論　文

# 上位層プロトコルがネットワークトラヒックの自己相似性に与える影響

住田　義明[†]　　大崎　博之[†]　　村田　正幸[†]　　宮原　秀夫[†]

## Effects of Upper-Layer Protocols on Self-Similarity of Network Traffic

Yoshiaki SUMIDA[†], Hiroyuki OHSAKI[†], Masayuki MURATA[†], and Hideo MIYAHARA[†]

あらまし　これまでの研究で，LAN や WAN 上で計測されるトラヒックが自己相似性を有することが指摘されている．しかし，なぜネットワーク上のトラヒックに自己相似性が発生するか，またトラヒックの自己相似性がネットワークの性能にどのような影響を与えるかは十分に明らかにされていない．また，トラヒックの自己相似性が，上位層のプロトコルによる再送処理等も含めたファイル単位の転送遅延時間やスループットなど，ユーザが直接感じるネットワークの性能にどのような影響を与えるかについても検討が行われていない．そこで本論文では，上位層に TCP/IP を用いたネットワークにおいて，ネットワークを流れるトラヒックに自己相似性が発生する原因を，シミュレーション手法を用いて明らかにする．また，トラヒックの自己相似性がパケット転送遅延時間やパケット棄却率といったネットワークの性能にどのような影響を与えるかについても評価を行う．さらに，トラヒックの自己相似性が，ユーザが感じるネットワーク性能にどのような影響を与えるかをについても検討を行う．その結果，トラヒックの自己相似性は，スイッチでのパケット棄却率の影響を受けること，および上位層に TCP/IP の制御がある場合にはネットワークに加わるトラヒックの自己相似性が保存されることを明らかにする．また，ファイル単位の平均転送遅延時間や実効スループットのような，平均的なネットワーク性能はあまり自己相似性の影響を受けないが，ファイル単位の 99.9% 転送遅延時間は自己相似性によって大きく影響されることを示す．

キーワード　ネットワークトラヒック，自己相似性，ハーストパラメータ，TCP/IP，サービス品質

## 1. はじめに

これまで，ネットワークを流れるトラヒックのモデル化を行なう際には，ポアソン到着モデルや ON/OFF ソースによる多重化モデル，MMPP，MAP といったマルコフモデルが広く用いられてきた．しかし，最近，イーサネット上のトラヒックを測定した結果 [1]，ネットワークを流れるトラヒックは自己相似性 (self-similarity) と呼ばれる性質を持っており，従来のマルコフモデルでは十分にモデル化できないことが指摘されている [2]．さらに，VBR (Variable Bit Rate) の動画像トラヒックも自己相似性を持つことが報告されている [3]．これにより，自己相似性を有するトラヒックのための数学モデルに関する研究や，そのような特性を持つトラヒックの多重化に関する研究も盛んに行われている．しかし，例えば，イーサネット上のトラヒックが自己相似性を有するというのはあくまで観測の結果であり，そのような結果が得られる原因をまず調べることが重要である．また，ユーザに対する性能品質がトラヒックの自己相似性によってどのように変化するのかを知る必要がある．

ネットワーク上で観測されたトラヒックがなぜ自己相似性を有するのか，また，自己相似性を有するトラヒックが，ネットワークの性能にどのような影響を与えるかについてもいくつかの研究が行なわれている．例えば [4] では，ネットワークを流れるデータサイズ (ファイルサイズ) の分布が裾野部分の大きな分布 (heavy-tailed distribution) に従うために，それらのデータが多重化された結果，ネットワーク上ではトラヒックが自己相似性を示すと説明されている．文献 [5, 6] では，TCP (Transmission Control Protocol) のような上位層のプロトコルの影響で，ネットワークトラヒックの自己相似性が変化することが指摘されている．さらに，トラヒックが自己相似性を持つ場合，自己相似性を持たない場合に比べて，パケット棄却率などといったネットワークの性能が低下することも示されている．

しかし，TCP のような上位層のプロトコルが存在する場合に，以下の検討が十分に行われていないのが現状である．

- なぜネットワークを流れるトラヒックの自己相似性が変化するか
- 例えば，上位層に TCP が存在する場合，TCP のどの制御 ( フロー制御や，パケット再送制御など ) によって自己相似性が変化するのか
- ネットワークを流れるトラヒック量や，ボトルネックとなる帯域幅，ネットワーク内のスイッチが持つバッファサイズといった，さまざまな要因がトラヒックの自己相似性にどのような影響を与えるか
- ネットワーク内部でトラヒックの自己相似性の度合が変化した結果，ネットワークを利用するユーザが感じるサービス品質がどのように影響されるか

そこで本論文では，上位層に TCP が存在する場合に，なぜトラヒックの自己相似性が変化するのか，トラヒックの自己相似性がネットワークの性能にどのような影響を与えるのか，さらにユーザに対する性能品質にどのような影響を与えるのかを，シミュレーション手法を用いて明らかにする．シミュレーションモデルとしては，サーバー・クライアントモデルを用いて，サーバーからクライアントへ転送されるデータサイズの分布がネットワークで観測されるトラヒックの自己相似性にどのような影響を与えるかについて評価を行う．また，トラヒックの負荷や，ボトルネックリンクの帯域，スイッチのバッファサイズを変化させた場合に，トラヒックの自己相似

[†] 大阪大学大学院 基礎工学研究科, 豊中市
Department of Infomatics and Mathematical Science,Graduate School of Engineering Science, OsakaUniversity, Toyonaka, Osaka 560-8531, Japan

性がどのように変化するかについても評価を行う．特に，上位層で再送制御などの処理をを行わないUDPの場合と比較して，TCPの制御によってどれだけ自己相似性が変化するかに着目した評価を行う．また，ネットワークの性能指標として，サーバーからクライアントまでのパケット転送遅延時間，ネットワーク内でのパケット棄却率などを用いる．

以下，2. 章において，自己相似性の定義と，自己相似性の度合をあらわすハーストパラメータを推定する手法を紹介する．次に，3. 章において，本論文で用いるシミュレーションモデルについて説明する．4. 章では，シミュレーション結果に基づいて，TCPがトラヒックの自己相似性に与える影響，およびトラヒックの自己相似性がネットワークの性能に与える影響について，さまざまな観点から検討を行う．最後に，5. 章において，本論文のまとめと今後の課題について述べる．

## 2. 自己相似性

本章では，自己相似性の定義と，自己相似性の度合を表すハーストパラメータについて簡単に説明する．厳密な定義については，例えば [7] を参照されたい．

### 2.1 自己相似性の定義

ある離散時間上の確率過程 $X = \{X_t : t = 0, 1, 2, \ldots\}$ において，

$$X_n^{(m)} = \frac{1}{m} \sum_{j=(n-1)m+1}^{nm} X_j \qquad (1)$$

となる，新たな確率過程 $X^{(m)} = \{X_n^{(m)} : n = 0, 1, 2 \ldots\}$ を考える．この時，$X^{(m)}$ の自己相関関数 $r^{(m)}(k)$ が十分大きな $m$ に対して以下の性質を満たすとき，確率過程 $X$ は自己相似性を有すると定義される [7, 8]．

$$r^{(m)}(k) \sim k^{-(2-2H)} \text{ただし} \frac{1}{2} < H < 1$$

ただし，記号 $\sim$ は $k \to \infty$ としたときに両辺が近似的に比例することを示す．なお，ここで $H$ は自己相似性の強さをあらわす指標であり，ハーストパラメータ (Hurst Parameter) と呼ばれている．

