

サービスの停止時間を短縮する プログラム実行環境のプリコピー移送手法の評価

有蘭 里奈¹ 澤田 優真¹ 黒木 勇作¹ 横山 和俊¹ 谷口 秀夫²

概要: クラウドコンピューティング環境は、広域分散システムとして実現されることが多く、その効率的な利用のため、アプリケーションプログラムを別の計算機へ移送することが頻繁に発生する。我々は、プログラム実行環境を効率的に移送するため、ファイルのアクセス情報を基にしたプログラム実行環境の移送方式を提案している。提案方式では、プログラムのファイルアクセスを監視し、プログラムとプログラムが利用するファイル群を特定する。その後、特定したファイル群を転送する。この時、移送対象のファイル群を分割して転送するプリコピー移送手法を用いることで、プログラムの実行中断を抑制し、サービス停止時間を短くしている。本稿では、分割して転送するステップについて、実際のアプリケーションプログラムとしてカーネル make を用いて評価した結果を報告する。

1. はじめに

クラウドコンピューティング環境は、広域分散システムとして実現されることが多く、その効率的な利用のため、アプリケーションプログラム (以下 AP と略す) の実行環境を別の計算機へ移送することが頻繁に発生する。移送手法の代表的なものとしては、仮想マシンを移送する方法がある [1][2]。しかし、この手法では仮想マシン上の全ファイルを移送するため、目的の AP が必要とするファイル以外のファイルも移送先に転送させる必要があり、時間や移送サイズの面から非効率である。そこで、我々はファイルのアクセス情報を基にしたプログラム実行環境の移送方式を提案している [3]。

提案手法では、AP のファイルアクセスを監視し、AP だけでなく AP が利用するファイル群も特定する。その後、AP と共に特定したファイル群を転送する。このファイル群の転送の時、移送対象のファイル群を分割して転送するプリコピー移送手法を用いることで、プログラムの実行中断を抑制し、サービス停止時間を短くしている。提案手法の分割転送では、AP を動作させたまま転送を行う「プリコピーフェーズ」と、AP を停止させて転送する「最終コピーフェーズ」の2段階で全てのファイルの転送を行う。AP のサービス停止時間を短縮するためには、最終コピーフェー

ズでのデータ量 (以降、最終コピー量と呼ぶ) を削減する必要がある。プリコピーフェーズであらかじめファイルを転送しておくことで最終コピー量を削減することができるが、プリコピーフェーズにおいて転送済のファイルに対して更新が発生した場合、そのファイルを再送する必要がある。つまり、最終コピー量を削減するためには、最終コピーフェーズで転送するファイル群に含まれる再送ファイルを少なくする必要がある。このため、我々は、ファイルサイズとアクセス頻度から、ファイル転送順序を制御し、プリコピー量とファイルの再送を抑制する提案を行っている。本稿では、実際の AP の例としてカーネル make を用いて評価し、提案手法の転送ステップの最終コピー量について述べる。

2. プリコピー移送手法

提案するプリコピー移送手法では AP のファイル群へのアクセス情報を監視する特定ステップと、特定したファイル群の転送を行う転送ステップの2つのステップから構成される。

2.1 特定ステップ

特定ステップでは、転送対象となる AP のファイルへのアクセスを監視し、その AP が利用するファイル群を特定し転送対象とする。具体的には AP が発行する open システムコールを十分な時間監視し、ファイル名とアクセス種別 (Read-Only, Read-Write) を記録する。記録した情報から転送対象となるファイルを決定する際には、対象となるファイルを転送すると影響を受けるファイルとの共有関係

¹ 高知工科大学情報学群
School of Information, Kochi University of Technology

² 岡山大学大学院自然科学研究科
Graduate School of Natural Science and Technology,
Okayama University

