平成 30 年度 卒業論文概要			
所　　属	機械情報工学科	指導教員	光來 健一
学生番号	15237068	学生氏名	村田 時人
論文題目	複数ホストにまたがって動作する仮想マシンの障害対策		

1　はじめに

近年、クラウドコンピューティングのサービス形態の一つである IaaS 型クラウドでは、大容量のメモリを持つ仮想マシン（VM）も提供されるようになっている。例えば、Amazon EC2 の High Memory インスタンスは 12TB のメモリを持つ VM を提供しており、ビッグデータの解析やインメモリデータベース等に用いられている。VM が動作するホストをメンテナンスする場合は、VM を別のホストにマイグレーションする必要がある。一方、大容量メモリを持つホストをマイグレーションするために移送先ホストとして十分な空きメモリを持つホストを常に確保しておくことは、コスト面から望ましくない。そこで、複数のホストにメモリを分割して転送する分割マイグレーション [1] が提案されている。しかし、複数のホストを用いるため、ホストの障害の影響を受ける可能性が高くなる。障害対策としてチェックポイント・リストアと呼ばれる手法が用いられているが、従来のチェックポイントを行うと、ホスト間で大量のメモリのやりとりが発生するためオーバーヘッドが大きい。また、従来のリストアでは複数ホストに分割された状態で VM を復元することができない。

本研究では、複数ホストにまたがって動作する VM の柔軟で効率のよいチェックポイント・リストアを可能とするシステム D-CRES を提案する。

2　分割マイグレーション後のホストの障害

IaaS 型クラウドでは大容量メモリを持つ VM が提供されており、このような VM はビッグデータの解析などに利用されている。VM が動作しているホストのメンテナンスや負荷分散を行う際には、VM を別のホストにマイグレーションする必要がある。マイグレーションでは、VM のメモリを移送元ホストから移送先ホストへ転送するため、移送先ホストには VM のメモリよりも大きな空きメモリが必要となる。しかし、大容量のメモリを持つ VM に対して、十分な空きメモリを持ったホストを常に確保し続けるのはコストの面から難しい。そこで、VM を複数のホストに分割してマイグレーションする分割マイグレーション [1] が提案されている。分割マイグレーションでは、VM 本体とアクセスが予測されるメモリをメインホストに転送し、メインホストに入りきらないメモリをサブホストに転送する。

分割マイグレーション後はメインホスト上で VM が動作し、サブホストはメインホストにメモリを提供する。この VM は分割メモリ VM と呼ばれる。VM がサブホストに存在するメモリを必要とした場合には、当該メモリをメインホストに転送 (ページイン) する。同時に、メインホスト上のメモリの内、今後アクセスされないことが予測されるメモリをサブホストに

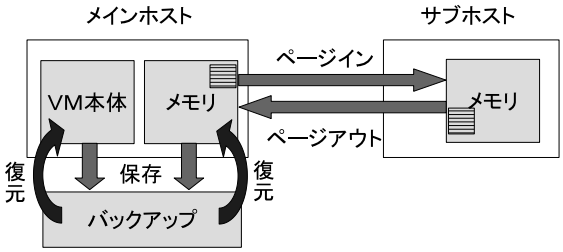


図 1: 従来のチェックポイント・リストア

転送 (ページアウト) する。このようなリモートページングを行えるようにするために、メインホストはネットワーク・ページテーブルを用いる。このテーブルはどのメモリがどのホストに存在するかを管理する。一方、サブホストではページ・サブテーブルを用い、サブホスト上のメモリを管理する。VM がメインホスト上に存在しないメモリにアクセスした際には、Linux の userfaultfd 機構を用いてアクセスを検出し、リモートページングの処理を行う。

このように、分割メモリ VM は複数のホストにまたがって動作するため、1 台のホスト上で動作する VM と比較して、ホストの障害の影響を受ける可能性が高くなる。従来の障害対策としては、チェックポイント・リストアと呼ばれる手法が用いられてきた。この手法は定期的に VM の状態をディスクにバックアップとして保存 (チェックポイント) し、障害発生時には最新のバックアップから VM を復元 (リストア) する。しかし、分割メモリ VM に従来手法を適用すると二つの問題が生じる。一つは、図 1 のようにチェックポイント時に大量のリモートページングが発生し、オーバーヘッドが大きいことである。これは、メインホスト上のメモリは従来通りに保存されるが、サブホスト上のメモリは一度メインホストにページインしてからメインホスト上で保存されるためである。もう一つの問題は、従来手法では複数のホストに分割された状態で VM を復元することができないことである。そのため、リストア時には十分な空きメモリを持ったホストが必要となる。

3　D-CRES

本研究では、複数ホストにまたがって動作する分割メモリ VM の柔軟で効率のよいチェックポイント・リストアを可能にするシステム D-CRES を提案する。D-CRES のチェックポイントでは、メインホストに存在するメモリはメインホストにおいて、サブホストに存在するメモリはサブホストにおいてディスクへの保存を行う。各ホストでメモリの保存を行うことで、サブホストとメインホスト間でリモートページングを発生させないようにすることができる。また、各ホストで並列にメモリを保存することで高速なチェックポイントが実現できる。D-CRES のリストアは、複数のホストに分割された状

