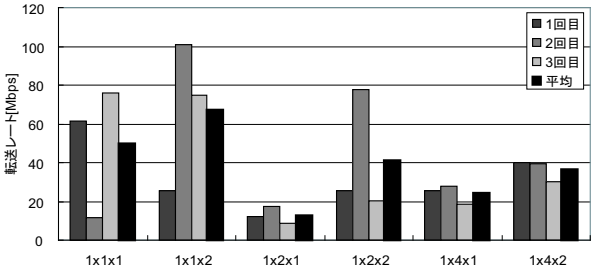

図 7: FLA IPv6 実験結果

4.1.1 IPv4 および IPv6 上の データ転送

システム構成による転送速度の比較を，図 6(IPv4), 図 7(IPv6) に示す．本実験によりデータレゼボワールが実環境の IPv6 ネットワーク上で動作する事が検証された．実験時，特に IPv6 では経路途中のネットワークが安定しておらず (図 8が TCP のパケット再送率)，性能が安定せず，実験結果のばらつきも大きくなっていた．特に，ストリーム間での転送速度の不均衡が顕著であり，1,2 本の未終了のストリームが細く長く通信を続けるためピーク性能に比して平均性能が悪くなる原因となった．

ただし，FLA 実験では IPv4 と IPv6 で経路が一部違うため，性能差の原因がプロトコルであるとは必ずしも断定できない．

4.1.2 TCP/IP の送信制御

前節, ストリーム間の転送速度の不均衡は TCP の輻輳制御のために起こると推測し IPv6 上で データ転送を行い，システムの上流のスイッチのポートで tcpdump をとり，TCP シーケンス番号および確認応

