5D-01

仮想マシンのリソースをミリ秒精度で管理する手法の提案と評価

金子　雄†　　伊藤　俊夫†　　前川　智則†
†株式会社東芝　研究開発センター　ネットワークシステムラボラトリー

1　はじめに

クラウドコンピューティングを構成する技術の1つに，動的リソース割当て技術がある．Virtual Machine (VM) が必要とするリソース (CPU やメモリなど) を予測し，最小限のリソースを割当てることで，物理マシンのリソースを有効利用する技術である．これにより，物理マシン数の削減や，リソース管理の自動化による運用コスト削減などの効果が得られる．

一方，クラウドの適用先は拡大しており，制御システムへの適用も検討されている [1]．制御システムは一般的な Web アプリケーションよりも，動作のタイミングに対する要件が厳しい．例えば遠隔地からビル群のセンサデータを収集するアプリケーション [2](以降，クローラと記述) は，情報収集のための通信タイミングを1ミリ秒精度で保証する必要がある．

クローラのようなアプリケーションの動作のタイミングを VM 上で保証するためには，任意の1ミリ秒におけるリソース不足を回避するべきだと考える．そこで我々は，仮想マシンのリソースをミリ秒精度で管理する手法を提案する．

2　提案手法

提案手法を説明する．以降，提案手法のことを”shiba”というコードネームで記述する．

shiba は VM のリソース使用量が周期性を持つと仮定する．制御システムのアプリケーションは，周期的に動作することが多いためである．shiba はアプリケーションの動作周期を N 個のタイムスロットに分割する．タイムスロット幅 tw は最小で1ミリ秒を想定する．また，動作周期とタイムスロットには番号を割り振る．以降，動作周期番号が i，タイムスロット番号が j であるタイムスロットのことを $S_{i,j}$ と記述する．j は0から $N-1$ までの整数である．

shiba は各 $S_{i,j}$ においてリソース使用量 $C_{i,j}$ を計測し，記憶する．タイムスロット幅 tw の間に計測処理が完了しない場合は，計測した値から各タイムスロットの使用量を推定する処理を行う．そしてタイムスロット番号 j における使用量履歴の最大値 $maxC_j$ を，目標利用率で商算した結果を割当量 $A_{i,j}$ とする (式 1).

$$A_{i,j} = \frac{\max\{C_{x,j}, x = i-v, i-v+1, \cdots, i-1\}}{\text{目標利用率}} \quad (1)$$

v と目標利用率はチューニングパラメータである．v を大きくすると，より多くの使用量履歴から $maxC_j$ を計算するため，割当量は増加する．また，目標利用率を小さくすると割当量は増加する．shiba は計算した $A_{i,j}$ を，$S_{i,j}$ の開始直前に割当てる．

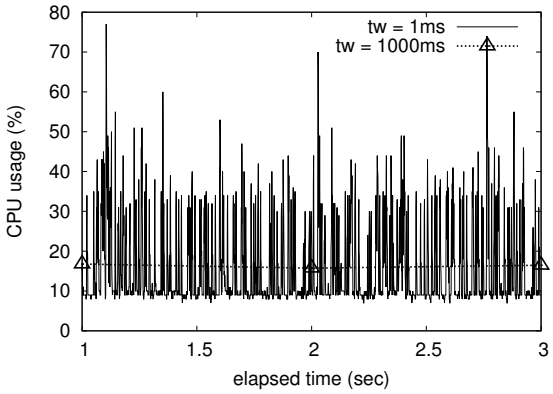

図 1: ある 2 秒間における CPU 使用量

3　評価

Xen*を対象とし，shiba を実装した．クローラを VM 上で実行し，shiba によるリソース管理を実施した．評価に用いた物理マシンの CPU は Intel Xeon E5-2603 (1.8GHz, 8core), OS は Ubuntu 14.04.1 とした．Xen はバージョン 4.4.0 を使用し，credit scheduler の *timeslice*† を1ミリ秒とした．クローラは 10 ミリ秒に1度の頻度でビル模擬装置と通信を行うように設定した．

1ミリ秒間隔で CPU 使用量を計測した結果を図1に示す．1秒間隔の使用量に変換した結果も示す．1ミリ秒間隔で計測することで，数ミリ〜数十ミリ秒ごとに発生する使用量のピークを検出できている．

目標利用率 (target utilization) を変化させ，動作保証率と CPU 利用率を評価した結果を図2と図3に示す．動作保証率とは，クローラの通信タイミングの誤差を

A Virtual Machine Resource Management on the Millisecond Time Scale
†Yu KANEKO　†Toshio ITO　†Tomonori MAEGAWA
†Network System Laboratory, R&D, TOSHIBA Corp.

