

# 通知を行う負荷管理システム

多川 悠太<sup>1,a)</sup> 串田 高幸<sup>1</sup>

**概要：**現在の世の中、周りにはゲーム機、パーソナルコンピュータ、スマートフォン、沢山の電子機器が溢れている。中でもスマートフォン、パーソナルコンピュータは、株取引や天気予報、ネットニュース、オンラインゲームでネットワークが使用されている。これらのものに対しては特にレスポンスの速さが求められる。このレポートではレスポンスの速さに関わる負荷について通知するシステムを扱う。

## 1. はじめに

今日、ネットワークを利用するシステム、機器は世界で広く使用、利用されている。例として、パーソナルコンピュータシステムのアップデート、Web サイトの運営、パーソナルコンピュータでのサイトの閲覧、スマートフォン上で動作するゲームアプリケーションが挙げられる。ネットワークを利用しているシステム、機器にとってサービス利用時にかかる負荷は重要な課題である。ウェブブラウザである Firefox, Googlechrome, InternetExploer では外部からのウイルス攻撃、サイト利用者の通信状況、サイトへのアクセスの集中などの負荷がスムーズなアクセスを妨げる要因である。このような負荷に関する問題を解決する手段としてキャッシュ分散処理や、ロードバランサーの実装、DNS サーバーを利用した負荷分散があり、解決するための提案、実装がされている。このような負荷分散システムがある一方、さくらインターネットを始め独自の負荷分散技術を提供する会社、企業が増えている。本論文では、2 章で負荷分散を利用した既存技術について紹介を行い、3 章以降で今回の負荷率に応じて通知を行うシステムについて報告する。

## 2. 関連研究

Randles らはウェブ配信サービスは予測されていたほど利用されていないことを述べた [1]。この理由はウェブ配信サービスの複雑性と技術的な専門知識を必要とすることとサービスを保守、管理の困難により発生している。これにより特定のサーバーを集中管理することが難しくなったとしサービス、ハードウェア提供のサービス、IoT(モノのインターネット) に対して先に挙げた項目を達成するためには

分散ガバナンスが必要不可欠であると述べ、負荷分散をより簡単に実現するために実現可能な三つの方法を検討している。1 つ目は自然発生現象であるミツバチの採餌に直接基づいており、大規模な SOA (企業の業務システムなどの設計様式の一つで、システム全体を利用者側から見たソフトウェアの機能単位である「サービス」(service) の組み合わせによって構築すること) でのサーバ割り当てを利用して実験効果が最大になるようにしている。最適なりソース割り当ての為にアプリケーション層で自己組織化、2 つ目はジョブをノードと置き、閾値よりも短い場合ランダムな近隣にジョブが割り当てられるというものである、3 つ目はジョブを負荷の少ないノードに割り振るためにシステムの再配線 (アクティブクラスタリング) を使用して負荷分散のパフォーマンスを向上させるものである。クラスタリング (与えられたデータを自動的に分類する手法

Mohamed らはキャッシュを分散させることにより負荷分散を行う方法を提案している [2]。これを分散クラウドキャッシュと命名しているが、分散クラウドキャッシュは全ての作業を単一のキャッシュに依存するのではなく、キャッシュ及びダウンロードプロセスを複数のノードに分散することにより、ダウンロードのパフォーマンスが向上し、クラウドデータセンターのオーバーヘッドが削減されると述べている。

ParamitaParida らの論文では電力消費量、実行待機時間、マシンの障害などの過負荷によるシステム障害の問題を解決するためには負荷分散が必要だと述べ、クラウド内での負荷分散アルゴリズムとその効果についての解説をしている [3]。そのアルゴリズムを考えるにあたってコスト、複雑性、速度、システムの移植性、セキュリティのような要因を顧慮する必要性があり、スケーラブルで多くの研究者が従うクラウド環境の単一ホストのシステムアーキテクチャの一つが図 1 : クラウド内の単一ホストアーキテクチャであ

