

ネットワーク遅延を考慮した WWWサーバベンチマークシステムの開発

西村 敦隆[†] 中山 貴夫[†] 中村 豊^{††} 砂原 秀樹[†]

[†] 奈良先端科学技術大学院大学
〒630-0101 奈良県生駒市高山町 8916-5
^{††} 大阪大学大学院
〒567-0047 大阪府茨木市美穂ヶ丘 5-1

E-mail: †{atsuta-n, takao-na}@is.aist-nara.ac.jp, ††yutaka-n@rd.center.osaka-u.ac.jp, †suna@wide.ad.jp

あらまし WWW によるさまざまなサービスが提供されるにつれ、大規模な WWW サーバの安定運用が重要となる。大規模な WWW サーバの安定運用にはサーバシステムの性能上限を知ることが必要であり、その調査にはベンチマークシステムを用いることが一般的である。しかし、既存のベンチマークシステムは LAN 環境でテストを行うため、現実の環境におけるネットワーク遅延が考慮されていない。そのため、ベンチマークシステムの結果が現実の環境を反映しないことがある。そこで、実際のネットワーク環境に近い実験環境を実現するため、複数のサーバ・クライアント間のコネクション毎に異なる遅延時間を挿入したベンチマークシステムを提案、実装する。そして、提案したシステムをパケットモニタで観測することにより、既存のベンチマークシステムとの比較を行う。

キーワード WWW サーバ, ベンチマーク, ネットワーク遅延, ENMA

Development of a WWW Server Benchmark System with Network Delay

Atsutaka NISHIMURA[†], Takao NAKAYAMA[†], Yutaka NAKAMURA^{††}, and Hideki SUNAHARA[†]

[†] Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara, 630-0101 Japan
^{††} Graduate School of Engineering Science, Osaka University
5-1 Mihogaoka, Ibaraki, Osaka, 567-0047 Japan

E-mail: †{atsuta-n, takao-na}@is.aist-nara.ac.jp, ††yutaka-n@rd.cenr.osaka-u.ac.jp, †suna@wide.ad.jp

Abstract Stable operation of a large-scale WWW server becomes important as providing various WWW services. Server administrators must know maximum performance of a server system for its stable operation, and they usually investigate server performance using a benchmark system. With a conventional benchmark system, the server system performance is measured in the LAN environment. The LAN environment, however, does not reflect the network delay that the real environment has. Therefore, building an experiment network more near actual environment, we design and implement a new benchmark system that employs method with inserting a different delay time every connection between the server and the client. We also compare our system and conventional system by observing using packet monitoring system.

Key words WWW server, benchmark, network delay, ENMA

1. はじめに

インターネットの急速な普及に伴い、WWW を用いたさまざまなサービスが提供されるようになってきている。たとえばオリンピックやワールドカップなどの中継、ホテルや航空券の予約や、ショッピングでの利用などである。これらのサービスが一般的になり、社会基盤の一つとして重要な役割が求められるようになるにつれ、ユーザは安定したサービス提供および品質の向上を要求するようになってきている。一方、サービス提供者は、ユーザの要求に応えるため、サービス品質の維持、向上に努めなければならない。このため、サーバ管理者はWWW サーバのキャパシティプランニングについて検討する必要がある。

キャパシティプランニングとは、サービスを安定してユーザに提供するために必要なハードウェアおよびソフトウェアの規模、性能を見積ることである。WWW サーバのキャパシティプランニングに対しては、「何人のユーザにサービスを提供するには、この程度のサーバが必要である」、「何人のユーザが存在する場合、リクエストはこの程度到着する」といった項目が挙げられる。これらの項目はWWW サーバの長期的計測をもとにある程度の予測が可能である。しかし、さらに具体的なキャパシティプランニングのためには、「このサーバシステムでは、1秒間あたり何リクエストまでを処理することができる」という項目を明確にする必要がある。つまり、サーバシステムの性能上限を知ることが重要となる。サーバシステムの上限を調査するためには、一般的にベンチマークシステムが用いられる。しかし、既存のベンチマークシステムでは、それらの示す結果が実際に運用されるサーバシステムに適用できないことがある。この原因として、

