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.

広域グリッドコンピューティングにおけるデータ転送プロトコル GridFTP のパラメータ設定方法に関する検討

伊藤　建志† 　大崎　博之† 　今瀬　真†

† 大阪大学 大学院情報科学研究科
〒 565-0871 大阪府吹田市山田丘 1-5
E-mail: †{t-itou,oosaki,imase}@ist.osaka-u.ac.jp

あらまし　近年、地理的に分散した計算機資源をネットワークにより接続することにより、計算機資源の有効利用を図るとともに、大規模な科学技術計算を可能にする、広域グリッドコンピューティングが注目を浴びている。広域グリッドコンピューティングでは、大容量のファイルを転送するために GridFTP と呼ばれるデータ転送プロトコルが用いられている。GridFTP は、既存の TCP の問題点を解消するため、以下のような機能を持っている。まず、 TCP のスロースタートフェーズにおける転送レートの立ち上がりを速くし、さらに TCP の輻輳回避フェーズにおいて高いスループットを達成するため、複数の TCP コネクションを並列に確立できる。また、ネットワークの帯域遅延積に応じて、GridFTP サーバと GridFTP クライアント間で TCP ソケットのバッファサイズを動的に交渉することができる。しかし、最適な並列 TCP コネクション数や、TCP ソケットのバッファサイズの大きさについては、これまで十分な検討が行われていない。そこで、本稿では TCP コネクション数と TCP ソケットバッファサイズに着目し、 GridFTP の最適なパラメータ設定を定量的に明らかにする。
キーワード　グリッドコンピューティング、GridFTP、TCP (Transmission Control Protocol)、パラメータ設定

On Parameter Tuning of Data Transfer Protocol GridFTP in Wide-Area Grid Computing

Takeshi ITOU†, Hiroyuki OHSAKI†, and Makoto IMASE†

† Graduate School of Information Science and Technology, Osaka University
1-5 Yamadaoka, Suita, Osaka, 565-0871 Japan
E-mail: †{t-itou,oosaki,imase}@ist.osaka-u.ac.jp

Abstract　In recent years, a wide-area Grid computing has got a lot of attention. In the wide-area Grid computing, by connecting computational resources distributed geographically via networks, the computational resources can be used efficiently and large-scale scientific and engineering computation become possible. In the wide-area Grid computing, a data transfer protocol called *GridFTP* is used for large file transfer. GridFTP has the following mechanisms for solving problems of the existing TCP. First, for accelerating the start-up in TCP's slow start phase and achieving high throughput in TCP's congestion avoidance phase, multiple TCP connections can be established in parallel. Second, according to the bandwidth-delay product of a network, the TCP socket buffer size can be adjusted between GridFTP server and client. However, in the literature, sufficient investigation has been performed neither on the optimal number of TCP connections, nor the optimal configuration of the TCP socket buffer size. Therefore, in this paper, we quantitatively investigate the optimal parameter configuration of GridFTP, in particular, in terms of the number of TCP connections and the TCP socket buffer size.
Key words　Grid Computing, GridFTP, TCP (Transmission Control Protocol), Parameter Tuning

1 はじめに

現在、インターネットでは、トランスポート層通信プロトコルとして TCP (Transmission Control Protocol) が広く使用されている [1]。GridFTP は、トランスポート層通信プロトコルとして TCP を用いることを前提として設計されている [2–4]。しかし、TCP は 1970 年代に設計された古い通信プロトコルである。TCP が近年のネットワークの急速な高速化に対応できないなど、TCP に関するさまざまな問題点が指摘されている。

例えば、現在の TCP Reno (TCP バージョン Reno) はパケット棄却が発生するまでネットワーク中の輻輳を検出できないため、大量のパケットが棄却されてしまう。ネットワークが高速化し、

ネットワーク内部のルータが持つバッファサイズが増大するにつれ、棄却されるパケット量が膨大となり、TCP のスループットが大きく劣化してしまう。

GridFTP は、既存の TCP の問題点を解消するため、複数の TCP コネクションを並列に確立したり、送信側ホストと受信側ホスト間で TCP ソケットバッファサイズを交渉するなどの機能を持つ [2–4]。しかし、これらの有効性に関する検討は十分に行なわれておらず、また、このような対策によって TCP の問題点の一部が解消されるにすぎない [5, 6]。

GridFTP は、既存の TCP の問題点を解消するため、以下のような機能を持っている。(1) TCP のスロースタートフェーズにおける立ち上がりを速くするため、複数の TCP コネクションを並列に確立できる、(2) ネットワークの帯域遅延積に応じて、送信側ホスト–受信側ホスト間で TCP ソケットバッファサイズを調整できる。ただし、最適な並列 TCP コネクション数や、最適な TCP ソケットバッファサイズの設定については、これまで十分な検討が行われていない [7, 8]。そこで本稿では、特に、TCP のコネクション数と TCP ソケットバッファサイズに着目し、最適なパラメータ設定を明らかにするとともに、GridFTP の性能限界を定量的に示す。

