仮想マシンPCクラスタのロードバランスに関する評価と実クラウドへの適用の検討

豊島　詩織† 　山口　実靖†† 　小口　正人†

† お茶の水女子大学　〒 112-8610 東京都文京区大塚 2-1-1
†† 工学院大学　〒 163–8677 東京都新宿区西新宿 1–24–2
E-mail: †shiori@ogl.is.ocha.ac.jp, ††sane@cc.kogakuin.ac.jp, †††oguchi@computer.org

あらまし　高度 IT 社会の進展に伴いデータの管理や IT コストの問題が深刻になっている．そこで本研究では手元のクラスタ使用状況を観察し，リソースが不足している場合は外部のクラウドのリソースを動的に使用するといった，クラスタのスケーラブルな運用のためのミドルウェア構築を目指す．クラスタシステムには PC クラスタの各計算ノードに仮想マシンを配置した仮想マシン PC クラスタ，またストレージアクセスには IP-SAN を導入することで，サーバとストレージ間の広域環境における通信を低コストで実現することが期待される．本稿においてクラウドには商用のクラウドサービスである Amazon EC2 を用い，ローカルのクラスタでデータインテンシブアプリケーションを実行した場合，一定以上の負荷がかかる場合にリモートサイトへジョブを負荷分散することにより実行時間が早くなることを確認した．

キーワード　仮想マシン, クラウドコンピューティング, iSCSI, リモートストレージアクセス

Analyzing performance of storage access optimization with virtual machine migration

Shiori TOYOSHIMA†, Saneyasu YAMAGUCHI††, and Masato OGUCHI†

† Ochanomizu Univesity　2-1-1 Otsuka, Bunkyouku Tokyo 112-8610 JAPAN
†† Kogakuin University　1–24–2 Nishi-shinjuku, Shinjuku-ku, Tokyo, 163–8677 Japan
E-mail: †shiori@ogl.is.ocha.ac.jp, ††sane@cc.kogakuin.ac.jp, †††oguchi@computer.org

Abstract　For the management and process of explosive quantity of information, it is required to construct a scalable system in consideration with resource usage. It will be possible to connect server and storage at low cost by introducing virtual machine PC cluster and IP-SAN. When I/O processing is heavy, application performance degrads due to overhead of remote storage access. Thus in this study, we propose a technique to migrate virtual machine to a remote site that stores data, for the purpose of load balancing and optimization of storage access, instead of iSCSI remote access. Hereafter since we have interested in using real cloud computing, we have measured basic performance of Amazon EC2.

Key words　virtual machine, cloud computing, iSCSI, remote storage access

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，よりスケーラブルなリソース管理の実現が望まれている．そこで期待が高まっているのがクラウドコンピューティングである．クラウドコンピューティングにおいてユーザはリソースを利用するだけであるため，システムの運用コストが大幅に削減できる．そして必要なときに必要な分だけリソースを利用することが可能となる．このクラウドコンピューティングのメリットを活かし，本研究では使用しているクラスタのシステム状況をモニタリングし，急激に大量のキャパシティが必要となる場合に外部のクラウドリソースへ負荷分散するミドルウェアの構築を目指す．負荷分散先としてクラウドコンピューティングリソースを用いることで，ユーザのニーズに応じてリソースを大幅に増減できることが期待される．またセキュリティポリシにより社外にデータを置けないユーザでも、データは社内に保存したまま、計算能力だけクラウドから借りることが可能になる．

動作させるアプリケーションとしてデータインテンシブアプリケーションを想定しているため，システム評価においてもそれを使用した．

2. 研究背景

2.1 仮想マシン PC クラスタ

クラスタシステムはワーカノードに仮想マシンを配置した PC クラスタとし，仮想化ソフトには Xen [1] を使用した．Xen は図 1 に示すように複数の OS を動かす為の基盤となるプラットフォームのみを提供することで仮想マシンのオーバヘッドを少なくし，物理マシンに近い性能が発揮されるよう工夫されている．オープンソースとしては非常に高性能で，現在ビジネスユースにおいても用いられるようになっている．

仮想マシンモニタが仮想化のための土台となり，その上で Domain と呼ばれる仮想マシンが動作する．ホスト OS として動いているものが Domain0，ゲスト OS として動いているものが DomainU で，Domain0 は実ハードウェアへのアクセスやその他のドメインを管理する特権を持つ．