<sup>1</sup> 東京工科大学コンピュータサイエンス学部  
CDSL, TUT, Hachioji, Tokyo 101-0062, Japan  
<sup>a)</sup> C0117191

る。  
Chen らは人気を集めている Web サイトはユーザーのアク

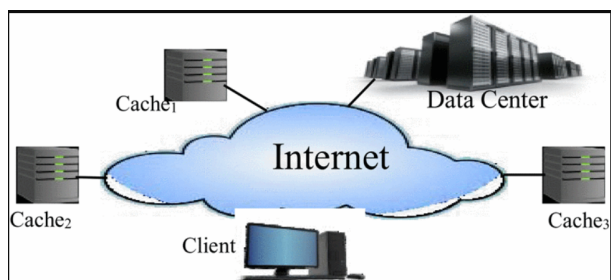


図 1 クラウド内の単一ホストアーキテクチャ

セスの数に比例して負荷が掛かり応答時間を増幅させていることを問題として挙げ、分散された複数のサーバーノードで構成される Web システムアーキテクチャに焦点を当て、クライアント要求をノード間で分散する 1 つ以上のメカニズムを検証している [4].

Philip らはクラウドコンピューティングの負荷分散モデルについて比較研究をしている [5]. クラウドコンピューティング環境での負荷分散のためのゲーム理論モデルとアルゴリズムに関する研究がほとんどないことに気付きゲーム理論を使用したクラウドパーティションの概念に基づいたパブリッククラウドの負荷分散モデルを検証している。

### 3. 提案

今回のシステムは外部のブラウザ上で動作する監視サービスを使用しないことで、CentOS 以外の外部サービス、ソフトウェアを使用するために導入しなくてよい。具体的には、監視サービスを提供しているサーバがダウンしてしまう災害時にも使用することが出来、導入コストがかからないという点である。勿論、この提案の類似のソフトウェアや、今回のような目的で作られているソフトウェアは存在するが、今回の CentOS を導入するだけで負荷率を通知することができるソフトウェアは存在しないと考えたため、今回の提案を行う。

図 2 が今回の提案のシステム構成図である。今回はサーバがダウンしても利用できるというメリットを達成するため、ネットワークに繋がっていないローカル環境でも利用出来るようにする。そのため、ホストユーザー上で VMware(CentOS) を利用することを前提に提案を行う。また、簡易的に素早く負荷をかけられるようにするため疑似的な負荷をかけるシェルスクリプトを使用する。疑似負荷がかかった時に予め定めた閾値を超えた負荷率の場合のみ監視ソフトウェアを通して、通常時の平均負荷率よりも負荷が大きくなっていることを Gmail や slack といったメールサービスシステムやコミュニケーションツールに対して通知を行う通知ソフトウェアを作成する。補足として疑似負荷をかけるシェルスクリプトと監視ソフトウェアは同じ