まず、TCP の連続時間モデルを多重化することにより、GridFTP の連続時間モデルを導出する [9]。さらに、定常状態を考え、GridFTP の性能を数学的解析によって明らかにする。TCP は、ネットワーク中のパケット棄却率によってウィンドウサイズを変化させる、いわゆるフィードバック型の制御である。そこで我々は、複数の TCP コネクションを独立の連続時間システムとしてモデル化する。さらに、我々は、これらの連続時間システムを組み合わせることにより、GridFTP の解析モデルを得る。これにより、特に TCP のコネクション数と TCP ソケットバッファサイズに着目し、GridFTP の最適なパラメータ設定を明らかにするとともに、GridFTP の性能限界を定量的に示す。

本稿の構成は以下の通りである。まず、2 章で、TCP ソケットバッファサイズの自動交渉や並列データ転送といった、GridFTP の重要な機能を簡単に紹介し、GridFTP の未解決の問題点について述べる。3 章で、TCP 輻輳回避機構を連続時間システムとしてモデル化することにより、GridFTP の解析モデルを導出し、GridFTP の最適なパラメータ設定を定量的に明らかにする。また、解析結果およびシミュレーション結果を比較し、数学的解析の妥当性を検証する。最後に、4 章でまとめと今後の課題を述べる。

2 GridFTP

本章では、GridFTP の概要および GridFTP の未解決の問題点を説明する。

GridFTP は、Grid における共通の大容量のデータ転送プロトコルとして、現在、GGF (Global Grid Forum) [10] において標準化が進められている。GridFTP は、IETF において標準化され、現在、インターネットで広く用いられている FTP (File Transfer Protocol) [11–13] を拡張したプロトコルである。GridFTP には、通常の FTP が持つ機能に加えて、さらに以下のような機能が追加されている : TCP ソケットバッファサイズの自動交渉、並列データ転送、データ転送のサードパーティー制御、部分ファイル転送、セキュリティ、信頼できるデータ転送のサポート。

以下では、特に、TCP ソケットバッファサイズの自動交渉と並列データ転送の概要、および GridFTP において未解決の問題を説明する。

2.1 TCP ソケットバッファサイズの自動交渉機能

GridFTP では、SBUF (Set Buffer Size) コマンドにより、クライアントがサーバの TCP ソケットバッファサイズを設定することができる。また、GridFTP では、ABUF (Auto-Negotiate Buffer Size) コマンドにより、サーバとクライアント間で TCP ソケットバッファサイズを交渉することにより設定することができる。既存の TCP 実装の多くは、TCP ソケットバッファサイズを固定的に割り当てる (例えば、64 Kbyte) ため、TCP ソケットバッファサイズをネットワークの帯域遅延積に応じて設定することにより、スループットの向上が期待できる。

ただし、ABUF コマンドをどのように実装するかについては、これまで十分に検討が行われていない。例えば、Globus Toolkit に含まれている GridFTP 実装には、ABUF コマンドの機能は実装されていない。文献 [3] では、サーバとクライアント間で、計測用のトラヒックを生成することにより、ネットワークのラウンドトリップ時間および利用可能帯域を測定する手法の一例が紹介されている

ABUF コマンドをどのように実装するかについては、さらなる検討が必要である。ネットワークの利用可能帯域は時間とともに変動する。また、利用可能帯域を正確に計測するためには、多量の計測用のトラヒックを生成する必要がある。このような理由から、文献 [3] で紹介されているような、アクティブ計測により、単純に帯域遅延積をもとに TCP ソケットバッファサイズを設定する方式では不十分と考えられる。

2.2 並列データ転送機能

GridFTP では、SPAS (Striped Passive) または SPOR (Strieped Data Port) コマンドを用いることにより、複数の TCP コネクションを並列に確立することができる。これにより、単一のファイルを、単一のサーバもしくは複数のサーバから、複数の TCP コネクションを通して転送することができる。複数の TCP コネクションを集約することにより、単一の TCP コネクションを用いる場合に比べて、より高いスループットを実現できる [6]。

これは、以下の理由によって説明できる。(1) 複数の TCP コネクションを集約することにより、TCP の輻輳回避フェーズにおいて、競合する他の TCP コネクションよりも、より大きな帯域を利用できる [14]。これは、TCP の輻輳回避フェーズでは、AIMD 型のウィンドウフロー制御が行われており、パケット棄却率が小さいネットワークでは、複数の TCP コネクションを集約することにより有利にデータ転送を行うことができるからである。(2) 複数の TCP コネクションを集約することにより、1 ファイルの転送に利用できる TCP ソケットバッファサイズの総量が大きくなる。これは、N 本の TCP コネクションを集約することにより、N 倍の TCP ソケットバッファサイズを利用できるからである。(3) 複数の TCP コネクションを集約することにより、TCP のスロースタートフェーズにおける、転送レートの立ち上がりが速くなる。スロースタートフェーズでは、1 ラウンドトリップ時間ごとに、輻輳ウィンドウが 2 倍になる。このため、N 本の TCP コネクションを集約することにより、転送レートの立ち上がりが N 倍になる。

