

QoS 統計データを利用した動的サーバ選択法

間瀬 憲一^{†a)} 栗林 孝行^{††} 津野 昭彦^{†††}

A Dynamic Server Selection Method Using QoS Statistics

Kenichi MASE^{†a)}, Takayuki KURIBAYASHI^{††}, and Akihiko TSUNO^{†††}

あらまし 本論文では、ミラー化された Web サイトからファイル等をダウンロードする場合に、ダウンロード時間が最小となるサーバを選択する観点から、サーバ選択法を考察する。具体的には、クライアント側でサーバ選択を行う方法を対象として、QoS 統計データに基づく動的サーバ選択法を提案する。過去の QoS データを曜日別、時間帯別に集計することにより、従来のサーバ選択法より高度に過去の蓄積データを利用することを特徴としている。インターネットを利用した実験結果をもとに検討・評価を行う。まず、提案するサーバ選択法のパラメータの最適な値を検討する。次に、本サーバ選択法と既存のサーバ選択法の比較・評価を行う。評価尺度として、平均・最大ダウンロード時間といった客観評価指標に加えて、ユーザ感覚を反映したダウンロード断念率、ストレス量等を用いる。これらの結果から、本サーバ選択法の優位性を示す。

キーワード サーバ選択, ミラーサーバ, QoS, ダウンロード, コンテンツ配信

1. まえがき

今日、インターネットを用いた情報収集はビジネスや日常生活に欠かせない手段となっている。アクセス系を含めたネットワークのブロードバンド化により、多量のデータや映像コンテンツをインターネットで配信することも可能になり、コンテンツ配信ビジネスが発展している。特定のコンテンツに対してユーザからの要求が集中すると、コンテンツを配信するサーバやネットワークがふくそうし、ユーザへの迅速なコンテンツ提供が困難になる場合が考えられる。そこで、同一のコンテンツをインターネットに広く分散配置し、各サイトに置かれたサーバからコンテンツを提供す

る形態が大量のコンテンツ配信要求を効率良く処理し、ユーザの要求に迅速に対応するために有効な手段となる。このような方式はミラーサーバ方式と呼ばれる。ミラーサーバ方式では、ユーザからのコンテンツ要求に対してミラーサーバの一つを選択し、選択されたサーバがコンテンツ配信を行う。すなわち、サーバ選択が必要になる。大別すると、サーバ側でサーバ選択を行う方法とクライアント側でサーバを選択する方法がある。サーバ側で選択する方法においては、まずユーザのコンテンツ要求を受け付ける必要があり、その受付サーバ自体がボトルネックとならない工夫が必要であるが、負荷の少ないサーバを選択することが可能である。一方、クライアント側で選択する方法では、クライアント側で各サーバの何らかの指標を測定・推定・評価し、その指標の最大（最小）のサーバを選択するものである。この方法はサーバの負荷等を直接把握することは困難であるが、コンテンツ配信に関するサービス品質（QoS）を指標とすることにより、ユーザからみて最適なサーバを選択できる可能性がある。本論文は、クライアント側でサーバ選択する方法を対象としている。

クライアント側でサーバ選択する方法に関して、従来いくつかのアプローチが提案されている。ここでクライアントとは、コンテンツを要求するクライアント

[†] 新潟大学工学部情報工学科, 新潟市

Department of Information Engineering, Faculty of Engineering, Niigata University, 2-8050 Ikarashi, Niigata-shi, 950-2181 Japan

^{††} 株式会社 NTT データビジネス開発事業本部, 東京都

IT Business Development Sector, NTT DATA Corporation, Toyosu Center Bldg., 3-3, Toyosu 3-chome, Koto-ku, Tokyo, 135-6033 Japan

^{†††} 日本電信電話株式会社 NTT ネットワークサービスシステム研究所, 武蔵野市

NTT Network Service Systems Laboratories, NTT Corporation, 9-11 Midori-cho 3-chome Musashino-shi, 180-8585 Japan

a) E-mail: mase@ie.niigata-u.ac.jp

そのものだけでなく、2.1に述べるようにクライアントに代わってサーバ選択を行うエージェント等を含めて広い意味で用いている。大別すると、静的な方法、動的な方法、統計的な方法に分類される[1]。静的な方法には、traceroute [2]等のツールを使って各サーバまでのホップ数を測定し、最小ホップで到達可能なサーバを選択する方法が含まれる。最小ホップ数は頻繁には変化しないため、この方法は静的な方法に分類される。静的な方法はネットワークのリアルタイムの状態を考慮することができない[3]。ネットワークのリアルタイムの状態を反映する方式は動的な方法と呼ばれる。例えば、ping, tcping等のツールを使って、サーバまでの往復遅延、利用可能帯域等のQoSをリアルタイム測定し、QoS最良のものを選択する[3], [4]。すべてのルータやサーバがping, tcpingに反応するとは限らず、この方法の適用性には限界がある。動的な方法はネットワークの瞬間的な状態での限られた測定情報のみに基づいてサーバ選択を行うため、正しいサーバ選択ができるとは限らない。過去のQoSデータ(QoS統計データ)に基づいてサーバ選択を行う方法は統計的な方法と呼ばれる[1], [5]~[9]。これには、プローブトラフィックに対するQoSを測定する方法(アクティブ方式)[5], [6]と実際のトラフィックに対するQoSを測定する方法(パッシブ方式)[1], [5], [7]~[9]がある。従来の統計的な方法では、過去のデータを曜日別、時間帯別に利用することや、測定パラメータの最適化等に関して、十分には考慮されていない。統計的な方法はネットワーク設備の障害や設備増減、異常トラフィック等のネットワークの状態変化に対応困難な面がある。このため、統計的な方法で一定数の候補サーバを選択し、そのなかから動的な方法によりサーバ選択することが考えられる。これは組合せによる方法と呼ばれる[1], [10], [11]。

本論文では、ミラー化されたWebサイトからファイル等をダウンロードする際、クライアント側でサーバ選択を行う方法を対象として、QoS統計データを利用した動的サーバ選択法を提案する。この本選択法は上記の分類において、組合せによる方法に属する。組合せによる方法に関しては、これまで検討例は少ない[1]。また、そこで用いるQoS統計データの最適利用に関して、前述のように十分な検討は行われていない。本論文は、対象となるミラーサーバに関する過去のQoSデータを曜日別、時間帯別に集計することにより、従来のサーバ選択法より高度に過去の蓄積デー

タを利用することを特徴としている。インターネットのトラフィックは電話網と同様、曜日や時間帯を反映して変動傾向を有することはよく知られた現象であり、サーバ選択においてもこの特性を把握し利用することが有効と考えられる。また、本論文では、従来の評価で用いられるスループットやダウンロード時間といった客観的なQoSだけでなく、ユーザ感覚も考慮した新たな評価尺度の考察も試みている。