### 2.2 ハーストパラメータの推定方法

ある観測されたプロセスが自己相似性を持つかどうか，さらに自己相似性を持つ場合そのプロセスのハーストパラメータの値がいくらかを推定する方法が数多く提案されている [7, 9]．ここでは，本論文で使用する 3 種類の推定方法，Variance-Timeプロット，R/Sプロット，Whittleの推定法について簡単に説明する．

Variance-Timeプロットでは，まず，観測されたプロセス $X$ からプロセス $X^{(m)}$ を式 (1) を用いて生成する．このプロセスが自己相似性を有する場合，$X^{(m)}$ の分散を $\mathrm{Var}[X^{(m)}]$ とすると，$\mathrm{Var}[X^{(m)}] \approx am^{2H-2}$ ($a$ は定数) となる性質を持つ．この性質を利用して，$\log(\mathrm{Var}[X^{(m)}])$ を $\log m$ に対してプロットし，十分大きな $m$ に対して適合する直線の傾き $2H-2$ を求めることによって，ハーストパラメータを推定する．

また，R/Sプロットでは，観測されたプロセス $X$ から，

$$R(n)/S(n) = \frac{1}{S(n)}\{\max(0, W_1, W_2, \ldots, W_n) - \min(0, W_1, W_2, \ldots, W_n)\}$$

ただし，

$$W_l = (X_k, X_2, \ldots, X_k) - k\overline{X}(n)$$

を計算する．ここで，$\overline{X}(n)$，$S(n)$ はそれぞれ $X_1, \ldots, X_n$ の平均および標準偏差である．自己相似性を持ったプロセスの場合，$n$ が無限大に近付くにつれ，$R(n)/S(n)$ の平均 $E[R(n)/S(n)]$ が $E[R(n)/S(n)] \approx an^H$ ( $a$ は定数) となる．この性質を利用して，$\log(R(n)/S(n))$ を $\log n$ に対してプロットし，適合する直線の傾き $H$ を推定する．

Variance-Timeプロットおよび R/S プロットでは，与えられた確率過程が自己相似性を持つかどうかの判定や，おおよそのハーストパラメータの値を推定するには有効であるが，統計的に厳密な方法でハーストパラメータの値を求めることができない．一方，Whittleの推定法では，統計的な手法でハーストパラメータを推定し，信頼区間を求めることが可能である．対象とするプロセス $X$ がガウス過程である場合に，$X$ のスペクトル密度を $f(\lambda; z)$ ($z$ は $H$ に依存する)，ピリオドグラムを $I(\lambda)$ としたとき，

$$W(z) = \int_{-\pi}^{\pi} \frac{I(\lambda)}{f(\lambda; z)} d\lambda$$

を最小化する $z$ を求め，これより $H$ を求めるという方法である [9]．なお，プロセス $X$ がガウス過程でない場合は，式 (1) により求めた $X_n^{(m)}$ が $m \to \infty$ の場合ガウス過程に収束するという性質を用いて，$X_n^{(m)}$ に対して Whittle の推定法を適用する．

## 3. シミュレーションモデル

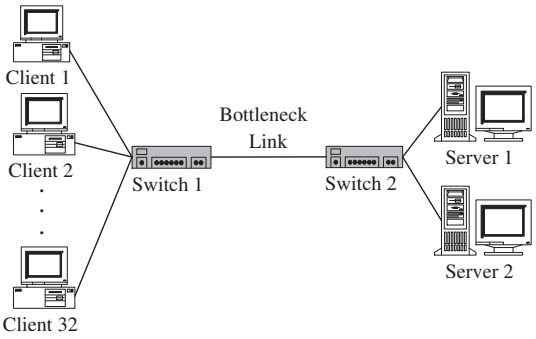


図 1 シミュレーションモデル

図 1 に，本論文で用いるシミュレーションモデルを示す．32 台のクライアントが，2 つのスイッチを経由して 2 台のサーバに接続されているようなサーバー・クライアント型のモデルを考える．各クライアントは，ランダムに選択したサーバにファイル転送を要求し，サーバはこれを受けてクライアントにファイルを転送する．ファイルはパケットに分割して転送されるが，本シミュレーションではパケットサイズ (IP データグラム長) を 1 K バイトの固定長としている．スイッチに到着したパケットは一旦バッファに蓄えられ，FIFO (First In First Out) により到着順にボトルネックリンクに出力されるものとする．上位層のプロトコルとして，TCP (TCP Reno) もしくは UDP を用いる．サーバからクライアントに転送されるファイルサイズの平均値は，UNIX のファイルサイズの平均値に合わせて

22 K バイトとする [10] . なお，クライアントが前回要求したファイルを受信し終えてから，次のリクエストを発生するまでの時間 (リクエスト間隔) の平均は 3.2 秒 とする . また，ファイルサイズの分布は，指数分布 (Exponential Distribution) もしくはパレート分布 (Pareto Distribution) に従うものとし，リクエスト間隔の分布は指数分布に従うものとする .なお，パレート分布の確率分布関数は以下の式で与えられる .

$$P[X \leqq x] = 1 - \left(\frac{k}{x}\right)^{\alpha} \quad 0 < \alpha,\ 0 < k \leqq x \qquad (2)$$

パレート分布は，裾野部分が大きい分布であり，上式の $\alpha$ が小さいほど裾野部分が大きくなる . また，$k$ は確率変数のとり得る最小の値を示す . なお，4. 章におけるシミュレーションでは，パレート分布が生成するファイルサイズの平均値を一定としながら $\alpha$ の値を変化させるため，$\alpha$ の値に応じて $k$ の値も変化している . 例えば，ファイルサイズの平均が 22 K バイトであるため，$\alpha = 1.05$ ならば $k = 2,151$ バイト，$\alpha = 1.95$ ならば $k = 10,975$ バイトとなる . なお，ON 期間もしくは OFF 期間がパレート分布に従うような ON/OFF ソーストラヒックを無限本多重化して得られたトラヒックは，そのハーストパラメータが $(3-\alpha)/2$ で与えられるような FGN (Fractional Gaussian Noise) となることが知られている [4, 11] .

