様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

平成　２９　年　　６　月　１５　日現在

機関番号：　１２６０５
研究種目：　若手研究(B)
研究期間：　2015～2016
課題番号：　１５Ｋ１５９６７

研究課題名（和文）ビッグメモリアプリケーションを考慮した仮想マシン移送に関する研究

研究課題名（英文）Live Virtual Machine Migration for Big Memory Applications

研究代表者
山田　浩史（Yamada, Hiroshi）

東京農工大学・工学（系）研究科（研究院）・准教授

研究者番号：００５７１５７２

交付決定額（研究期間全体）：（直接経費）　　3,000,000円

研究成果の概要（和文）：本応募研究課題ではビッグメモリアプリケーションが稼働している仮想マシン(VM)を移送する方式について研究する．開発する方式はハイパフォーマンス分野において成熟しつつあるGPU (Graphics processing units) プログラミング技法を活用する．プロトタイプを Xen 上に実装し，関係データベースサーバである PostgreSQL，インメモリデータベースである Memcached，またメモリインテンシブなワークロードを模した LMBench などを稼働させ，実験を行なった．結果として，最大で 16.7 倍の移送時間高速化に成功した．

研究成果の概要（英文）：This research explores mechanisms for live migration suitable for virtual machines with big memory applications. The proposal mechanism leverages GPGPU programming techniques sophisticated in the high performance community. We implement a prototype on Xen 4.2.1 and conduct several experiments using real-world applications including PostgreSQL and Memcached. The experimental results show that migration time of our prototype is up to 16.7x shorter than that of the regular live migration on Xen.

研究分野：　システムソフトウェア

キーワード：　仮想マシン移送

１．研究開始当初の背景

ビッグデータ時代と呼ばれるようになった今日では，ビッグメモリアプリケーションと呼ばれる大量のメモリを必要とするアプリケーションが注目を浴びている．ビッグメモリアプリケーションの特徴は，数 GB～数百 GB のメモリを確保してその中にデータを配備しながら処理を進めていく部分にある．たとえば，VoltDB や Memcached といったインメモリデータベースは低速なディスクへのアクセスを避け，メモリに全てのデータを配置することで大量のリクエストを高速に処理する．また，GraphLab といったグラフ解析アプリケーションでは，大量のログからグラフをメモリ上に作り上げて各ノードの相関を算出する．近い将来，こうしたアプリケーションは広く普及し，クラウド環境などのデータセンタ上で稼働することが予想される．

２．研究の目的

本研究課題ではビッグメモリアプリケーションが稼働している仮想マシン(VM) を移送する方式について研究する．仮想マシン移送は VM を稼働させたまま別の物理ホストに移動することができ，VM の再配置を可能にする．これによって負荷分散や消費電力削減，物理マシンのメンテナンスが容易となるなどのメリットがあり，円滑なデータセンタ管理を促す必須の技術である．実際に Google などのデータセンタでは仮想マシン移送を使って大量の物理サーバや VM を管理している．しかし，これまでの仮想マシン移送手法ではビッグメモリアプリケーションが稼働している VM を従来通りに移送することは難しい．既存手法は数 GB 程度の仮想マシン (VM) を想定しており，これらのアプリケーションが稼働する VM を移送すると 1)．仮想マシン移送に要する時間が長くなりこれまでのような迅速な再配置が困難になる，2)．移送時間が長くなることに伴って CPU 使用量が増大してしまい移送に関わるホスト上で稼働している VM の性能を劣化させる，といった問題が生じる．これらによって，VM の再配置を用いたデータセンタの管理が従来通りにできず，仮想マシン移送を用いたデータセンタの管理が困難になってしまう．本課題では上記 2 点の問題点を克服し，ビッグメモリアプリケーションが稼働する環境においても，仮想マシン移送による恩恵を享受できるようにすることを狙う．

開発する方式はハイパフォーマンス分野において成熟しつつある GPU (Graphics processing units) プログラミング技法を活用する．GPU 上で画像処理以外の演算をさせる計算を GPGPU と呼び，CUDA といった開発用ライブラリを用いて流体力学シミュレーションや地震解析を高速に実行できる．本研究では GPGPU を駆使し，仮想マシンの移送処理の一部を GPU にオフロードする．具体的には，移送元の GPU では仮想マシンのメモリ内容の圧縮処理を，移送先の GPU ではメモリの伸張処理をする．これにより，転送に要するデータ量，および移送時の CPU 負荷増大の支配的要因であるネットワーク転送処理を大幅に削減できる．

３．研究の方法

初年度は，提案方式の詳細設計およびプロトタイプの実装を行う．さらに，本年度内にプロトタイプの完成度を高めることを目指す．実ソフトウェアに対する提案方式の有効性を確かめるために，オープンソースで広く利用されている仮想マシンモニタである Xen に組み込む形で実現する．その上で稼働させる OS カーネルは Linux とし，その上でメモリを確保した後にゼロで初期化したり，ランダムにアクセスしたりといった単純なワークロードを用意し，プロトタイプの挙動を細やかに確認していく．具体的には以下の 3 項目を実施する．

(1) ***Xen のアーキテクチャを対象に提案方式を設計する***：Xen では Domain 0 と呼ばれる特殊な仮想マシンが存在し，その上で稼働しているプロセスが仮想マシン移送を実行している．このプロセスに 1)．移送対象の VM のメモリを圧縮，伸長する処理を GPU 上で実行する機構，2)．CPU と GPU との非同期性を活かすために，移送対象の VM のメモリ取得部分，GPU 処理部分，メモリ転送部分をパイプライン化して CPU と GPU とが並列に動作できるようスレッド化する．GPU ドライバとして，Domain 0 上での稼働実績のある Nouveau という GPU ドライバを，また GPGPU ライブラリ実行基盤として Gdev を用いる．