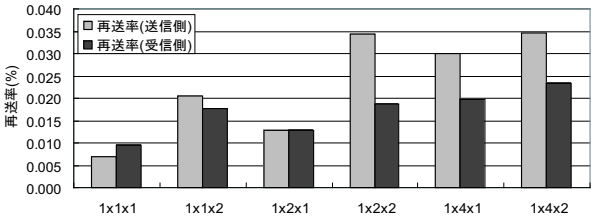

図 8: FLA IPv6 再送率

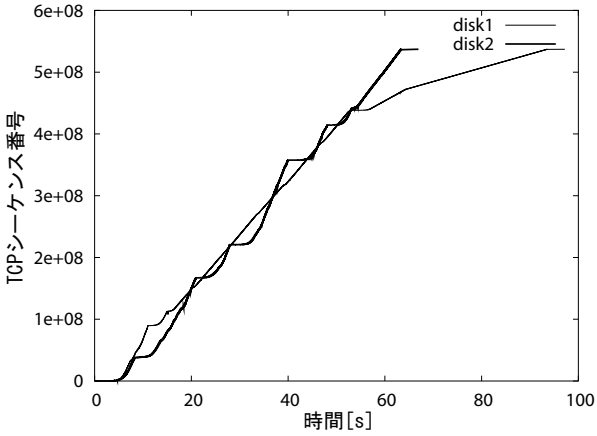

図 9: IPv6 上，1x2x1 構成での TCP シーケンス番号の推移

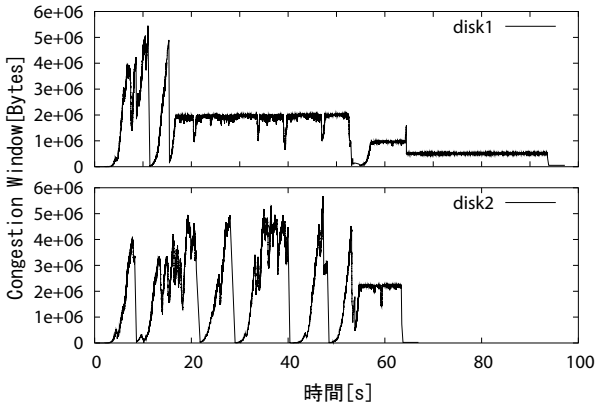

図 10: IPv6 上，1x2x1 構成での Outstanding な送信データ量の推移

答待ちパケット数[*8]の推移を調べた．図 9，図 10に 1×2×1 構成の 2 本の TCP ストリームに関する送信側 TCP シーケンス番号の推移と送信パケットのうち確認応答パケット未着のパケット数の推移をそれぞれ示す．図 10のピーク値は，その時点での Congestion Window サイズを表わしていると考えられる．たとえば，disk2, 9sec でわかるように送信遅延時間が大きくパケット損失の Fast Retransmit による補償がうまく機能せず，再送タイマのタイムアウトが発生していると考えられる．

disk1 では 16sec, 57sec, 64sec で輻輳ウィンドウのサイズが望ましいところより低い値でほぼ平衡状態に陥りシーケンス番号の増加速度が単調減少し disk2 がほぼ 70sec で転送を終了したのに対し disk1 は同量のデータの転送に 約 100sec 必要となった．

4.2 東京大学-ボルチモア間の転送実験

東京大学とボルチモア間で IPv4 上で最大 1×*x*×2 構成 (Window Size は 32MB，データサイズは 51GB)

[*8] http://irg.cs.ohiou.edu/software/tcptrace/tcptrace.html

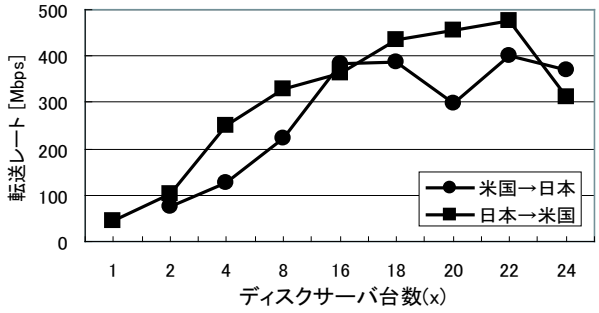

図 11: SC2002 IPv4 実験結果

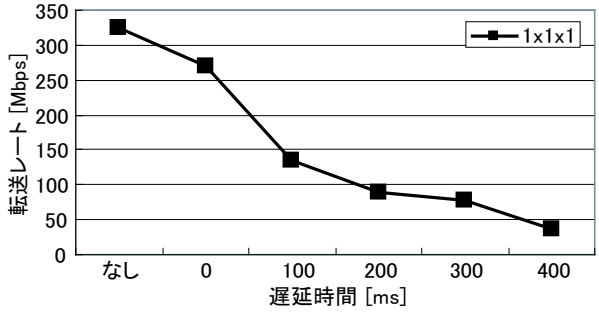

図 12: 遅延時間による性能低下

によるファイル転送の性能を図 11に示す．FLA との転送実験時に比べネットワークが安定していたが，帯域を共有していたためネットワークの状況が一定でなく性能にばらつきが見られる．ディスクサーバの台数が 8〜16 台，帯域の 60〜70%までは性能がほぼ比例して向上している．一方，帯域使用率が高くなると，ストリーム数に対するスケーラビリティが悪くなり，性能が不安定になる．

APAN の回線を占有して通信した時に L2 レベルの測定 (iSCSI 及び TCP/IP, イーサネットヘッダを含む) で，利用可能帯域の 91% にあたる平均 550Mbps 以上での持続的な性能を示した．一方，帯域を SC2002 に参加したグループで共用した高負荷環境では，224.8Mbps の性能を得た．

4.3 CometDrop による疑似遠距離高速通信実験

実験室で状況を再現するため，Comet Drop を使用し擬似的にパケットの転送を遅延させた場合の 1×1×1 構成で性能を図 12に示す．0ms 遅延で性能低下は Comet Drop の処理能力に起因している．往復遅延時間が 200ms で性能が約 0.28 倍になった．また，片道 100msec 遅延，0.01% パケット損失の環境を作るとほぼ，SC2002 の転送実験環境と同等な環境が構築できることがわかった．

