

仮想サーバの最適な配置方式の提案と評価

柳沢 満[†] 竹村 俊徳[‡]

^{†‡}日本電気株式会社 システム基盤ソフトウェア開発本部

1.はじめに

近年の企業システムでは稼動するサーバの台数が増加する傾向にあり、消費電力が高まっている。また、一般的なサーバの CPU 稼働率は平均 3 割程度と言われており、有効に使用されていないのが現状である。ユーザの要求性能を満たしながら、必要最小限のサーバ台数でシステムを自律的に稼動させる方法を考案した。

2. 課題と解決方法

負荷変動に応じた柔軟なサーバ活用が困難な原因の一つは、システムごとの実行環境 (OS、各種ミドルウェア等) の違いにより、サーバの再構築が容易にできない点である。このため、高負荷なサーバはシステム全体のサービスレベルを低下させる要因となり、低負荷なサーバはリソースの利用効率低下を招く。従来、負荷に応じたスケールアウト制御は行われているが、業務システム間で空きリソースを共有することはできなかった。この課題に対し、業務システムを仮想サーバ (VM) 上で稼動させることを前提とし、その上で各 VM が必要なリソースを確保しつつ、物理サーバ (SV) の空きリソースを削減できる、VM の最適な配置方式を提案する。

3. 自律制御機能

自律制御機能は、各 SV、VM の負荷変動を監視し、SV の高負荷・低負荷の状態に応じて、VM を移動させる。VM の移動先の SV は、VM 移動後に VM の負荷が集約された状態を維持しつつ、負荷の上限を超えない範囲で選択される。この VM 移動を負荷変動に応じて自律的に実行することにより、各 VM の処理能力を下げることなく、サーバ全体の利用効率を向上させる。

4. VM 移動による負荷変動のモデル化

物理サーバ SV_{src} 上で稼働する仮想サーバ VM を物理サーバ SV_{dest} に移動した場合の負荷は、以下の計算式に従って算出する。ここでリソース R を CPU, Memory, NW I/O, HDD I/O とし、 $U'(SV_{src}, VM, R)$ を VM 移動後の SV_{src} のリソース R の使用率、 $U'(SV_{dest}, VM, R)$ を VM 移動後の SV_{dest} のリソース R の使用率、 $U(SV, R)$ を SV のリソース R の使用率、 $C(SV, R)$ をリソース R の容量、 $U(SV_{src}, VM, R)$ を SV_{src} 上の VM のリソース R の使用率とすると、VM 移動前と移動後の SV のリソース使用率の変化を次式で定義する。これを図 1 に示す。

$$U'(SV_{src}, VM, R) = U(SV_{src}, R) - U(SV_{src}, VM, R)$$

$$U'(SV_{dest}, VM, R) = U(SV_{dest}, R) + U(SV_{src}, VM, R) \cdot \frac{C(SV_{src}, R)}{C(SV_{dest}, R)}$$

リソース使用率の分散 $V(SV_{src}, SV_{dest}, VM, R)$ は、次式で定義できる。

A method for optimal allocation of virtual machines and its evaluation
Mitsuru Yanagisawa[†] Toshinori Takemura[‡]

^{†‡} NEC Corporation

$$\frac{\{U'(SV_{src}, VM, R) - \bar{U}(VM, R)\}^2 + \{U'(SV_{dest}, VM, R) - \bar{U}(VM, R)\}^2}{2}$$

$$\bar{U}(VM, R) = \frac{U'(SV_{src}, VM, R) + U'(SV_{dest}, VM, R)}{2}$$

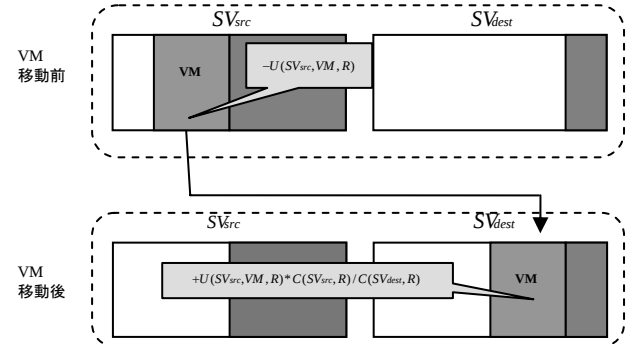


図1: VM移動によるリソースの使用率の変化

さらに、SVには複数のリソースが存在するので、リソース R に対する重み $w(R)$ を考慮し、VM 移動後の負荷のばらつき度合いを示す評価関数を次式で定義する。