映像コンテンツの配信に関して、ストリーミング技術を用いてオンライン再生を行う方式とデータファイルとしてダウンロードし、その後にユーザ側でオフライン再生する方式がある。前者の場合、利用者が感ずるQoSは映像品質(画素数、フレームレート等)になる。高品質の映像コンテンツを、その品質を低下させずにストリーミング配信を行うのは必ずしも容易ではない。後者の場合、利用者が感ずるQoSはコンテンツのダウンロード時間になる。いったんダウンロードしてしまえば、高品質の映像再生が容易であり、利用者が繰り返して再生を行う場合にも適しているといえる。本論文では、後者を対象とし、利用者からみたQoSとしてコンテンツのダウンロード時間に着目する。ここで、コンテンツダウンロード時間を利用者が要求を出してからコンテンツダウンロードを完了するまでの時間と定義する。この評価指標は映像コンテンツだけでなく、一般の大規模データファイルをダウンロードする場合にも適用できる。

本論文は以下のように構成される。2.では、QoS統計データを利用した動的なサーバ選択法のメカニズムを提案する。3.では、インターネットを用いた実験結果をもとに測定パラメータの最適化を行う。4.では、インターネットを用いた実験により提案するサーバ選択法と既存のサーバ選択法の比較・評価を行い、本選択法の有効性を示す。5.で得られた結果をまとめ、今後の課題について述べる。

2. システムの実現環境と提案するサーバ選択法

2.1 実現環境

ミラーサーバの運用形態として、コンテンツ配信事業者等がコンテンツ配信サーバを複数サイトに用意し、継続的に利用する場合を考える。クライアントは複数のサーバの一つからコンテンツ(以下、目的ファイルと呼ぶ)を取得する。この際、ファイルダウンロード時間が最小となるサーバを選択することが目標となる。

統計的な方法やそれをベースとする組合せによる方法では、定期的に QoS データを測定・収集する必要がある。クライアントがエンドユーザの場合、QoS データを管理するユーザの負担が大きい。また、個々のユーザが定期的に QoS 測定を行うため、ネットワークへの負荷も大きい。したがって、より現実的なのは、近隣のユーザを代表して QoS データを測定・収集するエージェント (QoS モニタと呼ぶ) をネットワークの各所に配置し、それをサーバ選択のためのクライアントとする形態である。このような QoS モニタはコンテンツ配信事業者がサービスレベルアグリーメント (SLA) 管理の観点から自社のコンテンツ配信 QoS を把握するために設ける場合や、コンテンツ配信事業者等から QoS 管理を請け負う事業者が設ける場合が考えられる。QoS モニタはコンテンツ配信事業者のシステムからの情報や過去のエンドユーザのアクセス履歴等に基づいて、対象とするコンテンツを特定するとともに、そのコンテンツを保有するミラーサーバの位置 (URL, IP アドレス等) を得る。このミラーサーバに対して QoS データの収集を行う。このような QoS モニタを前提とすると、このモニタにサーバ選択機能をもたせることが現実的である。このとき、エンドユーザがコンテンツ要求を行うと、その要求は QoS モニタに通知され、QoS モニタがサーバ選択を行い、その結果、選択されたサーバの位置がエンドユーザに通知される。具体的には、文献 [8], [9], [12], [13] 等の仕組みの利用が考えられる。以下では、このような QoS モニタも含めて、単にクライアントと呼ぶ。

コンテンツ配信を高速化する手段としては、ミラーサーバのほかにキャッシュサーバを用いる方法がある。キャッシュサーバ方式では、ユーザ側のネットワーク設備内にキャッシュサーバを設け、アクセス頻度の高いコンテンツをキャッシュする。この場合にはクライアントはキャッシュサーバにアクセスすることが基本となるので、クライアントがサーバ選択をする必要はない。しかし、キャッシュサーバをもてないユーザ、コンテンツ内容が時間とともに更新される場合のようにキャッシュサーバの利用が適さない場合等を考慮すると、キャッシュサーバのみではコンテンツ配信の高速化の問題に完全に対応することは困難であり、ミラーサーバ方式との併用も考えられる。この場合には、キャッシュサーバにないコンテンツを取得する段階で、クライアント側からのサーバ選択を適用することが考えられる。

2.2 提案するサーバ選択法

1. に述べたように、サーバ選択に関して組合せによる方法が考えられるが、従来、方式構成やメカニズムに関して、十分な検討は行われていない。また、QoS データの測定、集計に関するパラメータ設定の範囲や最適化に関しても体系的な検討は行われておらず、サーバ選択への効果も明らかではない。本論文では、QoS データの測定、集計の手順と方法を明確化し、曜日・時間帯別の利用を図るため、以下に述べるように、QoS サンプル、QoS 値、固定・移動平均等の概念を導入しサーバ選択法を構築する。

クライアントは対象となる複数のサーバから定期的に一定サイズのファイル (以下、プローブファイルと呼ぶ) をダウンロードして、その際のスループットを測定する。これを QoS サンプルと呼ぶ。QoS サンプルは例えば、10 分ごとに取得する。クライアントが目的ファイルを取得する場合、まず、その曜日と時間帯に対応する過去の QoS サンプルの平均値 (QoS 値と呼ぶ) を求める。ここで、過去とは、直近の k 週間 (例えば $k = 1 \sim 3$) を指す。QoS 算出区間の設定に関して、固定平均と移動平均を考慮する (図 1)。固定平均は時間軸を一定時間幅の区間ごとに区切り、各区間で QoS 値を求めるものである。移動平均は目的ファイルのダウンロード開始時刻を中心とした一定時間幅の区間において、QoS 値を求める方法である。例えば図 1 のように時間幅を 1 時間とし、11 時 10 分にファイルをダウンロードする場合、固定平均では、(a) のように 11 時～12 時までの QoS サンプルを使用して平均値を計算する。このため、ダウンロード時刻が対象区間の端になると QoS 値の時間変動が反映されにくくなる。一方、移動平均では、(b) のように 10 時

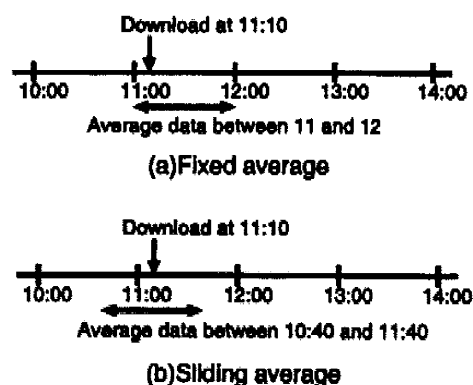


図 1 固定平均と移動平均の求め方 (1 時間平均の場合)

Fig. 1 Fixed average and sliding average.

