インタークラウドにおけるローカルストレージに基づくインスタンスマイグレーションに関する高速化手法の提案

山下 暁香[1]　Eng Keong Lua[2]　小口 正人[1]

概要：近年，実世界におけるデータ量の増加—ビッグデータにより，大量のデータがローカル端末ではなく，クラウド上で管理されるようになった．クラウドの技術としては，シングルクラウドとハイブリッドクラウドの進化系であるインタークラウドが 2, 3 年以内に実用段階になるであろうと言われている．インタークラウドでは，異なるプロバイダが管理する複数のクラウドがネットワークによって接続され，リソースを共有する仕組みになっている．クラウドコンピューティングには多くの利点があるが，セキュリティ面で無視できない問題点がある．例えば，ホストサーバにある OS が全ての仮想マシン（VM）を管理しているため，ホストサーバが攻撃されれば，ユーザのデータが失われる可能性がある．更に，マイグレーション中のデータ盗聴の危険性も報告されている．特に，インタークラウドでは，VM のマイグレーションは，IPsec などのセキュアなネットワークを通して行われるべきである．しかし，セキュリティの強度とマイグレーションの速度はトレードオフの関係にあり，セキュリティを強くすればマイグレーションの速度は遅くなる．本論文では，インタークラウドにおけるマイグレーションのセキュリティと速度の双方を両立するマイグレーション方法を提案する．具体的には，VM イメージの暗号化のタイミングと範囲を調節することでマイグレーションの速度向上を実現する提案手法を 3 通り実装し，評価実験を行った．最も効率的な提案手法においては，デフォルトのマイグレーション速度に対し，約 20% の時間でマイグレーションを完了できることを示した．更に，本論文における提案手法では，マイグレーション元と先のサーバが高遅延環境にあるほど効果的である．

A Study of Fast Instance Migration Method on Inter-cloud based on Local Storage

Akika Yamashita[1]　Eng Keong Lua[2]　Masato Oguchi[1]

1. はじめに

近年のデータ収集技術とデータ解析技術により，実世界とサイバー空間におけるデータや情報量が爆発的に増加している．その結果として，これらの大量の情報は，ユーザのローカル端末ではなく，クラウド上で管理されるようになった．クラウド上でデータや情報を管理する利点としては，以下の 3 点があげられる．(1) ユーザ個人が大容量のストレージや専用のソフトウェアを保持する必要がない，(2) ユーザがいつでも必要なときに，ネットワークを通して，要求するデータにアクセスすることが可能，(3) サーバ故障時や災害時に，クラウド上における仮想マシン (VM) やデータが他サーバに複製，マイグレートされるので，データや情報を失う可能性が極めて低い．

クラウドコンピューティング技術は，プライベートクラウドとパブリッククラウドが分離してそれぞれ使用される「シングルクラウド」，プライベートクラウドとパブリッククラウドがネットワークで接続された「ハイブリッドクラウド」と進化してきた．更に，インタークラウドでは，異なるクラウドプロバイダ間で VM をマイグレーションする環境の実現が期待されている．一方で，競争相手であるクラウドプロバイダがリソースをシェアするインタークラウ

[1] お茶の水女子大学
Ohanomizu University, 2-1-1 Otsuka, Bunkyo-ku, Tokyo 112–0012, Japan
[2] MONASH University
MONASH University,Wellington Road Clayton Victoria 3800, Australia

ドの実現は現実には難しいとする見解の人もいるが，本論文では，そのような批判に対し，インタークラウドのための技術は，異なるプロバイダが提供するクラウド間のみでなく，同じクラウドプロバイダの支店の間でも有効であることを強調したい．ローカルなクラスタ内における VM マイグレーションは，すでに，実用段階になっているが，プロバイダの支店間での VM マイグレーションについては，ローカルクラスタのようには効率的に行われていないのが現状である．よって，本論文において提案される技術は，インタークラウドだけでなく，同一プロバイダ内のクラスタ支店間でも有効である．

クラウドコンピューティング技術では，そのパワフルな処理能力や，大量のストレージを利用できる点，また，ユーザが特定のソフトウェアをインストールや設定する必要がないという点でたくさんの利点があるが，セキュリティという観点から，見逃せない欠点がある．例えば，サーバ上で稼働しているそれぞれの VM は管理 OS が管理しているので，1 つの VM に対する攻撃，または脆弱性が他の VM にも影響を及ぼす可能性がある．さらに，VM のマイグレーション時に，デバイス共有やライブマイグレーションによる VM の物理移動から派生する問題がある．VM のマイグレーション時には，メモリのイメージを転送するので，マイグレーションに対する攻撃を防ぐために，暗号化されたセキュアなネットワークを用いることが必須となる．