しかし、集約する TCP コネクション数 N が大きするぎると、

以下のような理由によりスループットが低下する。(1) TCP コネクションあたりのウィンドウサイズが小さくなり、タイムアウトが頻繁に発生してしまう。(2) サーバにおいて、TCP プロトコルスタックの処理に要するオーバヘッドが大きくなる。このため、ネットワークの状況に応じて、集約する TCP コネクション数 N の最適値を決定する必要がある。しかし、GridFTP において N をどのように決定するかについて、これまで十分な検討は行われておらず、未解決の問題のままである。

3 GridFTP の数学的解析

3.1 解析モデル

本稿では、文献 [9] で提案されているネットワークのモデル化手法を利用して、GridFTP のモデル化を行う。以下では、GridFTP の特徴のうち、並列データ転送をモデル化する。さらに、 GridFTP のモデルに対して、定常状態解析を行うことにより、特に TCP のコネクション数と TCP ソケットバッファサイズに着目し、GridFTP の最適なパラメータ設定を明らかにする。

以下では、データの転送元のホストを GridFTP サーバ、データ転送先のホストを GridFTP クライアントと呼ぶ。GridFTP は、並列データ転送や部分ファイル転送、データ転送の第三者制御をサポートしているため、一つの GridFTP クライアントに対して、複数の GridFTP サーバが存在することがある。以下では、制御チャンネル上のトラヒックは無視できるほど少ないと仮定し、データチャンネル上のトラヒックのみをモデル化する。

GridFTP のモデル化は以下のように行う。まず、 GridFTP サーバを TCP の輻輳制御機構の連続時間モデルを多重化することによってモデル化する。GridFTP が並列データ転送を行う時は、GridFTP サーバから GridFTP クライアントに複数の TCP コネクションが確立される。このため、集約される TCP コネクション数に応じて、複数の TCP の輻輳制御機構のモデルを GridFTP サーバに配置する。

3.2 定常状態解析

以下では、 GridFTP の流体近似モデルに対して定常状態解析を行うことにより、最適な並列 TCP コネクション数を導出する。

まず、 TCP ソケットバッファサイズ W が、TCP コネクション当たりの帯域遅延積 (TCP コネクションのスループット $\times$ ラウンドトリップ時間) よりも大きい場合を考える。この時、文献 [9] において、タイムアウト機構を含んだ TCP の、時刻 t におけるウィンドウサイズ $w(t)$ の変化が、流体近似法を用いて次式のようにモデル化されている。

$$\dot{w} = (1-p(t))\frac{w(t-R)}{w(t)R} - p(t)\frac{2}{3}w(t)\frac{w(t-R)}{R}\{1-p_{TO}(t)\} - p(t)\left\{\frac{4}{3}w(t)-1\right\}\frac{w(t-R)}{R}p_{TO}(t) \quad (1)$$

ここで、$p(t)$ は時刻 t におけるネットワーク中のパケット棄却率、R は TCP コネクションのラウンドトリップ時間である。また、$p_{TO}(t)$ は時刻 t において、TCP がタイムアウトによってパケット棄却を検出する確率であり、$p_{TO}(t) \simeq \min(1, 3/w(t))$ によって近似できる [15]。

定常状態における、ネットワークのパケット棄却率および TCP のウィンドウサイズ等を、それぞれ p^* および w^* のように表記する。定常状態では、$\dot{w}=0$ および $p(t)=p^*$ および $w(t)=w(t-R)=w^*$ となることから、式 (1) より、以下の関係が成立する。

$$\frac{3-p^*\left(3+2\,w^{*2}\right)+p^*\,(3-2\,w^*)\,w^*\,p^*_{TO}}{3\,R}=0 \quad (2)$$

ここで、定常状態における TCP のウィンドウサイズ w^* が 3 以上であると仮定する。この時、TCP がタイムアウトによってパケット棄却を検出する確率 p^*_{TO} は次式で与えられる。

$$p^*_{TO} \simeq \frac{3}{w^*} \quad (3)$$

そこで、式 (2)、(3) を w^* について解くことにより、定常状態における TCP のウィンドウサイズ w^* が次式のように得られる。

$$w^* \simeq \frac{1}{2}\left(-3+\frac{\sqrt{6+21\,p^*}}{\sqrt{p^*}}\right) \quad (4)$$