5 考察

データレゼボワールでは，ストレージの階層的ストライプにより超高速ネットワークの帯域に見合う I/O バンド幅を実現し，ストライプされたストレージ間で直接データを転送するため，複数ストリームによる並列データ転送が重要である．

高帯域・高レイテンシであっても回線品質が良い場合には有効帯域をマルチストリームでのデータ転送により利用可能である事が確認された．一方，高レイテンシ環境下では，僅かなパケット損失のストリーム間での転送レートのばらつきに与える影響が大きく，ピーク性能と平均性能の差が大きい．

ストリーム間での転送レートのばらつきの原因は，現在の TCP の送信 Window サイズの制御方式 [RFC2581] に由来すると考える．

TCP では通信開始直後の Slow Start でネットワーク帯域の推定を行う．1Gbps 級の帯域では，Slow Start 時の指数オーダでの Window サイズの増加に対して，Congestion Avoidance 時の線形オーダでの Window サイズの増加は非常に小さい．したがって，通信開始直後の Slow Start 時に発生するパケット損失のタイミングが，以後の通信 Window サイズを大きく支配する要因となる．

並列ストリームを使用した場合，主に全二重通信，スイッチにより構成される現在のネットワークでは，パケット損失のタイミングがストリーム毎に異なり，ストリームの転送レートとは無関係に発生する傾向にある．

また，TCP の送信 Window サイズの制御アルゴリズムは，ACK パケットの到着数に基づき，ACK パケットの到着時に Window サイズを変化させる．したがって，通信遅延時間が大きくなると，Window サイズの変化が悪くなる．また，超高速ネットワークでは，TCP は送信側はパケットをバーストに転送するため，受信側では ACK パケットを到着するパケット毎に返す事が出来ず，ACK パケット数が少なくなり，Window サイズの変化がさらに悪くなる．

したがって，超高速，高レイテンシ環境下では，ストリーム間で転送レートの不均衡が発生しやすく，一旦不均衡が生じると解消し難い．これは，TCP の輻輳制御の帯域を共有するストリームが等しい転送レートを得るという目的・性質に反する振舞いである．

6 関連研究

超高速・広域ネットワークでエンド-エンドの利用可能な帯域を有効に活用した信頼性のある通信を実現するため，TCP の送信 Window 制御の改善，及びストリームを複数用い並列にデータ転送する方法が一般に用いられている．

TCP の送信 Window 制御の改善としては, Window Scale オプション [RFC1323] を使用し，さらに，利用可能な帯域と通信遅延時間を見積り，送信バッファサイズを最適化する必須で，自動適応システム [10] が開発されている．

Floyd らによって TCP の超高速・高レイテンシ

環境に対する送信 Window 制御の改善方法が提案されている [7, 8]．Window サイズが大きい領域で，Congestion Avoidance 時の Window サイズの変化量を現在の Window サイズに応じて増減させ，Slow Start 時の Window サイズの成長の抑制する．

TCP の輻輳制御アルゴリズムに起因するネットワークトラフィックの自己相関による大域ネットワークの高速化に対するスケーラビリティの悪化，並列ストリームへの悪影響が指摘されている．[9]

GridFTP [11]，bbFTP [12] 等は，アプリケーション層で HTTP，FTP のファイルの一部分のみを転送する拡張により並列ストリームを生成する．ファイルシステムのオーバヘッドが避けられず，ストレージレベルでのストライピングとバルク転送の活用が困難である．

一方，データレゼボワールはストレージプロトコルレベルで並列ストリームによるデータ転送を実現しているため，ストライピングやアクセスの再配置といったデバイスレベルでの最適化により高効率が達成できる．また，近距離アクセスに対しては通常のデバイスとして振舞えるため，多様なシステムで使用でき，ユーザプログラムからシステムの存在を隠蔽できる．

ソフトウェアによる iSCSI の実装としては Linux-iSCSI [13],Cisco iSCSI Driver, IBM iSCSI Initiator がある．