特に，インタークラウドでは，異なるクラウドプロバイダ間で VM をマイグレーションする状況が考えられるので，マイグレーション時のセキュリティは一層重要である．しかし，セキュリティの強度とマイグレーションの速度はトレードオフの関係にあり，セキュリティを強くすればするほど，マイグレーションの速度は遅くなる．インタークラウドにおけるクラウド間の接続には，例えば，デフォルトで IPsec(security architecture for Internet Protocol) の使用が想定されるが，IPsec を使う場合，送出されるパケットは，小さなフラグメントに分割され，その一つ一つに対して，暗号化，復号が行われるため，効率が悪く，大きなサイズの VM をマイグレートするときには，多くの時間がかかる．本論文における評価実験では，セキュリティが確保されているネットワークプロトコルの代表例として，IPsec を用いた．

本論文では，VM をセキュアに，かつ，高速にマイグレートする手法を提案し，実機における評価実験によって，その性能向上率を示した．本論文における提案手法では，暗号化されたネットワークでマイグレーションをするのではなく，VM のイメージの必要部分を暗号化し，暗号化された VM をマイグレーション先のサーバに移動した後で復号し，VM イメージの差分を更新するという手法である．実機実験では，IPsec トンネルにおいて，通常のマイグレーションコマンドを利用する既存手法と，提案手法としては，通常の TCP/IP ネットワークにおいて，VM イメージ全体に対して暗号化処理を施す提案手法，VM イメージを圧縮することにより必要部分のみに対して暗号化処理を施す手法，そして，マイグレーション元とマイグレーション先の双方のディスクイメージの差分を抽出し，マイグレーション先のディスクを更新する手法の 3 通りの提案手法を実装した場合の静的な VM マイグレーション速度を比較した．実験の結果により，本論文における提案手法を用いる場合，最短で，デフォルトのマイグレーション手法の約 20% の時間でマイグレーションを完了できることを示した．さらに，本論文における提案手法は，マイグレーション元とマイグレーション先のサーバ間の物理的距離が長く，高遅延環境にあるほど，有効である．

構成：

2 節でインタークラウドにおけるマイグレーション技術の特徴を述べ，3 節で本論文における提案手法を説明する．4 節で実験概要と結果，考察を述べ，5 節で関連研究を紹介し，本提案手法の有効性を示す．そして，6 節で本論文をまとめる．

2. インタークラウド

2.1 インタークラウドのアーキテクチャ

図 1 に示したように，インタークラウドは，シングルクラウド，ハイブリッククラウドの発展形である．

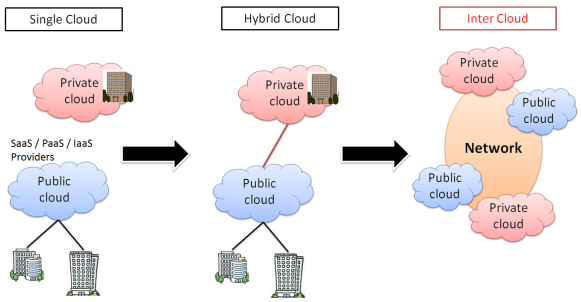

図 1　クラウドコンピューティング技術の進化

シングルクラウドでは，パブリッククラウドとプライベートクラウドが独立して存在する．プライベートクラウドは，会社や個人が，ローカルでプライベートなネットワークを用いて使用するものであり，パブリッククラウドは，SaaS (Software as a Service)，PaaS (Platform as a Service)，IaaS (Infrastructure as a Service) などのクラウドプロバイダによって，管理され，サービスが提供される．パブリッククラウドを使用するユーザは，クラウドプロバイダと契約をすることで契約に応じたリソースを利用できる．ハイブリッドクラウドでは，プライベートクラウドとパブリッククラウドがネットワークによって接続され，プライベートクラウド上のデータや情報は，少なくとも一時的にパブリッククラウドから見ることができる．このプラ

イベートクラウドとパブリッククラウドを接続するネットワークはセキュアでなければならない．インタークラウドでは，異なるプロバイダ間のクラウドが同じネットワークで接続されるので，ネットワークのセキュリティは一層重要になる．