40 分～11 時 40 分までの QoS サンプルを利用して平均値を計算する。このため、QoS 値の時間変動をより正確に反映することができる。次に、QoS 値に基づいて、上位から n 個 (n は整数) の候補サーバを選択する。選択された n 個の候補サーバに対して、目的ファイルのダウンロード時にリアルタイムの QoS 測定を行う。この結果に基づき、リアルタイム QoS が最良のサーバを選択する。なお、このように、段階的にサーバ選択を絞り込む方法は文献 [1] で提案されたものである。

定期的な QoS サンプルの測定とリアルタイムの QoS 測定に利用できるツールは種々あるが、ここでは、一例として、前者に wget [14]、後者に echoping [15] を使用した。それぞれのツールの特徴を付録 1. に示す。通常の利用では、Web ブラウザを用いてファイルをダウンロードすることが多い。wget はファイルダウンロード時のスループットを測定するツールであり、ファイルサイズ/スループットによりダウンロード時間が求まる。この値は Web ブラウザを用いてファイルをダウンロードする場合のダウンロード時間と高い相関をもつ (付録 2.)。echoping は往復遅延を測定するツールであり、QoS 測定時間を節約するために採用したが、wget を利用することも可能である。

Web サーバの一つのボトルネックはディスク処理である。ディスク内のファイルにアクセスが集中すると、ディスク IO ネットクにより、ファイルダウンロード時間が増加する。そこで、人気コンテンツをメモリ上に配置することにより、ディスク IO 処理が減り、ダウンロード時間が削減される。このような方式において、上述したサーバ選択法が的確に動作するためには、2.1 に述べたサーバ選択の実現環境のなかに、各ミラーサーバにおいて目的ファイルとプローブファイルを同一の蓄積媒体に配置するメカニズムを組み込むことが望ましい。ミラーサーバにより適切な負荷分散が図られているシステムでは、特定のサーバに過負荷が生ずることが避けられるので、ディスク IO 処理がボトルネックとなる可能性は少ない。このような場合には、Web サーバよりネットワーク側がボトルネックとなることが多いため、目的ファイルとプローブファイルの蓄積媒体が異なっても、サーバ選択上大きな支障は生じないと考えられる。

3. サーバ選択法のパラメータ最適化

3.1 プローブファイルのサイズ

QoS サンプルを得るためのプローブファイルのサイズについて検討する。四つのサーバ (日本: a, b と米国: f, g) から様々なサイズのプローブファイルをダウンロードするときのスループットを、wget を用いて 1 時間に 1 回ずつ測定する。測定期間は 1 週間とし、スループットの平均値を図 2 に示す。

これより、プローブファイルのサイズとともにスループットが増加するが、100 kbytes 程度で飽和する傾向をもつことがわかる。したがって目的ファイルのサイズが 100 kbytes より大きい場合にも 100 kbytes 程度のプローブファイルを用いることにより、目的ファイルのスループットと同等のスループットを把握することができる。すなわち、定期的に 100 kbytes のプローブファイルをダウンロードしておけば、そのときのスループットの値をベースとして目的ファイルダウンロード時のスループットを予測できることがわかる。しかし、プローブファイルのサイズが大きく測定周期が短いと、QoS サンプルを得るためにネットワークに加わる負荷は無視できない大きさになるおそれがあり、プローブファイルのサイズは極力小さく設定したい。サーバ選択で必要なのは、目的ファイルダウンロード時のスループットそのものの予測ではなく、高スループットのサーバの選択である。すなわち、プローブファイルダウンロードから得られる QoS サンプルによって、目的ファイルダウンロード時のスループット

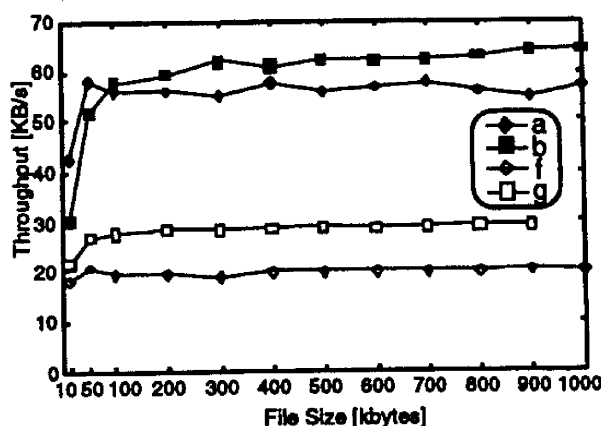


図 2 各ファイルサイズにおけるスループット (1 週間平均) (プローブファイルサイズ: 10～900 kbytes)

Fig. 2 Throughput for various file sizes (1 week average) (probe file size: 10～900 kbytes).

プットそのものを予測する必要はなく、目的ファイルダウンロード時に高スループットが期待できるサーバ選択のための指標となるデータが得られればよい。このためには、プローブファイルのスループットと目的ファイルのスループットの間に一定の相関関係があり、任意の2サイトにおいて、プローブファイルダウンロードに対してスループットが高いサイトは目的ファイルダウンロードに対するスループットも高くなる傾向があればよいと考えられる。

そこで、六つのサーバ（日本：a, b, c, d と米国：e, f）から10～100 kbytesのサイズのプローブファイルをダウンロードする場合のスループットを、wgetを用いて1時間に1回ずつ、測定した。測定期間は1週間とし、スループットの平均値を図3に示す。これより、多少のばらつきはあるものの、目的ファイルのサイズ1 Mbytes に対して、プローブファイルのサイズを10 kbytes としても、異なるサイト間でのスループットの大小関係がおおむね維持されることがわかる。よって、プローブファイルのサイズを10 kbytes としても、目的ファイルのダウンロードに対して高スループットとなるサーバ候補を選択できる可能性があることがわかる。そこで、以下の検討では、プローブファイルのサイズを10 kbytes 以下に設定する。前述のように、スループットはプローブファイルが100 kbytes 程度で飽和する傾向をもつので、目的ファイルが1 Mbytes より大きくなってもプローブファイルのサイズを10 kbytes より大きく設定する必要性は少ないと考えられる。プローブファイルを10 kbytes にすることは、4. におい

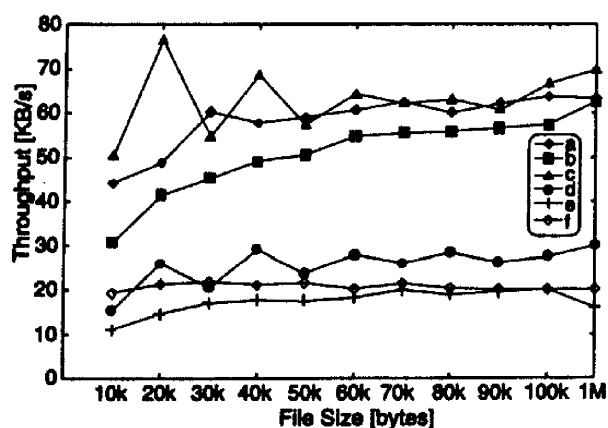


図3 各ファイルサイズにおけるスループット（1週間平均）
（プローブファイルサイズ：10～100 kbytes）

Fig.3 Throughput for various file sizes (1 week average) (probe file size:10～100 kbytes).