*The Xen Project. http://www.xenproject.org
†Xen の credit scheduler は *timeslice* 時間ごとに VM の CPU 量 (credit) を増やす．デフォルトは 30 ミリ秒．

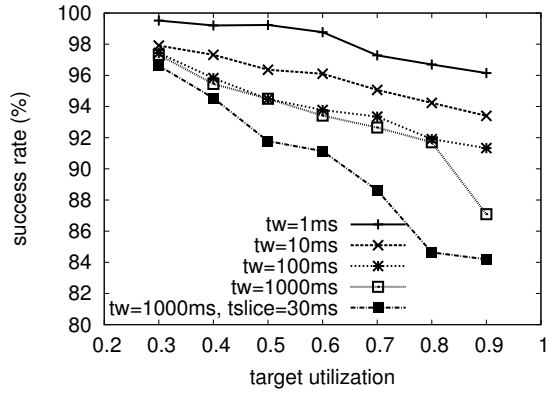

図 2: タイムスロット幅 *tw* と動作保証率の関係

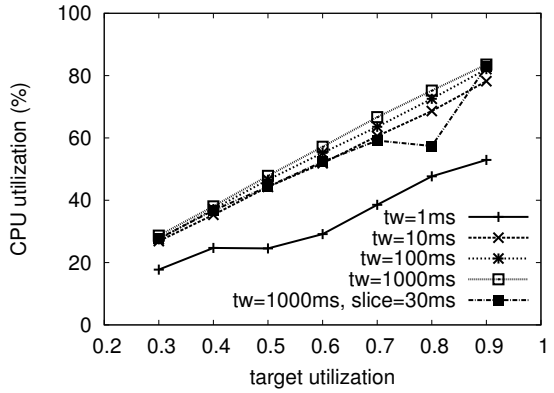

図 3: タイムスロット幅 *tw* と CPU 利用率の関係

1 ミリ秒以下に抑えられた確率である．CPU 利用率とは，割当量に対する使用量の比率である．各試行の評価時間は 30 分とした．

図 2 から，*tw* が小さいほど動作保証率が高いことがわかる．よって，ミリ秒精度でリソース管理を行うことは，ミリ秒精度の動作タイミングの保証に寄与すると言える．また，*timeslice* を 1 ミリ秒に変更したことも，動作保証率の向上に寄与している．一方，*tw* が 1 ミリ秒の場合に CPU 利用率は大きく低下した (図 3)．これは，CPU が不足した場合にクローラの動作タイミングが数ミリ秒遅延し，使用量のピークが本来とは異なるタイムスロットで観測されるためである．式 1 に示すように，shiba は各タイムスロットにおける使用量履歴の最大値を採用するため，ピークが観測されるタイムスロットが分散すると，多くのタイムスロットにおいて割当量が増加し，CPU 利用率が低下する．利用率を向上するための工夫が必要だと考える．

4 関連研究

shiba と同じくリソース使用量の周期性を利用するリソース割当手法に CloudScale[3] がある．タイムスロットにおける過去の平均使用量を割当量とする点が，shiba と異なる．CloudScale はリソース使用量の急増を検出した場合に，一時的に割当量を増やす工夫も加えている．ただし秒精度のリソース管理を想定しているため，ミリ秒精度に適用できるかは評価が必要である．

Peijie らは Xen を拡張し，リアルタイム性能を向上している [4]．hypervisor に独自拡張を加える手法は，システムのメンテナンス性を低下させる可能性がある．また，リアルタイム性能が向上しても，適切なリソース割当量を決定する問題は残る．

5 まとめと今後の課題

本稿では，仮想マシンのリソースをミリ秒精度で管理する手法を提案した．提案手法は制御システムの動作の周期性に着目し，ミリ秒間隔のリソース割当てを実現する．Xen を用いた評価により，アプリケーションの 1 ミリ秒精度の動作タイミングを 95%以上の確率で保証できることを確認した．

今後は，利用率向上に向けた検討，複数の VM を対象する場合の検討，Xen 以外の hypervisor への適用，既存研究との比較評価などを行う．

参考文献

[1] O. Givehchi, H. Trsek, and J. Jasperneite. Cloud computing for industrial automation systems - a comprehensive overview. In *Emerging Technologies Factory Automation (ETFA), 2013 IEEE 18th Conference on*, pp. 1–4, Sept 2013.

[2] 伊藤俊夫, 米良恵介, 金子雄, 松澤茂雄. 通信エンドの負荷ピークを低減するためのビル設備情報収集スケジュール作成方法. 電子情報通信学会技術研究報告. IN, 情報ネットワーク, Vol. 111, No. 197, pp. 77–82, aug 2011.

[3] Zhiming Shen, Sethuraman Subbiah, Xiaohui Gu, and John Wilkes. Cloudscale: Elastic resource scaling for multi-tenant cloud systems. In *Proceedings of the 2Nd ACM Symposium on Cloud Computing*, SOCC '11, pp. 5:1–5:14, New York, NY, USA, 2011. ACM.

[4] Peijie Yu, Mingyuan Xia, Qian Lin, Min Zhu, Shang Gao, Zhengwei Qi, Kai Chen, and Haibing Guan. Real-time enhancement for xen hypervisor. In *Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on*, pp. 23–30, Dec 2010.