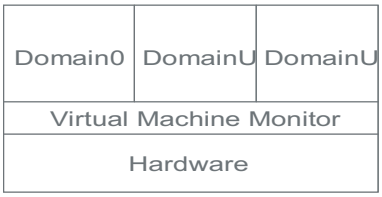

図 1　Xen の構造

2.2 IP-SAN

ストレージネットワークには SAN を使用した．近年情報システムにおいて処理されるデータの量が膨大になってきたことから，ネットワークストレージ技術が発展し，PC クラスタのストレージに SAN を用いることが多くなっている．SAN は，分散したストレージをネットワークで統合し，ストレージの集中管理とディスク資源の効率的な活用を可能にする．特に IP-SAN は Ethernet インタフェースと TCP/IP 対応ネットワークさえあれば導入でき，また専用網も含め広範囲に IP ネットワークのインフラが整備されているため長距離接続が可能で，クラウドコンピューティングなどの枠組を用い，計算機リソースのアウトソーシングに利用されることが期待される．

IP-SAN のプロトコルとしては iSCSI (Internet Small Computer System Interface) [2] を使用した．iSCSI の構造を図 2 に示す.iSCSI は SCSI コマンドを TCP/IP パケットの中にカプセル化することでブロックレベルのデータ転送を行う.Gigabit Ethernet/10Gigabit Ethernet が広く普及していくであろうことを考慮すると，IP-SAN をバックエンドに持つ PC クラスタ多くが使用されるようになると考えられる．

クラスタを仮想マシン PC クラスタとし，そのネットワークストレージに SAN(iSCSI) を導入することで，動作中のアプリケーションの状態を維持したまま仮想マシンを別のノード上へマイグレーションしたり，サーバとストレージの位置を分離した，リソースの柔軟な調達が可能となる．

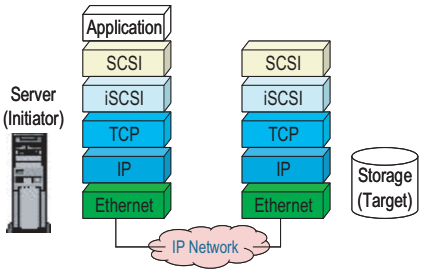

図 2　iSCSI の階層構造

2.3 クラウドコンピューティング

現在注目を集めているクラウドコンピューティングは，インターネットを介してアクセスするだけで，ネットワーク上に存在するサーバが提供するハードウェアやソフトウェアを利用できるサービス形態で，計算機リソースにおいては HaaS (hardware as a service) モデルが知られている．ハードウェアをクラウドから利用する場合，ユーザは実機を買い揃える必要がないので運用・管理コストの削減が可能になるほか，予めシステム規模が予測しづらいときなど，現在の使用状況に合わせてキャパシティを増減できるといったメリットがある．本研究ではこの特徴を利用し，クラスタで急激に大量のリソースが必要な場合にクラウドリソースへ負荷分散を行なう．

2.4 IP-SAN 統合型 PC クラスタ

文献 [3] は IP-SAN のプロトコルである iSCSI を使用し，クライアント-計算ノード間のフロントエンドと計算ノード-ストレージ間のバックエンドネットワークを同一の IP ネットワークに統合した IP-SAN 統合型 PC クラスタを構築している．

IP ネットワークを使用していることや，フロントエンドとバックエンドに同じネットワークを使用することから構築および管理コストの削減が期待されるが，ノード間通信とストレージアクセスで同じネットワークリソースを使用するため互いに衝突し，性能が低下する可能性が懸念された．このシステム上で並列データマイニングの HPA (Hash Partitioned Apriori [4] と PFP (Parallelized FP-growth)，科学技術計算の mpiBLAST [5] を動作させ， IP-SAN 統合型 PC クラスタの詳しい振舞を明らかにしている. 評価を行なった範囲では iSCSI のネットワークを統合してもネットワークバウンドにはならないということが分かっている．

3. 研究概要

3.1 実験環境

クラスタの各ノードのスペックを表 1 に示す．実験環境にてリモートアクセス環境を構築するため，iSCSI 通信時ローカルサイトとリモートサイト間には人工的な遅延を挿入する装置である Dummynet により往復遅延時間 (RTT) を挿入した．また各計算ノードには DomainU(virtual machine) を一つずつ配置した．