て実際にサーバ選択上効果が期待できることからその妥当性が確認される。ただし、環境やアプリケーションが異なれば、プローブファイルの最適サイズも見直すことが必要であろう。

3.2 最適パラメータの組合せ

(1) 実験内容

2. で提案したサーバ選択法は QoS 値に基づいてサーバ選択の候補を決める方式であり、このサーバ選択法が有効であるためには QoS 値そのものが適切に算出される必要がある。QoS 値はいくつかのパラメータを含んでいる。それぞれのパラメータの値を変化させた場合のサーバ選択の性能を実験的に評価することにより、最適なパラメータの組合せを選び出し、提案するサーバ選択法の最適化を行う。対象サーバを、デンマーク、エストニア、スウェーデン、スロバキア、香港〔2サイト〕、日本、メキシコ、アメリカ〔2サイト〕の計10サイトに設定した。実験期間を4週間とする。最初の3週間（2001年7月11日～31日）はQoSサンプルを取得するための期間である。次の1週間（2001年8月1日～7日）は目的ファイルをダウンロードする期間である。この期間において、各対象サーバからプローブファイルと1 Mbytesの目的ファイルを wget を用いて1時間に1回ダウンロードし、その際のダウンロード時間を測定する。測定期日と各パラメータの設定値を表1に示す。

(2) 実験結果と考察

初めに、QoS 値のみに基づいて、サーバを選択する場合に、最良のサーバ選択ができるように QoS 値算出パラメータを選定することを考える。目的ファイルの i 回目の測定において、最小ダウンロード時間のサーバ（最適サーバ）からの目的ファイルダウンロード時間を T_i 、最大 QoS 値のサーバからの目的ファイルのダウンロード時間を T'_i とするとき、ダウンロード時間の平均時間差 T_{avg} を次式で定義する。

$$T_{avg} = \frac{\sum_{i=1}^m |T_i - T'_i|}{m}$$

ここで、 m は測定回数である。平均時間差が小さいほど、適切なパラメータの組合せであるといえる。QoS サンプルの測定周期10分、QoS 値算出期間を直近1週間とした場合の実験結果を図4に、プローブファイルのサイズ10 kbytes、時間幅1時間、移動平均での実験結果を図5に示す。なお、QoS サンプル

表 1 QoS 値算出パラメータと条件

Table 1 Parameters and conditions for calculating QoS values.

QoS サンプルの測定周期	10 分	20 分	30 分
QoS 値算出の時間幅	30 分間	1 時間	2 時間
プローブファイルのサイズ	1 kbytes	5 kbytes	10 kbytes
QoS 値算出期間 (直近 k 週間)	1 週間	2 週間	3 週間
QoS 値算出区間	固定平均 移動平均		
測定対象サイト数	10 (ヨーロッパ 4, アジア 3, 北米 3)		
QoS サンプル収集期間	2001 年 7 月 11 日～31 日		
ダウンロード時間測定期間	2001 年 8 月 1 日～7 日		

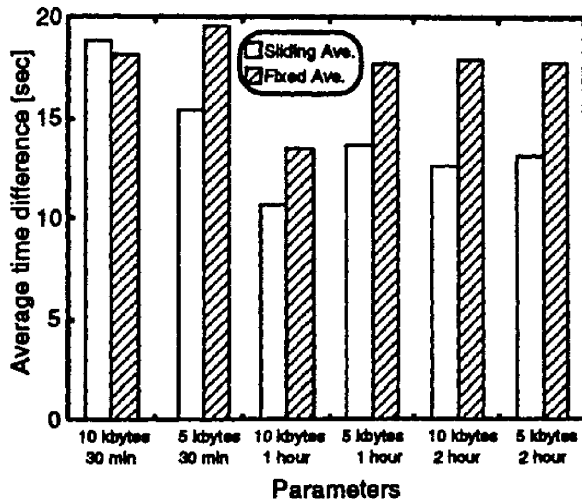


図 4 測定周期 10 分, QoS 算出期間 1 週間での実験結果
Fig. 4 Comparison of download performance (measurement interval is 10 min and QoS value is averaged over the latest one week).

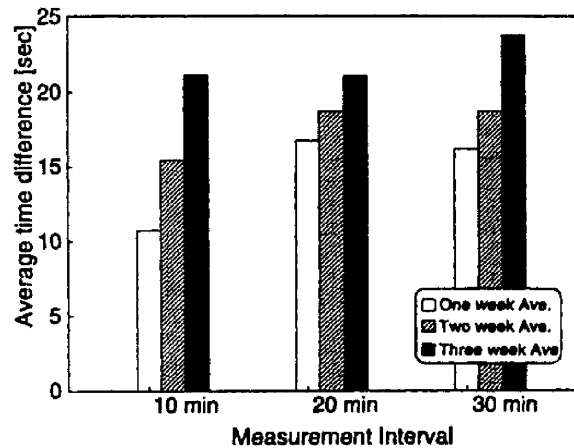


図 5 プローブファイル 10 kbytes, 時間幅 1 時間, 移動平均での実験結果
Fig. 5 Comparison of download performance (probe file size is 10 kbytes and QoS value is obtained for one hour based on the sliding average).

のファイルサイズが 1 kbytes の場合については、プローブファイルのサイズが小さすぎて wget による測定結果が極端な値で一定となってしまう、意味のあるスループットが得られなかったため、割愛した。図 4, 図 5 の結果から、

- プローブファイルのサイズ: 10 kbytes
- QoS サンプルの測定周期: 10 分
- QoS 値算出の時間幅: 1 時間
- QoS 値の算出区間: 移動平均
- QoS 値算出期間: 直近の 1 週間

のパラメータを組み合わせたときに、平均時間差が最小になることがわかった。

この理由としては、まず、プローブファイルのサイズが小さすぎると、wget での測定精度が低下してしまい、正確な結果が得られなくなる傾向がある。このためサイズが大きいほう（このケースでは 10 kbytes）が良い結果を与える。QoS サンプルの測定周期が大き

くなると、QoS 値の計算対象となるサンプル数が少なくなるため、安定した測定データが得られなくなる傾向がある。このため、実験範囲では 10 分の場合が最良の結果を与える。同様に、QoS 値算出の時間幅が 30 分ではサンプル数が少なくなるため、安定した測定データが得られなくなる傾向がある。2 時間ではサンプル数は多くなるが、トラヒックの時間変動をとらえにくい。このため、中間的な値の 1 時間が最も良い結果を与えると考えられる。QoS 値算出区間に関して、移動平均ではダウンロード時刻を中心として平均値を計算するため、QoS 値の時間変動を的確に反映することができるが、固定平均ではダウンロード時刻が対象区間の端になると QoS 値の時間変動が的確に反映されなくなるため、移動平均のほうが良い結果を与えると考えられる。QoS 値算出期間は直近の 1 週間が良い結果を与える。これはインターネットの状態は日々変化するため、数週間前の QoS サンプルの値はあまり参考にならないことを示していると考えられる。

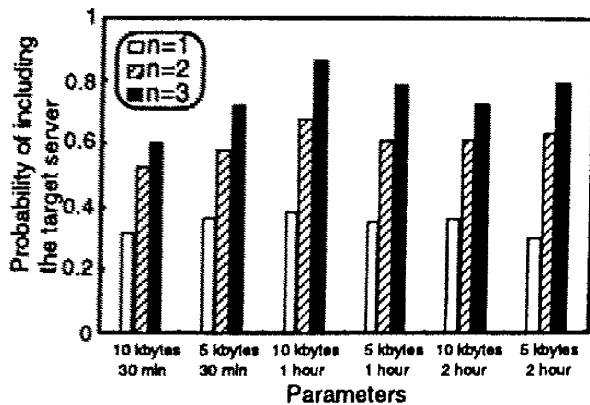


図 6 測定周期 10 分、直近 1 週間での実験結果

Fig. 6 Comparison of server selection performance (measurement interval is 10 min, QoS value is obtained based on the sliding average, and averaged over the latest one week).

