

複数のアプリケーション動作時における

キャッシュヒット率を考慮した動的 VM メモリ割り当て

Dynamic Memory Allocation based on Cache Hit Ratio for Virtual Machines with Multiple Applications

坂本 雅哉†
Masaki Sakamoto山口 実靖†
Saneyasu Yamaguchi

1. はじめに

データセンタ等における、サーバの消費電力の増加や設置スペースの肥大化が問題となっている。この問題に対する解決策の一つとして、仮想化技術を用いて複数のサーバ OS を一台の物理マシンに集約する手法がある[1]。

仮想化環境では、仮想マシン(VM)を停止させることなく割り当てメモリ量を変更することが可能である。一つの物理マシン上に複数の VM が稼働しており、それぞれの VM の負荷が時間により変化する場合、静的なメモリ割り当てを行うとメモリを効果的に活用できないことになる。負荷変動に対応するためには、動的に VM メモリ割り当て量を変更する必要がある。

仮想化ソフトウェアの Xen には、動的に VM メモリ割り当て量を変更させる機能として xenballoon がある。しかし、xenballoon がメモリ割り当てを行うために考慮する値は、VM 内のプロセスが使用するメモリ量のみであり、ページキャッシュとして利用されるメモリの量を考慮していない。よって xenballoon では適切に I/O 性能向上を行えないと考えられる。

本研究では、Xen を用いた仮想化環境におけるメモリ割り当て量とキャッシュヒット率についての調査を行い、キャッシュヒット率を考慮した動的 VM メモリ割り当て量の最適化手法[2]の複数アプリケーション動作時における性能評価を行う。

2. xenballoon

xenballoon は、Xen が持つ機能の一つで、VM に割り当てるメモリ量を動的に変更する機能である。VM 上でデーモンとして動作し、ゲスト OS 内のプロセスの推定メモリ使用量(Committed_AS)を監視してホスト OS への要求メモリ量を調整する[1]。

3. キャッシュヒット率

本章にて、VM の基本性能調査を行う。キャッシュヒット率と I/O 性能の関係について調査を行うため、VM 上でファイルアクセスベンチマークを実行し、スループットとキャッシュヒット率を測定した。

3.1 キャッシュヒット率の調査

本節にて、キャッシュヒット率と VM 上の I/O アプリケーションのスループットの関係について述べる。両者の関係を調査するため、VM 上でランダムアクセスベンチマーク (FFSB) を実行した。ベンチマークでは、VM 内ファイルシステム上に 1MB のファイルを 5000 個 (5000MB) 作成し、これらのファイルに対してランダムリードにてアクセスを行った。

実験に用いた物理マシンとホスト OS の仕様は CPU Intel

Celeron G530 2.40GHz, OS CentOS 6.3, kernel Linux 2.6.32.57, 仮想化システム Xen 4.1.2, メモリ 8GB, HDD 1TB である。仮想マシンとゲスト OS の仕様はカーネル Linux 2.6.18.8, メモリサイズは動的に変更する。

測定結果を図 1 に示す。図 1 より、キャッシュヒット率が低い環境では、メモリ割り当て量を増やしても性能向上が小さいことが分かる。一方キャッシュヒット率が高い環境では、メモリ割り当てによる性能向上が大きいことが分かる。

