

ヘテロジニアスなサーバ計算機環境における アプリケーションの特徴に基づいた仮想マシン再配置手法

藤井 淳† 和田 康孝‡ 近藤 正章†† 本多 弘樹†

† 電気通信大学 ‡ 早稲田大学 †† 東京大学

1 はじめに

近年、データセンタでは仮想化技術を用いて1台の物理サーバで複数の仮想マシン (VM) を動作させており、その際の課題の1つに VM の再配置が挙げられる。データセンタはヘテロジニアスなサーバ環境であることが多いため、VM 上で動作させるアプリケーションの処理性能は VM を配置するサーバによって異なると考えられ、VM 上で動作させるアプリケーションの特徴によって、その処理性能を最大にする配置サーバが変わり得るためである。

また VM 間ではサーバのメモリやキャッシュなどのリソースを共有しているため、同一のサーバに配置されている VM で動作させるアプリケーションの特徴の組み合わせによってもその処理性能は変わることが考えられる。

そこで本稿では、同じ物理サーバに配置する VM によって発生するリソース競合による影響、マイグレーションする VM とマイグレーション先サーバで既に配置されている VM によって発生するリソース競合による影響、アプリケーションとサーバ構成との相性の3つを考慮した VM 再配置手法を提案する。

2 関連研究

Xen や KVM などの仮想環境ではライブマイグレーションを用いて、ある物理サーバ上で稼働中の VM を停止させずに別の物理サーバ上に移動させることが可能であることから、VM の再配置を行うことができる。

VM 再配置の研究は数多く行われており、Xu ら [1] は、ライブマイグレーションにかかる時間と、同時に配置される VM 間の影響を数理モデルを作成することで見積もり、再配置を行う手法を提案している。

Novaković ら [2] は、動作中の VM をコピーして物理サーバ上で単体動作させた場合の性能を取得し、コピー元の VM でリソース競合が発生しているかを検知することで再配置を行う手法を提案している。

本稿では、ヘテロジニアスなサーバ構成を考慮する点で上記の研究とは異なる。

3 提案手法

同じ物理サーバに配置されている VM 間でリソース競合が発生すると VM の性能が低下するため、リソース競合が発生しているかを判断する必要がある。また、ヘテロジニアスなサーバ環境の場合には VM を配置するサーバによって VM の処理性能を最大にする配置サーバが異なるため、それを考慮した VM の配置が必要である。そこで運用前と同じ物理サーバに配置されている VM 間でのリソース競合の検知と、VM の処理性能を最大にする配置サーバの判別を行うためのモデルを作成し、運用時に用いる。

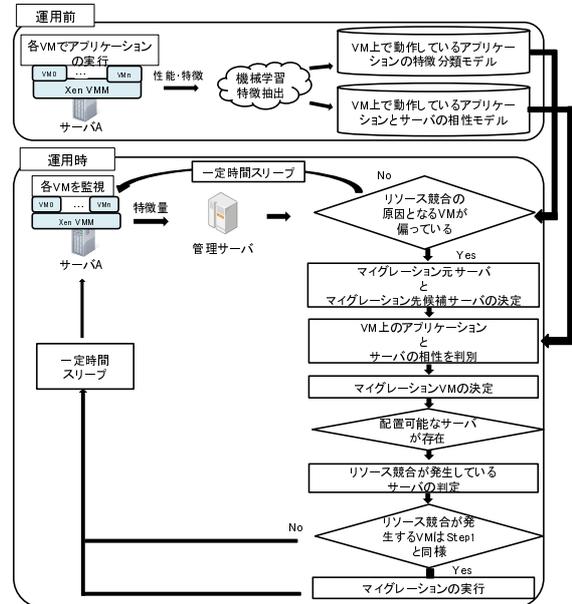


図 1: 提案手法の概要

同じ物理サーバに配置されている VM 間でのリソース競合の検知には、アプリケーションの特徴分類モデルを作成することで VM 上で実行中のアプリケーションの特徴を分類し、共有リソースを使用する特徴のアプリケーションを実行する VM が同じ物理サーバで複数実行されている場合には、リソース競合が発生しているサーバと判断する。