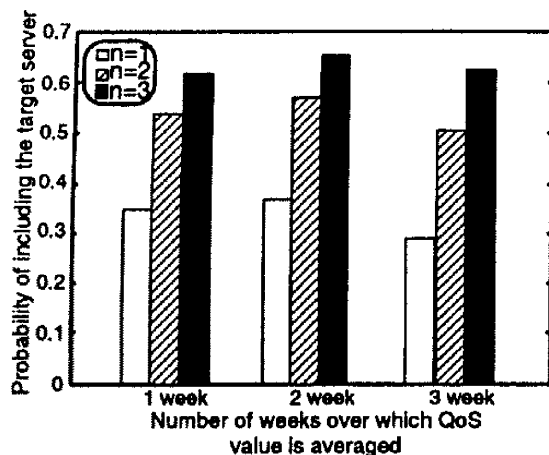


図 7 QoS サンプル 10 kbytes、測定周期 10 分、時間幅 1 時間での実験結果

Fig. 7 Comparison of server selection performance (probe file size is 10 kbytes, measurement interval is 10 min, QoS value is obtained for one hour based on the sliding average).

次に、提案するサーバ選択法に直接関係する性能として、QoS 値に基づき上位 n サーバを選んだときに最適サーバが含まれる確率を求めた。測定期日とパラメータの設定範囲は表 1 と同様である。QoS 値算出区間を移動平均とする。QoS サンプルの測定周期 10 分、QoS 値算出期間として直近 1 週間での実験結果を図 6、QoS サンプル 10 kbytes、QoS サンプルの測定周期 10 分、QoS 値算出の時間幅 1 時間での実験結果を図 7、プローブファイルのサイズ 10 kbytes、QoS 値算出の時間幅 1 時間、QoS 値算出期間として直近 1 週間での実験結果を図 8 に示す。図 6 ～ 図 8 の結果

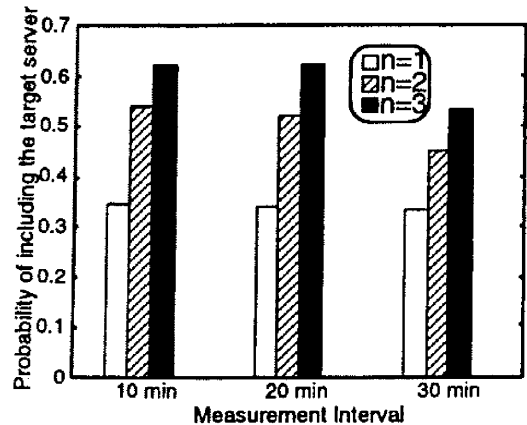


図 8 プローブファイル 10 kbytes、時間幅 1 時間、直近 1 週間での実験結果

Fig. 8 Comparison of server selection performance (probe file size is 10 kbytes, QoS value is obtained for one hour based on the sliding average and averaged over the latest one week).

からも同様に、前に示したパラメータの値の組合せが最良の結果を与えることがわかる。

更に今回のパラメータの組合せが最適であることを検証するために、プローブファイルのサイズが 10, 15 kbytes の場合、QoS サンプルの測定間隔が 10, 5 分の場合について追加の実験を行い、測定結果の比較を行ったが、平均時間差にほとんど差がないということがわかった。このことから、測定間隔を 5 分間隔、プローブファイルのサイズを 15 kbytes としても測定精度の向上はあまり見込めないと考えられるので、今回の実験環境では、上述した測定パラメータの値の組合せがほぼ最適であると考えられる。この場合（プローブファイルのサイズ：10 kbytes、QoS サンプルの測定周期：10 分）、仮に対象サーバ数を 1,000 とすると、QoS データ取得のために消費される平均帯域は、133 kbps 程度である。これはブロードバンドアクセスの環境下では許容範囲と考えられる。

4. サーバ選択法の比較・評価

4.1 比較するサーバ選択法

2. では、組合せ法によるサーバ選択法の一つとして、QoS 統計データを利用したサーバ選択法を提案し、3. では実験結果をもとに QoS 値に関する測定パラメータの最適化を行った。本章では、提案するサーバ選択法が実際にサーバ選択を行ううえでどれだけ有効であるか、ということについて評価するため、実験を通して既存のサーバ選択法との比較・評価を行う。比較の

対象とするのは次のサーバ選択法である。

【選択法 A】 2.2 に述べた提案するサーバ選択法である。3. の結果に基づき、QoS 値に関する各パラメータの値を次のように設定する。すなわち、プローブファイルのサイズ：10 kbytes、QoS サンプルの測定間隔：10 分、QoS 値算出の時間幅：1 時間、QoS 値算出区間：移動平均、QoS 値算出期間：直近の 1 週間とする。また、echoping で用いるファイルサイズを約 260 bytes、候補サーバ数を $n = 3$ とする。

【選択法 B】 直近 1 週間の QoS 値のみからサーバを選択する。したがって、統計的な方法であり、選択法 A において $n = 1$ の場合に相当する。

【選択法 C】 最小ホップ数のサーバを選択する。

【選択法 D】 選択法 A と同様に 3 候補サーバを選び、そのなかからランダムに 1 サーバを選択する。

【選択法 E】 全サーバに対して振られた一定の順番により、1 サーバを選択する（ラウンドロビン選択）。

4.2 サーバ選択の実験

(1) 実験内容

対象サーバを、デンマーク、エストニア、香港、メキシコ、スウェーデン、スロバキア、アメリカの計 7 サイトに設定した。実験期間を 2 週間とする。最初の 1 週間（2001 年 12 月 14 日～20 日）は選択法 A、B に必要な QoS サンプルを取得するための期間である。次の 1 週間（2001 年 12 月 21 日～27 日）は目的ファイルをダウンロードする期間である。この期間において、各対象サーバから 1 Mbytes の目的ファイルを wget を用いて 1 時間に 1 回ダウンロードし、その際のダウンロード時間を測定する。同時に 4.1 に示した五つのサーバ選択法を用いてサーバ選択を行い、サーバ選択の妥当性を評価する。選択法 B、C、D、E の場合、ダウンロード時間とは wget によって測定されるダウンロード時間（スループットから換算）である。選択法 A の場合には、目的ファイルのダウンロード直前に、3 個の対象サーバに対して順次 echoping を行う時間を wget のダウンロード時間に加えたものをダウンロード時間としている。