3.2 基礎実験概要

文献 [6] では，構築した仮想マシン PC クラスタにおいてストレージ機能を遠隔サイトで利用するシステムを構築し，

表 1　Experimental setup：PCs

OS	initiator :Linux 2.6.18-53.1.14.el5(CentOS5.3)
CPU	initiator：Intel (R) Xeon(TM) 3.6GHz target：Intel (R) Xeon(TM) 3.6GHz
Main Memory	initiator(Domain0)：2GB initiator(DomainU)：2GB target：4GB
iSCSI	initiator：iscsi-initiator-utils target：iSCSI-Enterprise-Target
Monitoring Tool	Ganglia　[9]

並列データマイニングの HPA，データベースベンチマークの OSDL-DBT3 (Open Source Development Labs Database Test-3) [7] という 2 種類の並列データ処理アプリケーションを動作させ，アプリケーション実行時にクラスタのノード間通信や I/O の実行を観察した．

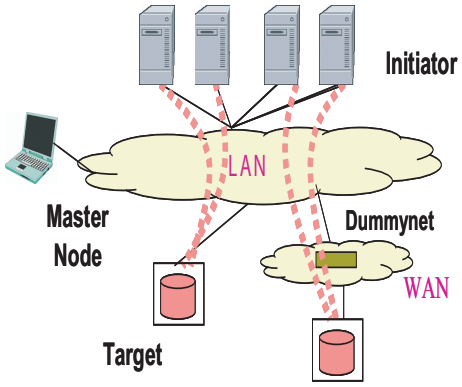

図 3　基礎実験：実験環境

HPA は相関関係抽出の代表的なアルゴリズムである Apriori アルゴリズムを元に作られたアプリケーションである．ジョブをワーカに振り分け，比較演算処理を繰り返し行なった後全ノード間通信を行なうため，全体として計算量やネットワーク通信量が多いアプリケーションである．

OSDL-DBT3 は意思決定支援システムをシミュレートする TPC-H [8] を簡略化したベンチマークであり，データベースへのデータの追加・削除，繰り返しのクエリ問い合わせが行なわれる I/O バウンドなアプリケーションである．

3.3　基礎実験結果

HPA は大量のトランザクションデータを処理するデータマイニングではあるが，全体として I/O バウンドではないため，遠隔ストレージアクセスのローカルサイト間とリモートサイト間の RTT が大きくなっても実行時間にあまり差は見られなかった (図 4)．

一方 OSDL-DBT3 は連続的なデータベースへのアクセスが発生する I/O インテンシブなアプリケーションであり，実験からも RTT が大きくなるにつれて実行時間が増大することが確認された (図 5)．

以上から HPA のような I/O バウンドではない並列アプリケーションの場合，PC クラスタにおいてストレージを遠隔サイトに配置しても，十分実用的な性能を発揮できるが，OSDL-DBT3 のような I/O バウンドなアプリケーションでは性能の劣化が著しい．上記の基礎実験に基づき，本実験では高遅延ネットワークを介したクラウド環境上にデータがある場合，同一 LAN 上に仮想マシンごとアプリケーションをマイグレートし，負荷分散とストレージアクセス最適化を行う手法を提案し，評価を行った．

