5J-6

iSCSI における TCP パラメータとアプリケーション実行性能の相関関係評価

千島　望 †　　豊田 真智子 †　　山口 実靖 ‡　　小口 正人 †
† お茶の水女子大学　　‡ 東京大学生産技術研究所

1. はじめに

近年，インターネット技術の進展などにより蓄積され利用されるデータ容量が爆発的に増加している．これに伴いストレージの増設，管理コストの増大が問題となっている．そこでストレージネッワークが登場し，その代表的なものとして FC-SAN(Fibre Channel - Storage Area Network) がある．また，SAN に IP ネットワークを利用した IP-SAN として iSCSI が期待されている [1].

iSCSI は，これまで DAS(Direct Attached Storage) で使われてきた SCSI コマンドを TCP/IP パケット内にカプセル化することにより，サーバ (initiator) とストレージ (target) 間でデータの転送を行う．しかし，iSCSI は複雑な階層構成でのプロトコルスタックで処理されており，また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．また，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ず，最大限の性能が発揮できるよう TCP パラメータなどを制御することが求められる．

これまで iSCSI 環境におけるシーケンシャルアクセスの性能評価が行われ，TCP パラメータの制御手法が提案及び評価されてきた [2][3]．しかしアプリケーションを実行した時の iSCSI 環境における TCP の振舞についてはあまり知られていない．そこで本研究では，iSCSI を用いたストレージアクセスを含むアプリケーションの実行時に，TCP パラメータとアプリケーション実行性能にどのような相関関係が存在するか検討した．

2. Linux TCP 実装

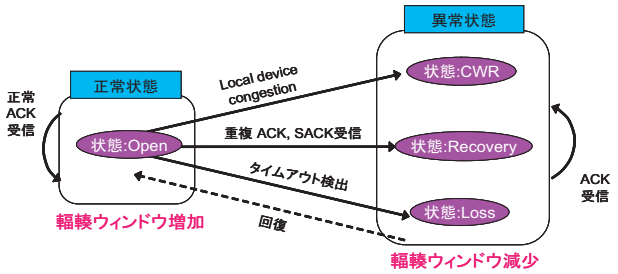

図 1: LinuxTCP の状態遷移

TCP では，通信能力の制御にウィンドウサイズという概念を用いている．ウィンドウサイズとは，ホストが ACK なしに一度に送信できるデータのサイズで，TCP ヘッダに含まれる．また，このウィンドウサイズは，データの送信側では輻輳ウィンドウ，受信側では広告ウィンドウと呼ばれ，このどちらか小さい方がウィンドウサイズとして用いられる．広告ウィンドウは現在の受信ウィンドウの空き容量を示しており，ACK で送信側に送られる．一方，輻輳ウィンドウは送信側の制御パラメータで，ネットワークの混乱を回避するため送信側が自主的に制限する値である．輻輳制御ではこの輻輳ウィンドウが利用されている．

本実験で用いた Linux OS においては，通信時の状態が正常であれば ACK 受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 (図 1)．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溢れることによる Local Congestion エラーを検出した場合（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．また，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，そのウィンドウの値を使い切ら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．

3. PostMark

本研究では，ネットワークストレージを用いる評価アプリケーションとして PostMark を利用した．PostMark はファイルシステムのベンチマークソフトウェアで，NetApp 社が実装/配布を行っている [4]．PostMark は主にインターネットサーバプログラムの性能評価を想定している．ファイルを作成した後，ファイルに対する読み書きの操作を繰り返すベンチマークであり，I/O の性能評価という側面が強い．

設定を変更できるものは次の値である．

- ファイル数　　・ブロックサイズ
- 試行回数　　・読込追記と作成削除の比
- ファイルのサイズ　　・ランダム数発生種

本研究では，アプリケーションとしてこの PostMark を用い，さらにファイル数，試行回数，ファイルのサイズなどの設定を変更した．

4. アプリケーションの実行と輻輳ウィンドウの評価

4.1 実験概要

iSCSI ストレージアクセスを行うイニシエータとストレージを提供するターゲットの間を Gigabit Ethernet で接続して実験システムを構築した（図 2）．イニ

Evaluation of correlation between TCP parameters and application performance on iSCSI
† Nozomi Chishima, Machiko Toyoda, Masato Oguchi
‡ Saneyasu Yamaguchi
Ochanomizu University (†)
Institute of Industrial Science, The University of Tokyo (‡)