7 まとめ

本稿では，データレゼボワール・システムの高レイテンシ超高速ネットワーク上でのファイル転送の性能を評価した．IPv4 に加え，IPv6 上での本システムが動作する事を検証した．通常の TCP によるデータ転送では，通信レイテンシの増加に従ってストリーム当りの性能は低下するが，デバイスプロトコルレベルでの通信の並列化及び最適化により複数ストリームの並列データ転送によって転送バンド幅の増大と安定化を達成した．日米回線を使用した実験により，12,000km 以上の遠距離通信でピーク性能 550Mbps，ネットワーク帯域の 91%以上の持続的通信を実現した．

複数ストリームの並列データ転送では，TCP の送信 Window サイズの制御を独立して行うと，ストリーム間での転送レートのばらつきが無視できない事が確認された．今後は，Window サイズをストリーム間で相関的に制御する方式を開発する事が重要であろう．

8 謝辞

日米通信実験にあたり東京大学・加藤 朗氏，米国富士通研究所・益岡竜介氏，MAX・Dan Magorian 氏，APAN，Abilene，メリーランド大学，SCinet ネットワーク担当者諸兄に感謝します．データレゼボワールプロジェクトは科学技術振興調整費先導的研究基盤整備「科学技術研究向け超高速ネットワーク基盤整備」の一環として研究開発が行われ，科学技術振興事業団 CREST による研究領域「情報社会を支える新しい高性能情報処理技術」研究課題「ディペンダブル情報処理基盤」の補助により実験を実施した．

参考文献

[1] K. Hiraki, M. Inaba, J. Tamatsukuri, R. Kurusu, Y. Ikuta, H. Koga, A. Zinzaki, “Data Reservoir: Utilization of Multi-Gigabit Backbone Network for Data-Intensive Research”, SC2002, Nov. 2002. http://www.sc2002.org/paperpdfs/pap.pap327.pdf

[2] R. Kurusu, M. Sakamoto, Y. Ikuta, K. Hiraki, M. Inaba, J. Tamatsukuri, H. Koga, A. Zinzaki, “Data Reservoir, Multi-Gigabit Data Transfer Facility, Its Design and Implementation”, Proc. PDCAT, pp. 100-108, Sept. 2002.

[3] 平木敬, 稲葉真理, 玉造潤史, 来栖竜太郎, 生田祐吉, 古賀久志, 陣崎明, “超高速ネットワーク用データ共有システム: データレゼボワールの性能評価”, SWoPP, Aug. 2002.

[4] K. Hiraki, M. Inaba, J. Tamatsukuri, R. Kurusu, Y. Ikuta, H. Koga, A. Zinzaki, “Data Reservoir: A New Approach to Data-Intensive Scientific Computation”, Proc. ISPAN, pp. 269-274, May 2002.

[5] 稲葉真理, 来栖竜太郎, 玉造潤史, 古賀久志, 陣崎明, 生田祐吉, 酒井英行, 平木敬, “Data Reservoir: A very high-speed Long distance file sharing facility for Scientific data processing”, Proc. HPCS, IPSJ, pp. 81-88, Jan. 2002.

[6] http://www.ietf.org/internet-drafts/draft-ietf-ips-iscsi-20.txt, Internet Draft, IETF, Jan. 2003.

[7] http://www.ietf.org/internet-drafts/draft-floyd-tcp-highspeed-01.txt, Internet Draft, IETF, Aug. 2002.

[8] http://www.ietf.org/internet-drafts/draft-floyd-tcp-slowstart-01.txt, Internet Draft, IETF, Aug. 2002.

[9] W. Feng, P. Tinnakornsrisuphap, “The Failure of TCP in High-Performance Computational Grids”, SC2001, Nov. 2001.

[10] E. Weigle, W.C. Feng, “A Comparison of TCP Automatic Tuning Techniques for Distributed Computing”, Proc. HPDC-11, pp. 265-272. Jul. 2002.

[11] http://www.globus.org/datagrid/gridftp.html

[12] http://doc.in2p3.fr/bbftp/

[13] http://linux-iscsi.sourceforge.net/