

可用性とレスポンスタイムを考慮した VM の再配置手法

1275099 柏原 星司 【分散処理 OS 研究室】

VM Relocation Method Considering Availability and Response Time

1275099 Joji KASHIWABARA 【Distributed System and Operating System Lab.】

1 はじめに

近年、クラウドコンピューティングが急速に普及している。クラウド事業者は安定したサービスを提供するにあたって SLA として稼働率を保証するが多い。また近年、クラウドアプリケーション提供者は SLO というサービスが満たすべき目標を設定するが多く、SLO においてレスポンスタイムが設定されるが多い。このように可用性を満たしながらレスポンスタイムを最小化するような手法が求められている。菊森ら [1] は、可用性を考慮しながら電力の最小化を目指す動的配置手法について検討している。この研究において、ホストの高負荷による電力上昇について考えられている。しかし、高負荷によるレスポンスタイムの上昇については考慮されておらず、レスポンスタイムについての評価も行われていない。そこで、本研究では可用性を考慮しながら、レスポンスタイムの最小化を目指す再配置手法について提案する。

2 システムモデル

2.1 可用性モデル

先行研究と同様に、可用性モデルとしてデュプレックスシステムを前提とする。デュプレックスシステムは一般に現用系と予備系が N 対 1 の構成として考えられているが、本研究では 1 対 1 構成を前提とする。また今回は先行研究と異なり、待機方式による差は生じないため、定期的に現用系と予備系が同期をとるようなホットスタンバイ構成を想定する。

2.2 レスポンスタイムモデル

レスポンスタイムは一般に待ち行列理論に基づいて近似できることが知られている。特に、複数の VM が 1 つのホストに対してリクエストを送るような環境では、M/M/1 モデルを用いてレスポンスタイムが近似できることが知られている。ここで、本研究ではすべてのホストが同機種である環境について考える。つまり全てのホストのサービスは等しいサービス率を持つような環境を想定する。また、すべての現用系 VM、予備系 VM は CPU の平均到着率、I/O の平均到着率をそれぞれ持ち、CPU と I/O の負荷率の和に応じてレスポンスタイムが

増加するものとする。以上のようなシステムモデルを基にシミュレーションを行う。

3 提案手法

レスポンスタイムへ与える上昇量が同程度となるような到着率を求めることによってレスポンスタイムの低下を目指す配置手法を提案する。

3.1 制約条件

菊森らの研究 [1] では制約条件を 3 つ設けており、その中の可用性に影響する制約条件として以下の 2 つが設けられていた。本研究においても想定する可用性モデルは同一であるため、同様に以下 2 つの制約条件を設ける。

1. あるサービスについて現用系の VM と予備系の VM が同一のホストに配置されないこと
2. あるサービスの現用系と予備系の配置の組み合わせが任意の他のサービスの配置の組み合わせと同一にならないこと

3.2 基本的な考え方

レスポンスタイムへ与える上昇量が同程度となるような配置にすることによって、CPU、I/O 片方のリソースの偏りによるレスポンスタイムの増大を防ぎながら、レスポンスタイムへの影響が低い到着率を多く占めることができる。よって可能な限り上昇量が同程度となるような配置を目指す。具体的に、I/O のレスポンスタイムへ与える影響が CPU の 2 倍である場合を考える。この時それぞれの到着率がレスポンスタイムへ与える上昇量は図 1 のようになる。この時、図中 a を平均到着率として I/O の到着率を a から少し減少させた値 a' について考える。I/O の到着率を a' とした場合 I/O の上昇量が CPU の上昇量が同程度となるのは到着率が b の時である。ここで、I/O 到着率を a' 、CPU 到着率を b とした配置にすることによって多くの到着率を占めながらレスポンスタイムに対して効率的に配置することができる。

その一方、レスポンスタイムへ与える上昇量が同程度になるような配置ができなかった VM については I/O の到着率が平均と比べ少し上昇するが、CPU の到着率を平均と比較して著しく低下させることができる。そのような VM については、できるだけ CPU と I/O が平滑

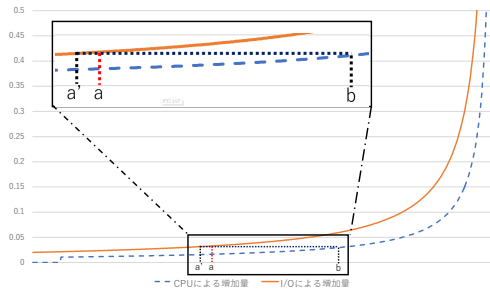


図1 レスポンス増加量に対する勾配

表1 シミュレーションパラメータ

パラメータ	値
ホスト数	100 台
サービス発生間隔	20 分
サービス発生率	15%
サービスレート	1500
平均到着率	350
シミュレーション時間	10 日
再配置間隔	2h

