

## 複数仮想マシン上におけるアプリケーションの性能予測のための階層性能モデルの検証 Layered Model for the Performance Evaluation of Applications on VMs

沼田 絵梨子<sup>†</sup> 木村 大地<sup>†</sup> 河津 正人<sup>†</sup>

Eriko Numata Daichi Kimura Masato Kawatsu

### 1. はじめに

システムインテグレーション (SI) では、顧客要求 (機能要求, 非機能要求) を十分に満たしたシステムを迅速に提供することが求められている。SI のプロセスの効率性と高品質性を向上させるためには、システム開発の早い段階で顧客の要求を評価できることが必要である。特に、性能は見積もりを間違えると手戻りが発生しやすく、設計段階で評価できることが重要になってくる。また、今後の SI の方向性として、クラウドを利用した拡大が考えられ、仮想化システムの性能評価がますます重要になってくる。

設計段階の仮想化システムの性能を評価する場合、仮想化によるオーバーヘッドを考慮することが、評価精度の高さを担保するには重要である。また、SI では低コストでのシステム実現のため、仮想マシン (VM) の最適な配置を求められるので、様々な配置パターンの性能を評価する必要がある。そのため、性能評価を迅速に行うには、これら数多くの性能評価を効率的に行えることが重要である。これより、仮想化システムを対象に設計段階の性能評価を行うためには、仮想化によるオーバーヘッドを考慮しつつ、配置パターン毎に必要な VM での性能評価を効率的に行えることが課題と考えられる。

設計段階のシステムの性能評価方法として、モデルを用いたアプローチがある。例えば、ホワイトボックスアプローチ (WB アプローチ: システムのメカニズムを模倣した性能モデルによる性能評価) や、ブラックボックスアプローチ (BB アプローチ: システムの実測値に基づいた性能モデルによる性能評価) が挙げられる。これらのアプローチにより仮想化システムの性能を予測している既存研究 ([1], [2]) も複数存在するが、BB アプローチを用いると、VM の構成が変更される毎に実測値の再計測が必要となるため、性能評価の効率化は難しい。また、もう一方の WB アプローチを用いた場合、仮想マシンモニタ (VMM) のエミュレーションや物理リソースでのリソース競合などの複雑なメカニズムを反映したモデルの作成には多くの時間が掛る。その上、これらのメカニズムの複雑さが増すほど、評価精度を担保したモデルの精緻化は困難になる。これより、どちらかのアプローチだけを用いるのでは、性能評価の効率化と評価精度を担保するという課題を解決することは困難であるといえる。

これらの課題に対して我々は、モデル化を階層に分けた“階層性能モデル”を用いることを提案している [3]。本稿では、複数 VM 上で動作するアプリケーションの性能予測 (ディスク I/O を対象) により、本アプローチの検証を行う。これにより、評価精度の高さを担保しつつ、効率的な性能評価方法として有効であることを示す。

### 2. 階層性能モデル

我々の提案しているアプローチ (図 1) は、システムを階層的な構造として捉えた性能予測方法 [4] の考え方を、仮想化システムに適用した。具体的には、仮想化システムを階層的な構造 (VM が動作している層: アプリケーション層, VMM と物理リソースを併せた層: ハードウェア層) と定義し (図 1 の (a)), 各層で個別に性能モデルを作成する (図 1 の (b))。そして、作成した各層の性能モデルを合成する (図 1 の (c)) ことにより、各 VM 上におけるアプリケーションの性能を予測する。

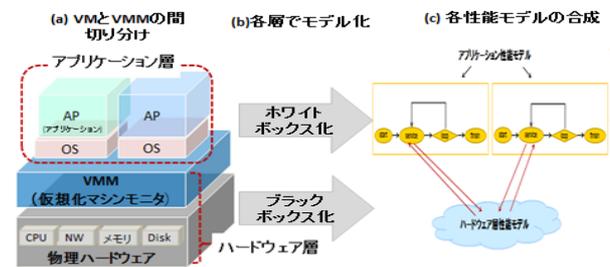


図 1. 階層性能モデルの概略

#### 2-1. 各層の性能モデルの作成方法

・アプリケーション層 (AP 層) : AP 層では、各 VM をホワイトボックスとして扱う。具体的には、VM 上で動作するアプリケーションの設計情報を用い、待ち行列やペトリネット等により AP 性能モデルを作成する。複数の VM が AP 層で動作している場合は、VM 毎に AP 性能モデルを作成する。また、HW 層と AP 層間で入出力のインターフェースを切り分けている。これにより、VM の構成変更には、変更部分のモデルのみの置換で対応することができる。