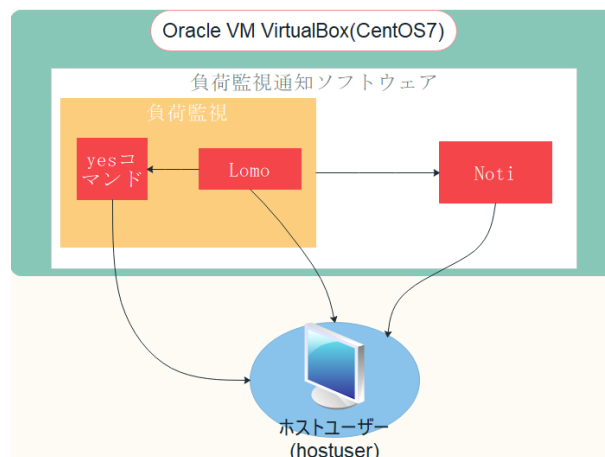


図 2 システム構成図

システムにまとめられているため「負荷監視」ソフトウェアに入っていて、負荷監視と通知ソフトウェアを合わせて今回提案する負荷監視通知システムとなっている。

### 4. 実装

今回のシステムは VM にかかる負荷を監視して予め定めた負荷率で Gmail や slack に通知するソフトウェアである。まず今回の実装環境は Oracle VM Virtual box であり、VM は CentOS を使用する。次に今回、負荷監視に使用するシェルスクリプトを実装する。シェルスクリプトとはアプリケーションソフトウェアを作成するための簡易的なプログラミング言語のことであるが、これに通知ソフトウェア Noni を組み合わせてホストユーザーに対して閾値を超えた負荷を通知する監視ソフトウェアである Lomo を実装する。Lomo では vmstat コマンドやコマンドというシェルコマンドを使用しプログラムを作成したコードを利用している。

- **yes コマンド**・・・疑似負荷をかけるシェルスクリプト。yes コマンドを使用して一時的な負荷を段階的にかけていく
- **Lomo**・・・**Load monitoring** から取った。これは疑似負荷スクリプトによって閾値を超える負荷を監視し、閾値を超えた負荷が存在した場合、通知ソフトウェアに超えた負荷の「負荷が発生した時間・負荷の大きさ」を情報として渡す、閾値として CPU のアイドル値を参照するようにした
- **Noni**・・・**Notification** から取った。Lomo から渡された情報を予めコード中に登録してあるユーザー（ここでは **hostuser** を想定）の Gmail, slack に対して通知を行う。なお、今回の論文では未実装である

負荷監視ソフトウェアのプログラムのコードの一部が図 3 である。図 3 の vmstat コマンドをシェルスクリプトとして組み込んでおくことで一秒毎に三回、つまり三秒間の負荷観測を行う。他のプログラムコードはホストユーザーの CPU 負荷率をログ (log) としてファイルを出力する。その

## vmstat 1 3 > \$VMSTAT\_LOG

図 3 負荷監視ソフトウェアのプログラムコードの一部

出力されたログファイルを監視し、負荷率が閾値を超えた時、Gmail,slack に「あなたの利用している VM に急激な負荷が掛かっている」という一種の危険通知アラートシステムとして通知を行うものである。なお今回は監視し通知するプログラムは組み込んでいない。

## 5. 評価と議論

前章の実装である負荷監視についての検証、その評価を行った。その結果が図 4,5,6 である。図 4 は閾値 60 で yes コマンドを一回も実行していない状態、一回実行した状態、二回実行した状態毎のデータで,Lomo で予め決めた閾値を超えた時の CPU の負荷状態を vmstat を使いファイルに出力したものである。図 5 は閾値 70 を超えた場合だけ vmstat の結果を保存するという条件にし、それ以外の方法や条件は同じにした。前提条件である閾値を閾値 60, 閾値 70 と変化させた場合、どのような変化があるのかという検証を行うためである。また 60,70 での負荷のグラフを比較した後、図 6 で一番負荷が高い状態の二つを比べている。この結

