

クラウド上での CDN 事業展開における 仮想サーバのスケーラビリティを活用した配信サーバ群選択

長尾 翼[†] 春本 要^{††} 滝根 哲哉^{†††}

^{†,††,†††} 大阪大学大学院工学研究科 〒565-0871 大阪府吹田市山田丘 2-1

E-mail: [†]t-nagao@post.comm.eng.osaka-u.ac.jp, ^{††}harumoto@eng.osaka-u.ac.jp,

^{†††}takine@comm.eng.osaka-u.ac.jp

あらまし クラウドプロバイダが提供しているコンピュータ資源を利用して CDN 事業を展開する際、配信するコンテンツに対する遅延時間や転送速度等の QoS 要求を考慮しつつ、利用コストを低減できるような配信サーバ群の選択が重要である。クラウドプロバイダは多様なスケールの仮想サーバをコンピュータ資源として提供しており、配信サーバ群の選択においてこれを活用するためには、配信サーバの位置だけでなくコンテンツ配信にかかる負荷に適したスケールの仮想サーバを適切に選択しなければならない。しかし、既存の配信サーバ群選択手法では、配信サーバの位置しか考慮されていなかった。本稿では、配信サーバの位置に加え、仮想サーバのスケーラビリティを活用した配信サーバ群選択手法を提案する。また、実際のクラウドプロバイダに近いコスト設定を用いた数値実験を行い、提案手法の有効性を評価する。

キーワード クラウド, CDN, 配信サーバ群選択

1. はじめに

近年、スマートフォンやタブレットといった個人用情報端末の普及に伴い、Web ページや動画などのコンテンツに対するアクセスが増加していることから、インターネット上でのコンテンツ配信サービスが注目されている。従来より、安定したコンテンツ配信のために、CDN (Content Delivery Network) が広く用いられてきた。CDN では、地理的に分散させた配信サーバに、配信したいコンテンツの複製を配置する。これによって、コンテンツを要求するユーザは、自身の近くの配信サーバからコンテンツの配信を受けることができ、配信の際の遅延時間を抑えることが可能となる。また、コンテンツへのアクセスが各配信サーバに分散されることから、配信サーバやネットワークの負荷も抑制される。

一方で、クラウドコンピューティングと呼ばれる、新たなコンピュータ資源の利用形態が普及してきている。これは、自身でコンピュータ資源を導入し運用するのではなく、クラウドプロバイダによって世界各地に設置されたデータセンタで提供されるコンピュータ資源を利用することで、情報システムの構築や運用にかかるコストが抑えられるという利点をもつ。クラウドコンピューティングでは仮想サーバ、ストレージ、ネットワーク等のコンピュータ資源の実際の使用量に応じた従量制課金が採用されているため、適切なスケールのコンピュータ資源を選択することによって利用コストを最適化することが可能である。

このクラウドコンピューティングを用いた、CDN 事業の展開が注目されている [1] [2] [3] [4]。クラウドプロバイダが世界各地に設置したデータセンタから適切な仮想サーバを選択して配信サーバとして利用することで、設備投資を行うことなく CDN

事業を展開することが可能である。

CDN において配信されるコンテンツには、遅延時間や転送速度といった、コンテンツに対する QoS 要求が存在する。これを考慮せずに配信サーバ群を配置すると、ユーザはサービスに不満を覚えて利用なくなり、事業収入の減少につながる。そのため、配信するコンテンツに対する QoS 要求を満たした配信が可能となるように配信サーバ群を配置することが必要である。また、多数の配信サーバを配置するとコンテンツ配信にかかるコストが上昇し、事業支出が増加する。したがって、クラウド上での CDN 事業展開によって利益を生み出すためには、配信するコンテンツに対する QoS 要求を満たした上で、コンテンツ配信にかかるコストを可能な限り抑えた配信サーバ群の選択が非常に重要である。