従って、定常状態における TCP のスループット T^* は次式で与えられる。

$$T^* \equiv \frac{w^*}{R} \simeq \frac{1}{2\,R}\left(-3+\frac{\sqrt{6+21\,p^*}}{\sqrt{p^*}}\right) \quad (5)$$

ここで、並列 TCP コネクション数を N、ボトルネックリンクの帯域を B とする。TCP ソケットバッファサイズが、TCP コネクション当たりの帯域遅延積よりも大きいという仮定のもとでは、定常状態において、TCP コネクションの総スループット $N\,T^*$ が、ボトルネックリンクの帯域 B と等しくなる。つまり、式 (5) より、

$$B \;=\; N\,T^* \simeq \frac{N}{2\,R}\left(-3+\frac{\sqrt{6+21\,p^*}}{\sqrt{p^*}}\right) \quad (6)$$

という関係が成立する。これを p^* について解くことにより、定常状態におけるネットワーク中のパケット棄却率が、並列 TCP コネクション数 N、TCP のラウンドトリップ時間 R、ボトルネックリンクの帯域 B によって与えられる。

$$p^* \simeq \left(-2+\frac{2\,B\,R}{N}+\frac{2}{3}\left(\frac{B\,R}{N}\right)^2\right)^{-1} \quad (7)$$

上式より、定常状態におけるネットワーク中のパケット棄却率 p^* は、TCP コネクション 1 本あたりの帯域遅延積 $(B\,R/N)$ で決まることがわかる。

定常状態における、ネットワーク中でのパケット棄却率が p^*、TCP コネクションのスループットが T^* であれば、$T^* \times p^*$ だけのパケットがネットワーク中で廃棄されるため、TCP コネクションの実効スループット (グッドプット) G^* は、式 (5)、(7) より次式で与えられる。

$$G^* \;\equiv\; T^*(1-p^*) \quad (8)$$

一方、TCP ソケットバッファサイズ W が、TCP コネクション当たりの帯域遅延積 (TCP コネクションのスループット $\times$ ラウンドトリップ時間) よりも小さい場合を考える。この時、TCP ソケットバッファサイズの制限により、TCP コネクションはウィンドウサイズを十分大きくすることができない。このため、TCP コネクションのスループット T^* は次式で与えられる。

$$T^* \;=\; \frac{W}{R} \quad (9)$$

TCP ソケットバッファサイズ W が、TCP コネクション当た

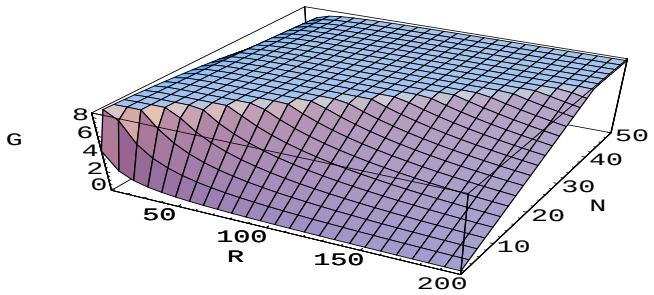

図 1　定常状態における GridFTP のグッドプット $N \times G^*$ (ラウンドトリップ時間 R と並列 TCP コネクション数 N が与える影響) ($B = 100$ [Mbit/s], W = 64 [Kbyte])

Fig. 1　GridFTP goodput ($N \times G^*$) (effect of round-trip time R and the number of parallel TCP connections N) ($B = 100$ [Mbit/s], W = 64 [Kbyte])

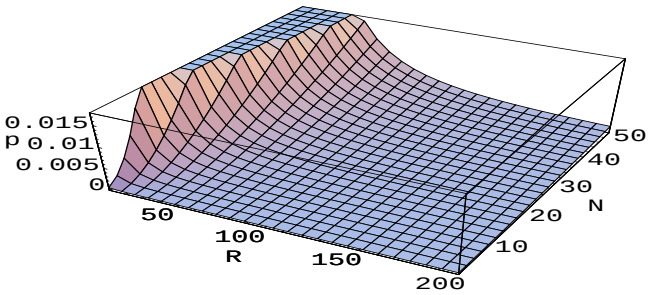

図 2　定常状態における GridFTP のパケット棄却率 p^* (ラウンドトリップ時間 R と並列 TCP コネクション数 N が与える影響) ($B = 100$ [Mbit/s], W = 64 [Kbyte])

Fig. 2　GridFTP packet loss probability p^* (effect of round-trip time R and the number of parallel TCP connections N) ($B = 100$ [Mbit/s], W = 64 [Kbyte])