- ワークロードが一定でない
- CGI など、動的に変化するページの影響
- ネットワーク遅延、帯域の影響

などが挙げられる。我々は、これらの中でも特にネットワーク遅延に注目した。既存のベンチマークシステムは、LAN 環境でテストを行うことが一般的である。そのため、実際にサーバシステムが運用されるインターネット上での挙動とは異なり、ネットワーク遅延の影響がベンチマークシステムの出力に影響していると考えられる。つまり、実際に運用されているサーバとクライアントである各ユーザとのネットワーク的距離が一定でないため、LAN 環境における一定のネットワーク遅延でのベンチマーク結果は、

実際のサーバシステムの挙動を示すものにならないと考えられる。

そこで我々は、ネットワーク遅延を考慮したWWW サーバベンチマークシステムを開発する。まず、2章で既存のベンチマークシステムについて述べる。次に3章で実際のネットワーク遅延特性について、実際に運用されているサーバとLAN 環境での計測結果を示す。4章で我々が提案するネットワーク遅延を考慮したWWW サーバベンチマークシステムの設計、実装について述べる。5章では提案するベンチマークシステムと既存のベンチマークシステムとの比較を行い、ネットワーク遅延を考慮することによる結果の違いを示す。最後に6章でまとめと今後の課題を述べる。

2. 関連研究

これまでにWWW サーバのベンチマークシステムはいくつか開発されている。主なものを以下に挙げる。

WebStone

WebStone [1] は、要求するドキュメントサイズの分布、そのアクセス頻度やクライアント数などを設定することのできるフリーのベンチマークツールである。

SPECWeb

SPECWeb [2] は、商用のベンチマークツールであり、固定のプロファイリングでテストを行うことにより、統一的な指標を出すことができる。

httperf

httperf [3] は、WWW サーバの処理能力以上のHTTP/1.1 リクエストを生成し、その負荷を維持することができる。また、独自のワークロードジェネレータにより、さまざまな負荷を設定することができるベンチマークツールである。今回、我々の提案するベンチマークシステムとの比較で、httperfを用いた。

Web Polygraph

Web Polygraph [4] は、プロキシサーバ用のベンチマークシステムとして開発されたものであり、専用のクライアントとサーバを用いてその間のプロキシサーバの性能を測定するシステムである。

このようにさまざまな特徴を持つベンチマークシステムが開発されているが、これらのシステムは、一般的にLAN 環境でテストを行うため、その結果が実際に運用されるWWW サーバに適用できない可能性がある。

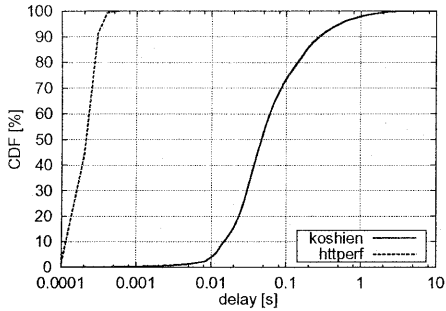


図1 遅延時間の累積頻度分布

3. 現実のネットワーク遅延

我々は、既存のWWWサーバベンチマークシステムが、現実の環境に適用できない主な原因としてネットワーク遅延に注目した。図1に示すのは、第83回全国高等学校野球選手権大会のインターネット中継に用いられたWWWサーバにおける遅延時間(実線)と、LAN環境でhttpperfを用いたWWWサーバのベンチマークテストの実行時における遅延時間(破線)の累積頻度分布である。この遅延時間は、ENMA [6]を用いて計測した。ENMAとは、パケットモニタを用いたWWWサーバ性能評価ツールである。ENMAでは、リアルタイムでHTTPリクエスト到着レート、HTTPセッション数、コネクション到着レート、トラヒック、パケット数などを観測することができる。また、TCPコネクションログやHTTPリクエストログを解析することにより、さまざまな性能指標を調査することが可能なツールである。ここでの遅延時間は、ENMAのコネクションログに記録される図2における(3) - (2)の時間としている。これはサーバ・クライアント間のラウンドトリップタイム(RTT)に相当する時間である。