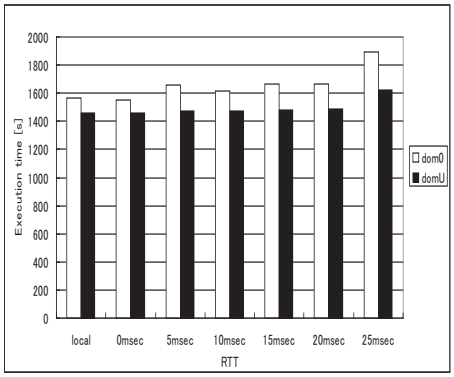

図 4　基礎実験：HPA

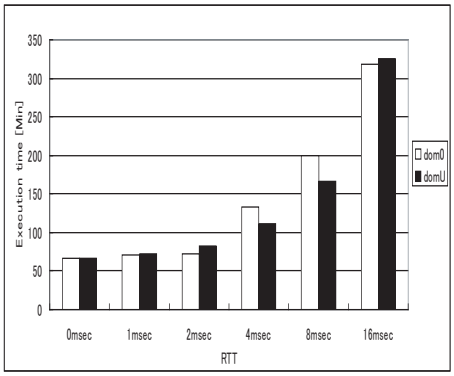

図 5　基礎実験：OSDL-DBT3

4.　実験結果と考察

基礎実験より，高遅延ネットワークを介したクラウド環境上にデータがある場合は大きな性能低下が起こることが確認された．そのためクラウド環境と同一 LAN 上に仮想マシンをマイグレートし，データが存在する遠隔サイトにてストレージにアクセスし計算処理させる実験を行なった．サーバ（計算ノードと Initiator）が 6 台，ストレージ (Target) が 2 台の仮想マシン PC クラスタ環境において，図 6 にマイグレーション前の実験環境図を示す．ローカルサイトにはサーバが 4 台とストレージが 1 台，遠隔サイトにはサーバ 2 台とストレージ 1 台がそれぞれ LAN で接続されている．iSCSI 遠隔アクセス環境を想定し，基礎実験と同様に 2 つの LAN の間には Dummynet により RTT が 1msec，2msec，4msec，8msec，16msec の遅延を挿入した環境で実験を行なった．

まず図 7 にそれぞれの RTT において仮想マシンをローカルサイトからリモートサイトにマイグレートする時間を測定した．RTT0msec から 8msec までは 21 秒となり，最も遅延の大きい

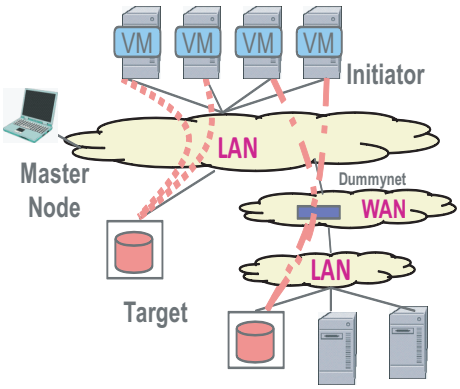

図 6　実験環境 (マイグレーション前)

RTT16msec の環境では 52 秒という結果になった.

図 8 にサーバをローカルからリモートにマイグレートする時間とその後 OSDL-DBT3 を実行した合計時間を比較した実行時間を示す. 比較対象として基礎実験において DomainU にジョブを与えた際の実行時間を示す. この結果から RTT が長くなるにつれて基礎実験で OSDL-DBT3 を実行した場合との差が大きくなることが分かる.

アプリケーション実行はサーバをリモートサイトにマイグレートすることで, 遅延無しの iSCSI 環境でストレージアクセスを行なうことができる. よって RTT が長くなるにつれてサーバをデータが配置されているリモートサイトにマイグレートした後計算処理を行なう効果が大きい.

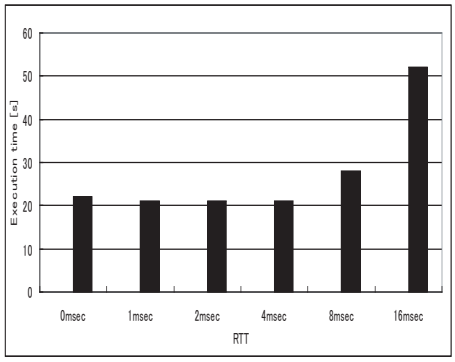

図 7　マイグレーション時間

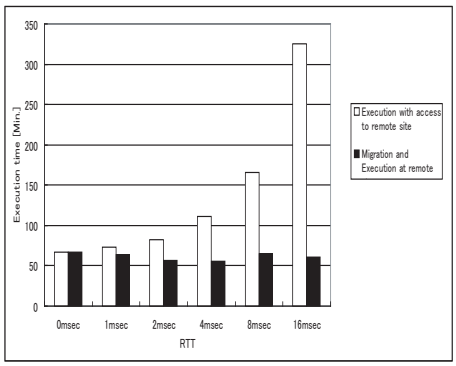

図 8　実 行 時 間

5.　実クラウド Amazon EC2 への負荷分散の検討

5.1　リモートサイトのリソース利用時のデータの配置

本研究では, ローカルクラスタにおいて負荷が高くなった場合にリモートサイトからリソースを借りて負荷分散を行うことを検討している. ただし本研究では主にデータインテンシブアプリケーションの実行を想定しているため, リソースとしては, 計算処理を行うサーバだけでなく, データおよびストレージをどのように取り扱うか考えなければならない.