(2) 実験結果と考察

実験結果に基づき各サーバ選択法の平均ダウンロード時間を図 9 に、最大ダウンロード時間を図 10 に示す。各選択法の有効性について評価するために、各時間帯においてダウンロード時間最小となる最適サーバを選択する理想法の結果も併せて示す。

図 9、図 10 の結果から、選択法 E の特性が最悪で

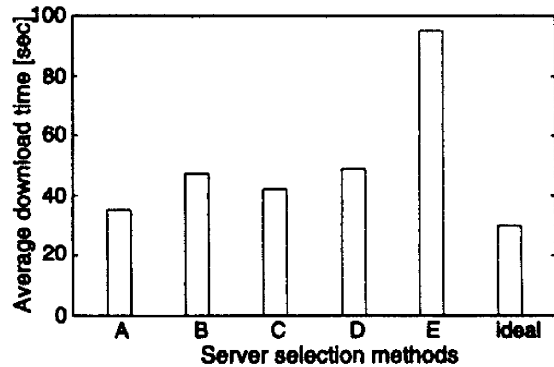


図 9 平均ダウンロード時間の比較

Fig. 9 Comparison of average download times.

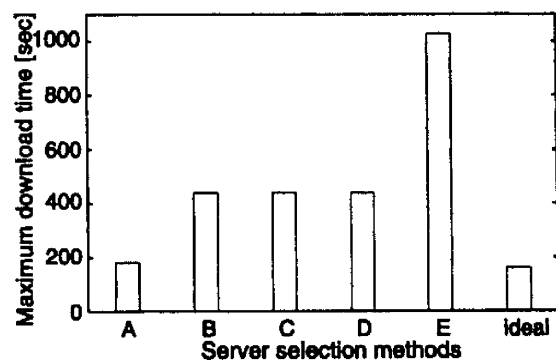


図 10 最大ダウンロード時間の比較

Fig. 10 Comparison of maximum download times.

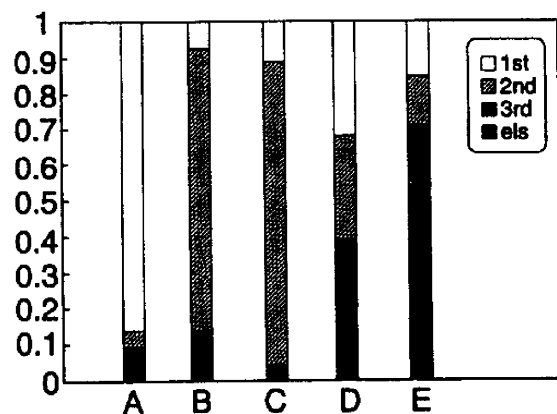


図 11 選択サイトの順位の比較

Fig. 11 Comparison of server selection correctness.

ある。選択法 E はラウンドロビン選択であるから、全対象サーバの平均的な特性が反映する。したがって、これが最悪であることは、各サーバからのダウンロード時間にはばらつきがあり、サーバ選択を適切に行うことの重要性を示している。選択法 A の平均・最大ダウンロード時間は理想法にかなり近く、他の選択法

より優れていることがわかる。図 11 は各選択法が選択するサーバが実際の目的ファイルのダウンロードにおいてダウンロード時間最小 (1 位), 2 位, 3 位, 4 位以下のサーバのいずれを選択したかの比率を示している。この結果から, 選択法 A では 1 位のサーバを 80% 以上の割合で選択できていることがわかる。他の選択法ではこの割合がかなり低く, 選択法 A の優位性が明らかである。図 11 より選択法 B, C は 2 番目のサーバを選択する割合が高いことがわかる。このことから, 1 週間前の QoS 値と実際のダウンロード時間, 及び, ホップ数と実際のダウンロード時間の間には高い相関があるが, 最適サーバを選択するための決め手になるほどではないことがわかる。また, 選択法 D の性能が選択法 A に比べて低いことから, リアルタイムでの QoS 測定が最適サーバ選択に重要な要因になっていることがわかる。

次に, ファイルをダウンロードする際に許容できる最大時間 (ダウンロード許容時間) が与えられており, それを越える場合にはファイルのダウンロードを断念するモデルを考える。各選択法における断念率を上記の実験データに基づいて評価した。その結果を図 12 に示す。選択法 A, B, C の特性は許容時間が 40 秒以上では同程度であるが, 40 秒より小さくなると, 選択法 A の断念率は他の選択法に比べて, かなり小さくなることがわかる。更に, ファイルのダウンロード時間が増えるとストレス量が増えるモデルを考える。一例として, ストレス量 S とダウンロード時間 D の関係に次式が成立すると仮定する。ここで, D_{\max} はダウ

ンロード許容時間, m はパラメータである。一例として $m = 5$ とし, S と D の関係を図 13 に示す。

$$S = \left(\frac{D}{D_{\max}} \right)^m \quad (1)$$

式 (1) を用いてダウンロード許容時間に対する各選択法のストレス量を実験データに基づいて評価した。その結果を図 14 に示す。ダウンロード許容時間に対するストレス量の特性に関して, 選択法 A は他の選択法に比べて, 各段に優れた特性をもつことが明らかである。このように, 単に平均ダウンロード時間, 最大ダウンロード時間のような客観特性で比較するだけでなく, ダウンロード断念率やストレス量といったユーザの立場からの評価指標を考慮することにより, ユーザ感覚により適合したサーバ選択法の評価が可能

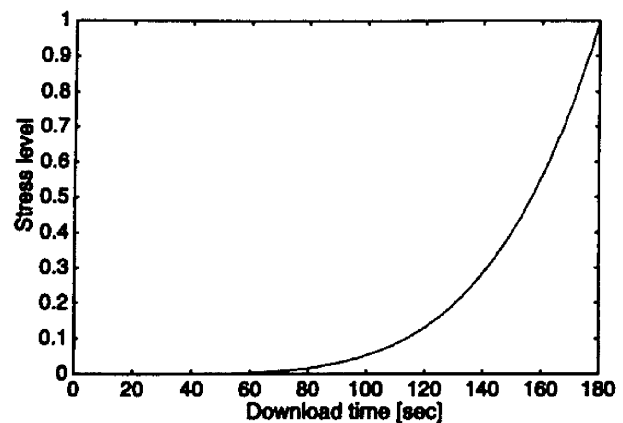


図 13 ファイルダウンロード時間とストレス量との関係 (ダウンロード許容時間 180 秒の場合)

Fig. 13 Relation between file download time and stress level when download give-up ratio is 180 s.