この問題に対して、文献 [1] では、コンテンツに対する QoS 要求として、配信の際の遅延時間を考慮した手法を提案している。しかし、CDN で配信されるコンテンツは様々であり、コンテンツの種類によって QoS 要求が異なる場合も存在する。また、クラウド上で提供される仮想サーバのコスト設定には、仮想サーバを使用する際の定額コストが存在するが、文献 [1] におけるコスト設定ではそれが考慮されていない。それに対し、文献 [5] では、仮想サーバの使用コストが存在するときには、異なる QoS 要求を考慮して配信サーバ群の選択を行うことで、配信の際のコストを抑えることができる場合が存在することが示されている。

しかしながら、現実的に考えると、文献 [5] にはいくつか不十分な点が存在する。まず、配信サーバの位置のみを考慮しており、その位置に存在するデータセンタ内のどのスケールの仮想サーバを使用するかということは考慮されていない。本稿では、ユーザからのリクエストサイズやコンテンツに対する QoS

要求に応じてデータセンタ内の仮想サーバのスケールを選択可能な配信サーバ群選択手法を提案する。

また、文献 [5] では遅延時間に関する QoS 要求について、各ユーザへのコンテンツ配信が設定した遅延時間を満たしているかないかの 2 状態で考えている。しかし実際には、サービスに対するユーザの不満は、配信の際の遅延時間が長ければ長いほど大きくなると考えられる。さらに、どのような状況においても QoS 要求を満たさないコンテンツ配信が最小になるように配信サーバ群を選択するため、少数のアクセスしか処理しない配信サーバが選択されてしまい、配信コストが上昇してしまうという問題がある。そこで提案手法においては、QoS 要求をどの程度満たしているかを示す指標として QoS 違反度を新たに定義し、QoS 違反をある程度許容した配信サーバ群選択が行えるようにする。

また、提案手法の評価に際しては、現実のクラウドプロバイダのコスト設定をモデルとした数値実験を行い、クラウド上での CDN 事業展開における提案手法の有効性を示す。

本稿の構成は以下の通りである。まず、2 章で既存手法における想定環境と定式化について紹介する。次に、3 章で既存手法の問題点を考慮した手法を提案する。さらに、4 章で数値実験により提案手法の有効性の検証を行う。最後に、5 章で本研究のまとめと今後の課題について述べる。

2. 既存手法

文献 [5] では、クラウド上の仮想サーバの利用を想定した環境において、配信するコンテンツに対する QoS 要求として遅延時間に関する QoS 要求と転送速度に関する QoS 要求という 2 種類の QoS 要求を考慮した配信サーバ群選択手法を、0-1 整数計画問題として定式化している。本章ではその想定環境と定式化の方法について述べる。

2.1 想定環境

図 1 に文献 [5] における想定環境を示す。想定領域内に、配信するコンテンツを格納するオリジンサーバ、地理的に分散して存在するサーバ候補、コンテンツをリクエストする多数のユーザが存在する。これらのサーバ候補の中から、各コンテンツに対する QoS 要求を考慮した上で、配信に必要なコストが最小となるように配信に用いるサーバを選択することを考える。コンテンツに対する QoS 要求として、遅延時間に関する QoS 要求と、転送速度に関する QoS 要求という 2 種類の QoS 要求を考慮する。遅延時間に関する QoS 要求とは、ユーザからのリクエストを受信してからその応答がユーザに到着するまでの遅延時間が定められた時間以内であることを要求するものである。一方、転送速度に関する QoS 要求とは、ユーザへのコンテンツ配信を定められた速度以上で転送することを要求するものである。どちらの QoS 要求についても配信するコンテンツの種類によって要求する QoS レベルが異なる場合がある。また、一方の QoS 要求だけをもつものと、両方の QoS 要求をもつものがある。