本論文では $\alpha$ を変化させるが，現実に $\alpha$ が取り得る値については以下のような結果が得られている . ネットワーク上のトラヒックには直接関係しないが，トラヒックソース自体の自己相似性に関する研究として，文献 [12] がある . ここでは，Web のドキュメントサイズの分布は，その裾野部分のみに着目すれば $\alpha = 1.12$ になるという結果が示されている . また文献 [10] をもとに，UNIX のファイルサイズ分布の特性を最小自乗法 (Least Squares Estimator) [13] で調べたところ，裾野部分に注目すると$\alpha = 1.294$ のパレート分布に相当することが分かった .

本シミュレーションにおいて，スイッチのバッファサイズは 4 K から 512 K バイト，スイッチ間のリンクの帯域は 150 Kbps から 15 Mbps と変化させて評価を行う . 端末とスイッチ間のリンクの帯域は全て 10 Mbps とする . また，ネットワークに加わるトラヒックの負荷は，

$$(\text{負荷}) = (\text{クライアント数}) \times \frac{(\text{ファイル実転送時間})}{(\text{ファイル実転送時間}) + (\text{リクエスト間隔})}$$

ただし，

$$(\text{ファイル実転送時間}) = \frac{(\text{ファイル長})}{(\text{帯域})}$$

と定義する . 実際には，TCP の再送制御によって，ファイル転送遅延時間 (コネクション設定から最後のパケットの ACK をサーバが受け取るまでの時間) は増大する . なお，最大ウィンドウサイズは，これによってスループットが抑えられないように十分大きな値 (20 Kbyte) とした . 本論文では，リクエスト間隔を変化させることによって負荷を 0.2 から 1.3 と変化させて評価を行う . このシミュレーションモデルでは，スイッチに接続されているリンクがボトルネックリンクとなるため，ボトルネックリンクにおけるトラヒック特性に着目した評価を行う . シミュレーションには ns (Network Simulator) [14] と呼ばれるシミュレーションパッケージを一部変更して使用した .

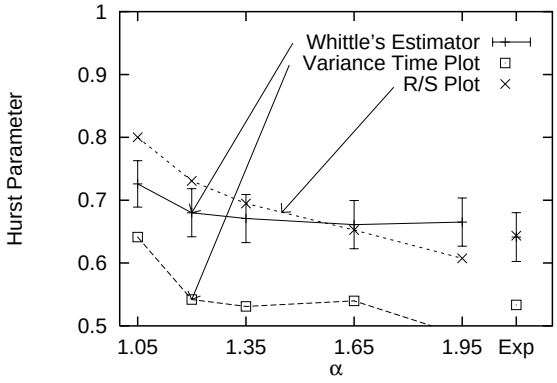


図 2: $\alpha$ とハーストパラメータの関係 (UDP の場合)

## 4. シミュレーション結果

### 4.1 上位層のプロトコルが UDP の場合

本節では，上位層で再送制御などの処理を行わない UDP の場合について評価を行う . これは，次節以降で示す TCP を用いた場合との比較のためである . 以降のシミュレーションでは，特に記述のない場合， スイッチのバッファサイズを 128 K バイト，ボトルネックリンクの帯域を 1.5 Mbit/s ，ネットワークに加わるトラヒックの負荷を 1.1 とする .

#### 4.1.1 ファイルサイズ分布の影響

図 2 に，式 (2) で与えられるパレート分布の，裾野分布の大きさを決定するパラメータである $\alpha$ と，ボトルネックリンクで観測したトラヒックのハーストパラメータとの関係を示す . 図中では，Variance-Time プロット，R/S プロット， Whittle の推定法を用いて求めたハーストパラメータの値を示している . なお， Whittle の推定法を用いて得られたハーストパラメータの値には，95% の信頼区間を示してある . なお，図中右端の “Exp” は，ファイルサイズ分布が指数分布に従う場合の値を示す (分布がパレート分布の場合と区別するため，点のみをプロットしている) . 上位層が UDP の場合は，スイッチに到着したトラヒックがそのまま多重化されてリンクに出力されるため，ファイル転送時を ON 期間，リクエスト間隔を OFF 期間と考えると，ボトルネックとなるリンクを流れるトラヒックは ON/OFF ソーストラヒックを多重化したものになると考えることができる . ON 期間がパレート分布に従う場合，理論的にはハーストパラメータの値はパレート分布のパラメータ $\alpha$ を用いて $(3-\alpha)/2$ と近似できる [4, 8] . また，ON/OFF 期間の分布がともに指数分布に従う場合には，ハーストパラメータの値は 0.5 で与えられる [4, 8] . これは，本論文で行ったシミュレーションのファイルサイズ分布がパレート分布，およびファイルサイズ / リクエスト間隔がともに指数分布の場合に対応する . しかし，シミュレーションより得られたハーストパラメータの値はこれらの値と大きく異なっている (図 2) . つまり，ファイルサイズ分布がパレート分布に従う場合，ハーストパラメータの値は $\alpha$ が大きいほど小さくなっているが，その値は $(3-\alpha)/2$ より小さい . また，ファイルサイズとリクエスト間隔がともに指数分布に従う場合には，ハーストパラメータの値がおよそ 0.64 と，0.5 よりも大きな値となっていることが分かる . これは，スイッチのバッファあふれにより，

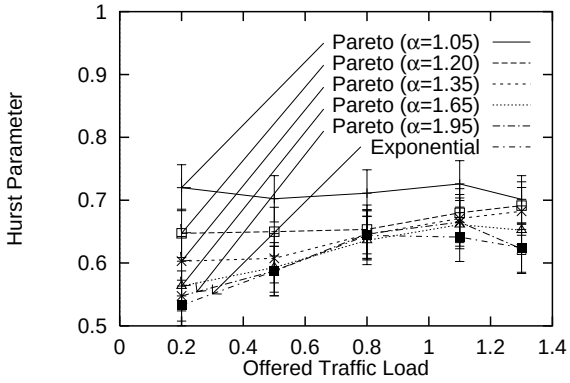


図 3: 負荷とハーストパラメータの関係 (UDP の場合)

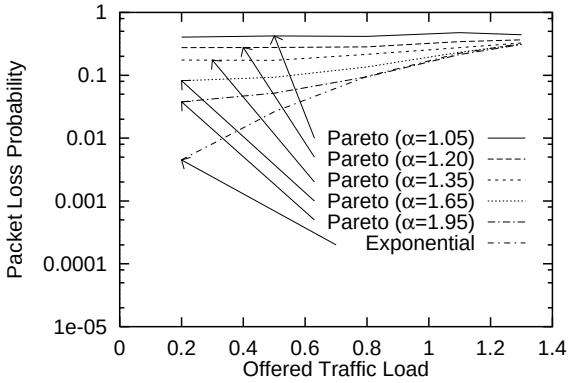


図 4: 負荷とパケット棄却率の関係 (UDP の場合)

10% 以上の大量のパケット棄却が発生しており，ボトルネックに到着するトラヒックの特性が変化してしまうためである．その結果として，ボトルネックで多重化されたトラヒックは，自己相似性が弱まっていると考えられる．