インタークラウドの構成を**図 2** に示した．インタークラウドは，Cloud Exchange，Cloud Broker, Cloud Coordinator といった要素から構成されている [1]．Cloud Exchange は Cloud Broker からのクエリを収集，Cloud Coordinator によって公開された，利用可能なリソースの評価を行う．Cloud Coordinator は，VM のスケジューリング，アロケーションにおけるメカニズムをモニタする．Cloud Broker は，ユーザとクラウドのインタフェースの役割を果たし，ユーザからのクラウドのサービスのクエリを解決する．Cloud Coordinator によって収集された利用可能なリソースの情報は Cloud Exchange に渡され，ユーザによるクエリは，Cloud Broker によって処理される．

2.2 インタークラウドにおける VM マイグレーション

クラウドコンピューティング技術には，2 種類のマイグレーションがある．通常の（静的な）マイグレーションとライブマイグレーションである．通常のマイグレーションにおいては，VM はマイグレーション時に，マイグレーション元で一時停止され，マイグレーション先において，再起動する．それに対し，ライブマイグレーションにおいては，VM を停止させず，稼働したまま，マイグレーションを行う．本論文で扱うインタークラウドにおける VM マイグレーション時の要件として，VM のマイグレーション中の VM 自体のセキュリティ，ネットワークのセキュリティの確保とローカルストレージの利用が挙げられる．

インタークラウドでは，異なるクラウド間でネットワークを通して VM のイメージがやりとりされる．今までのマイグレーションは，1 つのクラウドのクラスタ内で共有ストレージが使用されてきたが，インタークラウドにおけるマイグレーションでは，マイグレーション元とマイグレーション先のホストが遠隔にあることが想定され，これらのホストが遠隔アクセスで iSCSI などの共有ストレージを用いることは，非現実的である．よって，本論文における実験では，それぞれの VM が用いるストレージは，ローカルストレージとした．

インタークラウドにおけるマイグレーションに用られるネットワークとしては，たとえば，IPsec などのセキュアなネットワークである必要がある．IPsec では，パケットは IP 層で暗号化され，安全に転送される．送信側と受信側で SA(Security Association) に基づき鍵を決定し，受信側では IKE(Internet Key Exchange) が認証に使用される．IPsec とは，インターネット上の 2 地点間に仮想的なトンネルを作り，そこに IP パケットを通すものである．トンネルを通すパケットには，暗号をかけることで，パケットの中身が第 3 者によって盗聴されることを防ぐ．トンネルの入り口である IPsec のゲートウェイは，LAN から受け取ったパケットを暗号化し，トンネルの出口のゲートウェイを宛先にした IP パケットに暗号化したデータを入れ（カプセル化），宛先 IP アドレスへ転送する．トンネル出口のゲートウェイは，受信パケットからカプセル化をほどいて，暗号化されたパケットを取り出し，送信側と同じ暗号鍵を用いて，復号する．IPsec トンネルを通して VM をマイグレートする場合，データは，IP 層で IP パケットに分割され，それぞれの分割されたパケットに対して，暗号化処理と復号処理が行われる．IPsec を通して VM をマイグレーションをしている間，セキュリティは強いが，マイグレーションにかかる時間が長くなることが想定されるので，IPsec は非効率的である．

このマイグレーションの速度とセキュリティの両立に関する問題は解決されなければならない．本論文ではインタークラウドにおいて，セキュアに，かつ，高速にマイグレーションを実現する手法を提案した．

3. 提案手法

前節で述べたように，既存のマイグレーション手法においては，セキュアなネットワークを用いなければならないが，IPsec のようなセキュアなネットワークを用いる場合，小さなセグメントに分割されたパケットに対して暗号化処理を行うため，非効率である．また，IP 層のような下位層で暗号化処理を行う場合，上位層から渡されたデータの内容は理解できず，逐次暗号化するのみである事から，データの中身が関わらず，結果として全てのデータを暗号化しなければならないため，性能面で不利であると考えられる．よって，本論文では，VM イメージの転送速度を速めるために，IPsec などのセキュアなネットワークを用いず，VM イメージを暗号化することで通常のネットワークを通してマイグレーションする手法を提案する．

インタークラウドにおける既存のマイグレーション手法を**図 3** に示した．

手順は以下の通りになると考えられる．ライブマイグレーションの場合は，マイグレーション元と先のホストにおける VM の停止と再起動の処理は無く，マイグレーション中も，VM は稼働し続ける．

(1) マイグレーション元であるホストサーバにおける VM の停止
(2) IPsec トンネルを通して，メモリとディスクの情報をマイグレーション先のクラウドへ転送
(3) マイグレーション先クラウドにおける VM の再起動

既存手法に対して，IPsec を使用しない提案手法を**図 4** に示した．IPsec を用いてそれぞれの IP パケットに対して暗号化処理を施すのではなく，VM のマイグレーション前

