

# 仮想化におけるゲスト OS の統計情報を用いた動的メモリ割り当て

石原 吉晃<sup>†</sup> 芝 公仁<sup>†</sup><sup>†</sup> 龍谷大学理工学部

## 1 はじめに

近年、仮想化環境の普及により、1 台の計算機上で複数のオペレーティングシステム (OS) が動作する環境が増加している。仮想化環境でメモリの負荷が動的に変化する場合、最適なメモリ量は動的に変化し、静的なメモリ割り当てでは十分にメモリを活用することができない。この問題を解決するためには各仮想計算機 (VM) の最適なメモリ量を動的に決定し、それぞれに割り当てる必要がある。

仮想化ソフトウェアの Xen に実装されている xenballoon[1] では、VM が確保しているメモリ量を最適なメモリ量として扱い、動的メモリ割り当てを行っている。結果、xenballoon ではプロセスが確保したメモリ量に対応した割り当てを実現している。しかしこの手法では、ディスクキャッシュが考慮されていないため I/O 性能が劣化する場合がある。また、無条件にメモリの割り当てを行うため、物理メモリ量を超えた割り当てが行われる可能性がある。他にも文献 [2] では、xenballoon を改善してディスクキャッシュの確保を行っている。また、ディスクキャッシュの確保だけでなく全 VM の優先度を決定してメモリの配分を行っている。しかしこの手法では、キャッシュヒット率を取得するためにホストとゲストの OS を改変する必要がある。また、プロセスが使用するメモリ量は考慮されていない。

本稿では、プロセスのメモリ使用量とディスクキャッシュの両方を考慮した動的メモリ割り当ての手法を提案する。本手法では、VM が提供している情報のみを使用するため OS を改変する必要はない。

## 2 提案手法

提案手法はゲスト OS が提供している情報を使用して、各 VM に対してそれぞれ最適なメモリ量を決定する。また、それに基づき動的メモリ割り当てを行う。

### 2.1 最適なメモリ量

提案手法では、プロセスが使用するメモリ量とディスク I/O 性能を考慮して最適なメモリ量を決定する。最適なメモリ量  $W$  は、式 (1) のように、VM 上のプロセスが使用しているメモリ量と提案機構が持っている可変パラメータ Margin Size を合計した値である。

$$W = \text{使用中のメモリ量} + \text{MarginSize} \quad (1)$$

使用中のメモリ量は、物理メモリが未割り当てのページも含まれている。本手法では、使用中のメモリ量を考慮して VM に最適なメモリ量を設定することで、VM 上のプロセスが使用するメモリ量を割り当てる。

ディスク I/O への考慮は可変量のパラメータ Margin Size を使用する。この値は VM が十分なキャッシュを確保できるように提案機構が決定する。

### 2.2 構成

提案手法では、全 VM 上で Balloon Controller という機構が動作する。この機構は、前節で述べた Margin Size の制御と動的メモリ割り当てを行う。

Margin Size の制御は、VM に対して UP か DOWN の状態を付与し、その状態に応じて Margin Size の増減を行う。状態が UP の場合、キャッシュ量が増加している、または最近使用されたキャッシュが増加している場合に Margin Size を増加させる。そうでない場合は、状態を DOWN に変更する。DOWN の場合、キャッシュ量が増加していない、または最近使用したキャッシュ量が減少していない場合に Margin Size を減少させる。そうでない場合は、Margin Size を増加させ、状態を UP に変更する。以上の処理は状態変化の有無に関わらず 5 秒間隔で行う。

動的メモリ割り当ては、前節で述べたように VM 上のプロセスが使用しているメモリ量と Margin Size を合計して VM の最適なメモリ量を求めて、その値を用いてメモリ割り当てを行う。また、以上の処理は 1 秒間隔で実行する。

提案手法では、上記のように Balloon Controller が動作することで、プロセスが使用するメモリ量とディスク I/O を考慮して各 VM に最適な量のメモリを割り当てる。

Dynamic VM Memory Allocation for Guest Operating Systems using Statistics information

Yoshiteru Ishihara<sup>†</sup>, Masahito Shiba<sup>†</sup>

<sup>†</sup> Faculty of Science and Technology, Ryukoku University

表 1 物理計算機の構成

OS	Debian 9.2
Kernel	Linux-4.11.2
CPU	Core i5-6600 3.30GHz
CPU Core	4
Memory	4GB
HDD	2TB