各サーバ候補には利用する際に必要なコストが設定されている。図 2 に示すように、サーバの使用に必要な定額コスト、格

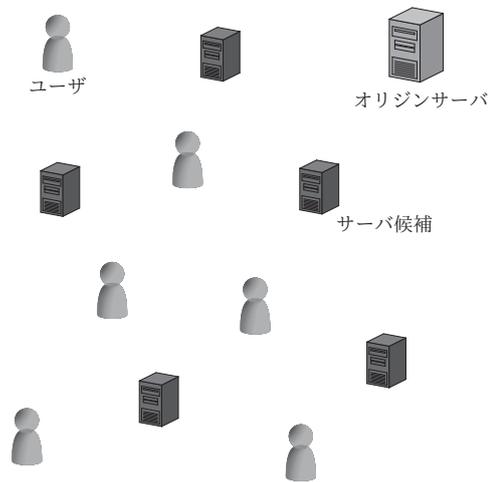


図 1 既存手法における想定環境

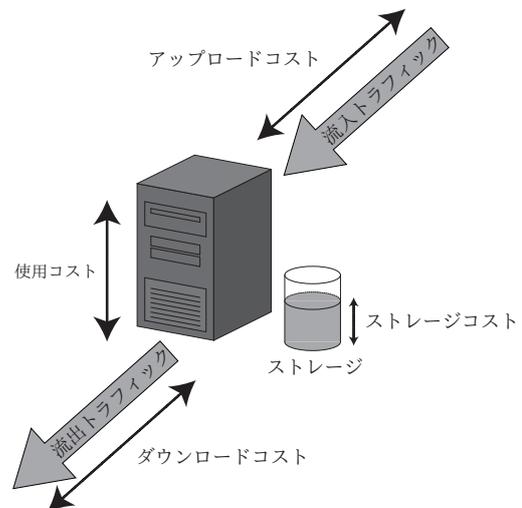


図 2 サーバ候補の利用コスト

納されたコンテンツのストレージ量に応じた従量コスト、サーバに流入するトラフィック量に応じた従量コスト、サーバから流出するトラフィック量に応じた従量コストが存在する。これらをそれぞれ使用コスト、ストレージコスト、アップロードコスト、ダウンロードコストと呼び、これら 4 つのコストを総称して利用コストと呼ぶ。

2.2 定式化

オリジンサーバを C_0 、サーバ候補の集合を $C = \{C_1, C_2, \dots, C_n\}$ 、ユーザの集合を $U = \{U_1, U_2, \dots, U_m\}$ とし、サーバ候補を文字 λ, μ 、ユーザを文字 ρ を用いて表す。オリジンサーバ、サーバ候補とユーザをまとめてノードと呼び、文字 α, β, γ を用いて表す。配信するコンテンツは QoS 要求によって l 種類に分類され、このうち k 種類のコンテンツが転送速度に関する QoS 要求をもつ。 k 種類のコンテンツの中には、遅延時間に関する QoS 要求と転送速度に関する QoS 要求の両方をもつものも含まれている。 l 種類の QoS 要求の集合を $Q = \{Q_1, Q_2, \dots, Q_l\}$ 、 k 種類の QoS 要求の集合を $Q' = \{Q_1, Q_2, \dots, Q_k\}$ とする。すなわち、 $k \leq l$ 、 $Q' \subseteq Q$ であり、QoS 要求を文字 ω を用いて表す。以上の記号の定義と文字

表 1 定式化における記号の定義と文字の意味

記号	定義	文字	意味
C_0	オリジンサーバ	λ, μ	サーバ候補
\mathcal{C}	サーバ候補の集合	ρ	ユーザ
\mathcal{U}	ユーザの集合	α, β, γ	ノード
$\mathcal{Q}, \mathcal{Q}'$	QoS 要求の集合	ω	QoS 要求