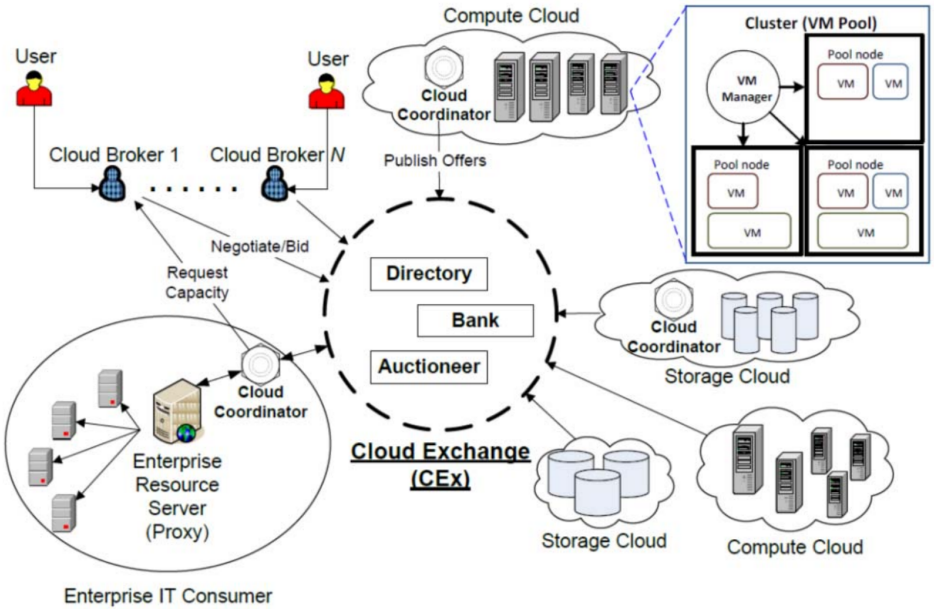

図 2　インタークラウドの構成

に，VM イメージの必要部分のみを暗号化し，SCP (Secure Copy Protocol) を用いて暗号化された必要部分をマイグレーション先のサーバへ転送し，マイグレーション先サーバで復号処理を行って，差分を更新する．VM イメージ自体を暗号化しているので，IPsec トンネルを使わなくても，第３者によって盗聴される心配はない．また，VM イメージの転送に実験では広く用いられているファイル転送であることから SCP を用いたが，既に暗号化されたデータであるため，RCP や FTP などの非暗号化ファイル転送を用いても，セキュリティ上は問題ない．提案手法は以下の通りである．

(1) マイグレーション元であるホストサーバにおける VM の停止
(2) マイグレーション元において，VM イメージをあらかじめ暗号化
(3) SCP を用いて暗号化された VM のイメージを転送
(4) マイグレーション先であるホストサーバにおいてイメージを復号
(5) VM の再起動

なお，本論文における実験は全て静的なマイグレーションを用いたもので，VM を停止させずに宛先ホストサーバへ移動するライブマイグレーションについては今後の課題とする．

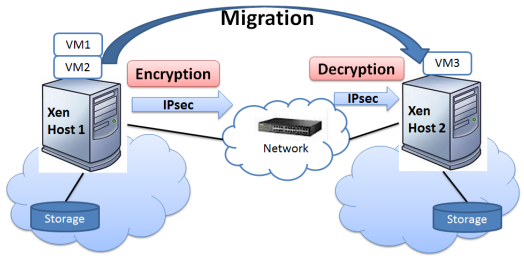

図 3　インタークラウドにおけるマイグレーションの既存手法

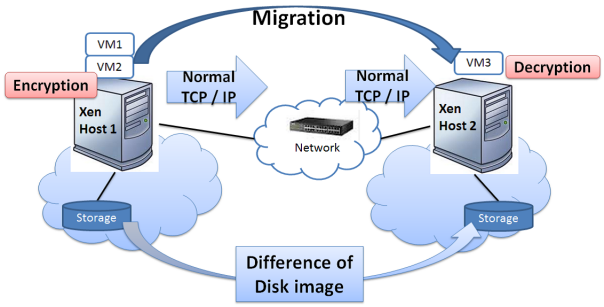

図 4　インタークラウドにおけるマイグレーションの提案手法

4.　評価実験

本論文の評価実験では，VM マイグレーションについて，3 通りの提案手法を実装し，既存手法と提案手法の全 4 通りのマイグレーション時間を比較した．

本論文における実験では，2 つの高遅延環境に存在する異なるクラウド間で VM をマイグレーションする状況を想定している．高遅延環境にあるクラウド間で iSCSI などのストレージ共有を行うのは非現実的であるため，マイグ