(2) ***設計方針に従って実装する***：まずは，移送プロセスから Gdev を利用できるよう Xen を拡張した上で，その次に移送処理のスレッド化へと移る．一つのモジュールのコーディングが終了したら，すぐに次の実装に移らずにそのモジュールのテストを入念に行う．Xen や Linux のソースコードは数十万行を超えており，不具合が起きた際の原因特定が極めて困難である．

(3) ***プロトタイプの完成度を高めるために人工的なワークロードを用意してテストする:***プロトタイプのコーディングが一通り終了した段階で，すぐに実アプリケーションを実行するのではなく，人工的なワークロードを用意してプロトタ

イプの挙動を確認する．数十 GB のメモリを確保するプロセスから始まり，すべてを 0 で初期化するプロセス，確保したメモリ領域にランダムアクセスするプロセス，一部分を更新するプロセスといった単純な振る舞いをするプロセスを稼働させて，プロトタイプを用いて移送する．Linux や Xen を含め，現在のソフトウェアは様々なコンポーネントから形成されているため，これらとプロトタイプが競合することなく動作するかを確認する．

平成 28 年度には，前年度に作成したプロトタイプを用いて，実アプリケーションを稼働させたときの有効性を定量的に評価する．

(1) ***クラウド環境を模した環境を構築する：***VM 移送用計算機，ストレージサーバを利用して Amazon EC2 を模した環境を構築する．Amazon EC2 で提供されている VM のスペックを参考に同スペックの VM を複数用意する．特に R3 インスタンスと呼ばれる，メモリ容量の多い VM を一通り用意して稼働させる．

(2) ***実アプリケーションを稼働させてプロトタイプを用いて移送する：***上記作業で用意した VM 上に実アプリケーションを稼働させてプロトタイプを動作させる．実アプリケーションとしては，広く利用されている関係データベースである PostgreSQL，インメモリデータベースである Memcached を稼働させる．これらのアプリケーションについても単純な入力から始まり，最終的には現実的な入力を与えながらプロトタイプを動作させる．

４．研究成果

平成 27 度は提案方式の詳細設計およびプロトタイプの実装を行った．提案方式の有効性を確かめるために，オープンソースで広く利用されている仮想マシンモニタである Xen に組み込む形で実現した．その上で稼働させる OS カーネルは Linux を対象とし，その上でメモリを確保した後にゼロで初期化したり，ランダムにアクセスしたりといった単純なワークロードを用意し，プロトタイプの挙動を細やかに確認した．具体的には次のとおりである．Xen では Domain 0 と呼ばれる特殊な VM が存在し，その上で稼働しているプロセスが仮想マシン移送を実行している．このプロセスに 1).移送対象の VM のメモリを圧縮，伸長する処理を GPU 上で実行する機構，2)．CPU と GPU との非同期性を活かすために，移送対象の VM のメモリ取得部分，GPU 処理部分，メモリ転送部分をパイプライン化して CPU と GPU とが並列に動作できるようスレッド化する．第一段階として，圧縮アルゴリズムには Run Length Encoding を採用している．シンプルなワークロードとして，VM を Idle の状態，VM のメモリページを 0 で埋め尽くすプロセス (Best case)，VM のメモリページを全て異なるようカラーリングするプロセス(Worst case)を稼動させたところ，Best case で最大 4 倍，Worst case でも 1.2 倍のスピードアップに成功した．

平成 28 年度では，前年度作成したプロトタイプを用いて，実践的な圧縮アルゴリズムの実装，および実アプリケーションを稼働させたときの有効性を定量的に評価した．実験にはクラウド環境を模した環境を構築して実験した．具体的には，VM 移送用計算機，ストレージサーバを利用して擬似的なクラウド環境を構築し，その上で実験を行った．開発した圧縮アルゴリズムは 2 種類であり，圧縮型移送方式で広く利用されている，Delta Compression および Fixed-chunking Deduplication である．用意した VM 上に実アプリケーションを稼働させてプロトタイプを利用して移送を行なう．VM のコンフィギュレーションは，実クラウド環境である Amazon Elastic Cloud Computing にて提供されるインスタンスを元に決定している．実アプリケーションとしては，関係データベースサーバである PostgreSQL，インメモリデータベースである Memcached，またメモリインテンシブなワークロードを模して LMBench を稼働させ，実験を行なった．これらの結果として，Delta Compression においては最大 3.7 倍，Fixed-chunking Dedeuplication においては 16.7 倍の高速化に成功した．

５．主な発表論文等

（研究代表者、研究分担者及び連携研究者には下線）

〔学会発表〕（計 3 件）

1．河村裕太，山田浩史：データベース管理システムと協調した仮想マシン移送，第 137 回システムソフトウェアとオペレーティングシステム研究会，ホテルモントレ沖縄（沖縄県・国頭群恩納村），12 pages，2016．

2．直井由樹，山田浩史：GPU を用いたデータ圧縮型仮想マシン移送のアクセラレーション，第 13 回ディペンダブルシステム研究会，ポスター発表，ホテル水葉亭（静岡県熱海市），2015．

3．直井由樹，山田浩史：データ圧縮型仮想マシン移送における GPU アクセラレーション，第 134 回システムソフトウェアとオペレ

ーティング・システム研究会，ビーコンプラザ別府国際コンベンションセンター（大分県・別府市），14pages，2015．

６．研究組織

(1)研究代表者

山田浩史（Yamada，Hiroshi）
東京農工大学・大学院工学研究院・准教授
研究者番号：00571572