の意味は以降で何度も使用するため、これらを表 1 に示す。

次に、遅延時間に関する QoS 要求について考える。サーバ候補 λ からユーザ ρ へコンテンツを配信する際の遅延時間は、その距離 $L_{\lambda\rho}$ に比例すると仮定する。QoS 要求 $\omega \in \mathcal{Q}$ によって定められる遅延時間 T_ω に対応する距離を B_ω としたとき、 $L_{\lambda\rho} \leq B_\omega$ となるようなサーバ候補 λ とユーザ ρ の組 (λ, ρ) はエッジであるといい、エッジの集合を \mathcal{E}_ω と定義する。QoS 要求 ω が転送速度のみを要求している場合においても、 T_ω を十分に大きな値とすることで、QoS 要求 ω におけるエッジの集合 \mathcal{E}_ω を考えることができる。また、他のサーバ候補を経由することなく、直接コンテンツを配信可能なサーバ候補 λ とサーバ候補 μ の組 (λ, μ) について、すべての QoS 要求 ω におけるエッジであると定義する。 \mathcal{E}_ω に含まれるエッジのみを使用する配信の遅延時間は、QoS 要求 ω によって定められる遅延時間以下となることが保証される。

また、転送速度に関する QoS 要求について考える。サーバ候補 λ からユーザ ρ へコンテンツを配信する際の転送速度は、サーバの負荷 O_λ に反比例するとし、サーバの負荷 O_λ はそのサーバに対するユーザからのリクエストサイズの総和で表されると仮定する。転送速度に関する QoS 要求 $\omega \in \mathcal{Q}'$ によってサーバ候補 λ において定められる転送速度に対応する負荷を G_λ^ω としたとき、配信サーバとして利用するすべてのサーバ候補 λ について $O_\lambda \leq G_\lambda^\omega$ となるような配信における転送速度は、転送速度に関する QoS 要求 ω によって定められる転送速度以上となることが保証される。

さらに、サーバ候補 λ における使用コストを R_λ 、ストレージコストを S_λ 、アップロードコストを P_λ 、ダウンロードコストを D_λ とする。また、オリジンサーバ上でのコンテンツの更新を反映させるために、オリジンサーバがコンテンツの再配信を行う頻度を F とする。QoS 要求 ω をもつコンテンツについて、その配信サイズを W_ω 、ユーザ ρ からのリクエストサイズを w_ρ^ω とする。このとき、ノード α からノード β に QoS 要求 ω をもつコンテンツを配信する際にかかるコスト $V_{\alpha\beta}^\omega$ は、ノード β がサーバ候補であるかユーザであるかによって定義が異なる。 $V_{\alpha\beta}^\omega$ の定義を以下に示す。

$$V_{\alpha\beta}^\omega = \begin{cases} (S_\beta + P_\beta F + D_\alpha F)W_\omega, & \beta \in \mathcal{C} \\ w_\beta^\omega D_\alpha, & \beta \in \mathcal{U} \end{cases}$$

さらに、定式化を行うために必要となる変数を定義する。文献 [5] では、0-1 整数計画問題として定式化を行っているため、すべての変数が 0 か 1 の二値をとる。まず、オリジンサーバからユーザ ρ へ QoS 要求 ω をもつコンテンツを配信する際に、エッジ (α, β) を使用するか否かを二値変数 $x_{\alpha\beta}^{\rho\omega}$ で表す。すなわ

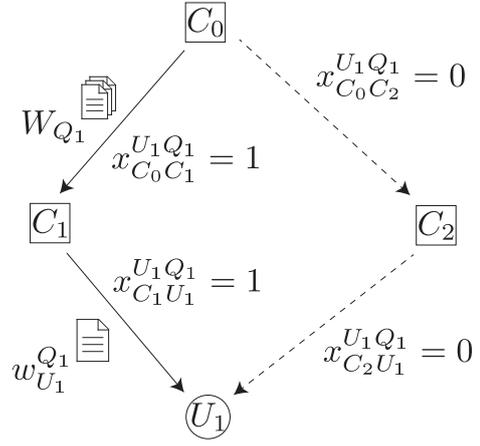


図 3 記号 $x_{\alpha\beta}^{\rho\omega}$ の例

ち、 $x_{\alpha\beta}^{\rho\omega}$ は以下で定義される。

$$x_{\alpha\beta}^{\rho\omega} = \begin{cases} 1, & \text{ユーザ } \rho \text{ への配信で } (\alpha, \beta) \text{ を使用する} \\ 0, & \text{ユーザ } \rho \text{ への配信で } (\alpha, \beta) \text{ を使用しない} \end{cases}$$