#### 4.1.2 ネットワークに加わるトラヒックの負荷の影響

パケット棄却率が，トラヒックの自己相似性に与える影響を明らかにするため，ネットワークに加えるトラヒックの負荷を 0.2 から 1.2 と変化させた時の，ハーストパラメータおよびパケット棄却率の値を，それぞれ図 3 および図 4 に示す．図 3 より，ファイルサイズ分布がパレート分布に従う時は，パラメータ $\alpha$ が小さい場合，負荷が変化してもハーストパラメータの値はほぼ一定であることが分かる．一方，$\alpha$ が大きい場合には，負荷が高くなるにつれてハーストパラメータの値も大きくなり，ハーストパラメータの値が 0.65 前後に近づいていることが分かる．これは以下のように説明できる．図 4 より，ファイルサイズ分布がパレート分布に従い，なおかつパラメータ $\alpha$ が小さい場合には，トラヒックの負荷に関係なくパケットの棄却がほぼ一定であることが分かる．一方，$\alpha$ が大きい場合，もしくはファイルサイズ分布が指数分布に従う場合には，トラヒックの負荷が高くなるにつれてパケット棄却率が上昇している．この結果，負荷 1.1 前後では，$\alpha$ が小さい時と同程度のパケット棄却が起こっていることが分かる．つまり，パケット棄却率が小さい状況では，ボトルネックに到着するトラヒックの特性が，そのままスイッチから出力されるトラヒックの特性として保存され，この結果ハーストパラメータは ON/OFF ソースを多重化した場合の理論的な値に近づいていると考えられる．一方，パケット棄却率が非常に大きい状況では，スイッチから出力されるトラヒックの特性は，スイッチのバッファでパケットが棄却されるパターンによって決定されるため，ハーストパラメータの値が一定値に近づくと考えられる．

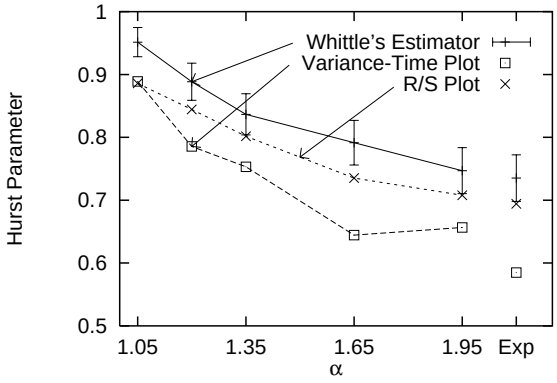


図 5: $\alpha$ とハーストパラメータの関係 (TCP の場合)

### 4.2 上位層のプロトコルが TCP の場合

本節では，上位層が TCP である場合のシミュレーション結果を示す．なお，4.1 節 (UDP の場合) の図 2，図 3 が，それぞれ本節の図 5，8 に対応する．

#### 4.2.1 ファイルサイズ分布の影響

まず，パレート分布のパラメータ $\alpha$ と，ボトルネックとなるリンクを流れるトラヒックのハーストパラメータの関係を図 5 に示す．上位層が UDP の場合 (図 2) と比較すると，TCP の制御が加わったことにより，特に $\alpha$ の値が小さい時にハーストパラメータの値が大きくなっていることが分かる．これは TCP の制御によってスイッチで棄却されるパケット数が大幅に減少したためと考えられる．つまり，スイッチに多重化して入力されるトラヒックの特性がそのまま保持され，スイッチから出力されるためと考えられる．また，ファイルサイズの分布が指数分布の場合は UDP の時と比べてハーストパラメータの値が少し大きくなっている (UDP の場合 0.64，TCP の場合 0.73)．TCP の再送制御によってファイルの転送に要する時間が長くなるため，ファイル転送時間分布の裾野部分が大きくなる．その結果，ファイルサイズ分布が裾野部分の長い分布に従う場合と同等の効果をもたらし，トラヒックの自己相似性が強くなっていると考えられる．

図 6 に，$\alpha$ を変化させた時の，サーバーからクライアントまでの平均パケット / ファイル転送遅延時間を示す．これらの値にはパケット棄却が発生した場合の TCP による再送処理時間も含んでいる．つまり，パケット転送遅延時間は，サーバがパケットを送出してから，そのパケットに対する ACK を受け取るまでの時間である．また，ファイル転送時間は，サーバがファイルを構成する先頭のパケットを送出してから，最後のパケットに対する ACK を受け取るまでの時間である．なお，パレート分布のパラメータ $\alpha$ が小さい場合には，シミュレーションを行うごとに値のばらつきが大きくなる．そのため，ここでは，30 回シミュレーションを行った結果の平均値をプロットし，併せて図中には 95% の信頼区間も示している．

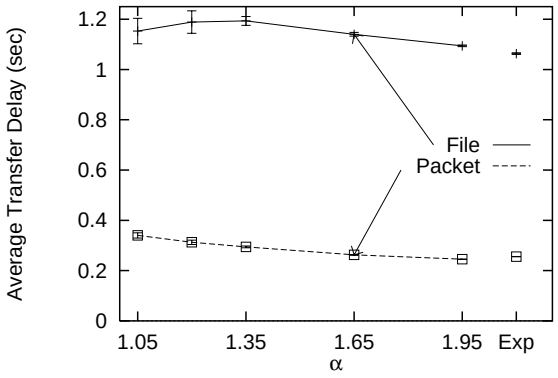


図 6: $\alpha$ と平均パケット / ファイル伝送遅延時間の関係 (TCP の場合)

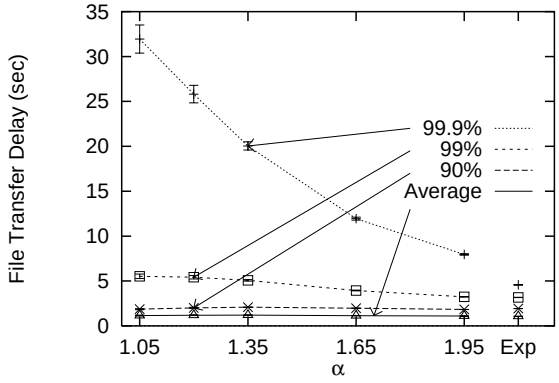


図 7: $\alpha$ と $n\%$ ファイル転送遅延時間の関係 (TCP の場合)

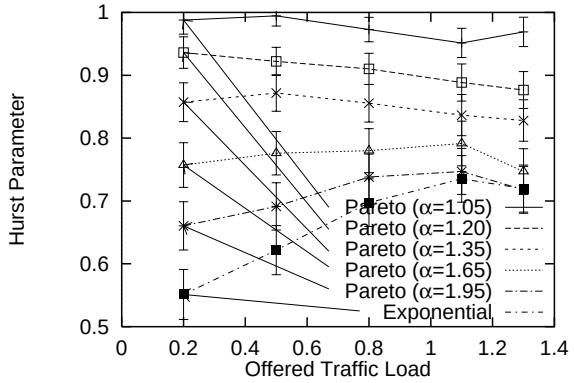


図 8: 負荷とハーストパラメータの関係 (TCP の場合)