レーション元とマイグレーション先のホストサーバは，それぞれローカルストレージを用いる．

評価実験で用いた 4 つの手法の手順は以下の通りである．

既存手法　：「migration」コマンドを用いた既存のマイグレーション．

提案手法 A　：マイグレーション前にマイグレーション元のホストサーバにおいて，VM イメージ全体を暗号化し，マイグレーション先のホストサーバで復号

提案手法 B　：マイグレーション前にマイグレーション元のホストサーバにおいて，VM イメージ全体を圧縮した後に暗号化し，マイグレーション先のホストサーバにて復号と解凍

提案手法 C　：マイグレーション前にマイグレーション元と先のホストサーバ間の VM イメージの差分を抽出し，パッチファイルを作成し暗号化，マイグレーション後に，マイグレーション先のホストサーバにおいてパッチファイルを適用し，VM イメージの差分を更新

提案手法 A は VM イメージ全体を暗号化するので，3 通りの提案手法のうち，もっともマイグレーション時間が長くなる．提案手法 B は，VM イメージを圧縮しているので，提案手法 A と比較して暗号化するデータ量が減少し，提案手法において，最低限短縮できるマイグレーション時間を示す．さらに，提案手法 C では，マイグレーション元と先の VM のディスクイメージの差分のみを抽出しているので，提案手法のうち，最短のマイグレーション時間を実現できる．

なお，既存手法と提案手法 C は，あらかじめマイグレーション先のストレージに必要なデータが存在することを前提としている．一方，提案手法 A と B は，ストレージのデータもマイグレーション時に暗号化して送るため，あらかじめデータを配置しておく必要はない．

4.1　評価実験環境

4.1.1　Xen

クラウドを管理する仮想化ソフトウェアとして，それぞれのホストサーバに Xen [2] を導入した．Xen は一つのハードウェアでマルチ OS の並列処理と管理を提供する．Xen では，それぞれの VM は Domain と呼ばれ，Xen ハイパーバイザは一つまたは複数の OS をサポートし，物理 CPU のスケジューリングを行う．ホスト OS は Domain-0（dom0）と呼ばれ，Dom0 上に新たに作られたゲスト OS，つまり，仮想マシン VM は Domain-U（domU）と呼ばれる．それぞれのホストサーバでは，一つの dom0 が複数の VM である domU を管理しており，dom0 は，ハイパーバイザが起動し，認証が完了すると，自動的に起動する．全ての物理ハードウェアは dom0 から直接アクセス可能であり，システムの管理者は dom0 を通して，全ての domU にログインすることが可能である．

マイグレーション元と先である 2 台の Xen ホストサーバのスペックは表 1，サーバ上の VM，つまり，それぞれの Domain のスペックは表 2 の通りである．4GB のメモリと 222GB のディスクを搭載した 64bit 環境のホストサーバ上で，それぞれの 4GB 分のサイズの VM を作成し，それぞれの VM にメモリ 2GB を割り当てた．

表 1　Xen ホストサーバ 1 と 2 の設定

OS	Linux 2.6.32-5-xen-amd64 and xen-4.0-amd64
Distribution	Debian GNU / Linux 6.0.2
CPU	Intel(R) Xeon(R) CPU 3.60GHz
Memory	4 GByte
Disk	222 GByte

表 2　Xen ホストサーバ上における VM の設定

OS	Linux 2.6.32-5-xen-amd64 and xen-4.0-amd64
Distribution	Debian GNU / Linux 6.0.2
CPU	Intel(R) Xeon(R) CPU 3.60GHz
VCPU	1 core
VCPU Memory	2 GByte
Disk	4 GByte

Xen ホストの 2 ノードの間には，高遅延環境にある 2 つのクラウド環境を模擬するために，Dummynet を挟んだ．Dummynet は，高遅延通信を人工的に発生させる装置である．Dummynet の端末のスペックは表 3 の通りである．

表 3　Dummynet

OS	FreeBSD 6.4-RELEASE
CPU	Intel(R) Xeon(R) CPU 3.60GHz
Disk	64 GByte

既存手法の実験では，VM を IPsec トンネルを通してマイグレーションを行った．IPsec は，openswan [3] のパッケージをインストールすることにより導入した．IPsec の詳細設定は表 4 の通りである．既存手法における IPsec の暗号化アルゴリズムと提案手法における VM イメージの暗号化については，公平な比較のため，双方とも AES 128bit 鍵を用いた．提案手法における暗号化と復号の実行には，OpenSSL [4] パッケージを利用した．