図 3 に $x_{\alpha\beta}^{\rho\omega}$ の例を示す。オリジンサーバ C_0 からユーザ U_1 へ、QoS 要求 Q_1 をもつコンテンツの配信を行うことを考える。 U_1 の近くに存在する C_1 と C_2 の二つのサーバ候補のうち、配信サーバとして C_1 が選択されたと仮定する。このとき、エッジ (C_0, C_1) とエッジ (C_1, U_1) は使用されるため、 $x_{C_0C_1}^{U_1Q_1} = 1$ 、 $x_{C_1U_1}^{U_1Q_1} = 1$ となるが、エッジ (C_1, C_2) とエッジ (C_2, U_1) は使用されないため、 $x_{C_1C_2}^{U_1Q_1} = 0$ 、 $x_{C_2U_1}^{U_1Q_1} = 0$ となる。

次に、この $x_{\alpha\beta}^{\rho\omega}$ に対して、すべてのエッジ (α, β) について

$$x_{\alpha\beta}^{\rho\omega} \leq y_{\alpha\beta}^\omega, \quad \forall \rho \in \mathcal{U}$$

を満たす二値変数 $y_{\alpha\beta}^\omega$ を定義する。定義から、 $y_{\alpha\beta}^\omega$ は以下の意味をもつ。

$$y_{\alpha\beta}^\omega = \begin{cases} 1, & \text{任意のユーザへの配信で } (\alpha, \beta) \text{ を使用する} \\ 0, & \text{どのユーザへの配信でも } (\alpha, \beta) \text{ を使用しない} \end{cases}$$

すなわち、 $y_{\alpha\beta}^\omega$ は QoS 要求 ω をもつコンテンツの配信の際に、エッジ (α, β) を使用するユーザが存在するか否かを表す変数である。これを用いて、サーバ候補 λ の負荷 O_λ を、サーバ候補 λ に対するユーザからのリクエストサイズの総和として、以下の式 (1) で表すことができる。

$$O_\lambda = \sum_{\omega \in \mathcal{Q}'} \sum_{(\lambda, \rho) \in \mathcal{E}_\omega} y_{\lambda\rho}^\omega w_\rho^\omega \quad (1)$$

さらに、QoS 要求 ω をもつコンテンツの配信について、サーバ候補 λ の使用に関する二値変数 z_λ^ω を以下で定義する。

$$z_\lambda^\omega = \begin{cases} 1, & \omega \text{ をもつコンテンツの配信で } \lambda \text{ を使用する} \\ 0, & \omega \text{ をもつコンテンツの配信で } \lambda \text{ を使用しない} \end{cases}$$

このとき、すべてのサーバ候補 λ について

$$y_{\lambda\alpha}^\omega \leq z_\lambda^\omega, \quad \forall (\lambda, \alpha) \in \mathcal{E}_\omega$$

が成立する。前述の通り、転送速度に関する QoS 要求を満たした配信を行うためには、配信サーバとして利用するすべてのサーバ候補 λ について $O_\lambda \leq G_\lambda^\omega$ が成り立っていればよい。このことは、 z_λ^ω を用いることで、以下の制約条件式として表現可能である。

$$z_\lambda^\omega O_\lambda \leq G_\lambda^\omega, \quad \forall \omega \in \mathcal{Q}'$$

また、 z_λ^ω に対し、すべてのサーバ候補 λ について

$$z_\lambda^\omega \leq z_\lambda, \quad \forall \omega \in \mathcal{Q}$$

を満たす二値変数 z_λ を定義する。定義から、 z_λ は以下の意味をもつ。

$$z_\lambda = \begin{cases} 1, & \text{任意のコンテンツの配信で } \lambda \text{ を使用する} \\ 0, & \text{どのコンテンツの配信でも } \lambda \text{ を使用しない} \end{cases}$$