表 2 仮想計算機の構成

OS	Ubuntu 17.10
Kernel	Linux-4.13.0
Virtual CPU Core	1
Virtual Memory	3072MB (上限値)

### 3 評価

提案機構を Linux カーネルと VMM である QEMU を使用した環境に対して実装した。動的メモリ割り当ては、QEMU に実装されている Balloon の機能を使用して行う。Balloon は、VM のメモリ上で膨張、収縮してメモリの動的割り当てを実現している。また、VM 上の情報は /proc/meminfo から取得を行った。前章で述べたプロセスが使用しているメモリ量は Committed\_AS、キャッシュ量は Cached、最近使用したキャッシュ量は Active(file) を使用している。

本章では、I/O 量に応じて提案機構が適切に Margin Size を制御できているか確認するために実験を行う。実験では、VM を 2 台使用し、各 VM 上でファイル読み込みを行うプロセスを動作させる。本プロセスは 1 万個のファイルから一定数のファイルを選択して読み出しを行う。各ファイルの容量は 1MB で、読み出し対象のとなるファイルの数は 10 分ごとに変化させる。読み出すファイルの数は 500, 5000, 2000 と 1000, 4000, 1000 の値を各 VM で使用する。また、ファイルの選択は一様分布乱数で行う。

表 1, 2 に実験で使用する環境を示す。実験環境では、物理メモリ量が 4096 MB で、各 VM は 3072 MB 以下のメモリが動的に割り当てられる。

図 1, 2 に実験の結果を示す。図中の opti\_size は VM 上で動作しているプロセスが読み込むファイルの容量で、MemTotal は VM に割り当てられたメモリ量である。図 1, 2 から読み出すファイル量が変化する約 600 秒でそのファイル量に対応して VM に割り当てられたメモリ量が増加している。また読み出すファイルが減少して必要なメモリ量が減少する約 1200 秒でも、遅れてはいるが不要なメモリが回収されて、VM のメモリ量が減少している。以上のことから I/O 量の増減に

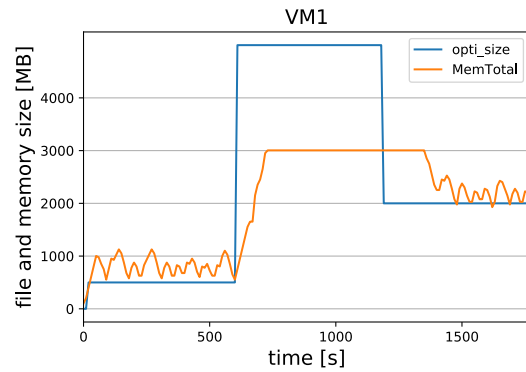


図 1 読み出すファイル量が 10 分ごとに 500, 5000, 2000 と変化する VM のメモリ量変化

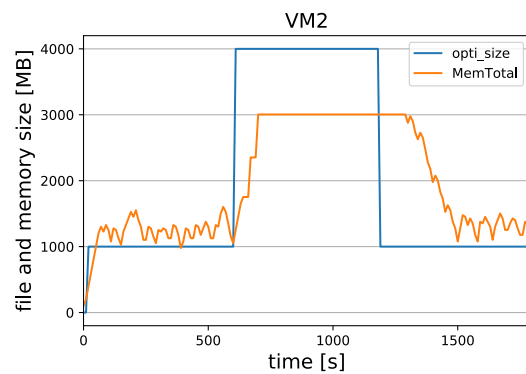


図 2 読み出すファイル量が 10 分ごとに 1000, 4000, 1000 と変化する VM のメモリ量変化

対応して Balloon Controller が適切に Margin Size を制御していることが確認できた。

### 4 おわりに

本稿では、キャッシュの変化に応じてメモリを割り当てる手法を提案した。提案手法により、I/O 量に応じて VM に適切なメモリ量が割り当てられる。

今後は、物理メモリ量を考慮した Margin Size 制御の検討を行い、よりメモリを活用できる動的割り当ての手法を実現を目指す。

### 参考文献

- [1] Stephen Spector, (2008) "MemoryOvercommit", <https://blog.xenproject.org/2008/08/27/xen-33-feature-memory-overcommit/>, (accessed 2017-12-15)
- [2] "日名川幸矢, 竹内洸祐, 山口実靖" 仮想化環境におけるキャッシュヒット率を考慮した VM メモリ割り当て", 研究報告マルチメディア通信と分散処理 (DPS) 2013-DPS-154(17), 1-7, 2013-03-07