命令トレーサ:TURUNGを用いたWWWサーバの性能分析

小宮山 彰一郎† 内藤 義郎† 吉澤 康文‡
東京農工大学大学院工学研究科† 東京農工大学工学部‡

1 はじめに

WWW サーバを始めとする，ネットワークサービスを提供するサーバマシンは，クライアントからの要求の受付に対して応答をスムーズに処理することが要求される．この要求を満たすためには，サーバマシンの処理速度を向上させる方法と，サーバマシンのプログラムオーバヘッドを軽減させる方法がある．我々は，特に後者の方法に注目し，サーバプログラムの性能を分析し，向上させることを目的としている．ネットワークサーバのように，オペレーティングシステムへの依存度が高いプログラムの性能を向上させるためには，トランザクションの使用する資源量を定量的に測定しなければならない．そこで我々は，命令単位の動作を記録するトレーサ:TURUNG と解析ツール群:TURUNG Tools を開発した．本論文では TURUNG と TURUNG Tools を用い，WWW サーバの性能を定量的に分析した結果を報告する．分析したトランザクションは，HTML ファイルの転送と CGI プログラム実行であり，それぞれのプログラムのステップ数，関数プロファイリング，使用メモリページ数などから，処理のボトルネックの特定を行い，その改善方法についての考察を行う．

2 命令トレーサ:TURUNG

OS の内部処理を把握したり，評価することは，OS の目的と構成上困難である．この理由として，OS は，仮想的に高機能な計算機を作り出すことを目的としていることが挙げられる．また，OS が独立して計算機システムを構成しているのではなく，アプリケーションと密接な関係を持って成り立っているためである．さらに，割り込みなどの非同期イベントが発生するため，プログラムソースを参照するだけでは制御の流れを追跡できない場合がある．そこで，上記を解決するために，我々は評価と分析に適した命令トレーサ:TURUNG を開発した．以下に TURUNG の概要を述べる．

2.1 TURUNGの概要

TURUNG は，ソフトウェアで実現されたカーネルレベルの命令トレーサである．

Performance Analysis of World Wide Web Server using Instruction Tracer "TURUNG"
†Syoichiro KOMIYAMA,Yoshiro NAITO:Graduate School of Engineering,Tokyo University of Agriculture and Technology
‡Yasufumi YOSHIZAWA:Faculty of Engineering, Tokyo University of Agriculture and Technology

TURUNG は，プログラムの実行を機械語レベルでトレースする．記録する情報を次に示す．
・機械語命令列
・参照したメモリアドレス
・割込み
・コンテキストスイッチ

TURUNG では，性能評価を主眼においているため，記録対象を上記の 4 種に限定し，性能評価を行うために必要十分なデータを取得しながら，小さなオーバヘッドで動作することが可能である．

2.2 TURUNGの構成

TURUNG は，Intel i386 アーキテクチャ上で動作する Linux カーネルの Loaderble Kernel Module として実装された．TURUNG の構成を図 1 に示す．

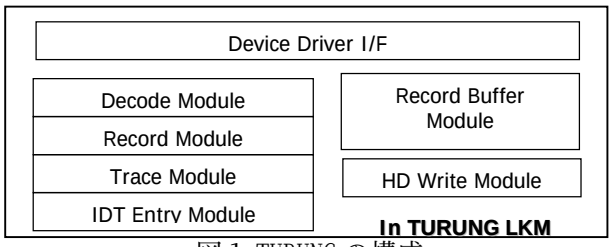

図 1 TURUNG の構成

2.3 TURUNG Tools

トレースデータを解析するために提供されているツールは，現在，レコード表示ツール，関数遷移表示ツール，プロファイリングツール（関数，メモリ）がある．

これらのツールは，TURUNG の出力したデータを統一的にアクセスするためのライブラリ群（TURUNG Library）を用い，アプリケーションとして実現される．

3 性能分析

TURUNG を用いて www サーバをトレースした．トレースは，次に示す 2 つの場合について行った．
・HTML ファイルを転送した場合
・CGI プログラムを動作させた場合

以下，それぞれの場合における分析結果を述べる．サーバは，CPU が Intel Celeron 433MHz，メモリを 128M 搭載したマシンを用い，OS は Linux(Ver.2.4.18) カーネル，www サーバには apache-1.3.26 を用いた．

3.1 HTML ファイルの転送

3.1.1 分析対象

クライアントからのリクエストを処理する部分をトレースした．また，転送ファイルサイズを変え，その処理量の変化を分析した．

3.1.2 分析結果

転送ファイルサイズが 6,040B，41,518B の場合の，httpd，共有ライブラリ，カーネルの各部分におけるダイナミックステップ数を，表 1 に示す．また，httpd，共有ライブラリ，カーネルにおいて，命令実行で参照されたページの局所性を表すグラフを図 2 に，データとして参照されたページの局所性を表すグラフを図 3 に，それぞれの統計情報を表 2 に示す．また，処理ステップの多かった関数を表 3 に示す．