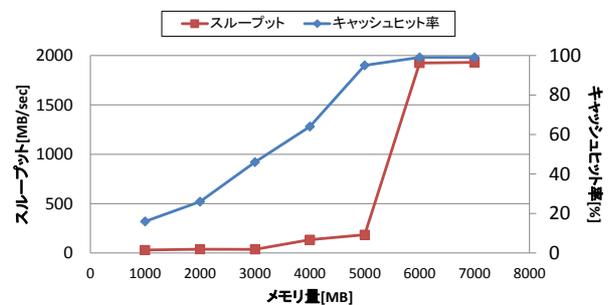


図 1 キャッシュヒット率とスループットの測定

4. キャッシュヒット率を考慮した動的 VM メモリ割り当て

本章にて、ゲスト OS のページキャッシュヒット率を監視し、それを元に VM にメモリ割り当てを行う手法[2]について述べる。

本手法では、各 VM のキャッシュヒット率の値をホスト OS にて集計し、以下の手順により VM メモリ割当量を決定する。

1. 全 VM のメモリ割り当て量を 10% 減少させ、これも再配分用メモリとする。
2. 上記 1 にて得た再配分用メモリをキャッシュヒット率が 100% 未満の VM に各 VM のキャッシュヒット率により比例配分する。

5. 性能評価

動的 VM メモリ割り当て手法[2]の有効性を検証するために性能評価実験を行った。本章にて、性能評価結果について述べる。

5.1 評価方法

実験では、Xen を用いて 1 台の物理マシン上に 3 台の仮想マシンを立ち上げ、全 VM 上でベンチマーク 4 種類 (FFSB, postmark, TPC-H, dd コマンドによるシーケンシャルリード) を実行し I/O 性能を測定した。各 VM でそれぞれのベンチマークを実行し、すべてのベンチマークが終了するまでの時間、各 VM のメモリ割り当て量とキャッシュヒット率の推移を観察した。実験に用いた物理マシンと仮想マシンの仕様は 3 章と同様である。

†工学院大学大学院工学研究科電気電子工学専攻
Electrical Engineering and Electronics, Kogakuin University Graduate School

5.2 I/O 性能の測定

図3から図7に動的VMメモリ割り当て手法と静的メモリ割り当て手法と xenballon の性能を示す。静的メモリ割り当て手法とは全 VM のメモリを 1706 [MB](均等割り当て)で固定したものである。図より静的メモリ割り当て手法と xenballon より動的 VM メモリ割り当て手法の方がすべてのベンチマーク終了までの時間が短いことがわかる。

図8にVMのメモリ割り当て量の推移を示す。グラフ内で用いている文字はTがTPC-H, FがFFSB, Dがddコマンド, Pがpostmarkを示し, Sがベンチマークデータサイズであり, 数字は1が約1[GB], 2が約2[GB], 3が約3[GB]を示している。縦線は太線がベンチマークソフトの切り替わりを, 細線がベンチマークデータサイズの切り替わりを示している。

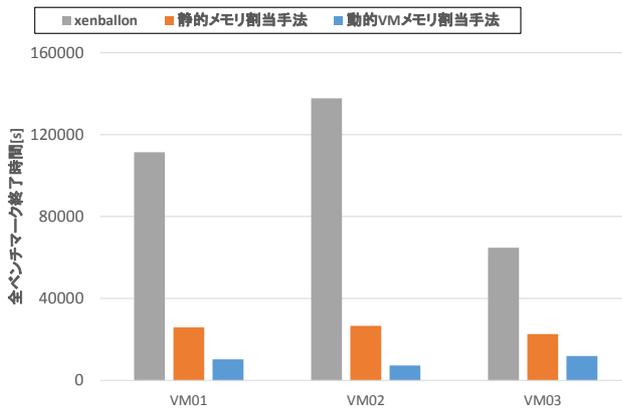


図3 性能評価 (すべてのベンチマークの合計)

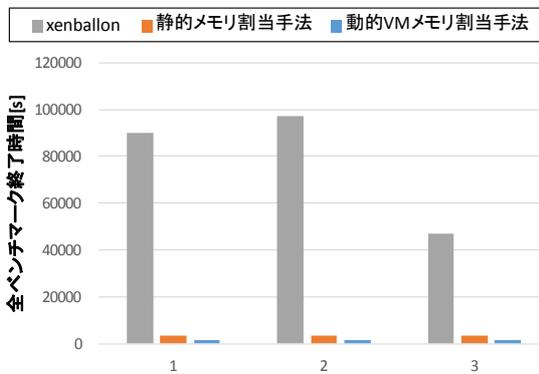


図4 性能評価 (FFSB)