これを用いて、配信の際に必要なコスト DC (Delivery Cost) を以下の式 (2) で表すことができる。

$$DC = \sum_{\omega \in \mathcal{Q}} \sum_{(\alpha, \beta) \in \mathcal{E}_\omega} y_{\alpha\beta}^\omega V_{\alpha\beta}^\omega + \sum_{\lambda \in \mathcal{C}} z_\lambda R_\lambda \quad (2)$$

文献 [5] における配信サーバ群選択手法は、配信の際に必要なコストを最小化するために、 DC を目的関数として以下のように定式化される。

$$\begin{aligned} P_1 : \text{minimize} \quad & DC \\ \text{subject to} \quad & \forall \rho \in \mathcal{U}, \forall \omega \in \mathcal{Q}, \\ & \sum_{(\alpha, \beta) \in \mathcal{E}_\omega} x_{\alpha\beta}^{\rho\omega} - \sum_{(\gamma, \alpha) \in \mathcal{E}_\omega} x_{\gamma\alpha}^{\rho\omega} = \begin{cases} 1, & \alpha = C_0 \\ -1, & \alpha \in \mathcal{U} \\ 0, & \alpha \in \mathcal{C} \end{cases} \\ & x_{\alpha\beta}^{\rho\omega} \leq y_{\alpha\beta}^\omega, \quad \forall (\alpha, \beta) \in \mathcal{E}_\omega \\ & y_{\lambda\alpha}^\omega \leq z_\lambda^\omega, \quad \forall (\lambda, \alpha) \in \mathcal{E}_\omega \\ & z_\lambda^\omega \leq z_\lambda, \quad \forall \lambda \in \mathcal{C} \\ & z_\lambda^\omega O_\lambda \leq G_\lambda^\omega, \quad \forall \lambda \in \mathcal{C}, \forall \omega \in \mathcal{Q}' \end{aligned}$$

ただし、 O_λ , DC の定義はそれぞれ式 (1), 式 (2) で与えられる。なお、最後の制約条件式は線形化が可能である [5]。

3. 提案手法

文献 [5] の想定環境と定式化には、不十分であると考えられる点が存在する。まず、文献 [5] では、想定領域内に分散して存在するのはサーバ候補であり、データセンタではない。各サーバ候補が存在する位置には、それぞれ 1 台のサーバ候補のみが存在するものと想定しており、同じ位置に複数のサーバ候補が存在することを想定していない。しかし、現実のクラウドプロバイダは、世界各地にデータセンタを分散して設置し、そのデータセンタ内では様々なスケールの複数の仮想サーバが提供される。大きいスケールの仮想サーバは、利用するためにより多くの使用コストを必要とするが、配信サーバとして利用する際にはより大きな負荷に対応することができる。このような仮想サーバのスケールビリティを考慮した配信サーバ群選択を行

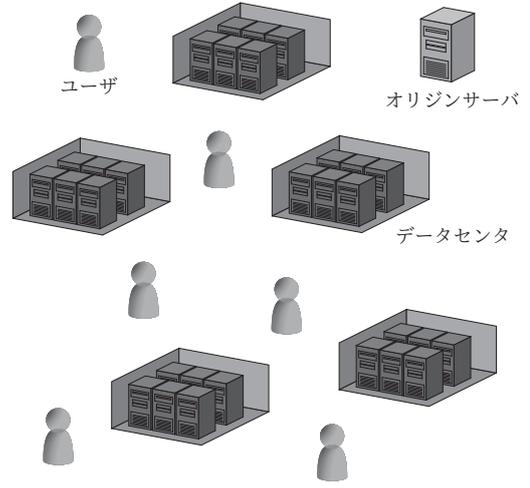


図 4 提案手法における想定環境

うことは、クラウド上で CDN 事業を展開する際には非常に重要である。したがって、本稿では想定領域内に多数存在するのはサーバ候補ではなくデータセンタであり、そのデータセンタ内からスケールの異なる仮想サーバを配信サーバとして選択可能であるとする。本稿における想定環境を図 4 に示す。