VM の処理性能を最大にする配置サーバの判別には、VM 上で実行するアプリケーションとサーバ構成の相性を用いる。本稿で相性とは、VM 上のアプリケーションを実行する際の相対的な MIPS 値を示し、高い MIPS 値を示すサーバはそのアプリケーションと相性が良いとする。相性の判別には、VM 上で動作しているアプリケーションとサーバの相性モデルを作成することで判断する。

提案手法の概要について図 1 に示す。まず、運用前に各サーバ構成上の VM においてアプリケーションを実行し、実行中のパフォーマンスカウンタの値を取得する。取得した値を特徴量として VM 上で動作しているアプリケーションの特徴分類モデルと、VM 上で動作しているアプリケーションとサーバの相性モデルを作成する。

そして運用時には、管理サーバが各物理サーバ上に配置されている VM のパフォーマンスカウンタ値を取得し、アプリケーションの特徴分類モデルを用いてリソース競合が発生するアプリケーションを実行する VM であるかを判断する。リソース競合が発生するアプリケーションを実行する VM の数が特定のサーバに偏っている場合に、リソース競合が発生するアプリケーションを実行する VM の数を均一になるように再配置を行い、マイグレーション先が複数存在する場合にはアプ

†Jun Fujii ‡Yasutaka Wada ††Masaaki Kondo †Hiroki Honda
†The University of Electro-Communications
‡Waseda University
††The University of Tokyo

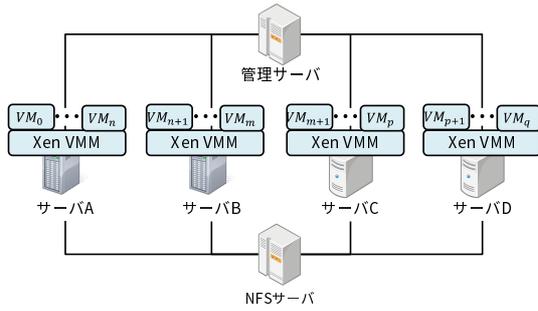


図 2: 作成した実験環境

表 1: 実験に使用したサーバのサーバ構成

サーバ名	搭載 CPU	コア数	CPU 周波数 (GHz)	L3 Cache (MB)	メモリ (GB)	メモリ周波数 (MHz)
サーバ A,B	Xeon X5675	6	3.06(3.46)	12	12	1333
サーバ C,D	Xeon E5640	4	2.66(2.93)	12	12	1066

リケーションとサーバ構成の相性が最も良いサーバをマイグレーション先サーバとする。

リソース競合が発生するアプリケーションを実行する VM の数を均一にする理由としては、サーバ間の性能差が大き過ぎなければ、リソース競合の原因となる VM の数を同数になるようにマイグレーションしたほうが、処理性能が高くなると考えられるためである。

マイグレーション VM はマイグレーション先サーバ毎に、リソース競合が発生している VM の中で現在実行中のサーバと比べてマイグレーション先での MIPS 値の増加率が最も大きいと予測されるサーバとする。

4 評価実験

4.1 評価方法

提案手法の有効性を評価するために、図 2 に示す環境を作成した。各サーバのサーバ構成を表 1 に示す。

各物理サーバには予め VM が配置されており、各 VM 上ではモデルの作成に用いていないアプリケーションのうち 1 つを実行し、初期配置から何も行わない場合と、提案手法を用いて再配置を行う場合との全 VM が実行終了するまでの時間を比較する。今回想定する共有リソースはメモリと共有キャッシュであり、モデルの作成と評価には SPEC2006 のアプリケーションを用いて行う。各 VM には 1CPU コアを割り当て、CPU コア数以上に VM は作成しないものとする。