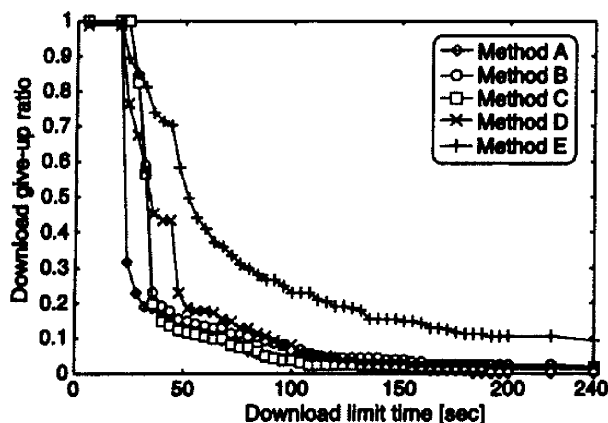


図 12 ダウンロード許容時間とダウンロード断念率との関係

Fig. 12 Download give-up ratio versus download limit time.

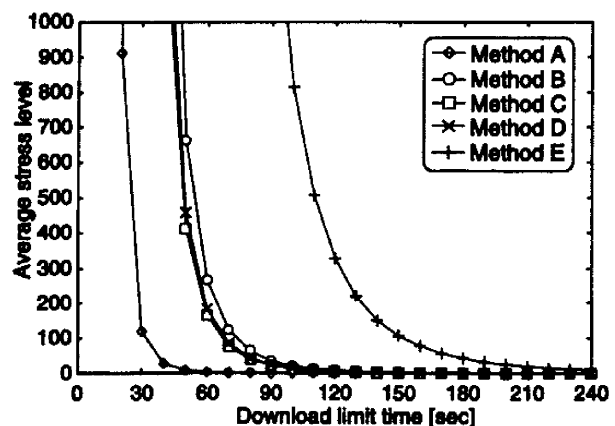


図 14 ダウンロード許容時間と平均ストレス量の関係

Fig. 14 Average stress level versus download limit time.

になるものと思われる。もちろんその妥当性は式 (1) の S と D の関係をアプリケーションの内容等も考慮し、適切にモデル化することが前提である。そのモデル化そのものは本論文の対象外であるが、人間工学的な知見の利用等も考えられよう。また、元来、人間の感覚には個人差やあいまい性が存在することを考慮すれば、式 (1) を厳密に求めることは必ずしも必要ではなく、人間の感覚特性を大雑把に表すことができればよい。それを用いて評価したストレス量に結果として大きな差があるのであれば、そのストレス量を基準にサーバ選択法を評価することは十分な意味があるといえるであろう。

以上に述べたように、ダウンロード時間、ダウンロード断念率、平均ストレス量の観点から、提案するサーバ選択法と従来法を比較し、これらの評価において提案手法の優位性が示された。実験条件が限られていることから、提案手法の優位性が一般的に証明されたわけではないが、サーバ選択法を検討するうえで、有効な選択肢の一つになりうるということが実証されたといえる。提案するサーバ選択法は、曜日、時間帯ごとに得られた QoS 履歴データに基づき、一定数の候補サーバを選択し、そのなかからリアルタイムの QoS 測定に基づきターゲットサーバを選択する。候補サーバ数 n が少なすぎると、候補サーバのなかに最良サーバが含まれない危険性が増える。 n が多すぎると、リアルタイムの QoS 測定によりユーザからみたダウンロード時間が増える。したがって、 n の値を適正に定めることが重要になる。今回の実験では、最適サーバが含まれる確率が高く、更にリアルタイム測定に要する時間も目的ファイルのダウンロード時間に比べて無視できるという条件から n の値を 3 に設定し、良い結果が得られた。この値は実際の利用環境に合わせて、適切に選択する必要があると考えられる。例えば、ミラーサーバの数がもっと多い条件では、 n の値を増やす案も考えられる。しかし、多くのミラーサーバがあるなかで統計的にベスト 3 に選ばれるようなサーバのすべてがファイルダウンロード時に良い状態ではないというのは考えにくい。このように考えると、 n の値は多くの環境のもとで、3 前後の比較的少ない値で十分のように思われる。具体的な検証は今後の課題である。

5. む す び

インターネット等を利用したコンテンツ配信ビジネスが登場し、同一のコンテンツを複数サイトに配置し

たサーバに分散し、サーバの一つを選択して、ユーザにコンテンツを配信することにより、ユーザ QoS やサービスの信頼性を改善する方式が注目されている。本論文では、アプリケーションとして、Web サイトから目的ファイルをダウンロードする場合を取り上げ、ダウンロード時間が最小となるサーバを選択する観点から、サーバ選択法を考察した。具体的には、クライアント側でサーバ選択を行う方法を対象として、QoS 統計データを利用した動的サーバ選択法を提案した。過去の QoS データを曜日別、時間帯別に集計することにより、従来のサーバ選択法より高度に過去の蓄積データを利用することの特徴としている。インターネットを利用した実験結果をもとに評価・検討を行った。まず、提案するサーバ選択法のパラメータの最適な値を検討した。次に、本選択法と既存のサーバ選択法の比較・評価を行った。評価尺度として、平均・最大ダウンロード時間といった客観評価指標に加えて、ユーザ感覚を反映したダウンロード断念率、ストレス量を用いた。この結果から、提案手法の優位性を示した。

本論文は提案するサーバ選択法と他のサーバ選択法の性能を極めて限られた実験条件のもとでファイルダウンロード時間という単一の尺度に基づき評価したものであり、一般的に提案手法の優位性を主張するものではない。提案手法で必要となる QoS 測定等のオーバーヘッドを含めたシステム全体としての総合的な評価は本論文の範囲外である。実際に提案手法の導入を検討するには、その環境条件に基づいて、効果を見極める必要がある。本論文は実証実験に基づき、サーバ選択法において有効な選択肢の一つを示し、その際の考え方や検討パラメータと手順を明らかにしたものである。

インターネット事業ではサービスレベルアグリーメントの導入が一般的になっており、QoS の定期的な測定はいずれにしろ不可欠になっている。したがって、QoS 測定の常時化、共通化等の枠組が実現できる環境では、提案手法における QoS 測定オーバーヘッドの問題は解消される可能性がある。インターネットバックボーンでは MPLS の導入等により、同一のサイトに対して、経路の選択が可能になる場合がある。提案手法はサーバ選択だけでなく、経路選択にも利用できる可能性があり今後の検討課題である。

