

金澤 美晴 洪沢 進†
茨城大学工学部情報工学科‡

1 はじめに

現在 Web などの普及に伴い、計算機のネットワークがますます広がってきている。それに伴って増え続けるクライアントの要求のにより、サーバの負担は大きくなり必要情報の取得が悪化している。複数の計算機がネットワークでつながっている分散計算機環境(以下分散環境と呼ぶ)ではサーバの負担の分散のため、コンテンツの複製配置による負荷分散の方法が多く研究されている [1, 2]。ミラーサーバの設置と異なり、コンテンツ毎に複製を行うので特にハードウェア資源を無駄に費やすことが無く有効的であるとされている。

本報告では、サーバの負荷要因の一つである CGI プログラム処理を対象とし、プログラム処理による応答時間の増加を改善するために、プロセスの処理時間やサーバ、複製配置先計算機の負荷状況を考慮した複製配置方法を提案し、応答時間の比較を行う。実験の結果、配置を行った場合に応答時間の改善が見られたが、重視する計算機の負荷情報により応答時間は異なることが明らかとなり、CGI プログラムの特徴をとらえることも必要になると思われる。

2 CGI 処理とその問題点

CGI 処理は、CGI(Common Gateway Interface) と呼ばれるインターフェイスを使用して外部プログラムを起動する方法が利用されている。外部プログラムは Web サーバとは異なるプロセスとして実行されるため、一般に負荷の大きいプロセス処理を伴う。要求がある毎にプロセスが起動されるため、複数の Web クライアントからのアクセスが集中してしまう場合、この負荷のためサーバの処理が極端に重くなってしまうという点がある。

*Dynamic Relocation Methods of CGI Programs Considering Computer System Loads

†Miharu Kanazawa, Susumu Shibusawa

‡Department of Computer and Information Sciences, Faculty of Engineering, Ibaraki University

3 計算機負荷を考慮した動的複製配置方式

分散環境において、要求された CGI プログラムを実行することによるサーバの負担を他の計算機に動的にプログラムの複製を配置し、処理させることにより応答時間の短縮を図る。分散環境にあるメインマシン(サーバ)以外の計算機(以下、予備計算機と呼ぶ)に、複製配置を行うに当たって、プログラムの配置先を予備計算機群の中から無闇に選択していたのでは、負荷が高くなっていった計算機に配置してしまった場合、応答時間の改善を得ることができないのは明らかである。そこで、各計算機の負荷状況を取得し、動的に配置先計算機を決定していく方法をとる。また、継続的に情報を得ることにより、過去の負荷の統計的値を利用することを考える。

各計算機のシステム状況は、それぞれに対して uptime コールと ps -au コールを実行し、その情報を利用する。uptime コールは 1 分前、5 分前、15 分前から現在までのロードアベレージ(システムの負荷)を取得できる。ps -au コールは全ユーザが実行しているプロセスの詳細情報を得ることができ、計算機上で実行されている各プロセス毎の CPU の利用率(%)と、メモリの利用率(%)を得ることができる。

3.1 複製配置アルゴリズム

CGI プログラム処理によるシステム負荷増加のため、悪化したサーバの応答時間を改善するために、step1 から step4 の作業を行う。

step1 一定期間毎にサーバ計算機と予備計算機のシステム状況を得る。各計算機で uptime と ps -au コマンドを実行し、その都度さらに、uptime は各時間毎に平均(以下システム負荷平均と呼ぶ)を計算し、ps -au は CPU とメモリの使用率の合計の平均(それぞれ以下 CPU 利用率平均、メモリ利用率平均と呼ぶ)を計算する。

step2 クライアントからの要求があった時、サーバの負荷平均値が閾値を越えていれば step3, step4 の処理を行う。越えていなければサーバで処理を行う。

step3 平均値が低い程、計算機負荷は低いので、システム負荷平均とCPU利用率平均とメモリ利用率平均の情報から、各々最小値を持っている予備計算機を挙げ、最も多く挙げられた計算機を選択する。各最小値に挙げられた計算機が全て異なる場合、各平均値の優先度を元に決定する。

step4 選択された計算機にサーバにあるCGIプログラムの複製を作り、その計算機上で処理を行わせる。複製されたプログラムは消去を行わず、次回要求が来た時に再利用をする。