表 4　IPsec トンネルの設定

モード	トランスポートモード
IPsec アルゴリズム	ESP
認証アルゴリズム	HMAC-SHA-1
暗号化アルゴリズム	AES 128 （Pre-Shared Key）

4.2　具体的な実験手順

各手法の具体的な実験手順を以下に示す．

4.2.1　既存手法：IPsec を用いたマイグレーション

(1) マイグレーション元ホストサーバにて VM(domU) を起動
(2) IPsec トンネルを張る（AES key 128 bit）
(3) マイグレーション先 Xen ホストサーバの IP アドレスを指定して，migration コマンドを実行

4.2.2　提案手法 A：VM イメージ全体暗号化

(1) マイグレーション元で domU を起動
(2) マイグレーション元で domU を停止
(3) openssl コマンドを用いて VM のイメージファイル（disk.img）全体を暗号化（AES 128bit 鍵）
(4) SCP コマンドで暗号化した VM のイメージファイルをマイグレーション先へ転送
(5) マイグレーション先で openssl コマンドを用いて VM のイメージファイル（disk.img）を復号
(6) マイグレーション先ホストサーバにて domU を再起動

4.2.3　提案手法 B：VM イメージ圧縮と暗号化

(1) マイグレーション元で domU を起動
(2) マイグレーション元で domU を停止
(3) VM のイメージを圧縮
(4) openssl コマンドを用いて圧縮済みの VM のイメージファイル（disk.img）を暗号化（AES 128bit 鍵）
(5) SCP コマンドで暗号化した VM のイメージファイルをマイグレーション先へ転送
(6) マイグレーション先で openssl コマンドを用いて VM のイメージファイル（disk.img）を復号
(7) VM のイメージを解凍
(8) マイグレーション先ホストサーバ domU を起動

4.2.4　提案手法 C：VM イメージの差分抽出

(1) マイグレーション元で domU を起動
(2) マイグレーション元で domU を停止
(3) マイグレーション元と先のホストサーバにて VM イメージの差分を抽出し，patch ファイルを作成（xdelta [5] パッケージを使用）
(4) 差分の patch ファイルを暗号化（AES 128bit 鍵）
(5) SCP で patch ファイルをマイグレーション先ホストサーバへ転送
(6) マイグレーション先ホストサーバにて patch ファイルを複合
(7) patch ファイルを VM イメージに適用して差分を更新
(8) マイグレーション先ホストサーバ domU を起動

なお，提案手法 C における差分抽出はマイグレーション元で行い，マイグレーション元とマイグレーション先に同じ状態の VM イメージが存在するという前提で実験を行っている．提案手法 C における「マイグレーション先」の VM においては特に処理が行われていない VM のイメージファイルの初期状態なので，マイグレーション時にマイグレーション元からコピーしてくる必要は無く，元々、マイグレーション元にコピーを置いておいた初期状態の VM のイメージファイルを利用している．つまり，提案手法では，まず，事前にマイグレーション元と先のホストサーバに同じ状態の VM イメージを配置した後に，実験手順を開始している．

従って既存手法と提案手法 C においては，マイグレーション開始時に同じ VM のイメージがマイグレーション元と先のホストサーバに存在し，この状態から，マイグレーション元の VM イメージに対して，約 30% の差分を加えた後にマイグレーションを行う評価実験となっている．ここで，30% の差分の設け方として，マイグレーション元とマイグレーション先のホストサーバに同じ VM イメージを配置し，マイグレーション元のホストサーバにおける VM 内に，VM の全体容量の 30% 分のファイルを作成した．