$$f(SV_{src}, SV_{dest}, VM) = \sum_{R \in \{CPU, Memory, NW I/O, HDD I/O\}} w(R) \cdot V(SV_{src}, SV_{dest}, VM, R)$$

5. VM 移動における最適な候補の選択方式

高負荷・低負荷の状態の SV から VM を移動させる場合、移動対象の VM と移動先の SV を決定する必要がある。自律制御機能は、各移動元 SV ごとに、移動元 SV 上の VM と移動先 SV のペアを候補として、以下の2条件で絞り込みを行い、残った各候補に対して、前述の評価関数による評価値を求めることにより、候補の順位付けを行う。

制約条件(必須条件)	物理的リソース制約条件、VM 共存条件
フィルタリング条件	ユーザのリソース要求条件等
評価関数	上記の条件を満たす候補が複数存在する場合に、評価関数で評価値を算出

順位の評価基準は2種類ある。(1) 移動元 SV が低負荷の場合は、移動先 SV の負荷の上限を超えない範囲で評価値が最大になる移動先候補を選択することにより、1 台の SV に VM を集約する。これにより、リソース利用効率が高められる。(2) 移動元 SV が高負荷の場合は、移動先 SV の負荷の上限を超えない範囲で評価値が小さい候補を優先することで、両 SV の負荷が均等に近づく(平衡化する)。これにより、VM の集約度を上げつつ高負荷を解消できる。

6. シミュレーションによる評価

評価実験は、上述の選択方式を実装したシミュレータで行った。シミュレータは、事前に作成した各 VM の1日当たりの負荷変動データを入力として、上述の選択方式によってシミュレータ上で VM を移動させた場合の各 SV の負荷の変化を算出するものである。シミュレーション

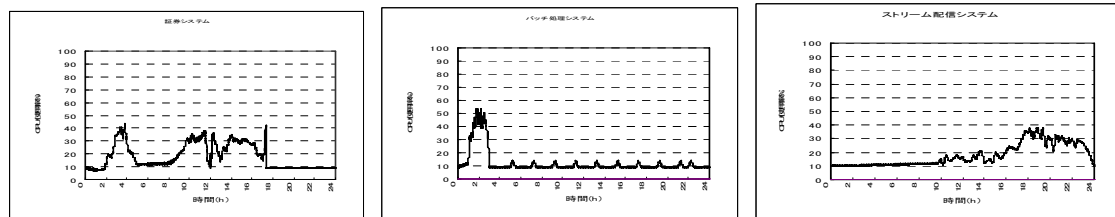


図2：VMのCPU使用率の負荷変動パターン(シミュレータへの入力データ)