・ハードウェア層 (HW 層) : HW 層はブラックボックスとして扱う。具体的には、ベンチマークテストにより得た実測値を用い、統計分析や機械学習により HW 層性能モデルを作成する。性能モデルの作成には、VM 上で動作すると想定したアプリケーションのリソース消費量の範囲でベンチマークテストを行い、そこで得た実測値を用いる。これにより、効率的に様々なケースの VMM のエミュレーションや物理ハードウェアでのリソース競合の影響を学習させることができ、これらの複雑なメカニズムへの深い知識を必要とすることなく、モデル化を行うことができる。また、実測値を学習させておくことで、各 VM で動作するアプリケーションを変更した場合でも、HW 層と AP 層との整合性を保つことができる。

#### 2-2. 階層性能モデルによる性能評価方法

本モデルを用いた性能評価 (図 1 の (c)) を行うために、まず、AP 層で動作する VM の台数分の AP 性能モデルを用意する。この AP 性能モデルを用いて、離散事象型シミュレーションを行う。各 AP 性能モデルのシミュレーションで HW 層のリソースが必要な場合、シミュレーションのプロセスは HW 層性能モデルに移る。この時、HW 層性能モ

<sup>†</sup> NEC 情報・ナレッジ研究所

Knowledge Discovery Research Laboratories,  
NEC Corporation

デルへの入力は、AP性能モデルで算出した値とし、この値を用いてHW層性能モデルにより処理される。HW層性能モデルでの処理が完了すると、プロセスをAP性能モデルに戻す。この一連の流れを、AP層でのシミュレーションが完了するまで繰り返す。最終的にシミュレーションで得られた入出力値を基に、VM上で動作するアプリケーションの性能を予測する。

### 3. 階層性能モデルを用いた性能評価の検証

提案方法の検証のために、KVMを用いて仮想化環境を構築し、VMを2台(VM1, VM2)用意した。検証として、2台のVM上でアプリケーションを同時実行した場合の、各VMでのスループット(本稿ではディスクのリソース消費量)を予測した。ディスクI/Oを予測対象とする理由は、仮想化システムにおいては、通常、リソースの中でディスクが最もボトルネックになりやすいからである。

#### 3-1. ベンチマークテストに用いるアプリケーション

各VM上で実行させるアプリケーションとして、ディスクへのアクセスパターンをパラメータで変更させることができるものを作成した。具体的には、アプリケーションはディスクに対して $p$ の確率でランダムリード、 $1-p$ の確率でシーケンシャルリードとして動作し、 $p$ は0から1までの間で変化する。

#### 3-2. 階層性能モデルを用いたシミュレーション内容

・AP性能モデル：待ち行列を用いて、AP性能モデル(図2)を作成する。このAP性能モデルをVM台数分用意し、離散事象型シミュレーションにより解く。このシミュレーション時にハードウェアのリソースが必要な場合は、後述のHW層性能モデルを用いた処理を行う。

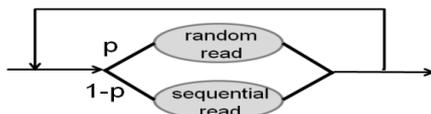


図2. アプリケーション性能モデル

・HW層性能モデル：HW層性能モデルとして、ベンチマークテストによる実測値の計測を行った。計測では、前述のアプリケーションを用い、ランダムリードだけが起る場合( $p=1$ )と、シーケンシャルリードだけが起る場合( $p=0$ )をベンチマークテストとして用いた。VM1とVM2上のアプリケーションの挙動は、ランダムリードとシーケンシャルリードの組み合わせ4通りを計測の対象とした。この計測で得た4つの実測値を用いて、HW層性能モデルを作成した。