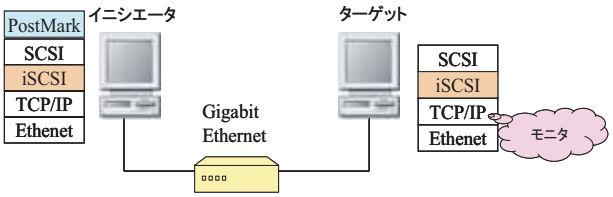

図 2: 実験システムの概要

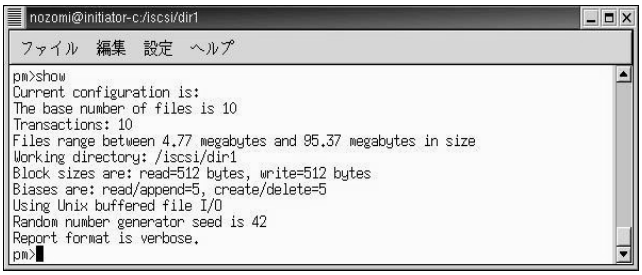

図 3: PostMark 設定状況

シエータとターゲットには，OS が Linux2.4.18-3，CPU が Intel Xeon2.4GHz，Main Memory が 512MB DDR SDRAM，NIC が Intel Pro/1000XT Server Adapter on PCI-X (64bit, 100MHz)，iSCSI は UNH-iSCSI ver.1.5.3 を用いた．この実験環境において iSCSI を起動し，PostMark の実行時に TCP 輻輳ウィンドウをモニタして，グラフを表示する．PostMark 設定状況はファイル数 10，試行回数 10，ファイルのサイズ 4.77MB〜95.35MB とした (図 3)．この場合，PostMark の実行中，10 個のファイルが作成され，その 10 個のファイルが繰り返しアクセスされる．

4.2 実行結果

PostMark には，read/write 速度やファイルの作成削除，読込追記などの結果が表示される．一方，モニタのグラフからは，輻輳ウィンドウの変化の様子がわかる．図 4，5 は PostMark で transaction 処理が行われている時の輻輳ウィンドウの変化の様子である．PostMark では read と write が繰り返し行われ，本研究では target 側の輻輳ウィンドウの様子を観察しているため，グラフの変化があまり見られないところでは，write 処理が行われていると思われる．また，輻輳ウィンドウが激しく変化しているところでは，read 処理が行われ，target 側から initiator 側へデータの送信が行われていることがわかる．

図 4 は，initiator 側の広告ウィンドウの設定をデフォルトの 64KB とした時のものである．この場合，輻輳ウィンドウは一定値以上に上がらず，輻輳ウィンドウが使いきれてない．一方図 5 は，広告ウィンドウを 8MB に設定した時のものである．この時，輻輳ウィンドウはある値で打ち切られてしまうことなく上昇し，ウィンドウが十分に使われていることが確認される．したがって，輻輳ウィンドウは広告ウィンドウによって制限され，それによって PostMark の実行結果にも影響が現れていることが分かった．また，PostMark でファイルサイズの変更を行うと，それに合わせて輻輳ウィンドウの最大値も変化していることがわかった．ファイルサイズが小さいときは，輻輳ウィンドウの値は小さく，ファイルサイズが大きいときは，輻輳ウィンドウ大きな値となった．

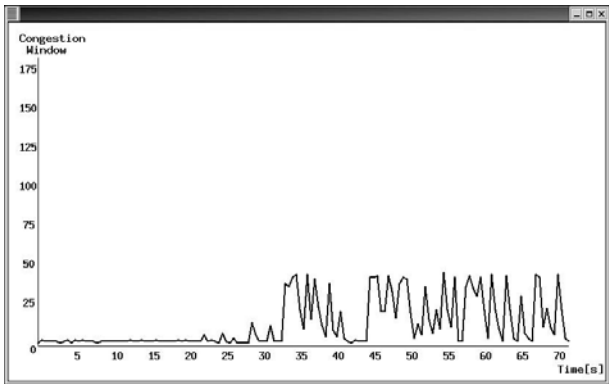

図 4: 輻輳ウィンドウの変化 (広告ウィンドウ:64KB)

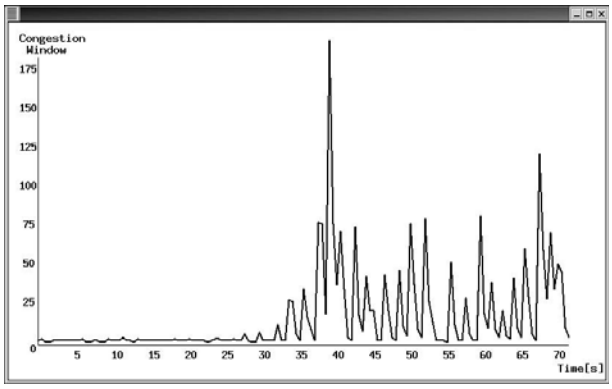

図 5: 輻輳ウィンドウの変化 (広告ウィンドウ:8MB)

5. まとめと今後の課題

iSCSI における PostMark の実行性能を測定し，その時の輻輳ウィンドウの時間変化を観測した．その結果，広告ウィンドウの値を制御することで，輻輳ウィンドウの値も制限できることが確認された．また，PostMark の設定の変更に合わせて，輻輳ウィンドウの値も変わることがわかった．今後は，initiator 側においても輻輳ウィンドウをモニタし，詳しく比較していく．さらに，TCP パラメータの制御を行い，それによりアプリケーションの実行性能にどのような違いが現れるのか解明したい．

参考文献

[1] 喜連川優，ストレージネットワーキング，オーム社出版局

[2] 豊田真智子，山口実靖，小口正人：”高遅延ネットワーク環境における iSCSI リードアクセス時の TCP 輻輳ウィンドウ制御手法の性能評価” SACSIS2005，pp.443-450，2005 年 5 月

[3] MachikoToyoda,SaneyasuYamaguchi,MasatoOguchi ”TCP Congestion Window Control on an iSCSI Read Access in a Long-Latency Environment”CSA2005，pp.170-175，Jul. 2005

[4] PostMark http://www.netapp.com/ftp/postmark-1_5.c