4 要求の応答時間

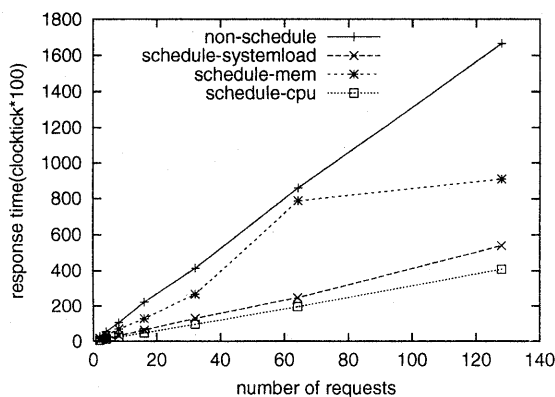


図 1: 要求回数と応答時間の変化

本研究には、複製配置にメインサーバを含め5台の計算機を利用し、等差数列の積を求めるCGIプログラムを作成し、複製を行わない場合と、行った場合を実験した。図1に、step2の開始からstep4の終了までの時間を応答時間として計測し、要求間隔は3秒、平均値更新間隔は5秒とし、要求回数を2から128に変化させ、それぞれ10回の平均を示す。図1において、スケジュールを行わない場合(non-schedule)以外のグラフは、step3の優先度を变化させた場合である。これらはそれぞれ、

- システム負荷の低い計算機を優先して処理を行わせる方法を取った場合 (schedule-systemload)
- メモリ利用率の低い計算機を優先して処理を行わせる方法を取った場合 (schedule-mem)
- CPU利用率の低い計算機を優先して処理を行わせる方法を取った場合 (schedule-cpu)

の応答時間を示している。横軸は要求回数であり、縦軸は要求回数毎の応答時間である。

要求数が増加するにつれて、比例して応答時間も増加していくが、複製配置を行わない場合と行った場合とでは、システム負荷平均を優先させた場合と、CPU利用率平均を優先させた場合で、複製配置を行った方が約3倍速いことがわかる。メモリ利用率平均を優先させた場合において、他の二つと比べて応答時間が大きかったのは、今回用いたCGIプログラムの性質によると考えられる。3つの方法のうち、どの方法が良いかについては、実行するCGIプログラムの特徴による。

5 終りに

本報告では様々な計算機負荷情報を考慮した複製配置方法について、応答時間の違いを考察した。どの情報を用いても応答時間の短縮につながったが、特にCPUの利用率を考慮した場合において最も改善を図ることができた。

本実験において、プログラムの処理のほとんどが予備計算機上で行われた。それはCGIプログラムの実行のメインとなる計算機で、複製配置のスケジューラの実行も行ったためと考えられる。結果、応答時間の縮小は図ることができたが、これはCGIプログラムをこのサーバに置く必要がないことを意味する。今後、スケジューラや負荷情報の取得を別の計算機で行う方法も考察が必要である。

各計算機の負荷情報を得る時、情報を収集し平均値を計算するまでのオーバーヘッドも大きなものであり、取得方法についても考察を必要とする。複製配置の実行契機となるメインサーバ負荷の閾値の決定方法や、事前に、実行されるCGIプログラムの性質を知ることなどによって、どの情報に優先度を高く持たせるかを決定できる機能も考慮する。

謝辞 ご討論頂いた茨城大学工学部情報工学科渋沢研究室の皆様へ感謝致します。

参考文献

- [1] 渡辺尚, 森章文, 山本幸寿, “分散オブジェクト指向システムにおける動的複製配置プロトコル”, 電子情報通信学会論文誌 D-1, Vol.J83-D-1, No.8, pp.862-872, 2000年8月.
- [2] 高倉健, “アクセス負荷の予測を用いた動的コンテンツ再配置方式”, 情報処理学会論文誌 Vol.40, No.3, pp.1285-1292, Mar. 1999.