

クラウドリソースへの負荷分散ミドルウェアにおけるリモートデータアクセスの検討

豊島 詩織[†]

山口 実靖[‡]

小口 正人[†]

[†]お茶の水女子大学

[‡]工学院大学

1. はじめに

高度 IT 社会の進展によりコンピュータシステムにおいて利用可能なデータの量が増大している近年、よりスケラブルなリソース管理の実現が望まれている。本研究ではクラスタのシステム状況をモニタリングし、負荷が高い場合には外部のクラウドリソースへ動的に負荷分散を行うミドルウェアの構築を目指す。本研究の特徴は負荷分散先のリソースとして伸縮性の高いクラウドを用いている点、そしてデータインテンシブアプリケーションを対象負荷として用いている点である。対象をデータインテンシブアプリケーションとしているため、負荷を測定する指標としては CPU の値ではなくディスク I/O を用いた。

ローカル環境における負荷が高い場合にネットワーク越しの遠隔リソースへ負荷を分散すること自体は、グリッドコンピューティングなどの枠組みで実現できるため、新しい考え方ではない。しかし遠隔リソースとしてクラウドを使用することで、ユーザのニーズに応じてリソースを大幅に増減できたり、従量制のコストがかかったり、セキュリティポリシーにより社外にデータを置けないユーザが、データは社内には保存したまま計算能力だけクラウドから借りるケースが出てくるなど、従来とは異なる特徴がある。本研究では東京大学生産技術研究所が Eucalyptus を使用して構築したプライベートクラウド (以下 Cloko) を使用した。

2. ローカルクラスタのシステム構成

ローカルシステムはクラスタのワーカノードに仮想マシンを配置した仮想マシン PC クラスタとした。クラウドコンピューティングでは仮想化が重要な構成要素となっており、クラウドの特徴であるフレキシブルなリソースの利用が可能となっている。またストレージアクセスには SAN のプロトコルである iSCSI を使用することで高速なデータアクセスや、データの集中管理ができる環境とした。

3. データ処理アプリケーション最適配置ミドルウェア

3.1 最適配置の概要

クラウドでは従量制のコストがかかることも特徴であるため、クラウドへの負荷分散を行う際にはアプリケーションの実行時間に制限が設けられた場合にローカルクラスタとクラウドコンピューティングリソース間でどのようにジョブを分けるのがいいのか、実行が制限時間内に終わりかつ最もコストパフォーマンスが良くなるよう検討する。システム環境については、ローカルクラスタでは使用できるマシン台数に制限があるという現実的な環境で、その不足分をクラウドリソースで補う。クラウド

側は使用ノード数の制限がない。またジョブは順次投入されていく環境を想定している。

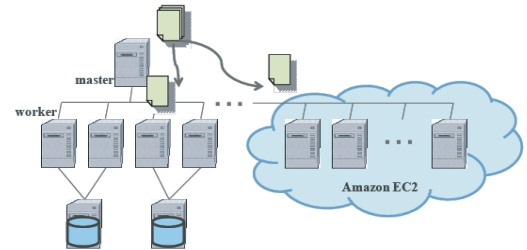


図 1 ローカルクラスタからクラウドリソースの調達

3.2 ミドルウェアの動作概要

ユーザはまずアプリケーション全体の実行制限時間を決定し、それを Limit time とする。ミドルウェアは Limit time を越えない範囲で、かつクラウドのコストが最も最小になるようローカルクラスタとクラウドに仕事を振り分ける。

ミドルウェアはローカルクラスタのディスク I/O を測定する Monitor 部とジョブを振り分ける Dispatch 部に分けられる。以下にそれぞれの仕事を示す。

【Monitor 部】

- 定期的にローカルクラスタの I/O 負荷を dstat コマンドを用いて測定。
- I/O 負荷によりジョブ量をチェック。
- ジョブの実行終了時間を見積もる。

【Dispatch 部】Dispatch 部では Monitor 部が見積もったジョブの終了時刻を元にクラウドへの仕事の振り分けを判断する。

- 順次投入されるジョブをローカルクラスタで実行。
- Monitor 部が予測したジョブの終了時刻が、Limit time を超える場合はクラウドに処理を依頼。
- クラウド内では Limit time 内で最もコストが低くなるようマシンの台数やジョブの配置を決定。

クラウドでのジョブの振り分けについては多くのマシンを借りたほうが実行時間が短くなるが、その分コストが課金される。そのため最もコストパフォーマンスが良くなるマシン台数を考える。このときコストは実行時間×台数で計算している。

3.3 ミドルウェアの精度

ミドルウェアの精度評価のためのデータインテンシブアプリケーションとしては PostgreSQL のベンチマークである pgbench を用いた。pgbench のデータを 5 秒ごとに 4 ユーザずつ合計 10 回を順次投入し、1 回毎にローカルクラスタとクラウドのどちらに投げるかを決定する。またトータルの実行時間について、ユーザがあらかじめ指定した制限時間 (Limit time) 内に終了させることを制約条件とする。

評価では上記のように pgbench を 10 回投入する際に、ミドルウェアを使用した場合と、ミドルウェアを使用せず理想的なリソース配置をした場合の振舞いを比較することでミドルウェア

(注): Load distribution middleware when executing data-intensive application

アの精度を測定する。ミドルウェアを使用した場合は、ローカルディスクの負荷状態から実行にどのくらいの時間がかかるかを見積り、Limit time より長くなる場合にはクラウドへ負荷を分散する。理想的な配置とは、単純にローカルクラスタとクラウドにおける pgbench の実行時間より判断した配置である。