りの帯域遅延積よりも小さい場合、TCP コネクションはボトルネックリンクの帯域を 100% 利用することができないため、バックグラウンドトラヒックの影響によるパケット棄却が無視できると仮定すれば、定常状態におけるパケット棄却率 p^* は

$$p^* = 0 \tag{10}$$

となる。従って、定常状態における TCP コネクションの実効スループット G^* は、

$$G^* \equiv T^*(1-p^*) = T^* = \frac{W}{R} \tag{11}$$

となる。

式 (8)、 (11) より、TCP ソケットバッファサイズ W、並列 TCP コネクション数 N、ラウンドトリップ時間 R、ボトルネックリンクの帯域 B が与えられた時の、全 TCP コネクションの実効スループットの総和 $\overline{G^*}$ は次式で与えられる。

$$\overline{G^*} \simeq \min\left(\frac{N\,W}{R}, \frac{N\,(1-p^*)}{2\,R}\left(-3+\frac{\sqrt{6+21\,p^*}}{\sqrt{p^*}}\right)\right) \tag{12}$$

従って、最適な並列 TCP コネクション数は、 式 (12) を最大化する N を求めることによって得られる。また、必要な TCP ソケットバッファサイズは、式 (12) が十分大きな値となる W として求めることができる。

3.3　数　値　例

以下では、定常状態解析の数値例をいくつか示すことにより、並列 TCP セッション数や TCP ソケットバッファサイズが、GridFTP の性能に与える影響を定量的に明らかにする。

まず、図 1 に、定常状態における GridFTP のグッドプット (全 TCP コネクションのグッドプットの総和) $N \times G^*$) を示す。ここでは、ボトルネックリンクの帯域 $B = 100$ [Mbit/s]、TCP ソケットバッファサイズ $W = 64$ [Kbyte] とし、TCP コネクションのラウンドトリップ時間 R と並列 TCP コネクション数 N が、GridFTP のグッドプットに与える影響を示している。また、この時の、定常状態における GridFTP のパケット棄却率を図 2 に示す。

これらの図より、GridFTP のグッドプットに関して以下のことが分かる。まず、図 1 より、ボトルネックリンクの帯域を、ほぼ 100% 利用するためには、並列 TCP コネクション数 N を

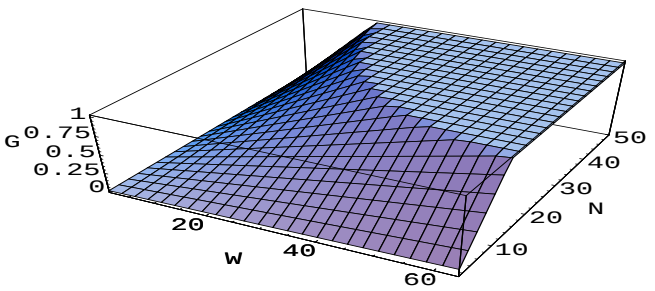

図 3　定常状態における GridFTP のグッドプット $N \times G^*$ (並列 TCP コネクション数 N と TCP ソケットバッファサイズ W が与える影響) ($B = 100$ [Mbit/ms], R = 100 [ms])

Fig. 3　GridFTP goodput $N \times G^*$ (effect of the number of parallel TCP connections N and TCP socket buffer size W) ($B = 100$ [packet/ms], $R = 100$ [ms])

ある程度大きくする必要があることがわかる。どの程度、並列 TCP コネクション数 N を大きくすれば良いかは、TCP コネクションのラウンドトリップ時間にほぼ比例して決まることがわかる。また、並列 TCP コネクション数がさらに大きくなるにつれ、GridFTP のグッドプットが少しづつ低下していることが分かる。この傾向は、ラウンドトリップ時間が小さい状況において特に顕著に表れている。これは、図 2 から分かるように、並列 TCP コネクション数 N が大きく、かつ TCP コネクションのラウンドトリップ時間 R が小さい領域において、パケット棄却率が大きくなっているためである。

次に、並列 TCP コネクション数 N と TCP ソケットバッファサイズ W が、定常状態における GridFTP のグッドプットとパケット棄却率に与える影響に着目する。並列 TCP コネクション数 N および TCP ソケットバッファサイズ W を変化させた時の、定常状態における GridFTP のグッドプットおよびパケット棄却率を、それぞれ図 3 および図 4 に示す。ここでは、ボトルネックリンクの帯域 $B = 100$ [Mbit/s]、TCP コネクションのラウンドトリップ時間 $R = 100$ [ms] としている。

図 5 より、ボトルネックリンクの帯域をほぼ 100% 利用するためには、TCP ソケットバッファサイズ B および並列 TCP コネクション数 N を適切に設定する必要があることがわかる。特に、