文 献

- [1] S.G. Dykes, K.A. Robbins, and C.L. Jeffery, "An Empirical evaluation of client-side server selection algorithm," Proc. IEEE Infocom 2000, 2000.
- [2] V. Jacobson, "traceroute," <ftp://ftp.es.lbl.gov/traceroute.tar.Z>.
- [3] M.E. Crovella and R.L. Carter, "Dynamic server selection in the internet," Proc. of the Third IEEE Workshop on the Architecture and Implementation of High Perf. Comm. Subsystems (HPCS'95), pp.158-162, Aug. 1995.
- [4] R.L. Carter and M.E. Crovella, "Server selection using dynamic path characterization in wide-area networks," Proc. of IEEE Infocom'97, April 1997.
- [5] M. Sayal, Y. Breitbart, P. Scheuermann, and R. Vingralek, "Selection algorithms for replicated web servers," Performance Evaluation Rev., vol.26, no.3, pp.44-50, Dec. 1998.
- [6] 真壁 知, 太田耕平, 加藤 寧, G. Mansfield, 根元義章, "ネットワークの負荷変動を考慮した動的なミラーサーバ選択方式," 信学論 (B), vol.J84-B, no.3, pp.435-442, March 2001.
- [7] S. Seshan, M. Stemm, and R.H. Katz, "SPAND: Shared passive network performance discovery," Proc. the First USENIX Symp. on Internet Technologies and Systems (USITS'97), Dec. 1997.
- [8] 廣津登志夫, 高田敏弘, 栗原 聡, 菅原俊治, "転送履歴を利用した URL Resolver における最適経路選択," 第 1 回インターネットテクノロジーワークショップ (WIT'98), Aug. 1998.
- [9] H. Kamiya, K. Ohta, N. Kato, G. Mansfield, and Y. Nemoto, "An improved content search engine — Usage of network configuration information —," Proc. IEEE TENCN '98, vol.1, pp.21-24, New Delhi, India, Dec. 1998.
- [10] A. Tsuno, Y. Toyama, K. Mase, and N. Karasawa, "Internet QoS measurement for the server selection," Technical Report of IEICE, CQ2000-18, pp.19-24, 2000.
- [11] K. Mase, A. Tsuno, Y. Toyama, and N. Karasawa, "A Web server selection algorithm using QoS measurement," Proc. of the IEEE International Conference on Communications (ICC2001), June 2001.
- [12] 下川俊彦, 吉田紀彦, 牛島和夫, "ネームサーバを用いた柔軟な負荷分散," インターネットコンファレンス '99, 107-116, Dec. 1999.
- [13] H. Miura, M. Yamamoto, K. Nishimura, and H. Ikeda, "Server load balancing with network support: Active anycast," Proc. of the 2nd International Working Conference on Active Networks (IWAN2000), pp.371-384, Oct. 2000.
- [14] <ftp://gnjilux.cc.fer.hr/pub/unix/util/wget/>
- [15] <ftp://ftp.pasteur.fr/pub/Network/echoping/>

付 録

1. 使用ツールの概要

(1) wget の概要

wget は UNIX 上で使用できるダウンロード支援ツールであり、ホームページ自動巡回ツールでもある。具体的には、URL を指定すると、ターゲットサーバの指定したページ (またはファイル) をダウンロードし、その際のスループットを得ることができる。

wget を用いる利点としては、次のようなことがあげられる。

- リトライ (リジューム) 機能を備えているため、ダウンロードを確実に実行することができる。
- 処理が軽く測定用に適している (CPU やメモリをあまり使わない)。
- FTP だけではなく、HTTP もサポートしている。

(2) echoping の概要

echoping は、遠隔マシンのサーバが応答しているかどうかのテストを行うとともに、往復遅延時間 (RTT) を測定するためのツールである。具体的には、ターゲットとなるサーバに対して http プロトコルを用いて echo パケットを送信することにより、往復遅延時間を測定している。echoping を用いる利点としては、次のようなことがあげられる。

- echoping はユーザレベルで利用可能。
- ターゲットとなる多数のサーバに対して同時に echoping を実行し、それらの往復遅延時間を測定することができる。
- echoping では http プロトコルを使用するため、ネットワークレベルに加え TCP/IP や Web サーバ等の高位レイヤを含めた性能を把握することができる。
- Web サーバであれば ping, tcping が使えない場合でも echoping は通常使用可能。

2. wget と Web ブラウザのダウンロード時間の相関

wget によるファイルのダウンロード時間と Web ブラウザによるファイルのダウンロード時間の相関について検証するため、以下の実験を行った。デンマーク、エストニア、香港、スウェーデン、スロバキア、アメリカ (2 サイト) の計 7 サイトを設定し、各サイトから両方式により 1 Mbyte のファイルを 10 回 (1 日 5 回ずつ 2 日間実施) ダウンロードした。実験結果を付

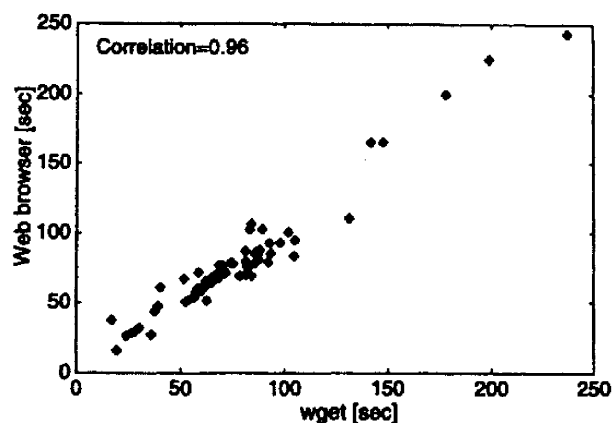


図 A-1 wget と Web ブラウザでのダウンロード時間の関係

Fig. A-1 Comparison of download time measured by wget and Web browser.

図 A-1 に示す。

実験の結果、wget によるファイルのダウンロード時間と Web ブラウザによるファイルのダウンロード時間の相関係数が 0.96 となり、相関がかなり高いことがわかる。

(平成 14 年 6 月 19 日受付, 9 月 27 日再受付)



津野 昭彦 (正員)

平 11 新潟大・工・情報卒, 平 13 同大学院博士前期課程了, 同年日本電信電話(株)入社, 現在, NTT ネットワークサービスシステム研究所ネットワークソフトウェアサービスプロジェクトに所属, e ビジネスサポートシステムの研究開発に従事。



間瀬 憲一 (正員:フェロー)

昭 45 早大・理工・電気通信卒, 昭 47 同大学院修士課程了, 同年電電公社(現, NTT)武蔵野電気通信研究所入所, 以来, 通信網構成法, ダイナミックルーティング, ネットワーク設計法, 通信品質, コンピュータネットワーク, マルチホップ無線ネットワーク等の研究に従事, 昭 53~54 米国コロンビア大客員研究員, 平 11 より新潟大・工・情報・科教授, 工博, 平 5 度本学会論文賞, 平 9 度第 13 回電気通信普及財団テレコムシステム技術賞受賞, 著書「マルチメディアネットワークとコミュニケーション品質」など。



栗林 孝行 (正員)

平 12 新潟大・工・情報卒, 平 14 同大学院博士前期課程了, 同年(株)NTT データ入社, 現在, ビジネス開発事業本部ビジネスプラットフォーム事業部に所属, オープンネットワークを利用したビル制御システムや, エネルギー管理システムなどの開

発に従事。