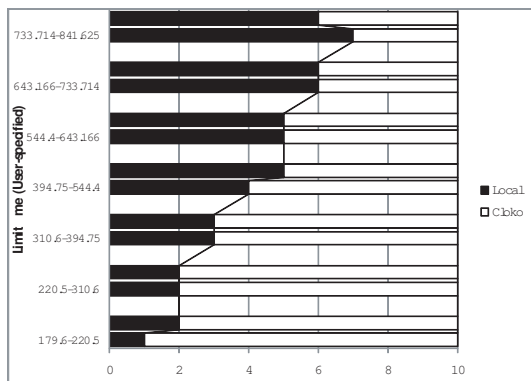


図 2 ミドルウェアを使用した際のジョブの振り分け

図 2 にローカルクラスタとクラウドの投げ分けにおいて、ユーザが指定したそれぞれの制限時間毎における最も理想的な振り分けをした場合（下）とミドルウェアを使用して振り分けた場合（上）、で 10 回中何回ローカルクラスタとクラウドへ投げたかを示す。7 回以上ローカルで実行するとモニタリングにおいてディスクアクセスが飽和となり、負荷が測定できないため 8 回以上ローカルで実行することはない。

この結果から、多少の誤差はあるものの、理想とかなり近い振り分けができていると言える。つまりローカルクラスタのモニタリング結果により、理想に近いジョブの振り分けができることが分かる。

4. データの配置

上記の実験では処理に必要なデータは常に処理を行うサーバ上にあることが前提であり、クラウドにジョブを投げる場合でも既にクラウド上にデータが存在することを仮定していた。しかし実際にリモートサイトのサーバに計算処理の一部をマイグレートする場合、処理を行うデータの扱い方については、以下の 2 つも考えらえる。

- (1) 処理のマイグレート時にオンデマンドでコピー
- (2) セキュリティポリシーや、データが巨大すぎるなどの理由からデータは常にローカルに存在

(1) については一般にリモートへのデータ転送はスループットが低いため、この方式はデータ量が多い場合には、性能低下を招く可能性がある。データ量が少ない場合やアプリケーション全体の処理時間が長い場合には有効であると考えられる。(2) の場合には、計算処理のみリモートサイトのリソースを利用しながら、データはローカルに置きリモートサイトからアプリケーション実行時にアクセスする事が考えられる。この場合には、クラウドサイトから計算処理サーバだけ借りれば良く、容易に負荷分散のマイグレーションが実現できる。ただしクラウドサイトとローカルクラスタの間の通信性能が全体の実行性能に大きな影響を与えるため、ネットワークの帯域幅が小さい場合やデータアクセス頻度が高いアプリケーションの場合には、

性能の大幅な低下が予想される。また、リモートサイトからローカルのストレージへのアクセスには制限がある場合もあり、これらの問題をクリアしなければならない。

このようにリモートサイトのリソースを利用して負荷分散を行う事を考える際、データインテンシブアプリケーションの場合には、データをどのように扱いつどこに置いて実行するか考える事が重要である。本稿では (2) の場合を想定しローカルクラスタとクラウドサイト間のネットワーク帯域とディスクアクセス性能を測定した。

図 3 にネットワークスループット、図 4 にディスクアクセスの結果を示す。ネットワークスループットの測定には、大きいサイズのファイルを転送する手法を用いた。またディスクアクセスの性能についてはベンチマークの dbench を使用した。

ディスクアクセスのグラフには比較としてクラウドのローカルディスクアクセス性能も示す。

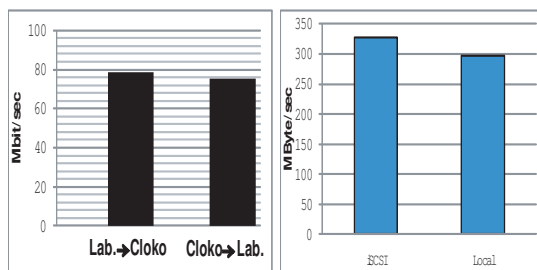


図 3 ネットワーク帯域 図 4 ディスクアクセス性能

以上のグラフよりネットワークについてもディスクについても十分実用的な性能がでていることが分かる。

5. まとめと今後の課題

スケーラブルなデータ処理システムを実現するため、手元のクラスタ負荷をモニタリングし、負荷が大きい場合は外部のクラウドコンピューティングリソースへ負荷分散を実現するミドルウェアを構築した。その際に対象をデータインテンシブアプリケーションを対象とし、システムのボトルネックとなるディスク I/O を負荷の判断指標とした。

ローカルクラスタとクラウド間のジョブの振り分けを行ったところ、ミドルウェアを使用した場合、理想に近い振り分けができていることが分かった。

本実験ではデータが常に処理を行うサーバ上にあることが前提だったが、今後はセキュリティポリシーなどの制約でデータは常にローカルに存在する場合や、処理のアウトソーシングと同時にデータのコピーを行うなど、より現実的な条件を考慮した実験を行う。

本研究は一部、文部科学省科学研究費特定領域研究課題番号 18049013 によるものである。

文 献

- [1] 豊島詩織、山口実靖、小口正人: "データ処理アプリケーションのクラウドリソースとローカルクラスタ間における負荷分散ミドルウェアの検討," SWoPP2010, CPSY-6, 金沢, 2010 年 8 月.