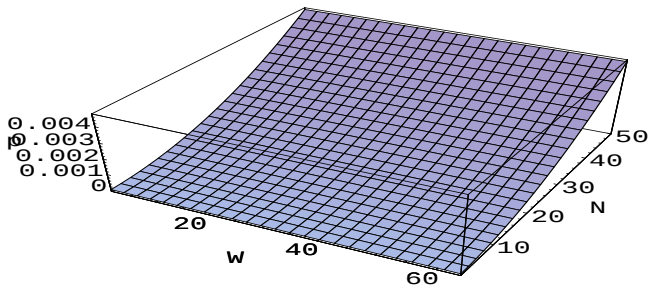

図 4　定常状態における GridFTP のパケット棄却率 p^* (並列 TCP コネクション数 N と TCP ソケットバッファサイズ W が与える影響) ($B = 100$ [Mbit/s], R = 100 [ms])

Fig. 4　GridFTP packet loss probability p^* (effect of the number of parallel TCP connections N and TCP socket buffer size W) ($B = 100$ [Mbit/s], R = 100 [ms])

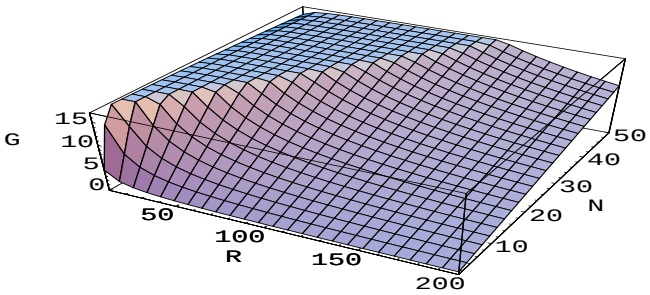

図 5　定常状態における GridFTP のグッドプット $N \times G^*$ (ラウンドトリップ時間 R と並列 TCP コネクション数 N が与える影響) ($B = 200$ [Mbit/s], W = 64 [Kbyte])

Fig. 5　GridFTP goodput ($N \times G^*$) (effect of round-trip time R and the number of parallel TCP connections N) ($B = 200$ [Mbit/s], W = 64 [Kbyte])

TCP ソケットバッファサイズ W が小さい (例えば 16 [Kbyte]) 場合には、並列 TCP コネクション数 N を非常に大きくとる必要があることがわかる。このため、並列 TCP コネクション数 N を小さく抑えることができるように、TCP ソケットバッファサイズ W を十分大きな値に設定することが望ましいと考えられる。

図 4 より、並列 TCP コネクション数 N が大きくなるにつれ、GridFTP のパケット棄却率が大きくなっていることがわかる。また、GridFTP のパケット棄却率は、TCP ソケットバッファサイズ W と独立であることもわかる。このことからも、GridFTP の制御パラメータ設定方法としては、TCP ソケットバッファサイズ W をできるだけ大きく取り、これにあわせて、ボトルネックリンクの帯域を 100% 利用できるように、並列 TCP コネクション数 N を設定すれば良いことがわかる。

次に、ボトルネックリンクの帯域の増加が、GridFTP のパラメータ設定方法にどのような影響を与えるかを明らかにする。図 5 は、図 1 において、ボトルネックリンクの帯域 B を 2 倍 ($B = 200$ [Mbit/s]) にした時の結果である。この図から、ボトルネックリンクの帯域が大きい時には、それに応じて並列 TCP コネクション数を増加させる必要があることがわかる。ただし、図 5 より、図 1 の場合と比較すると、並列 TCP コネクションを非常に大きくした時の、GridFTP のグッドプットの低下の割合が小さいことがわかる。これは、より高速なネットワークでは、TCP コネクション 1 本あたりの帯域遅延積が大きくなり、その結果、TCP のタイムアウトが発生しにくくなるためと考えられる。これは、ネットワークが高速化もしくは広域化するにつれ、GridFTP の制御パラメータ設定がより簡単になることを意味している。

最後に、解析結果とシミュレーション結果を比較することにより、近似解析の妥当性を示す。シミュレーションには ns-2 シミュレータ (バージョン 2.27) を用いた。シミュレーションでは、1 ホップの単純なネットワークトポロジーを用いた。ボトルネックとなるリンクの帯域を $B = 100$ [Mbit/s]、往復伝搬遅延を $\tau = 100$ [ms] または $\tau = 20$ [ms] とした。パケット長を 1500 [byte] とし、GridFTP サーバを、複数の FTP トラヒックを多重化することによってモデル化した。並列 TCP コネクション数 (多重する FTP トラヒックの本数) を変化させながら、60 [s] 間のシミュレーションを実行し、グッドプットおよびパケット棄却率を測定した。

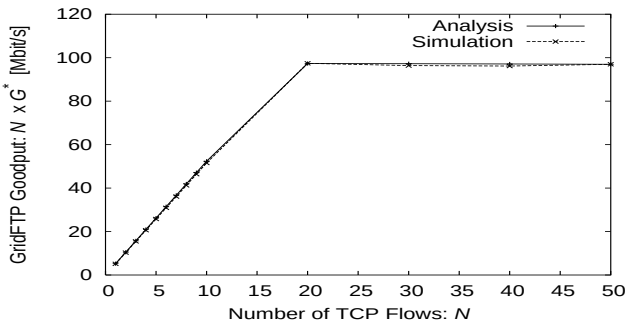

図 6　並列 TCP コネクション数と GridFTP のグッドプットの関係 ($B = 100$ [Mbit/s], $\tau = 100$ [ms], $W = 64$ [Kbyte])