平均パケット転送遅延時間に注目すると，上位層が TCP の場合，$\alpha$ が小さくなるにつれて平均パケット転送遅延時間が大きくなっていることがわかる．これは，$\alpha$ が小さくなるとパケット棄却率が大きくなり，その結果 TCP によって途中で棄却されたパケットの再送が行われるためだと考えられる．さらに，$\alpha$ が小さくなると，バッファの平均待ち行列長が大きくなるが，これもパケット転送遅延時間が大きくなる原因と考えられる．このように，たとえ上位層に TCP を使用していたとしても，トラヒックの自己相似性が強くなるに従ってパケットの転送遅延時間が大きくなってしまう．このため，ユーザから見たネットワークの性能品質 (例えばファイル転送遅延時間) はさらに劣化すると予想される．

しかし，図 6 より，平均ファイル転送遅延時間は$\alpha$ の値にかかわらずほぼ一定となっている．この現象は以下のように説明することができる．再送による遅延およびバッファでの遅延時間を無視すれば，TCP のスロースタート機能のため，サイズの小さなファイルは，大きいサイズのファイルに比べより早く転送される．例えば，ファイルサイズ 3 K バイトの場合，ファイル転送遅延時間は往復伝搬遅延時間の 2 倍になるが，11 K バイトの場合には，往復伝搬遅延時間の 4 倍となる．一方，20 K バイトのファイルでは，ファイル転送遅延時間は往復伝搬遅延時間の 5 倍となる．このように，ファイルサイズが小さい方が，ファイルサイズの差に対してファイル転送遅延時間の差が大きくなる．このため，ファイルサイズ分布がパレート分布に従う場合，$\alpha$ が小さい時には，小さなサイズのファイルの発生確率が大きくなるため，平均ファイルサイズは同じであるにも関わらず，$\alpha$ が小さい方が平均ファイル転送遅延が小さくなる．従って，再送およびバッファにおける遅延を考慮しないならば，$\alpha$ が小さい場合には平均ファイル転送遅延時間が小さくなると考えられる．しかし，図 6 では平均ファイル転送遅延時間が $\alpha$ の値にかかわらずほぼ一定となっている．これは，再送およびバッファにおける遅延を考慮しない場合のファイル転送にかかる平均遅延時間の差が，スイッチにおけるパケット棄却率の差 によって生じる再送のための遅延によって打ち消されているためであると考えられる．このことから，上位層のプロトコルが TCP の時には，たとえ入力トラヒックの自己相似性が強い場合でも平均ファイル転送遅延時間にはあまり影響を与えないと言える．

次に，図 7 に 90%，99%，99.9% のファイル転送遅延時間を示す．ここで，$n\%$ ファイル転送遅延時間とは，転送遅延時間分布のうち $n\%$ がその範囲以下に収まるような値を意味する．また図中には，比較のため平均ファイル転送遅延時間もあわせて示している．これより，$\alpha$ の値が変化しても平均ファイル転送遅延時間に影響はないが，$n\%$ ファイル転送遅延時間は $\alpha$ の値に影響されていることが分かる．特に，99.9% ファイル転送遅延時間に着目すると，$\alpha = 1.05$ の時の値は $\alpha = 1.95$ の時の 4 倍以上となっている．以上の結果から，入力トラヒックが自己相似性を持つ場合には，平均ファイル転送遅延時間にはあまり影響がないが，99.9% ファイル転送遅延時間がきわめて大きくなると言える．例えば，Web サーバからのドキュメント転送を考えると，ドキュメント分布の裾野部分が大きく，多重化されたトラヒックの自己相似性が強い場合でも，ドキュメントの平均転送遅延時間のみを見れば，自己相似性が通信品質に与える影響は小さいと言える．しかし，より高品質なサービスを目指す場合，例えば 99.9% 転送遅延時間の向上が望まれる．その場合には，自己相似性の影響による通信品質の劣化が顕著になる．ここで，さらなる品質向上のためには，ネットワークの通信能力に関する検討が不可欠になる．そこで， 4.3 節では，スイッチのバッファサイズやボトルネックリンクの帯域に着目した評価を行なう．

#### 4.2.2 ネットワークに加わるトラヒックの負荷の影響

図 8 に，ネットワークに加わる負荷とハーストパラメータの関係を示す．これより，$\alpha$ が小さい場合には，負荷が小さくなるにつれてハーストパラメータの値が増加していることが分かる．一方，$\alpha$ が大きい場合には，負荷が小さくなるに

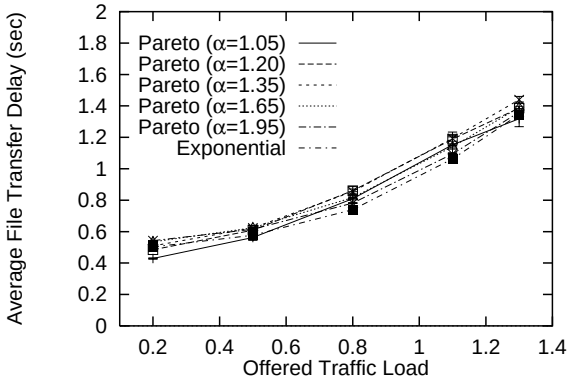


図 9: ネットワークの負荷と平均ファイル転送遅延時間の関係 (TCP の場合)

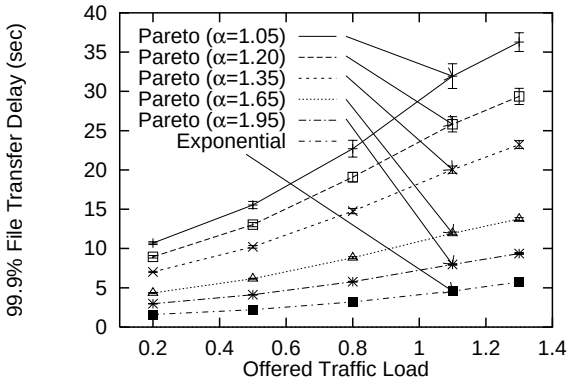


図 10: ネットワークの負荷と 99.9% ファイル転送遅延時間の関係 (TCP の場合)

つれてハーストパラメータの値が減少していることが分かる．これは，トラヒックの負荷が小さくなるにつれてパケット棄却率が小さくなり，その結果，ファイルサイズの分布がそのままボトルネックリンク上のトラヒックに影響しているためと考えられる．

次に，図 9 および図 10 に，負荷を 0.2 から 1.3 と変化させた時の，平均ファイル転送遅延時間および 99.9%ファイル転送遅延時間を示す．ここでは $\alpha$ とハーストパラメータとの関係を検討したときと同様，平均ファイルサイズを 22 K バイト，スイッチのバッファサイズを 128 K バイトとしている．平均ファイル転送遅延時間に注目すると，平均ファイル転送遅延時間は $\alpha$ の値にほとんど依存しないことが分かる．これは，図 6 の場合と同様，再送およびバッファにおける遅延を考慮しないファイル転送遅延時間の差を，パケット棄却率の差が打ち消しているためであると考えられる．