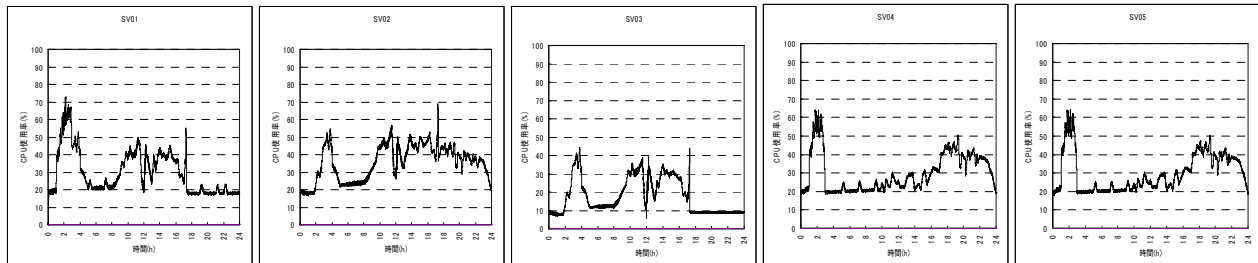


図3：VM移動を行わない場合の各SVのCPU使用率の変化

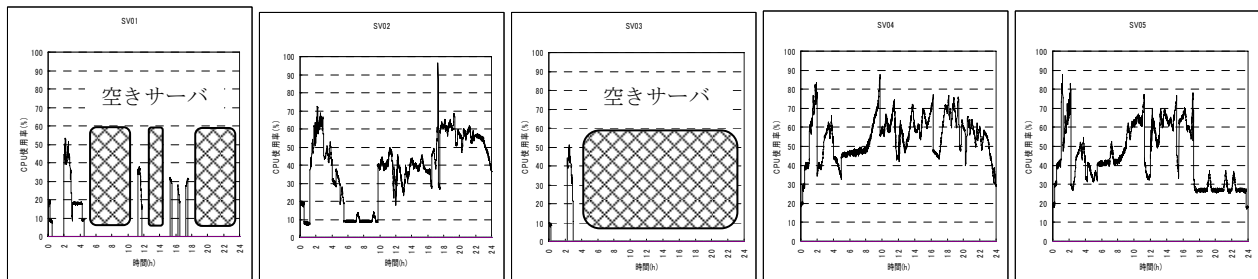


図4：提案方式を適用し、VM移動を行った場合の各SVのCPU使用率の変化

は、5 台の SV (SV01～05) 上で 9 台の VM を移動させるケースで行った。VM の初期配置は、4 台の SV 上に各 2 台の VM を配置し、残りの 1 台の SV 上に 1 台の VM を配置した。VM は、図 2 に示す 3 種類の異なる負荷パターンをとる。VM の負荷変動データは 30 秒間隔の CPU 使用率であり、9 台の VM は、3 台ずつが同じ負荷パターンをとるものとする。各グラフの横軸は時間(h)、縦軸は CPU 使用率(%)である。提案方式を適用せず、VM 移動を行わない場合の各 SV の CPU 使用率の変化は図 3 のようになり、各 SV の CPU 使用率は低く CPU が無駄になっている。これに対し、提案方式を適用して自律的に VM 移動を行った場合は図 4 のようになり、VM 移動の結果、VM の負荷が 3 台の SV に集中し CPU 使用率が向上している。同時に SV01 と SV03 では、SV 上に VM が存在せず空きサーバとなる時間帯(網掛け部分)が生じている。

7. 考察

サーバのリソース削減率を (SV 上に VM が存在しない時間の総和)/(全 SV の稼働時間の総和) として計算すると、

$$153720 / (86400 \times 5) = 0.355 \approx 36\%$$

となる。VM 移動を行わない場合は、常に 5 台のサーバが必要であるのに対し、提案方式では、83%の時間帯で 3 台のサーバでの処理 (= 余剰サーバの 4 割削減) が可能となっている。

8. 終わりに

本提案方式を使った VM 配置の最適化により、5 台中 3 台のサーバで稼働することが可能となり、1 サーバ当たり

の利用効率を向上させることができた。加えて、空きサーバの電源を切ることにより省エネルギーも実現できる。今後の課題としては、SV の削減率をさらに高めるための VM 移動時の候補の決定方式の改善と、急激な負荷上昇にも対応可能な負荷変動の予測方式の実現が挙げられる。引き続き、実機を用いた環境において、VM 移動時の負荷等を考慮した場合のサーバ削減効果を実証する予定である。

謝辞

本研究は、独立行政法人新エネルギー・産業技術総合開発機構 (NEDO) から委託を受け実施している「エネルギー使用合理化技術戦略的開発/エネルギー使用合理化技術実用化開発/サーバの利用効率向上による省エネルギー化の研究開発」の成果である。ここに感謝の意を表す。

参考文献

M. Nuttall, M. Sloman : Workload characteristics for process migration and load balancing, Proceedings of the IEEE International Conference on Distributed Computing Systems, ICDCS'97, May 1997.