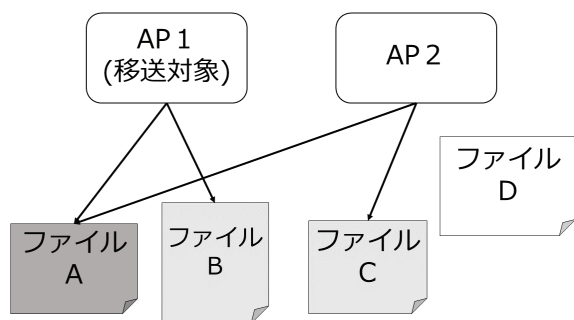


図 1 共有関係の特定

を追跡し、影響を受けるファイルも含めたファイル群を特定する。ファイルの共有関係を考慮した追跡の様子を図 1 を用いて説明する。図 1 に示すように、AP として AP1 と AP2 が実行されている。また、ファイル資源として、ファイル A、ファイル B、ファイル C、ファイル D がある。ここで AP1 が利用するファイルは、ファイル A とファイル B であり、AP2 が利用するファイルは、ファイル A とファイル C である。すなわち、ファイル A が 2 つの AP によって共有されている。ここで AP1 を転送する場合を考える。AP1 がファイル A に対し、何らかの書き込み処理を行う場合、AP2 がその情報に依存して動作している場合がある。そのため AP1 とファイル A、ファイル B のみを移送した場合、AP2 が AP1 からの情報を受け取ることができず、意図しない動作をする可能性がある。この可能性を考慮し、AP1 を移送する場合、追跡プロセスが、AP1、ファイル A、ファイル B に加え、ファイル A を共有する AP2 と、AP2 が利用しているファイル C も特定し移送対象とする。

2.2 転送ステップ

転送ステップでは、特定ステップで決定した移送対象のファイル群を転送する。このとき、移送対象をいくつかのグループに分割して転送する。本ステップでは、AP を動作させながら転送を行う「プリコピーフェーズ」と、AP を停止させて転送を行う「最終コピーフェーズ」の 2 段階のフェーズで全ファイルの転送を行う。これら 2 つのフェーズについて以下で説明する。

2.2.1 プリコピーフェーズ

プリコピーフェーズにおける具体的な転送例を図 2 に示す。図 2 では、ファイル群を、ある転送順序に基づいて、グループ A からグループ J までの 10 個のグループに分割している。プリコピーフェーズでは、このグループ単位で逐次転送する。プリコピーフェーズは AP を動作させながらファイルの転送を行うため、転送されたファイルに対して、再度更新処理が発生する場合がある。その場合は、更新された該当ファイルを転送対象に戻し、再送する。

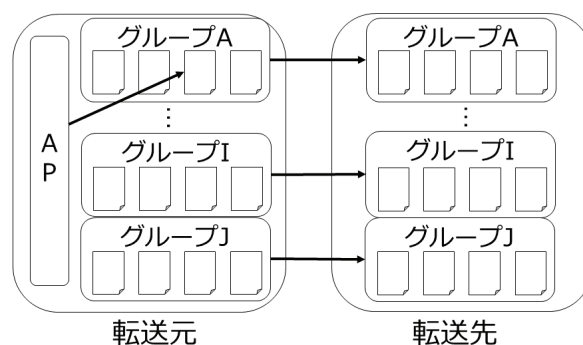


図 2 プリコピーフェーズ

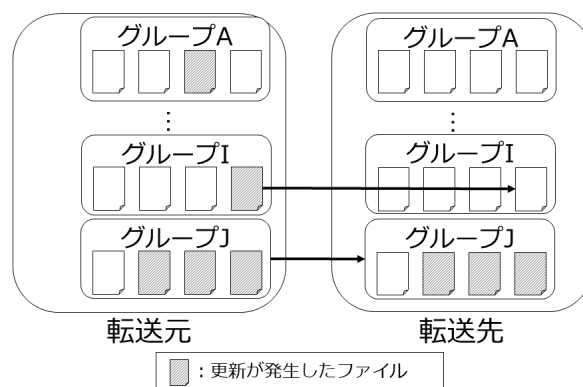


図 3 最終コピーフェーズ

2.2.2 最終コピーフェーズ

最終コピーフェーズにおける転送例を図 3 に示す。図 3 は、プリコピーフェーズの終了直後である。図 3 の網掛けで表わしているファイルは、プリコピーフェーズで一度転送されたものの、その後更新処理が発生し、転送対象に戻っているファイルである。最終コピーフェーズでは、これらの転送対象のファイルを全て、AP を停止して転送する。つまり、AP のサービス停止時間を短縮するためには、再送されるファイルを少なくし、最終コピー量を削減することが必要である。