Fig. 6　Number of parallel TCP connections N vs. GridFTP goodput (($B = 100$ [Mbit/s], $\tau = 100$ [ms], $W = 64$ [Kbyte])

まず、ボトルネックとなるリンクの往復伝搬遅延 $\tau = 100$ [ms] の時の結果を示す。図 6 および図 7 は、並列 TCP コネクション数を変化させた時の、GridFTP のグッドプットおよびパケット棄却率である。シミュレーション結果では、多重化した FTP トラヒックの総グッドプットを、GridFTP のグッドプットとして用いている。解析結果は、シミュレーションによって得られた、TCP コネクションの平均ラウンドトリップ時間をもとに計算した値である。これらの図から、本稿における GridFTP の定常状態解析により、GridFTP のグッドプットおよびパケット棄却率が正確に得られていることがわかる。

ボトルネックとなるリンクの往復伝搬遅延が小さい ($\tau = 20$ [ms]) 時の結果を、図 8 および図 9 に示す。伝搬遅延が大きい場合 (図 6 および図 7) と比較して、解析結果がシミュレーション結果から外れていることがわかる。特に、並列 TCP コネクション数が大きい時 ($N = 50$) の時に、パケット棄却率が 1.5 倍程度ずれていることがわかる。これは、以下のような原因によると考えられる。ネットワークの帯域遅延積が小さく、かつ並列 TCP コネクション数が多い状況では、TCP コネクションあたりウインドウズサイズが小さくなり、TCP のタイムアウトがより発生しやすくなる。しかし、本稿の解析では TCP がタイムアウトによってパケット棄却を検出する確率 p^*_{TO} を式 (3)

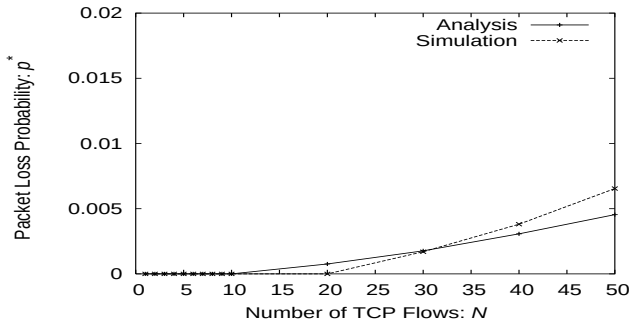

図 7　並列 TCP コネクション数と GridFTP のパケット棄却率の関係 ($B = 100$ [Mbit/s], $\tau = 100$ [ms], $W = 64$ [Kbyte])

Fig. 7　Number of parallel TCP connections N vs. GridFTP goodput (($B = 100$ [Mbit/s], $\tau = 100$ [ms], $W = 64$ [Kbyte])

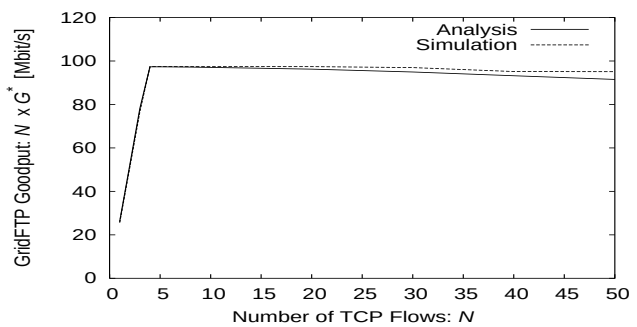

図 8　並列 TCP コネクション数と GridFTP のグッドプットの関係 ($B = 100$ [Mbit/s], $\tau = 20$ [ms], $W = 64$ [Kbyte])

Fig. 8　Number of parallel TCP connections N vs. GridFTP goodput (($B = 100$ [Mbit/s], $\tau = 20$ [ms], $W = 64$ [Kbyte])

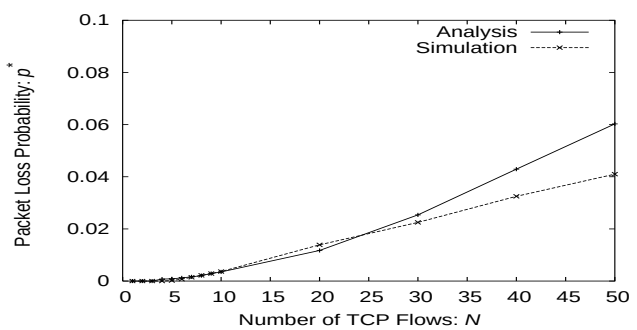

図 9　並列 TCP コネクション数と GridFTP のパケット棄却率の関係 ($B = 100$ [Mbit/s], $\tau = 20$ [ms], $W = 64$ [Kbyte])