しかし，99.9% ファイル転送遅延時間 (図 10) に着目すると，$\alpha$ の値が小さい時には，トラヒックの負荷が増加すると，99.9% ファイル転送遅延時間が急激に増加していることが分かる．例えば，負荷が 1.3 の場合には，$\alpha$ の値によって 99.9% ファイル転送遅延時間が 9.3 秒から 36.3 秒まで変化している．これは以下のように説明できる．高負荷時に巨大なファイルが発生すると，スイッチにパケットが連続して到着するために，スイッチの待ちパケット長が伸びてしまう．このため，$\alpha$ が小さい時には，スイッチにおいて頻繁にパケット棄却が発

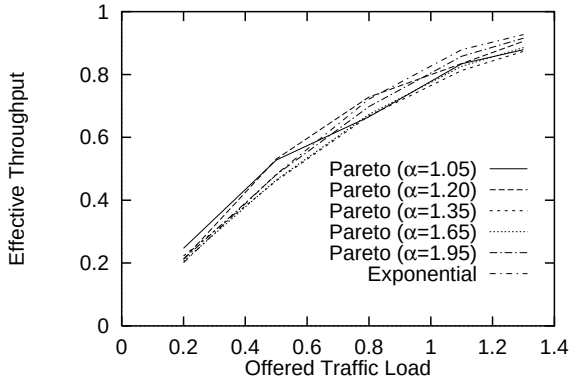


図 11: ネットワークの負荷と実効スループットの関係 (TCP の場合)

生し，TCP の再送制御によって遅延時間がさらに増大するためであると考えられる．

一方，ユーザが感じるネットワーク性能としては，ファイル単位の転送遅延時間の他に，ネットワークのスループットも重要である．そこで，図 11 にネットワークの負荷を変化させた場合にサーバからクライアントに転送されるトラヒックの実効スループットを示す．この図より，ネットワークに加わる負荷を増加させると，実効スループットがほぼ線形的に増加していることがわかる．これより平均ファイル転送遅延時間が $\alpha$ の影響をあまり受けないのと同様に，トラヒックの自己相似性はネットワークの平均的な性能にあまり影響を与えないといえる．

### 4.3 システムパラメータの影響

これまで，ネットワークに入力されるトラヒックの特性 (ファイルサイズ分布やトラヒックの負荷) を変化させた時に，ボトルネックとなるリンクで観測されるトラヒックの自己相似性がどのように変化するかについて検討を行った．その結果，ネットワークを流れるトラヒックの自己相似性は，パケット棄却率に大きく影響されることが分かった．しかし，例えばパケット棄却率はスイッチのバッファサイズやリンクの帯域に依存するため，トラヒックの自己相似性もこれらのシステムパラメータによって変化すると考えられる．そこで本節では，ファイルサイズの分布は平均 22 K バイトのパレート分布，さらにリクエスト間隔を平均 3.2 秒の指数分布に固定し，スイッチのバッファサイズおよびボトルネックとなるリンクの帯域を変化させてシミュレーションを行う．

#### 4.3.1 スイッチのバッファサイズの影響

まず図 12 に，スイッチのバッファサイズを 4 K バイトから 512 K バイトと変化させた時の，ボトルネックリンクで観測されるトラヒックのハーストパラメータを示す．図 12 より，バッファサイズが 4 K バイトときわめて小さい場合は，$\alpha$ の値にかかわらずハーストパラメータの値が 0.75 前後となっているが，これは以下のように説明できる．バッファサイズが 4 K バイトの場合，シミュレーションで用いた TCP のパケットサイズが 1 K バイトであるため，バッファに収容できるパケット数が高々4 個となり，パケット棄却率がきわめて大きく (およそ 10%) なっている．このため，入力されるトラヒックの特

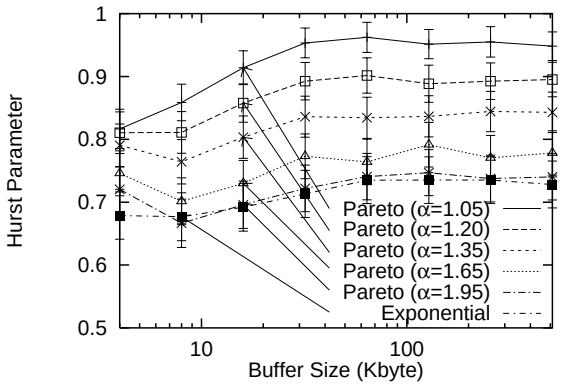


図 12: バッファサイズとハーストパラメータの関係 (TCP の場合)

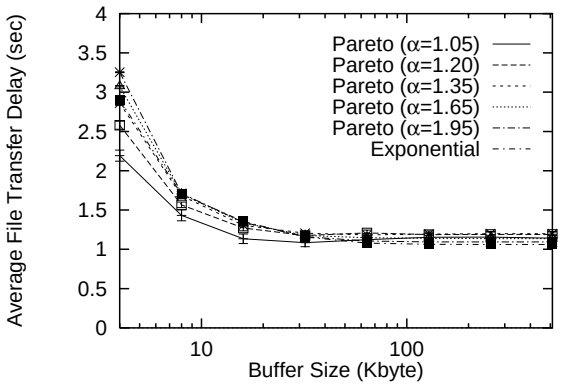


図 13: バッファサイズと平均ファイル転送遅延時間の関係 (TCP の場合)

性が失われ，なおかつバッファの待ち行列長の変動が単純であるために，$\alpha$ の値にかかわらずハーストパラメータが一定になっていると考えられる．一方，バッファサイズが大きくなると，それに伴ってハーストパラメータが大きくなっているが，バッファサイズが 100 K バイト程度を超えるとハーストパラメータの値が一定となっていることが分かる．これは，バッファサイズが 100 K バイト程度あれば，TCP のフロー制御によってほとんどパケットの棄却が発生せず，入力されるトラヒックの特性がそのままスイッチから出力されるためだと考えられる．

次に，この場合の平均パケット転送遅延時間について述べる．バッファサイズがおよそ 32 K バイトまでは，バッファサイズを大きくするにつれて平均パケット転送遅延時間が小さくなっている．しかし，バッファサイズがそれ以上大きくなると，平均パケット転送遅延時間が逆に大きくなっており，また，$\alpha$ の値が小さくなるほど遅延時間が大きくなっている．この現象は以下のように説明できる．TCP のウィンドウフロー制御により，バッファサイズが 32 K バイト程度あれば，スイッチにおいてパケット棄却がほとんど発生しなくなる．このため，それ以上バッファサイズを大きくすると，逆にバッファでパケットが待たされるために，平均遅延時間が大きくなると考えられる．さらに，$\alpha$ が小さい場合には，ファイルサイズ分布の裾野部分が大きいため，バースト的にパケットが到着し，指数分布の場合に比べバッファにおける遅延がより大きくなるため，平均パケット転送遅延時間が大きくなると考えられる．