スケールの異なる仮想サーバの選択を可能にするために、文献 [5] の定式化における記号と変数の定義を変更する。集合 $\mathcal{C} = \{C_1, C_2, \dots, C_n\}$ をサーバ候補ではなくデータセンタの集合とする。データセンタ λ 内では様々なスケールの仮想サーバが提供されているが、ストレージコスト S_λ , アップロードコスト P_λ , ダウンロードコスト D_λ はデータセンタごとに定義される。使用コスト R_λ については、仮想サーバごとに異なるため、データセンタ λ 内で最小スケールの仮想サーバの使用コストを表すとする。

この最小スケールの仮想サーバをスケール 1 の仮想サーバと呼ぶ。データセンタ λ 内で提供されているすべての仮想サーバは、このスケール 1 の仮想サーバの整数倍のスケールであるとし、 n 倍のスケールの仮想サーバをスケール n の仮想サーバと呼ぶ。スケール n の仮想サーバは、スケール 1 の仮想サーバの n 倍の使用コストを必要とするが、スケール 1 の仮想サーバの n 倍の負荷を処理可能であるとする。この仮想サーバのスケールの倍率を、本稿の定式化では変数 z_λ で表すことにする。すなわち $z_\lambda = 1$ のときには、データセンタ λ において配信サーバとして選択された仮想サーバはスケール 1 の仮想サーバであり、 $z_\lambda = 3$ のときには、スケール 3 の仮想サーバが選択されたことを意味する。したがって、文献 [5] では z_λ は二値変数であったが、本稿では非負の整数値をとる変数とする。なお、二値変数 z_λ^ω は提案手法では用いない。

データセンタ内で提供される仮想サーバにおける、転送速度に関する QoS 要求について考える。文献 [5] と同じく、サーバ候補 λ からコンテンツを配信する際の転送速度は、サーバ候補 λ の負荷に反比例すると仮定する。データセンタ λ 内のスケール 1 の仮想サーバは、その負荷が H_λ 以下であるとき、転送速度に関する QoS 要求を満たした配信が可能であるとする。す

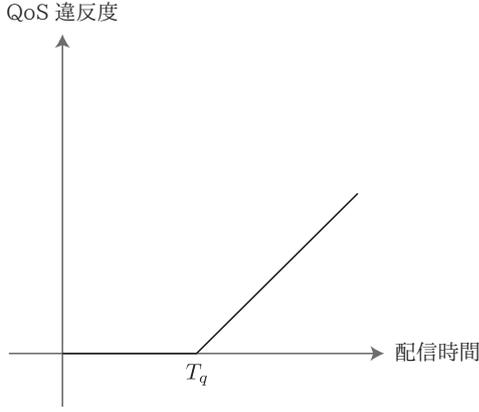


図5 配信時間に対する QoS 違反度の定義

なわち、スケール倍率 z_λ の仮想サーバでは、転送速度に関する QoS 要求を満たした配信が可能な負荷の上限値は $z_\lambda H_\lambda$ である。

以上の変更点を考慮し、 P_1 から以下の制約条件を取り除く。

$$y_{\lambda\alpha}^\omega \leq z_\lambda^\omega, \quad \forall (\lambda, \alpha) \in \mathcal{E}_\omega$$

$$z_\lambda^\omega \leq z_\lambda, \quad \forall \lambda \in \mathcal{C}$$

$$z_\lambda^\omega O_\lambda \leq G_\lambda^\omega, \quad \forall \lambda \in \mathcal{C}, \forall \omega \in \mathcal{Q}'$$