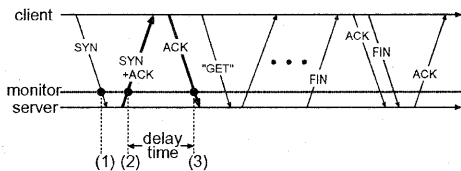


図2 遅延時間の定義

サーバ・クライアント間にルータおよびスイッチだけを配置したLAN環境では、1ms以下の遅延しか発生していない。それに対し、実際に運用されているWWWサーバにおける遅延時間は、ほとんどが

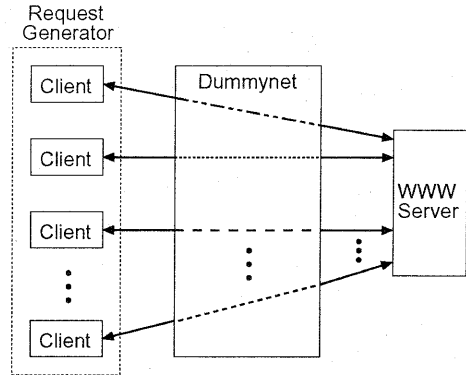


図3 提案するベンチマークシステムの構成

1ms以上であり、幅広く分布し、LAN環境での遅延時間の特性とは大きく異なることがわかる。我々は、この遅延時間の特性の違いがベンチマークの結果に影響を与えると考えた。

4. 提案するベンチマークシステム

前章で述べたように、既存のベンチマークシステムはLAN環境で実験を行うため、現実の環境でのWWWサーバおよびクライアントの挙動とは異なったものとなる。テスト用に構築したLAN環境では、現実のネットワーク環境をシミュレーションすることは困難である。

この問題を解決するため、現実に近いネットワーク環境を実現するWWWサーバベンチマークシステムを提案する。これは、複数の仮想クライアントとWWWサーバの間にコネクションを生成し、それぞれのコネクション毎に異なる遅延時間を挿入することにより、現実のネットワーク遅延をシミュレーションするシステムである。

提案システムは、複数のクライアントからHTTPリクエストを送信し、応答を受信するリクエストジェネレータおよび、ネットワーク遅延を実現するDummynet [5]から構成される。図3に提案するベンチマークシステムの構成を示す。

4.1 リクエストジェネレータ

DummynetをIPアドレスベースで遅延時間の設定を行う方法で用いるため、各コネクション毎に異なる遅延時間を実現するためには、複数のIPアドレスから、リクエストを送信しなければならない。そこで、alias機能で実現した複数のIPアドレスから、リクエストを送信できるように実装した。これにより、複数のIPアドレスからのHTTPリクエストの

それぞれに異なる遅延時間を設定することが可能となる。また、複数のクライアントをシミュレートするために多数の計算機を用いる必要もなくなる。

リクエストジェネレータの実行時には、WWWサーバへの各々のリクエストに対して、(1)コネクション要求開始時刻、(2)HTTPリクエスト送信時刻、(3)WWWサーバからの最初の応答受信時刻、(4)コネクション終了時刻、(5)最大同時コネクション数をアプリケーションレベルで記録し(図4参照)、実行終了時に統計情報を出力する。この(1)~(5)の時間を記録することにより、WWWサーバの性能指標の一つである応答時間、コネクション継続時間や同時コネクション数などを得ることができる。

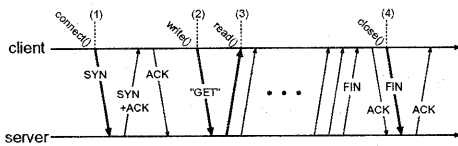


図4 リクエストジェネレータが記録する時間

4.2 ネットワーク遅延の実現

ネットワーク遅延はDummysnetを用いて実現する。Dummysnetは、ネットワークプロトコルのテストのために開発されたツールであり、帯域制御を実現するツールとしても用いられている。Dummysnetにより、キューと帯域幅の制限、パケットの遅延やパケットロスのシミュレーションができる。FreeBSDのカーネル内に実装されており、ルータもしくはブリッジとして用いることができる。