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配置については, 大きく分けて以下の 3 つのケースが考えられる. まず 1 つ目は, 処理に必要なデータについても処理のマイグレート時にオンデマンドでリモートサイトにコピーするケースである. 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. ただしコピーが終わったらリモートサイトにおいてデータへ高速アクセスが可能となるため, 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 つ目は, ローカルクラスタでの処理中に遠隔バックアップが行われているケースで, 計算処理のマイグレート時に, リモートサイトにも処理に必要なデータが既に存在する形となる. この場合にはデータアクセスに関する制約が無くなるため, 積極的にリモートサイトのリソースを利用した方が性能面では有利になると考えられる. 3 つ目のケースは, データが巨大すぎるため, 或いはローカルサイトにおけるセキュリティポリシなどの理由により, データをローカルサイトから外に出す事が出来ない場合である. この場合には, 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. 例えば Google 社が提供するクラウドサービスである Google Apps においては Secure Data Connector [10] という仕組みが提供されており, これを利用するとクラウドとローカルサイトの間にセキュアトンネルが構築され, クラウドからローカルサイトのデータに対し, 安全にアクセスを行う事ができるようになる. このケースの場合には, リモート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. ただしリモート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どこに置いて実行するか考える事が重要である. さらに上記の 1 つ目と 2 つ目のケースのように, データをリモートサイトに配置して計算処理を実行する場合には, リモートサイトにおけるストレージについても, 何

台用いデータをどのように配置すべきかについて検討する必要がある．

5.2 遠隔リソースとしてのクラウドの特徴

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利用した場合，以下のような点で従来とは異なる特徴がある．まずクラウドはオンデマンドで利用することができるという点である．これまでのシステム環境の場合，負荷分散を実行するためには，予め利用するリソースを決め準備をしておかなければならなかった．これに対しクラウドの場合には，基本的に利用したい時点でリソースが利用可能となるため，クラウドを利用した負荷分散の枠組みさえ構築しておけば，必要な時にオンデマンドでの遠隔リソースの利用が実現できる．また，クラウドは必要な時点で利用できるというだけでなく，必要な量のリソースを自由に指定できるという点も特徴と考えられる．特に，リソースの量や種類にはあまり制限が無い場合が多く，必要であれば極めて多くのリソースを獲得して利用することも可能であり，負荷分散の際にリソース量の制限をあまり考える必要がない．すなわち，例えばある時点で突然，利用するリソース量を膨大に膨らませるような使い方も可能で，そのような形は従来のシステム環境では殆んど想定されていなかった．ただしリソースの利用には，一般に従量性のコストがかかるという点もクラウドの特徴である．このことからクラウドを利用して負荷分散を考える場合，パフォーマンスを最大にするという指標に加えて，使用リソース量と実行時間に比例してかかるコストを低く抑えるという指標も考慮しなければならない．一般にコストとパフォーマンスはトレードオフの関係にあるため，例えば大量のリソースを投入すれば実行時間が速くなるとしても，コストの制限がある場合には適切なリソース量でバランスを取る必要がある．このような指標で最適化を行う事は，従来のシステム環境にはない新しい評価基準である．

5.3 Amazon EC2 の概要

本実験においては人工的に遅延を挿入し，クラウドコンピューティングを模擬したリモートサイトを構築していた．今後実際のクラウドコンピューティングとして Amazon EC2(Elastic Compute Cloud) [11] を使用することを検討しているため，その基本性能の測定を行なった．EC2 はインターネット上のサーバレンタルサービスである．バックエンドではサーバ仮想化技術が使われているため，自分で作った環境の OS イメージをまるごとバックアップしたり，さらにその環境のイメージを複製して同一の環境を持つサーバを複数稼動させたりすることが可能となる。また，急に負荷が増えた際なども，イメージファイルさえ作っておけば，数分で新しいマシンを起動できるためシステムに合わせて柔軟な運用が可能となる．