#### 3-2. 実験

階層性能モデルを用いたシミュレーション結果と、VM1とVM2上で上述のアプリケーションを同時実行した場合の実測値とを比較した(アプリケーションは確率 $p$ を0から1の範囲で任意の間隔で変化させ実行した。また、VM1とVM2の確率 $p$ は同一のものとした)。実験では、アプリケーションがファイルサイズ4Gのテキストファイルを、レコードサイズ16KBと512KBでディスクにアクセスする場合をそれぞれ対象とした。VM1での実験結果を図3に示す(VM1, VM2では同傾向の結果が得られている)。結果は、レコードサイズが16KBの場合では平均誤差2.60%、最大誤差6.12% ( $p=0.02$ )で、512KBの場合では平均誤差1.76%、最大誤差は3.04% ( $p=0.06$ )であった。また、レコードサイズが小さい場合はシークタイムが長く、

スループットが低下しやすい。そのため、レコードサイズ16KBと512KBの実測値を比較した場合、16KBではランダムアクセスの傾向が強く表れている。このように、レコードサイズにより異なる傾向が表れる場合でも、誤差数%程度で予測することができている。

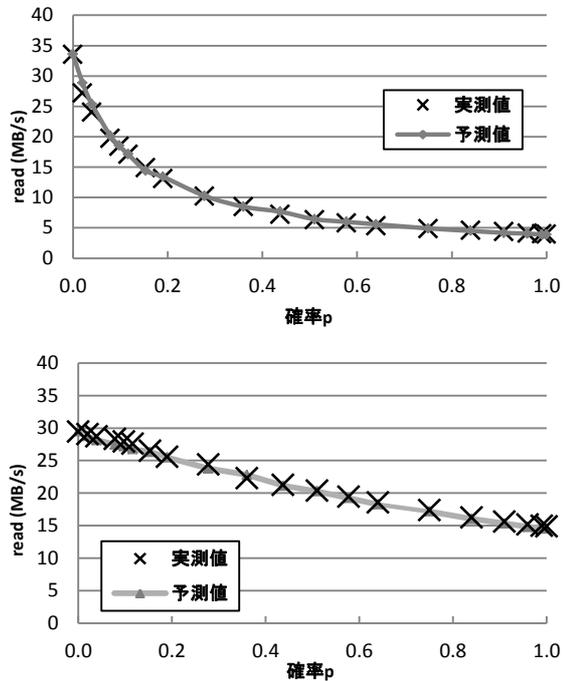


図3. 予測値と実測値の比較

(上図:レコードサイズ16KB, 下図:レコードサイズ512KB)

### 4. おわりに

本稿では、モデル化を2階層に分けた階層性能モデルの検証結果を示した。検証結果として、VM1とVM2上で同一のアプリケーションを動作させた場合、ディスクI/Oを誤差数%の精度で性能予測することができた。これより、提案手法は、論理的に効率的であり、かつ評価精度の高いアプローチとしての有効性があることを示せた。今後は、稼働させるVM台数を増やした実験を行うことで、より複数のプロセスが競合する場合の本性能評価方法の有効性を検証する予定である。

謝辞

本研究の一部は、総務省の委託研究「平成24年度災害時の確実な情報伝達を実現するための技術に関する研究開発(大規模通信混雑時における通信処理機能のネットワーク化に関する研究開発)」として実施中である。

#### 参考文献

- [1]Fabricio Benevenuto et al., "Performance Models for Virtualized Applications", ISPA 2006, vol.4331, pp.426-439 (2006)
- [2]Lei Lu, et al., "Untangling Mixed Information to Calibrate Resource Utilization in Virtual Machines", ICAC2011, pp.151-160(2011)
- [3]沼田絵梨子, 木村大地, 矢野尾一男, 複数VMの動作を考慮した階層構成モデルによる仮想化環境の性能予測, 第75回IPSJ全国大会(2013)
- [4]川西俊之, 紀一誠, 即時・待時混在システムの階層的な性能評価法, 2007-EVA-21, 2007, pp15-22