提案システムでは、リクエストジェネレータから送信するリクエストがすべてDummysnetを通過するようにDummysnetをルータとして用いる。そして、DummysnetのIPアドレス毎に異なる遅延を入れる機能により、コネクション毎に異なる遅延時間の挿入を実現した。

5. 実験

提案システムと既存のベンチマークシステムを用いて、ネットワーク遅延を考慮することによる違いを明らかにするための実験を行った。既存のベンチマークシステムとして、httpperfを用いた。

ネットワーク遅延を考慮することにより、1つのHTTPセッションの継続時間が長くなる。そのため、同じリクエストレートでWWWサーバにリクエストを送信すると、遅延を考慮しない場合に比べて、WWWサーバで処理中のリクエストが増加すると予

想される。そこで、本実験ではWWWサーバの処理中のコネクションの数に着目して計測を行った。計測には3章と同様、ENMAを用いた。

5.1 実験環境

実験を行ったネットワークの構成を図5に、使用した計算機の環境を表1に示す。WWWサーバにはApache[7](Ver.1.3.20)を用い、その設定はすべてデフォルトのままとした。また、WWWサーバに用いた計算機は、本実験で要求するリクエストレートに対応するため、最大ソケット数を増加させたカーネルを用いた。リクエストジェネレータに用いた計算機のカーネルはデフォルトのままで使用した。Dummysnetに用いた計算機は、遅延設定の精度を高めるためにHZ値をデフォルト値(100)の10倍にしたものを用いた。

Dummysnetはルータとして動作するよう設定し、リクエストジェネレータはDummysnetおよびスイッチを介して、WWWサーバと通信する。DummysnetおよびWWWサーバはスイッチに直接接続した。

また、スイッチのポートミラーの機能を用いて、WWWサーバが接続されているポートに流れるパケットをENMAで計測した。リアルタイムデータは、制御用ネットワークを介して、解析ホストに送信される。解析ホストは、WWWサーバの現在の状態を表示するグラフを生成する。

5.2 実験方法

表2に示すそれぞれの条件のもとで、クライアントから110リクエスト/秒のリクエストレートで合計100,000リクエスト(約15分間)を送信した。要求したファイルサイズは、約10KBである。遅延時間は、3章での実測結果を参考にして設定している。一定遅延の場合は実測結果の中央値である往復48msを、複数遅延の場合は累積頻度分布における遅延時間を100段階に分割し、100のクライアントにそれぞれに対応した遅延時間を設定した。そして、この時のWWWサーバをENMAで計測した。WWWサーバへ到着したHTTPリクエストをENMAにより観測した例を図6に示す。この図はENMAがリアルタイムに出力するグラフである。約110リクエスト/秒の一定の割合で、リクエストがWWWサーバへ到着していることがわかる。

5.3 結果

5.3.1 実験1(提案システム、遅延なし)

図7に、遅延設定なしで、IPアドレスの異なる100クライアントからリクエストを送信した時のENMAの出力する処理中のコネクション数のグラフを示す。

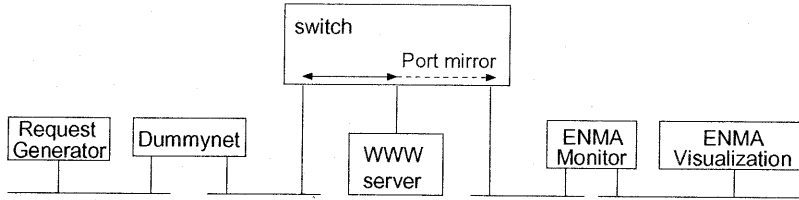


図5 実験環境

表1 実験に用いた計算機環境

	WWW server	Request Generator	Dummynet	ENMA
CPU	PentiumIII 800MHz	PentiumIII 800MHz	PentiumIII 1GHz × 2	PentiumIII 800MHz
Memory	512MB	512MB	512MB	512MB
OS	FreeBSD4.3-Release	FreeBSD4.3-Release	FreeBSD4.3-Release	FreeBSD4.3-Release
Kernel configuration	maxsocket=8232 maxuser=256	default maxuser=32	HZ=1000 maxuser=512	maxuser=1024