我々は今後，データインテンシブアプリケーションにおいてそのアプリケーション実行時間などの性能をネットワークスループットやディスクアクセス性能から見積もり，手元のクラスタと実クラウドで動的な負荷分散を行なうことを検討している．

EC2 では数種類のスペックの仮想マシンが提供されており，全部で 5 種類のインスタンス・タイプの中から選択することができる．本実験においてはまず，デフォルトである「スタンダードプラン」の small と,「High CPU プラン」の High-CPU Medium という 2 種類のスペックを使用した．それぞれのスペックは表 2 の通りとなっている．ECU(EC2 Compute Unit) は Amazon が定義した CPU リソース単位量である．

表 2　本研究で使用する EC2 のスペック

Instance	CPU	memory	storage	platform
Small	1ECU × 1	1.7GByte	160GByte	32bit
High-CPUMedium	2.5ECU × 2	1.7GByte	350Gbyte	32bit

5.4 Amazon EC2 の基本性能

5.4.1 Network Throughput

まずローカルクラスタの仮想マシンの DomainU 間と，EC2 の 2 種類のインスタンス間のネットワークスループットをそれぞれ測定した (図 9)．DomainU のメモリが 2GByte，EC2 におけるメモリは 1.7GByte であることを考えても，EC2 間のスループットはローカルクラスタのマシンに比べ遅いことが分かる．

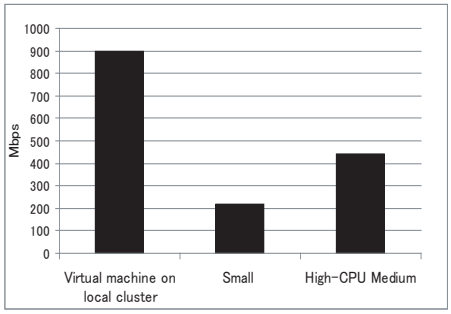

図 9　Network Throughput

5.4.2 Disk Access

データインテンシブアプリケーションの性能にはストレージアクセス性能の評価が不可欠となるため，ローカルクラスタの DomainU，EC2 において local disk，iscsi disk のアクセス性能を測定した (図 10，11)．

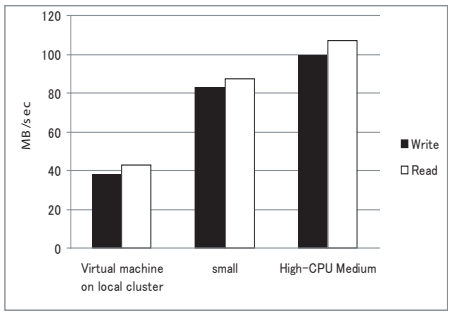

図 10　local disk

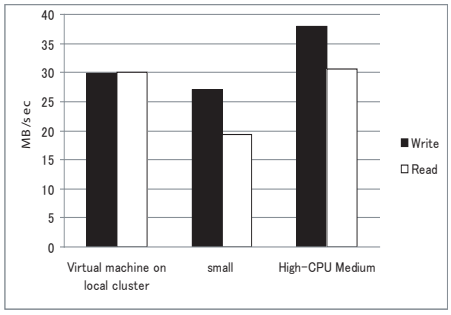

図 11　iSCSI disk

local ディスクアクセスについては実クラスタより EC2 インスタンスの性能がよいことが分かる．また iSCSI ディスクアクセスについては実クラスタと High-CPU Medium の性能が同程度となった．

5.4.3　ローカルクラスタと EC2 との通信

次に EC2(High-CPU Medium) とクラスタ間の Network Throughput を測定した (図 12)．結果から性能がよいと考えられるクラスタが送信側の性能が高いことが分かる．

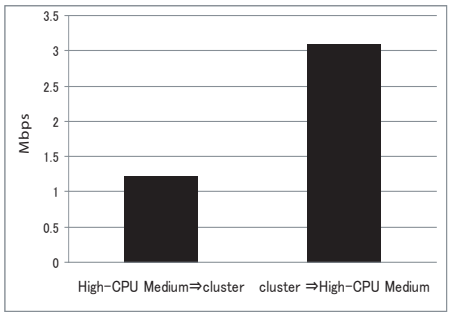

図 12　EC2 とクラスタ間の Network Throughput

6.　まとめと今後の課題

手元のクラスタリソースに対して，クラウドコンピューティングの枠組みにおけるリモートリソースを動的に使用する環境を構築するため，本実験においては data-intensive なアプリケーションに対する仮想化した PC クラスタの評価を行なった．基礎実験から，I/O インテンシブなアプリケーションにおいて遠隔アクセスを含む iSCSI 通信を行ったところ，RTT に伴い性能が大きく低下することが分かった．