しかし，ファイル単位の転送遅延時間に着目すると，以下に示すように，バッファの待ち行列長が伸びることによる転送遅延時間の増加はほとんど無視できるようになる．図 13 および図 14 は，それぞれスイッチのバッファサイズを変化させた時の，平均ファイル転送遅延時間および 99.9% ファイル転送遅延時間を示している．この図から分かるように，バッファサイズが 100 K バイト程度あれば，平均ファイル転送遅延時間はほとんど $\alpha$ によって影響されない．これは 4.2.1 節の場合と同様に，再送，およびバッファにおける遅延を考慮しない場合の平均ファイル転送遅延の差をパケット棄却率が打ち消していることが影響しているためだと考えられる．また，99.9% ファイル転送遅延時間 (図 14) については，$\alpha$ が小さいほどファイル転送遅延時間が大きくなっている．これは次のような理由による．バッファサイズが小さい場合にはパケット棄却が発

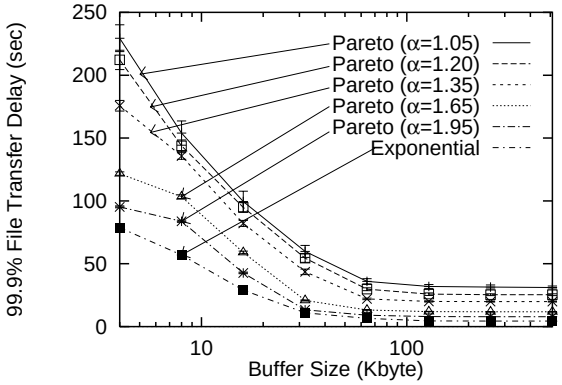


図 14: バッファサイズと 99.9% ファイル転送遅延時間の関係 (TCP の場合)

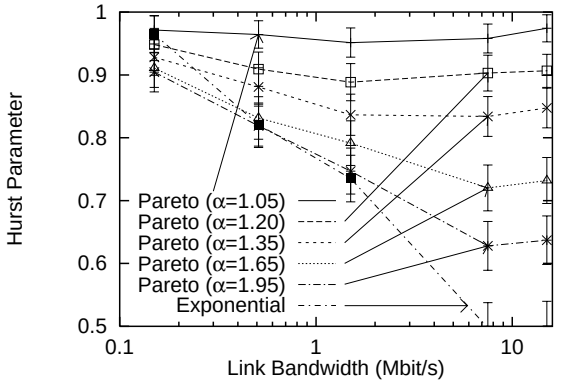


図 15: リンクの帯域とハーストパラメータの関係 (TCP の場合)

生し，TCP による再送処理による遅延が大きくなる．しかし，バッファサイズが大きい場合には，大きいサイズのファイルがバースト的に到着し，バッファでの遅延が大きくなるためである．ただし，$\alpha$ が小さな場合でもサイズの大きいファイルの発生確率は小さいため，図 13 のように平均遅延にはその影響はあまり現れない．しかし，例えば 99.9% 遅延にも着目して，より高品質なネットワークを目指そうとすれば，トラヒックの自己相似性も十分考慮する必要がある．

#### 4.3.2 ボトルネックリンクの帯域の影響

最後に，ボトルネックとなるリンクの帯域を 150 Kbit/s から

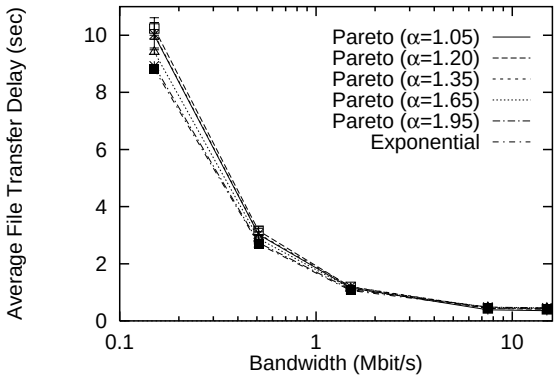


図 16: ボトルネックリンクの帯域と平均ファイル転送遅延時間の関係

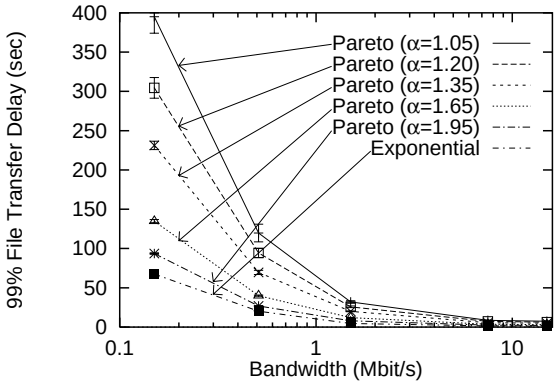


図 17: ボトルネックリンクの帯域と 99.9% ファイル転送遅延時間の関係

15 Mbit/s と変化させた時の，ボトルネックリンクで観測されるトラヒックのハーストパラメータを，図 15 に示す．なお，ここではリンクの帯域にあわせて平均リクエスト間隔を変更し，ネットワークに加わる負荷を常に 1.1 としている．図 15 より，ボトルネックとなるリンクの帯域が小さい場合には，$\alpha$ に関わらずハーストパラメータの値が大きくなっていることが分かる．これは，帯域が小さい場合，ボトルネックリンクは常に最大限利用される状態になるため，異なった時間スケールで観測しても，同じようにリンクが最大限利用されているからであると考えられる．

一方，帯域が大きくなるにつれ $\alpha$ の違いがハーストパラメータに与える影響が大きくなり，ハーストパラメータの値は ON/OFF ソースを多重化したときの理論的な値に近づいていることが分かる．これは以下のように説明できる．帯域が大きい場合には，スイッチのバッファあふれによるパケット棄却が発生しにくくなる ．また，ウィンドウサイズも十分に大きくなることができ，ファイルサイズ分布の特性がそのままボトルネックリンク上のトラヒックに反映されるためであると考えられる．

図 16 および図 17 に，平均ファイル転送遅延時間および 99.9% ファイル転送遅延時間を示す．これより，ボトルネックリンクの帯域が大きくなるにつれて，$\alpha$ の違いがファイル転送遅延時間に与える影響が小さくなっていることがわかる．これは以下のように説明できる．ボトルネックリンクの帯域が 150 Kbit/s の場合，平均ファイル転送遅延時間の違いは，スイッチのバッファにおける待ちパケット長 ($\alpha = 1.05$ の時は平均約 34 パケット，$\alpha = 1.95$ の時は平均約 27 パケット) の違いが原因となっている．しかし，ボトルネックリンクの帯域が大きくなるにつれて，バッファにおける遅延時間が相対的に小さくなるため，$\alpha$ の値による平均ファイル転送遅延時間の差が小さくなっていると考えられる．