表2 実験条件

	ベンチマーク	遅延設定
実験1	提案システム	遅延なし
実験2	提案システム	一定遅延 (往復 48ms)
実験3	httperf	遅延なし
実験4	httperf	一定遅延 (往復 48ms)
実験5	提案システム	複数遅延

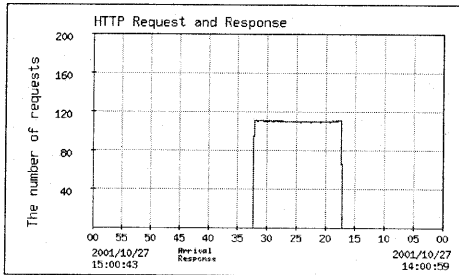


図6 WWWサーバへのリクエスト

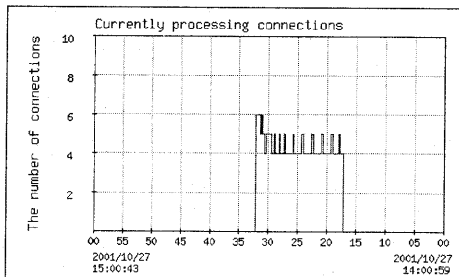


図7 提案システム, 遅延なし

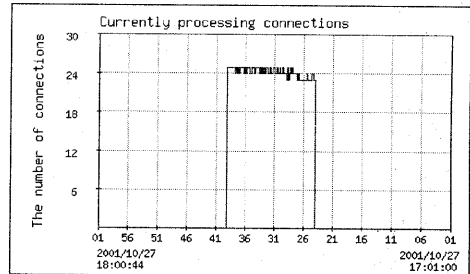


図8 提案システム, 一定遅延 (往復 48ms)

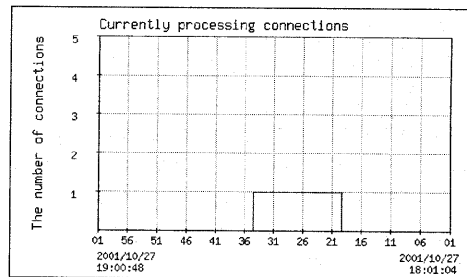


図9 httperf, 遅延なし

リクエストが要求されている時間帯で、4~6のコネクションを処理していることがわかる。

5.3.2 実験2 (提案システム, 一定遅延)

図8に、100のクライアントからの通信のすべて

に、片方向24ms (往復48ms)の一定の遅延時間を設定した場合の結果を示す。実験1と比較すると処理中のコネクションの数が大幅に増加し、一定に25程度のコネクションを処理していることがわかる。

5.3.3 実験3 (httperf, 遅延なし)

図9に、httperfを用いて、実験1と同じ条件でWWWサーバにリクエストを送信した時の処理中のコネクション数のグラフを示す。処理中のコネクション数は少なく、実験1と同様の傾向がみられる。

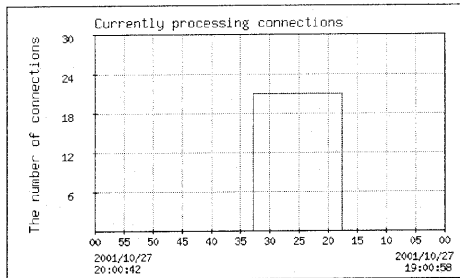


図 10 httpperf, 一定遅延 (往復 48ms)