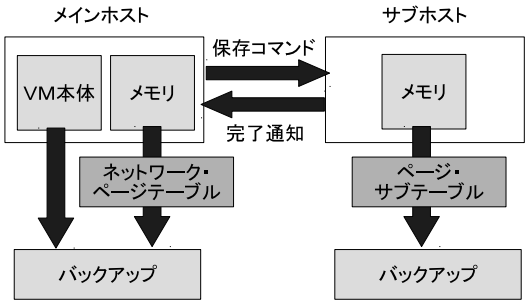


図 2: チェックポイント時の各ホストの動作

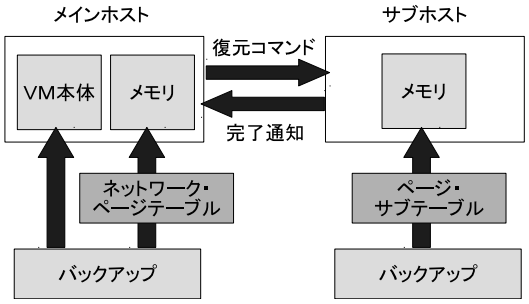


図 3: リストア時の各ホストの動作

態での VM の復元を可能にする。各ホストで保存したバックアップから、それぞれのホストごとに並列に復元を行う。

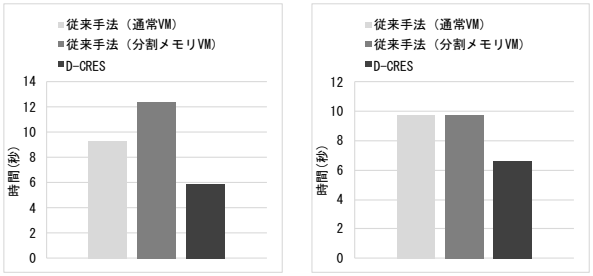
3.1 分割メモリＶＭのチェックポイント

分割メモリ VM のチェックポイント時の各ホストの動作を図 2 に示す。メインホストは VM を停止した後、サブホストのメモリの保存を並列に行うためにサブホストにチェックポイント・コマンドを送信する。次に、ネットワーク・ページテーブルを参照しながらメインホストに存在するメモリを保存する。サブホストに存在するメモリについてはアドレスとサブホスト ID のみを保存する。メモリの保存が完了すると、CPU やデバイスの状態などの情報を保存する。VM のすべての状態の保存が完了すると、サブホストからのコマンド完了通知を待って VM を再開する。

一方、サブホストはメインホストからチェックポイント・コマンドを受け取ると、ページ・サブテーブルを探索してサブホストに存在するメモリのアドレスとデータを保存する。すべてのメモリの保存が完了するとメインホストにコマンドの完了を通知する。

3.2 分割メモリＶＭのリストア

分割メモリ VM のリストア時の各ホストの動作を図 3 に示す。リストアを開始する前に、D-CRES はチェックポイント時と同じまたは大きなメモリを持つメインホストとサブホストを選ぶ。メインホストはサブホストのメモリの復元を並列に行うために、サブホストにリストア・コマンドを送信し、メモリとネットワーク・ページテーブルを復元する。メインホストのメモリについては、VM の対応するアドレスにデータを書き込み、ネットワーク・ページテーブルに登録する。サブホストのメモリについては、ネットワーク・ページテーブルにアドレスとサブホスト ID の対応を登録する。その後、復元したメモリを userfaultfd 機構に登録し、リモートページングが行えるようにする。登録が終わると CPU やデバイスの状態を復元し、サブホストからのコマンド完了通知を待つ。サブホストの復元が完了すると、リモートページングのためにサブホストとの接続を確立して VM を再開する。



(a) チェックポイント　(b) リストア

図 4: チェックポイント・リストアの性能

一方、サブホストはメインホストからリストア・コマンドを受け取ると、メモリを復元し、そのデータとアドレスの対応をページ・サブテーブルに登録する。復元が完了するとメインホストにコマンドの完了を通知する。

4 実験

2 台のホストで動作する分割メモリ VM に対して、D-CRES を用いたチェックポイント・リストアにかかる時間を測定した。比較のために、分割メモリ VM と 1 台のホストで動作する通常の VM に対して、従来手法のチェックポイント・リストアにかかる時間も測定した。実験には、Intel Core i7-7700 の CPU、8GB のメモリ、1TB の HDD、ギガビットイーサネットを搭載したマシンを 2 台用いた。各ホストではリモートページング用に変更を加えた Linux 4.4.169 および仮想化ソフトウェアの QEMU-KVM 2.4.1 を使用した。VM には 1GB のメモリを割り当て、分割メモリ VM のメモリはメインホストとサブホストに半分ずつ割り当てた。

図 4 に実験結果を示す。チェックポイント時間の測定結果より、D-CRES では従来手法で分割メモリ VM を保存するよりも 52% 高速に保存できることが分かった。通常 VM を保存する場合と比べても 36% 高速に保存できた。D-CRES を用いると各ホストで保存するメモリ量は半分になったが、メインホストで VM のその他の状態を保存するため、チェックポイント時間は半分にはならなかった。一方、リストアの実験結果より、D-CRES では従来手法より 30% 高速に VM を復元できることが分かった。

5 まとめ

本研究では、複数ホストにまたがって動作する分割メモリ VM の柔軟で効率のよいチェックポイント・リストアを可能にするシステム D-CRES を提案した。D-CRES では、各ホストで並列にメモリを保存・復元することができ、リストア後に複数のホストにまたがって VM を動作させることができる。

今後の課題は、VM を停止することなくチェックポイントを行えるようにすることである。また、リストアする先のホストの空きメモリがチェックポイント時と異なる場合には、VM の分割の仕方を変えるなど、より柔軟なリストアを可能にする。

参考文献

[1] M. Suetake, T. Kashiwagi, H. Kizu, and K. Kourai. *S-memV: Split Migration of Large-memory Virtual Machines in IaaS Clouds.* CLOUD 2018.