VM 再配置の有効性は初期配置のリソース競合が発生する VM の偏りかたに依存するため、いくつかの初期配置において実験を行う必要がある。そこで、リソース競合が発生する VM の数が特定のサーバに大きく偏っている場合、リソース競合が発生する VM の数が偏っていない場合、VM 上で動作するアプリケーションが途中で変化し、リソース競合が発生する VM の偏り方が変化する場合について評価を行う。

ここでは、リソース競合が発生する VM の偏り方が変化する場合の結果のみを示す。

4.2 リソース競合が発生する VM の偏り方が変化する場合

VM の配置と実行するアプリケーションを図 3 に示す。黒色の四角がメモリバウンドなアプリケーションを示しており、灰色の四角は CPU バウンドなアプリケーションを示す。アプリケーション名が複数書いてあるものは、上からひとつずつ実行を行う。マイグレーションができるようにサーバ B とサーバ C は実行可能な VM の最大数より少なく動作させている。

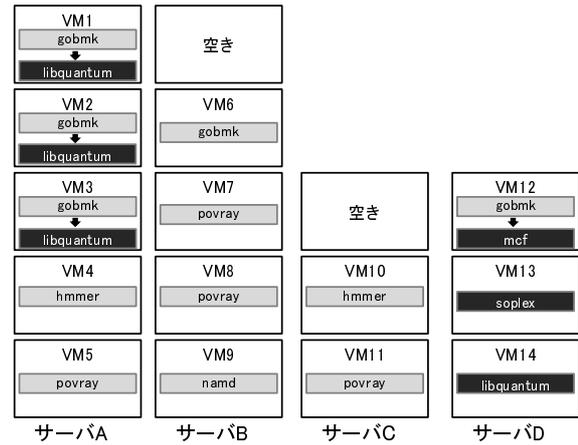


図 3: VM 上で実行するアプリケーションの特徴が変化する場合

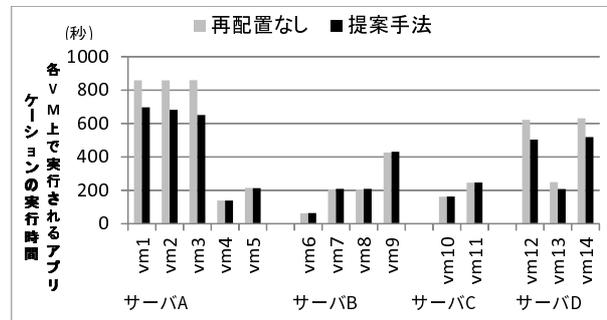


図 4: VM 上で実行するアプリケーションの特徴が変化する場合

リソース競合が発生する VM の偏り方が変化する場合の各 VM 上で実行されるアプリケーションの実行時間の結果を図 4 に示す。なにもしない場合と比べて提案手法を用いた場合には、最大で約 19% VM 上で実行されるアプリケーションの実行時間を短縮できた。

5 おわりに

同じ物理サーバに配置する VM によって発生するリソース競合による影響、マイグレーションする VM とマイグレーション先サーバで既に配置されている VM によって発生するリソース競合による影響、アプリケーションとサーバ構成との相性の 3 つを考慮した仮想マシン再配置手法を提案した。初期配置の際にリソース競合が発生する VM が偏っている場合、リソース競合が発生する VM の偏り方が変化する場合について実験を行い、その有効性を確認した。

今後の課題としては、より多くの種類の共有リソースを考慮した評価を行いたい。

参考文献

- [1] Fei Xu, et al. iAware: Making live migration of virtual machines interference-aware in the cloud. IEEE Transactions on Computers, Vol. 63, No. 12, pp. 3012–3025, Dec 2014.
- [2] Dejan Novaković, et al. DeepDive: Transparently Identifying and Managing Performance Interference in Virtualized Environments. Proceedings of the 2013 USENIX Conference on Annual Technical Conference, pp. 219–230, 2013