Fig. 9　Number of parallel TCP connections N vs. GridFTP goodput (($B = 100$ [Mbit/s], $\tau = 20$ [ms], $W = 64$ [Kbyte])

によって似的に与えているため、タイムアウトが頻発する状況下において正確なモデ化ができていないと考えられる。

ただし、図 8 からも分かるように、帯域遅延積の小さいネットワークでは、GridFTP の制御パラメータをそれほど慎重に設定しなくても、GridFTP は高い性能を示す。また、これまでに議論したように、GridFTP の制御パラメータ設定では、TCP ソケットバッファサイズ W を十分大きく設定し、並列 TCP コネクション数を、ボトルネックリンクの帯域をほぼ 100% 利用できる程度に大きく設定すればよい。このため、実際の GridFTP のパラメータ設定への適用を考えると、本解析は実用上問題ないと考えられる。

4　まとめと今後の課題

本稿では、数学的解析手法を用いることにより、並列 TCP コネクション数や TCP ソケットバッファサイズといった、GridFTP 制御パラメータ設定手法について議論した。まず、TCP の連続時間モデルを多重化することにより、GridFTP の連続時間モデルを導出した [9]。定常状態を考え、GridFTP のグッドプットおよびパケット棄却率を導出した。いくつかの数値例により、GridFTP の制御パラメータ設定方法としては、TCP ソケットバッファサイズ W をできるだけ大きく取り、これにあわせて、ボトルネックリンクの帯域を 100% 利用できるように、並列 TCP コネクション数 N を設定すれば良いことを示した。また、シミュレーション結果と解析結果を比較することにより、近似解析の妥当性を示した。

今後の課題としては、タイムアウト機構の正確なモデル化による近似解析の精度向上や、バックグラウンドトラヒックが存在する環境や、異なるバージョンの TCP が混在する環境など、より複雑でネットワーク環境における GridFTP の解析を行い、より一般的な制御パラメータ設定手法を明らかにすることなどがあげられる。

文　献

[1] J. Postel, "Transmission control protocol," *Request for Comments (RFC) 793*, Sept. 1981.

[2] The Globus Project, "GridFTP: universal data transfer for the Grid," *White Paper*, Sept. 2003. available at `http://www.globus.org/datagrid/deliverables/C2WPdraft3.pdf`.

[3] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, and S. Tuecke, "GridFTP: Protocol extensions to FTP for the Grid," *Internet Draft*, Mar. 2001.

[4] The Globus Project, "GridFTP update January 2002," 2002. available at `http://www.globus.org/datagrid/deliverables/GridFTP-Overview-200201.pdf%`.

[5] J. Semke, J. Mahdavi, and M. Mathis, "Automatic TCP buffer tuning," in *Proceedings of ACM SIGCOMM '98*, vol. 28, 1998.

[6] L. Qiu, Y. Zhang, and S. Keshav, "On individual and aggregate TCP performance," in *Proceedings of Internetl Conference on Network Protocols*, pp. 203–212, 1999.

[7] H. Sivakumar, S. Bailey, and R. L. Grossman, "PSockets: The case for application-level network striping for data intensive applications using high speed wide area networks," in *Proceedings of the 2000 ACM/IEEE Conference on Supercomputing*, 2000.

[8] T. J. Hacker and B. D. Athey, "The end-to-end performance effects of parallel TCP sockets on a lossy wide-area network," in *Proceedings of the 16th IEEE-CS/ACM International Parallel and Distributed Processing Symposium (IPDPS)*, Aug. 2001.

[9] 氏家 純也, 大崎 博之, 今瀬 真, "複数のフィードバックループをモデル化した TCP の特 性解析," 電子情報通信学会技術研究報告 (IN2003-90), pp. 1–6, Oct. 2003.

[10] "Global Grid Forum." `http://www.ggf.org/`.

[11] J. Postel and J. Reynolds, "File transfer protocol (FTP)," *Request for Comments (RFC) 959*, Oct. 1985.

[12] R. Elz and P. Hethmon, "FTP security extensions," *Request for Comments (RFC) 2228*, Oct. 1997.

[13] P. Hethmon and R. Elz, "Feature negotiation mechanism for the file transfer protocol," *Request for Comments (RFC) 2389*, Aug. 1998.

[14] S. Floyd, "Highspeed TCP for large congestion windows," *Internet Draft* `draft-ietf-tsvwg-highspeed-01.txt`, Aug. 2003.

[15] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: a simple model and its empirical validation," in *Proceedings of ACM SIGCOMM '98*, pp. 303–314, Sept. 1998.