4.3　実験結果と考察

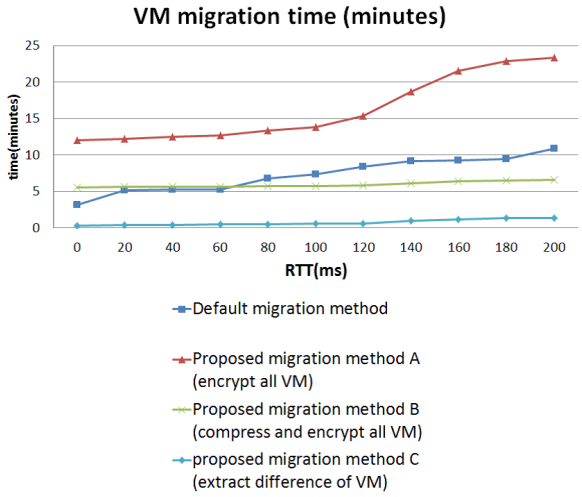

図 5　既存手法と提案手法における VM マイグレーション時間の比較

図 5 に既存手法と提案手法におけるマイグレーション時間の比較実験の結果を示した．縦軸は，それぞれの手法のマイグレーション時間，横軸は，ダミーネットで設定されている RTT(Round Trip Time) を示している．本論文の実験における RTT の値については，0ms〜200ms の値を用いた．インタークラウドで VM をグローバルに遠隔にあるサーバ（クラウド）へマイグレーションする場合を想定している．例えば，米国大陸横断の往復遅延時間は約 60ms である [6]．これを元に計算すると，東京-大阪間の RTT は 20ms 程度，東京-アメリカ西海岸は 120ms 程度，東京-ヨーロッパ諸国は 200ms 程度である．ただし，使用する回線の種類やネットワーク環境，地理環境などにより，距離のみで計算される RTT からある程度差が生じる．なお，全ての実験は，3 回施行した結果の平均をとっている．

グラフはそれぞれ，既存手法，提案手法 A，B，C の結果を示している．提案手法 A のマイグレーション時間は，全ての提案手法の中で最長の結果となっている．提案手法 B は，提案手法 A を可能な限り短縮したマイグレーション時間を示しており，提案手法 C では，VM イメージの差分のみを抽出しているので，既存手法と比較して，大幅な速度改善が示されている．

グラフ**図 6**，**図 7**，**図 8** はそれぞれの提案手法 C における固定時間を示している．縦軸は時間，横軸はそれぞれの手順を表している．なお，図 8 の縦軸の値が，図 6 と図 7 の縦軸の値と異なることを注意されたい．

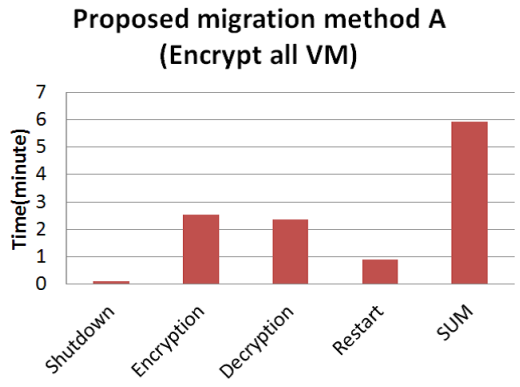

図 6　提案手法 A における固定時間

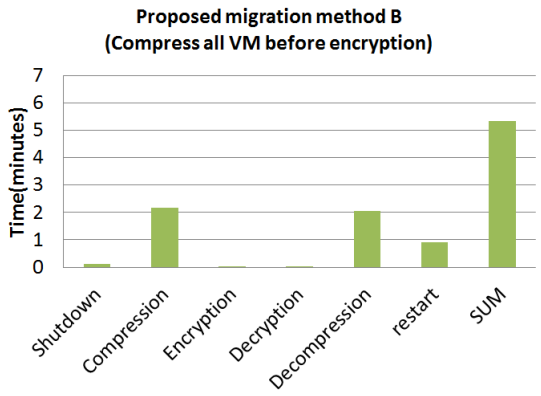

図 7　提案手法 B における固定時間

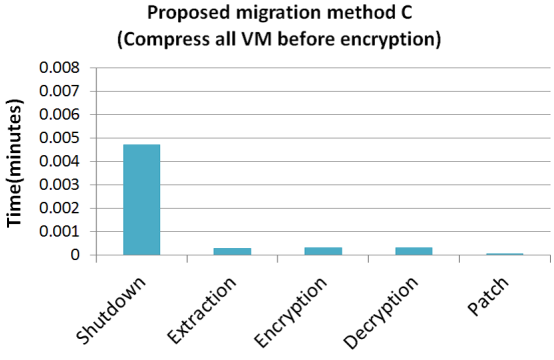

図 8　提案手法における固定時間

図 5 より，提案手法 A は，既存手法の約 2 倍のマイグレーション時間がかかっている．これは，VM イメージ全体に対する暗号化処理，複合処理に約 4 分以上の時間がかかっているためである（図 6）．既存手法と提案手法 B を比較すると，RTT が 60ms 以上の時に，提案手法 B の方が既存手法よりもマイグレーション時間が短くなっていることがわかる．なお，提案手法 B においては，VM イメージの圧縮と解凍の時間が約 4 分以上かかっており，ボトルネックとなっている（図 7）．

提案手法 C は，もっとも短い時間でマイグレーション時間が完了している．これは，VM イメージの差分のみを効率的に抽出できているためである．提案手法 C の固定時間図 8 についても，提案手法 A と C と比較すると短い．