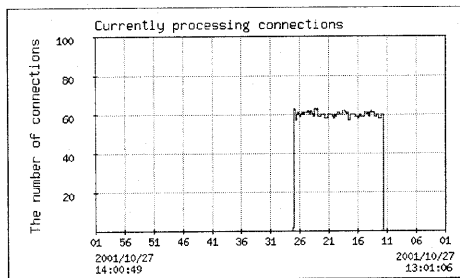


図 11 提案システム, 複数遅延

5.3.4 実験 4 (httpperf, 一定遅延)

次に, httpperf を用いて, 片方向 24ms (往復 48ms) の一定の遅延時間を挿入したときの結果を図 10 に示す. 実験 2 とほぼ同じような傾向があり 20 コネクション近くが一定に処理中となっている.

5.3.5 実験 5 (提案システム, 複数遅延)

図 11 に, 提案システムで, クライアント数を 100, 遅延時間を第 83 回全国高等学校野球選手権大会での実測値の割合と同等に設定した場合の結果を示す. 遅延時間を複数設定しているため, 処理中のコネクションが 60 程度で分布していることがわかる.

5.4 遅延時間

図 12 に, 実験 1~5 の WWW サーバ・クライアント間のネットワーク遅延の累積頻度分布を示す. 実験 1 および 3 では, 遅延時間を設定しなかったため, ルータとスイッチによる遅延だけとなり, 1ms 以下の遅延時間しかないことがわかる. 遅延時間を一定に設定した実験 2 および 4 では, 垂直な直線になっており, 遅延時間がすべてのリクエストにおいて設定した 48ms 付近であったことを示している.

実験 5 では, 実測データをもとに遅延時間を設定した. そのため, 実測データとほぼ同じ分布が得られ, 実環境の遅延時間を本システムで再現できていることがわかる.

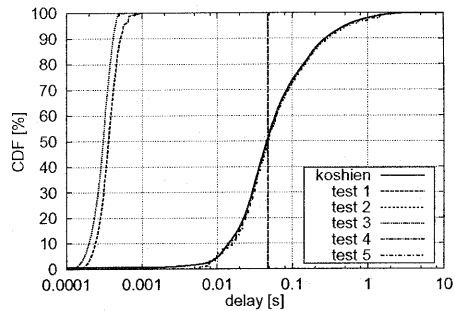


図 12 実験環境における遅延時間の累積頻度分布

5.5 考察

WWW サーバの性能指標として, これまで一般的には, リクエスト処理レートが用いられることが多かった. 本稿では, 実際に運用される WWW サーバの観測結果などから, ネットワーク遅延の影響をベンチマークシステムで評価することを検討した.

文献 [8] では, WWW サーバが飽和状態になると, WWW サーバ内部でコネクション処理が滞るため処理中のコネクションが増加し, WWW サーバのパフォーマンスに影響を与えることが報告されている. しかし, ネットワーク遅延がほとんどない LAN 環境では, WWW サーバがクライアントから次々に送られてくるリクエストの処理を完了するまでの時間が短くなり, WWW サーバ上で処理中のコネクションは増加しない (実験 1 および実験 3).

一方, 一定のネットワーク遅延を挿入した場合, リクエストの処理の完了までにかかる時間がネットワーク遅延の分だけ長くなる. そのため, ネットワーク遅延がほとんどない環境に比べると, WWW サーバ内で処理中のコネクションを多く保持することになるが, 遅延が一定であるため, 処理中のコネクション数もほぼ一定になる (実験 2 および実験 4).

実測結果に基づく遅延を挿入した実験 5 においては, 一定の遅延を挿入した実験 2 および実験 4 と比べて, 約 2 倍のコネクション数となっている. これは一定遅延の場合に比べて, 2 倍程度サーバに負荷を与えていることとなる. したがって, 現実の環境と LAN 環境では, 同程度のリクエストレートでも, サーバに対する負荷の与え方が異なることがわかる.