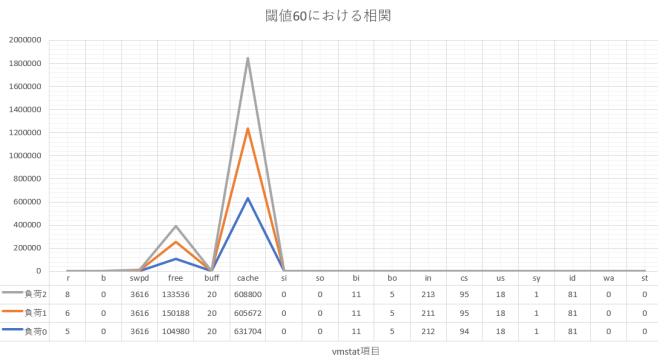


図 4 閾値が 60 の時の負荷相関

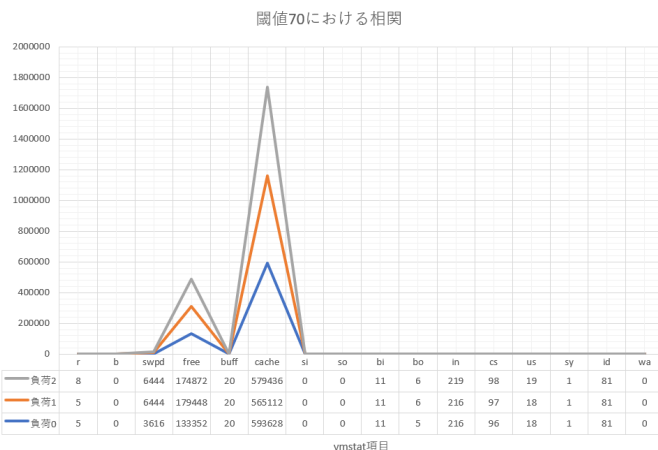


図 5 閾値が 70 の時の負荷相関

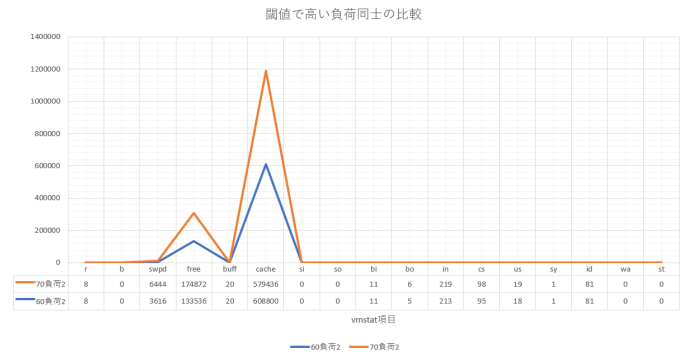


図 6 一番負荷が高い状態の比較

果から今回実装したシステムは閾値に基づいた処理を正確に行っているということが分かった。また閾値を 60,70 と比較がしやすいように変化させた。その結果、負荷が高い状態である閾値 60 の方が閾値 70 の状態と比べて cache の値が全体的に高く free の値が小さいため閾値に応じた判定が出来ている。上記の結果から今回の環境下での負荷は比例するものだということが分かった。本論文での負荷監視システムでは閾値を 60 として負荷の監視を行い、閾値を超えた場合という条件でログ出力を行っているが、閾値を 60,70 にした場合とどの様な理論に基づいて閾値を取るかという点に関して一考の余地がある。加えて、閾値を超えたものを通知するソフトウェアが必要であることは今後の課題である。

## 6. おわりに

今回の負荷通知システムは負荷を監視するために外部の監視ソフトウェアをコストをかけ導入しないで済むという点に重きを置いたソフトウェアである。外部の監視ソフトウェアのようにカスタマイズ出来る多機能性は無いが、必要最低限のシステムは揃っていて外部ソフトウェアを導入するためのコストを払わなくてよく、ローカル環境で負荷監視を行えるシステムである。Noni にあたるシステムの構築,Lomo が閾値を超えた時に Noni にアラートをさせるようにするための連携についての改良の余地はある。

## 参考文献

- [1] Randles, M., Lamb, D. and Taleb-Bendiab, A.: A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing, pp. 551–556 (2010).
- [2] Mishra, S. K., Sahoo, B. and Parida, P. P.: Load balancing in cloud computing: A big picture, *Journal of King Saud University - Computer and Information Sciences* (2018).
- [3] Al-Jaroodi, J. and Mohamed, N.: Distributed Cloud Cache (2018).
- [4] Song, S., Lv, T. and Chen, X.: A cooperative game method for load balancing in cloud based on cost-efficiency (2014).
- [5] Cardellini, V., Casalicchio, E., Colajanni, M. and Yu, P.: The State of the Art in Locally Distributed Web-

Server Systems, *ACM Comput. Surv.*, Vol. 34, pp. 263–  
311 (2002).