提案手法 C における差分抽出の方法について説明する．提案手法 C の差分抽出の際は，マイグレーション元と先のホストサーバにおける VM のイメージについて，4byte 以上一致する部分については圧縮し，残りの部分のみを抽出している．よって，patch ファイルのサイズが小さくなり，高速にマイグレーション先サーバへ転送することができる．

5.　関連研究

VM のマイグレーションに関する分野の研究では，主にマイグレーションのパフォーマンス向上が注目されてきた．特に，VM を停止させずにマイグレーションをするライブマイグレーションに関しては [7] や [8] で説明されているように，メモリのコピーとディスクの同期が重要な技術となる．[1] では，インタークラウドにおけるアプリケーションのサービスのスケーラビリティについて述べられており，複数のクラウドベンダー間のクラウドにおける VM の可動性について，確かな性能向上とコスト削減の手法が示されている．[9] では，インタークラウドにおける VM の可動性と負荷分散についての手法が提案されている．[10] では，クラウド間の VM マイグレーションにおける物理的なネットワークパフォーマンスの向上について述べられている．このように，インタークラウドにおける VM マイグレーションに関する研究では，主に，マイグレーション時のさまざまな要素の性能向上について注目されてきた．

VM のマイグレーションにおけるセキュリティの考慮に関しては，J. Rexford らが，ノーハイパーバイザを提案した [11]．ハイパーバイザをセキュアにすることは，セキュアなマイグレーションを実現する良い手法であるが，ここで提案されている手法は，現在の VM マイグレーションメカニズムの脆弱性に対して，十分ではない．

マイグレーション技術では，セキュリティとパフォーマンスの両立が大切であり，これらがトレードオフの関係にあることを注意する必要がある．

6. まとめと今後の課題

本論文では，インタークラウドにおいて，IPsec のようなネットワークのトンネルを使うのではなく，VM イメージ自体の必要な部分に対して暗号化と復号処理を施すことで，マイグレーション時のセキュリティとパフォーマンスを両立する手法を提案し，実機における比較評価実験を通して，提案手法の有効性を示した．もっとも性能向上が見られた提案手法では，既存手法に対して，約 20% の時間でマイグレーションが完了できることが分かった．さらに，本論文における提案手法では，特に，高遅延環境において有効である．また，VM イメージ自体を暗号化しているため，マイグレーション時のセキュリティも担保できている．

今後の課題としては，本提案手法をライブマイグレーションに適用する予定である．ライブマイグレーションについては，マイグレーション時のメモリの情報も暗号化する必要があり，ライブマイグレーションにおける効率的な手法を提案したい．

謝辞

本研究を進めるにあたり，MONASH 大学の Eng Keong Lua 教授に大変有用なアドバイスをいただきました．深く感謝いたします．

参考文献

[1] R. Buyya, R. ranjan, and R. N. Calheiros, “Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services,” in *the 10th International Conference on Algorithms and Architecture for Parallel Processing (ICA3PP2010)*, May 2010, pp. 13 – 31.

[2] “Xen project,” `http://xen.org/`.

[3] “Openswan,” `https://www.openswan.org/projects/openswan/`.

[4] J. Viega, M. Messier, and P. Chandra, “Openssl,” 2009, Network Security with OpenSSL.

[5] xdelta.org, “xdelta,” `http://xdelta.org/`.

[6] David G. Messerschmitt, “Understanding network applications,” 2000, Morgan Kaufmann Publishers.

[7] Diego Perez-Botero, “A brief tutorial on live virtual machine migration from a security perspective,” 2011, http://www.cs.princeton.edu/ diegop/courses.html.

[8] Jansen and Gerardus T., “Mechanism for inter-cloud live migration of virtualization systems,” 2012.

[9] K. Nagin, D. Hadas, Z. Dubitzky, A. Glikson, I. Loy, and B. Rochwerger adn L. Schour, “Inter-cloud mobility of virtual machines,” in *the 4th Annual International Conference on Systems and Storage Article(SYSTOR2011)*, May 2011, p. No. 3.

[10] M. Tsugawa, P. Riteau, A. Matsunaga, and J. Fortes, “User-level virtual networking mechanisms to support virtual machine migration over multiple clouds,” in *IEEE GLOBECOM Workshops (GC Wkshps 2010)*, December 2010, pp. 568–572.

[11] E. Keller, J. Szefer, J. Rexford, and R. B. Lee, “Nohype: Virtualized cloud infrastructure without the virtualization,” in *the 37th annual international symposium on Computer architecture (ISCA2010)*, 2010, pp. 350 – 361.