本稿では, 既存のベンチマークシステムで, ほとんど考慮されていなかった WWW サーバとクライアント間の遅延特性について検討した. 提案システムでは, 複数のコネクションに実環境に対応した複数

のネットワーク遅延時間を設定することにより、現実のネットワーク遅延特性をLAN環境で再現することが可能となった。それにより、より現実の環境に近い状態でのベンチマーク結果が得られるようになった。

6. おわりに

WWWサーバを安定運用させるためには、WWWサーバの性能限界を知ることが重要であり、その調査のためにベンチマークシステムが用いられてきた。既存のシステムではネットワーク遅延を考慮しないLAN環境でテストし、WWWサーバのリクエスト処理レートを性能指標として用いることが一般的であった。しかし、その結果は実際に運用するWWWサーバに適用できないことがあった。我々はその原因として、ネットワーク遅延が影響していると考えた。そのことを確認するために、WWWサーバとクライアント間に、複数のネットワーク遅延を設定することのできるWWWサーバベンチマークシステムのプロトタイプを実装し、ENMA(パケットモニタによるWWWサーバ観測システム)を用いて計測した。その結果、ネットワーク遅延を考慮した場合、ネットワーク遅延を考慮しない場合に比べて、WWWサーバの処理中のコネクション数が増加することが明らかになった。また、複数のコネクションに異なるネットワーク遅延を設定することによって、実環境のネットワーク遅延特性をLAN環境で再現できた。

今後検討する必要がある項目としては、

- 遅延特性のプロファイリング
- ネットワーク遅延以外の要素の影響の検討
- 処理中のコネクション数以外の性能指標の検討
- サーバの性能上限を定義する手法の検討

などが考えられる。

本稿では、現実のネットワーク遅延の一例として図1に示した実測結果をもとに遅延時間を設定した。しかし、より現実に近いネットワーク環境を実現するためには、ネットワーク遅延に関するこれまでの様々な研究[9][10]や、さらに多くの実測データの解析結果をもとにして、遅延設定のプロファイリングに関する検討が必要である。

また、今回は考慮しなかったネットワーク遅延以外の要素、たとえば、パケットロスや帯域幅の影響などについて検討することも今後の課題である。WWWサーバの性能指標に関しては、本稿では処理中のコネクション数に注目したが、その他の性能指標についても検討する必要があると考えられる。

謝 辞

本研究において、サーバの運営・ログの収集に御協力頂きました朝日放送、NTTスマートコネクト、CKPの皆様へ深く感謝致します。

文 献

- [1] Trent, G., and Sake, M. "WebSTONE: The First Generation in HTTP Server Benchmarking," February 1995.
- [2] Standard Performance Evaluation Corporation, "SPECweb 99 benchmark, performance results," <http://www.spec.org/osg/web99>, June 2000.
- [3] D. Mosberger and T. Jin, "httperf: A tool for measuring web server performance," in WISP, pp. 59-67, Madison, WI, June 1998, ACM.
- [4] Web Polygraph. <http://www.web-polygraph.org/>.
- [5] Rizzo, L. "Dummynet: a simple approach to the evaluation of network protocols" ACM Computer Communication Review 27, 1, January 1997.
- [6] 中村豊, 知念賢一, 砂原秀樹, 山口英. "ENMA:パケットモニタによるWWWサーバの性能計測システムの設計と実装" 電子情報通信学会論文誌, Vol.J83-D-I, No.3, pp.329-338, 3月1999.
- [7] Apache HTTP Server Project. <http://www.apache.org/>.
- [8] Yutaka Nakamura, Ken-ichi Chinen, Suguru Yamaguchi, Hideki Sunahara. "An Analysis of WWW Server Status by Packet Monitoring", IEICE TRANSACTIONS on Information and Systems, Vol.E83-D, No.5, May 2000.
- [9] V. Paxson, "End-to-end Internet packet dynamics," in Proc. ACM SIGCOMM, pp. 139-152, September 1997.
- [10] A. Fei, G. Pei, R. Liu, and L. Zhang, "Measurements on delay and hop-count of the Internet," in IEEE GLOBECOM'98 - Internet Mini-Conference, 1998.