3. 評価

実際の AP としてカーネル make を用いて、転送順序による転送結果の違いと提案手法の有効性をシミュレーションにより評価する。カーネル make で利用する AP 数とファイル数を表 1 に示す。転送速度は 5Mbps を想定し、Write システムコール 50 回をプリコピー 1 周期として、プリコピー 500 周期で最終コピーフェーズに移行するものとする。これを下記の 3 種類の転送順序で評価する。

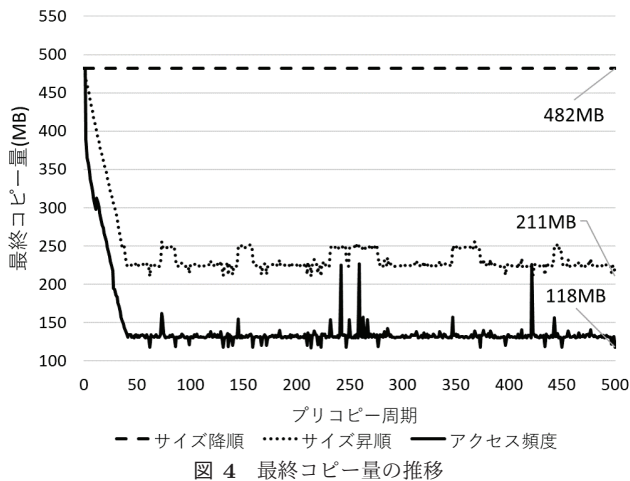


図 4 最終コピー量の推移

- ファイルサイズの降順 (サイズ降順と略す)
- ファイルサイズの昇順 (サイズ昇順と略す)
- アクセスされた回数の昇順 (アクセス頻度と略す)

表 1 カーネル make での AP 数とファイル数

対象 AP 数	対象ファイル数
88 個	13812 個

評価結果を図 4 に示す。横軸はプリコピーフェーズの周期数、縦軸は最終コピー量を表している。また、図 4 中の値は、各転送順序のプリコピーフェーズ終了時の最終コピー量を表している。結果としては、サイズ降順の最終コピー量が 482MB、サイズ昇順が 211MB、アクセス頻度が 118MB となった。また、全てのファイルの総サイズに対する最終コピー量の削減率は、アクセス頻度が約 64%、サイズ昇順が約 56%、サイズ降順が約 0%となった。サイズ降順の削減率が 0%になった理由として、アクセス頻度が高くサイズの大きいファイルの転送と再送を続けているため、最終コピー量の削減が見られなかったためである。サイズ昇順は、プリコピーを何周期か行った後に再送量がプリコピー量と同等またはそれ以上になってしまっているため、アクセス頻度ほどの削減が見られなかった。最も最終コピー量を削減することができたアクセス頻度は、アクセス頻度の高いファイルのプリコピーを後回しにしているため、再送量を抑えられた。また、提案手法の有効性に関しては、全容量の 482MB に対し最も削減できた場合の最終コピー量が 118MB となっており、約 64%の削減に成功しており、提案手法の有効性が確認できた。

4. おわりに

本稿では、カーネル make を用いて、特定ステップの評価及び、ファイルの転送順序による転送ステップの評価を行った。評価はファイルサイズの降順と昇順、ファイルへのアクセス頻度の 3通りの転送順序より、最終コピー量の削減率から提案手法が有効であることを明らかにした。結果

として、アクセス頻度順での転送において、最も削減率が高く、提案手法が有効であることが明らかになった。

謝辞

本研究の一部は、科研費科研費 (17K00107) の支援を受けて実施しています

参考文献

- [1] 広淵崇宏, マウリシオツガワ, 中田秀基, 伊藤智, 関口智嗣, “仮想マシンの超広域ライブマイグレーションにむけたベストエフォート型状態同期機構の試作”, 情報処理学会研究報告, 2012-OS-121(11), (2012).
- [2] 高橋一志, 笹田耕一, “差分転送を用いた高速な VM ディスクイメージ転送手法の提案”, コンピュータシステム・シンポジウム, Comsys2011, (2011).
- [3] 黒木勇作, 大西史洋, 横山和俊, 谷口秀夫, “サービスの停止時間を短縮するプログラム実行環境のプリコピー移送手法”, 情報処理学会研究報告, vol.2018-DPS-175, No.16, pp1-6 (2018).