になるように配置することによって CPU, I/O の偏りを防ぐことができ、レスポンスタイムへの影響を少なくできる。このような配置は、まず VM を CPU, I/O それぞれ降順にソートした配列を用いて、順にペアリングを行う。次に、作成したペアを総負荷の降順でソートし、グリーディ法によって配置することによって平滑的な配置を目指す。

3.3 評価

提案する配置手法のレスポンスタイムと障害復旧時間について評価する。比較対象に加えて以下の3つの手法を実装し評価する。

1. 提案手法による再配置
2. 再配置なし
3. グリーディ法による再配置
4. 遺伝的アルゴリズムによる再配置 (制約条件なし)

ここで遺伝的アルゴリズムは可用性による制約をなくしたものを用意し、疑似的な最適解として評価を行う。また、シミュレーションで用いるパラメータを表1に示す。

ここでは横軸をホストのサービス率に対する到着率である負荷率として、レスポンスタイムへの影響度が CPU:I/O=1:2 で、CPU と I/O の到着率が同程度である場合の結果について示す。結果を図2に示す。この結果より、レスポンスタイムが60%を超過した点からレスポンスタイムの低下が確認できる。特に到着率が80%に近いところではレスポンスタイムがグリーディ法や再配置をしなかった場合と比べて大きく向上させることができる。また最適な配置として考えている遺伝的アル

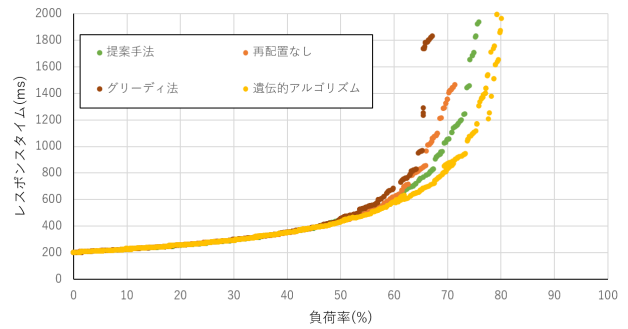


図2 レスポンスタイム 負荷率特性

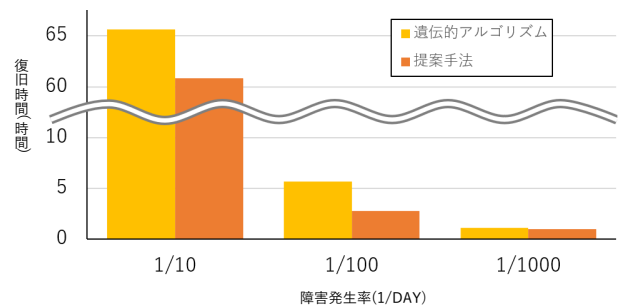


図3 復旧時間

ゴリズムと比較すると、レスポンスタイムは悪化しているものの到着率80%に近いところでもレスポンスタイムは1.3倍程度にとどまっている。

次に障害復旧時間についての評価を図3に示す。ここでは可用性の制約条件を設けている提案手法と、遺伝的アルゴリズムで障害復旧時間の検証を行う。

1日当たりの障害が発生する確率が1/10, 1/100の時に提案手法では遺伝的アルゴリズムと比較して、障害復旧時間を短縮することができている。特に障害が発生する確率が1/100の条件では復旧時間が50%以上も削減することができている。しかし、障害が発生する確率が1/1000の条件では復旧時間に大きな差が生じなかった。これはホスト数に対して障害が発生する確率が低く、可用性を考慮していない場合でも予備系が動作したためであると考えられる。

これらの結果より、提案手法の実現によって可用性の制約を満たしながらレスポンスタイムの低下を目指せることが確認でき、提案手法の有用性が確認できた。

4 おわりに

可用性とレスポンスタイムを考慮した仮想マシンの再配置手法を提案し、シミュレーションを用いて本手法の有効性について評価を行った。

参考文献

- [1] 菊森剛, 横山和俊, クラウドコンピューティングにおける可用性と消費電力を考慮した仮想マシン配置手法の提案, 情報処理学会 86 回全国大会 1X-02, (2024).