また図 17 は，$\alpha$ が小さい場合には，ボトルネックリンクの帯域を増やすことによって 99.9% ファイル転送遅延時間が大きく改善されることを示している．例えば，ファイルサイズの分布が指数分布に従う時には，ボトルネックリンクの帯域を 150 Kbit/s から 15 Mbit/s へと増加させたとしても，99.9% ファイル転送遅延時間は 67 秒から 1.12 秒にしか減少しない．一方，入力トラヒックが強い自己相似性を持つ場合 ($\alpha = 1.05$) には，395 秒から 7.47 秒へと大きく減少している．従って，ボトルネックとなるリンクの高速化は自己相似性を有するトラヒックの存在を前提とした場合，そのネットワークの高品質化という観点からも非常に意味があることが分かる．

## 5. まとめと今後の課題

本稿では，上位層に TCP が存在する場合に，なぜネットワークのトラヒックの自己相似性が変化するのか，トラヒックの自己相似性がネットワーク性能にどのような影響を与えるのかについて検討を行った．特に，上位層で特別な制御を行わない UDP の場合と比較することによって，TCP の制御によってどれだけ自己相似性が変化するかについて評価を行った．

その結果，上位層のプロトコルが UDP の場合には，スイッチでパケット棄却が大量に発生し，ボトルネックリンクに入力されるトラヒックの特性が失われる．このため，ボトルネックで観測されるトラヒックのハーストパラメータは，入力トラヒックの自己相似性の強さに関わらず一定の値に近づくことが明らかになった．一方，上位層プロトコルが TCP の場合には，スイッチでのパケット棄却が少ないため，ボトルネックリンク上で観測されるトラヒックのハーストパラメータは入力トラヒックの値に近くなっている．

また，ネットワークの負荷が上がるとパケット棄却が多くなり，入力トラヒックの特性が失われる．このため，ボトルネックで観測されるトラヒックのハーストパラメータは，入力トラヒックのハーストパラメータの理論値と異なることも明らかになった．さらに，スイッチのバッファサイズやボトルネックリンクの帯域を増やすことによって，パケット棄却が抑えられるため，ネットワークで観測されたトラヒックのハーストパラメータは，入力トラヒックの値に近づくことも明らかになった．

また，ネットワークに加わるトラヒックの自己相似性がユーザが感じるネットワーク性能にどのような影響を与えるかについても検討を行なった．その結果，平均ファイル転送遅延時間はトラヒックの自己相似性にあまり影響されないことが分かった．また，実効スループットも自己相似性の影響をほとんど受けないことが分かった．しかし，99.9% ファイル転送遅延時間は，トラヒックが自己相似性を持つ場合には非常に大きくなり，特にネットワークの負荷が高くなるにつれ急激に悪化することが明らかになった．また，トラヒックが自己相似性を持つ場合，ネットワークにおいてボトルネックとな

るリンクの帯域を増やすことにより，性能悪化を抑えることができるということも明らかになった．

さらに今後の課題として，実際のイーサネット上のトラヒックも自己相似性を有する原因を明らかにするために，より下位層のプロトコル (例えば CSMA/CD) の影響も考慮に入れた評価を行っていく必要があると考えられる．

## 文　献


[1] W. E. Leland and D. V. Wilson, "High time-resolution measurement and analysis of LAN traffic: Implications for LAN interconnection," in *Proceedings of IEEE INFOCOM '91*, (Bal Harbour, FL), April 1991.

[2] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 1–15, Feburary 1994.

[3] M. W. Garrett and W. Willinger, "Analysis, Modeling and Generation of Self-Similar VBR Video Traffic," in *Proceedings of SIGCOMM 94*, vol. 24, pp. 269–280, October 1994.

[4] W. Willinger, M. S. Taqqu, R. Sherman, and D. Wilson, "Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level," in *Proceedings of ACM / SIGCOMM*, pp. 100–113, August 1995.

[5] K. Park, G. Kim, and M. Crovella, "On the Relationship Between File Sizes, Transport Protocols, and Self-Similar Network Traffic," in *Proceedings of International Conference on Network Protocols*, pp. 171–180, October 1996.

[6] K. Park, G. Kim, and M. Crovella, "On the Cause and Effect of Self-Similar Network Traffic," tech. rep., Boston University Computer Science Department, May 1996.

[7] J. Beran, *Statistics for Long-Memory Processes. Monographs on Statistics and Applied Probability*. New York: Champman and Hall, 1994.

[8] B. K. Ryu, *Fractal Network Traffic: From Understanding to Implications*. PhD thesis, Columbia Univ., 1996.

[9] M. S. Taqqu, V. Teverovsky, and W. Willinger, "Estimators for long-range dependence: An empirical study," *Fractals*, vol. 3, no. 4, pp. 785–798, 1995.

[10] G. Irlam, "UNIX File Size Survey – 1993," September 1994. available at http://www.base.com/gordoni/ufs93.html.

[11] M. E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Prssible Causes," in *Proceedings of the 1996 ACM SIGMETRICS*, pp. 160–169, May 1996.

[12] M. E. Crovella and A. Bestavros, "Explaining World Wide Web Traffic Self-Similarity," Tech. Rep. TR-95-015, Boston Univ., October 1995.

[13] N. L. Jhonson and S. Kotz, *Continuous Univariate Distributions-1*. New York: Wiley-interscience, 1970.

[14] "LBNL network simulator (ns)." available at http://www-nrg.ee.lbl.gov/ns/.


（平成 x 年 xx 月 xx 日受付）


### 住田　義明　（学生員）

平成 9 阪大・基礎工・情報卒．現在，同大大学院修士課程在学中．トラヒック制御に関する研究に従事

### 大崎　博之　（正員）

平 5 阪大・基礎工・情報退学．平 9 同大学大学院博士課程了．同年より阪大・基礎工・助手．高速広帯域通信網の性能評価に関する研究に従事．工博．IEEE 会員．

### 村田　正幸　（正員）

昭 57 阪大・基礎工・情報卒．昭 59 同大大学院修士課程了．同年日本アイ・ビー・エム (株) 入社．同社東京基礎研究所を経て，昭 62 阪大・大型計算機センター・助手．平 1 阪大・基礎工・助手．平 3 同講師．平 4 同助教授．この間，待ち行列理論，システム性能評価，広帯域通信網などの研究に従事．工博．IEEE，ACM 各会員．

### 宮原　秀夫　（正員）

昭 42 阪大・工・通信卒．昭 47 同大大学院博士課程了．昭 48 京大・工・助手．昭 57 阪大・基礎工・助教授．昭 62 阪大・大型計算機センター・教授．平 1 阪大・基礎工・情報・教授．平 7 阪大・大型計算機センター・センター長併任．昭 58 ~ 昭 59 米国 IBM トーマスワトソン研究所客員研究員．システム性能評価，マルチメディアシステム，広帯域通信網，ネットワーク管理に関する研究に従事．工博．平 3 本学会論文賞．IEEE，情報処理学会各会員．