表 1　HTML ファイルの転送におけるステップ数

File Size	httpd	Lib	Kernel	Total
6,040	40,778 (24.2%)	32,794 (19.5%)	94,856 (56.3%)	168,428
41,518	41,485 (12.6%)	32,910 (9.8%)	254,774 (77.4%)	329,169

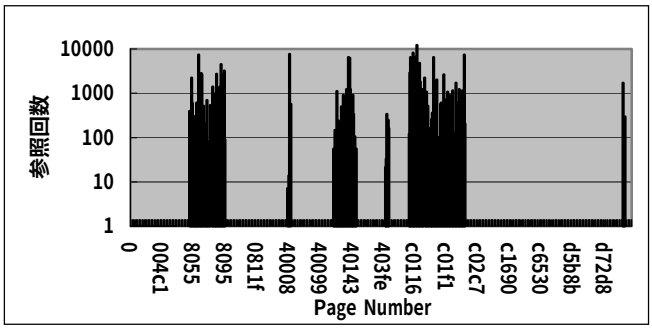

図 2　命令実行におけるページ参照回数

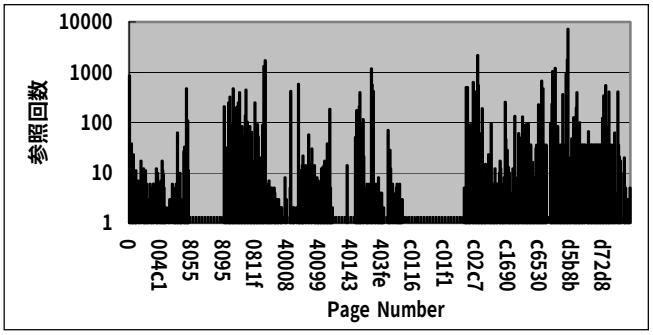

図 3　データ参照におけるページ参照回数

表 2　HTML の転送におけるページプロファイリング

	使用ページ数	平均参照回数
命令	155	1,096
データ	459	92

表 3　HTML ファイルの転送における関数プロファイリング

Name	Min	Max	Ave	Times	Total
csum_partial_ copy_generic	114	1,304	1,216	31	37,699
ap_vformatter	133	1,649	528	13	6,872
tcp_sendmsg	1,529	4,987	3,258	2	6,516

3.2 CGI プログラムの実行

3.2.1　分析対象

HTML ファイルの転送と同じく，クライアントからのリクエストを処理する部分をトレースした．実行した CGI プログラムは perl スクリプトである.apache では，CGI プログラムを実行する場合には新たに子プロセスを生成し，そのプロセスが CGI プログラムを解釈，実行する．

3.2.2 分析結果

CGI プログラムを解釈,実行するプロセスのユーザ，共有ライブラリ，カーネル，およびリクエストを処理する親プロセスまで含めたトータルのダイナミックステップ数を表 4 に示す．

表 4　CGI プログラムの実行におけるステップ数

User	Lib	Kernel	CGI total	Total
140,608 (8.2%)	592,780 (34.5%)	738,105 (43.0%)	1,471,493 (85.6%)	1,718,154

4　考察

表 1 より，HTML ファイルの転送を行う場合，ファイルサイズを大きくしても，apache の処理ステップはほぼ一定であることがわかる．表 3 より，カーネルの処理では TCP のチェックサム計算ルーチンがもっとも重いことがわかる．したがって，HTML ファイルの転送を行う場合の処理効率を向上させるためには，プロトコルスタックの改良が有効であると言える．

表 4 より，CGI プログラムを実行した場合は，CGI インタプリタの処理ステップ数が全体の処理の 85.6%を占めることがわかる．これは，perl スクリプトを解釈，実行するためのコストが大きいことを示している．

図 2，図 3 および表 2 より，データ参照されるページ数は命令参照されるページ数の約 3 倍多いことがわかる．また，データが配置されたページは，命令が配置されたページよりも平均参照回数が約 1/12 となっている．従って，命令参照の局所性よりも，データ参照の局所性の方が低い．これより，性能の向上には，データ参照の局所性を向上させることが重要であることがわかる．

5　おわりに

本論文では，TURUNG の概要を述べ，さらに，TURUNG を用い www サーバの性能を分析した．HTML ファイルを転送した場合と，CGI プログラムを実行した場合の性能を評価し，ボトルネックとなっている処理を示した．また，メモリ参照の局所性についても分析し，データ参照の局所性を高めることが重要であることを示した．

参考文献

[1]毛利公一，森本洋行，池谷敬之，吉澤康文，”命令トレーサを用いたネットワークサーバプログラムの性能評価”，情報処理学会研究報告 2000-OS-84, pp.165-171，2000.