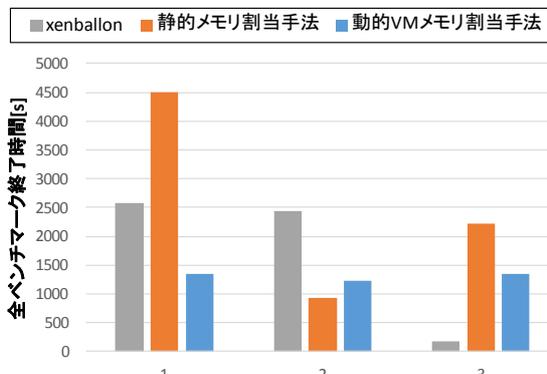


図5 性能評価 (dd コマンド)

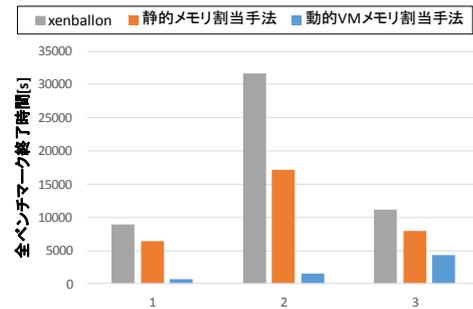


図6 性能評価 (postmark)

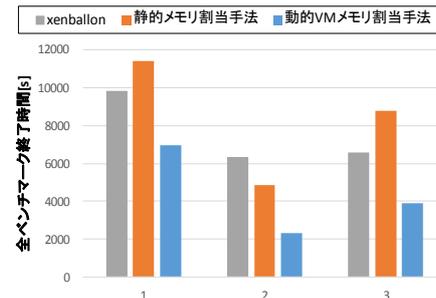


図7 性能評価 (TPC-H)

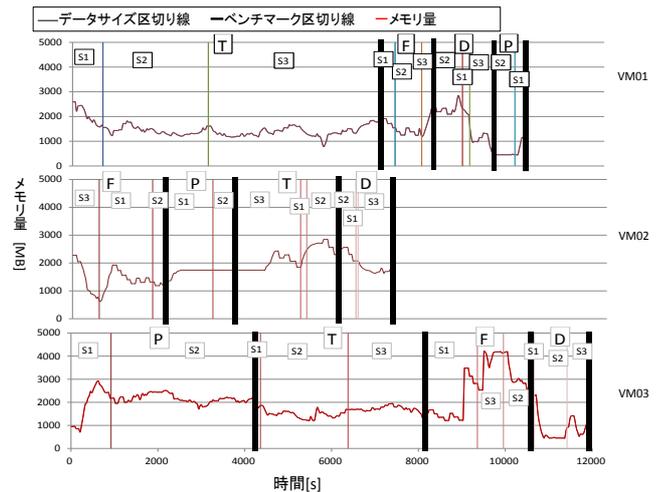


図8 ベンチマークデータサイズとVMのメモリサイズの推移

6 おわりに

本稿では, キャッシュヒット率を考慮した VM メモリ割り当て手法について述べ, 複数のベンチマークによる評価を行った. 評価の結果, 複数ベンチマーク環境において本手法が有効であることが確認された.

今後は, さらに多くのアプリケーションを用いての評価を行う予定である.

謝辞 本研究は JSPS 科研費 24300034, 25280022, 26730040 の助成を受けたものである.

参考文献

- [1] Stephen Spector, "Memory Overcommit", August 27, 2008, <http://blog.xen.org/index.php/2008/08/27/xen-33-feature-memory-overcommit/>
- [2] 坂本 雅哉, 山口 実靖, "読込み書込み比率を考慮した動的VMメモリ割り当てによるDBMS性能向上に関する一考察", 第13回情報科学技術フォーラム FIT 2014