このためローカルサイトの仮想マシンを遠隔サイトのサーバへマイグレートし，遠隔サイトにて直接ストレージアクセスと計算処理を実行する環境を構築したところ，マイグレートのコストを考慮しても，全体の実行時間がより高速になることが確認された．よって本研究で使用した OSDL-DBT3 のようなアプリケーションにおいては，リモートストレージにアクセスする負荷が大きい場合はサーバをリモートサイトに再配置し，そこで計算処理を行なう本手法は有効であることを確認した．

ここまでの実験では人工的な遅延を挿入した模擬的な遠隔リソース環境を利用していたが，次に実クラウドを遠隔リソースとして利用することを検討した．本研究ではデータインテンシブアプリケーションを対象としているため，リソースとしては計算処理を行うサーバだけでなくストレージも含まれ，データをどこにどのように配置するかという点を考慮して負荷分散を行う必要がある．これについて本論文では，計算処理のマイグレート時にデータもオンデマンドでリモートサイトにコピーする場合，遠隔バックアップなどによりリモートサイトにデータが既に存在する場合，データはローカルに置いたまま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場合の 3 つのケースについて検討を行った．また遠隔リソースとしてクラウドを利用した場合が，従来のグリッドコンピューティングなどの枠組みとは異なる点について議論した．そして実クラウドの例として Amazon EC2 を取り上げ，この基本性能を明らかにした．今後はクラスタシステムのスケーラブルなリソース管理を実現するミドルウェアの構築を目指して研究を進めて行く．そしてこれを EC2 などの実クラウドに適用し，本論文で述べたような様々なケースにおいて，実行時間やコストといった指標に対しそれぞれ最適な負荷分散を行える環境構築を進める．

謝辞　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] Xen : http://www.xen.org/
[2] iSCSI RFC: http://www.ietf.org/rfc/rfc3722.txt
[3] 原明日香、神坂紀久子、山口 実靖、小口正人: ”並列データマイニング実行時の IP-SAN 統合型 PC クラスタのネットワーク特性解析”, DEIM2009,2009 年 3 月
[4] 小口正人、喜連川優:”ATM 結合 PC クラスタにおける動的リモートメモリ利用方式を用いた並列データマイニングの実行”, 電子情報通信学会論文誌,Vol.J84-D-I, No.9, pp.1336-1349, 2001 年 9 月
[5] mpiBLAST:http://www.mpiblast.org/
[6] 豊島詩織、山口実靖、小口正人: ”仮想マシンマイグレーションによるストレージアクセス最適化に関する性能評価”,CPSY, 信学技報, Vol.109, No.296, CPSY2009-37, pp.13-18，京都，2009 年 11 月.
[7] OSDL-DBT3:http://ldn.linuxfoundation.org/
[8] TPC-H:http://www.tpc.org/tpch/
[9] Ganglia Monitoring System:http://www.ganglia.info/
[10] Google Secure Data Connector:http://code.google.com/intl/ja-JP/securedataconnector/
[11] Amazon Elastic Compute Cloud:http://aws.amazon.com/ec2/
[12] Aravind Menon,Alan L.Cox,Willy Zwaenepoel: ”Optimizing Network Virtualization in Xen”, USENIX Annual Technical Conference,2006 年
[13] Jose Renato Santos,Yoshio Turner,G.(John)Janakiraman,Ian Pratt:”Bridging the Gap between Software and Hardware Techniques for I/O Virtualization”, USENIX Annual Technical Conference,2008 年
[14] 谷村勇輔、小川宏高、中田秀基、田中良夫、関口智嗣: ”仮想クラスタに対する IP ストレージの提供方法の比較”,「ハイパフォーマンスコンピューティングとアーキテクチャの評価」に関する北海道ワークショップ (HOKKE),2007 年