そして、以下の制約条件を追加する。

$$y_{\lambda\alpha}^\omega \leq z_\lambda, \quad \forall (\lambda, \alpha) \in \mathcal{E}_\omega$$

$$O_\lambda \leq z_\lambda H_\lambda, \quad \forall \lambda \in \mathcal{C}$$

また、文献[5]では遅延時間に関する QoS 要求について、配信に要した時間がある値以下であればその QoS 要求は満たされ、そうでなければその QoS 要求に違反しているとしており、QoS 要求を満たしているかないかの2状態で考えていた。しかし実際には、サービスに対するユーザの不満は、配信の際の遅延時間が長ければ長いほど大きくなると考えられる。そこで本稿では、遅延時間に関する QoS 要求にどの程度違反しているかを表現する新たな指標として、QoS 違反度を定義する。これには様々な定義が考えられるが、本稿では図5のように、QoS 要求 $\omega \in \mathcal{Q}$ によって定められる遅延時間 T_ω 以内に配信可能な場合には0であり、QoS 要求で定められた遅延時間を超過した場合には超過した分だけ線形に増加するような QoS 違反度の定義を使用する。その他にも、ユーザの効用を考慮した一般の増加関数を用いた定義や、リクエストサイズと超過した時間の積を QoS 違反度とするような定義も考えられる。文献[5]と同じく、配信の際の遅延時間はサーバ候補からユーザまでの距離に比例するものとし、時間 T_ω に対応する距離 B_ω 、サーバ候補 λ からユーザ ρ までの距離 $L_{\lambda\rho}$ を用いると、サーバ候補 λ からユーザ ρ まで QoS 要求 ω をもつコンテンツを配信する際の QoS 違反度 $Q_{\lambda\rho}^\omega$ は以下で定義される。

$$Q_{\lambda\rho}^\omega = \max(0, L_{\lambda\rho} - B_\omega)$$

この QoS 違反度の定義を導入することで、エッジの定義をより簡単なものに変更することが可能である。文献[5]におけ

るエッジ (α, β) は、直接コンテンツを配信可能なサーバ候補の組か、時間 T_ω 以内に配信可能なサーバ候補とユーザの組であったが、後者の定義を前者にそろえ、直接コンテンツを配信可能なノードの組とする。すなわち、ノード α からノード β へ、QoS 要求 ω をもつコンテンツを配信時間に関わらず直接配信可能であるとき、 (α, β) はエッジとなる。

また、文献[5]では、コンテンツに対する QoS 要求を満たすことが可能なサーバが存在する場合には、必ずそのサーバを選択している。しかしながら、これでは少数のユーザにしか配信を行わないようなサーバ候補も、配信サーバとして選択されてしまう。現実的には、少数のユーザからのリクエストに対応するためだけに、新たな配信サーバを立ち上げて多大なコストをかけるとは考えにくい。したがって、ある程度の QoS 違反を許容することによって配信サーバ群の数を低減させることができるような制約条件を考える。QoS 違反度の合計値を QVD (QoS Violation Degree) とすると

$$QVD = \sum_{\omega \in \mathcal{Q}} \sum_{(\lambda, \rho) \in \mathcal{E}_\omega} y_{\lambda\rho}^\omega Q_{\lambda\rho}^\omega \quad (3)$$

と表すことができる。許容可能な QVD の上限値を $QVDLimit$ とすると、定式化に制約条件

$$QVD \leq QVDLimit$$

を追加することで、 QVD が $QVDLimit$ 以下であることが保証される。すべての QoS 要求が満たされた配信を行いたいときには、 $QVDLimit = 0$ とすればよい。また、目的関数を変更し

$$DC + \theta QVD$$

とする。ただし、係数 θ は十分に小さい値とする。これにより、 DC を最適化するような配信サーバ群が選択されると同時に、同じ DC となるような配信サーバ群の候補の中で QVD が最小となるような配信サーバ群を選択することが可能になる。

以上より、提案手法を定式化すると以下ようになる。

$$P_2 : \text{minimize } DC + \theta QVD$$

$$\text{subject to } \forall \rho \in \mathcal{U}, \forall \omega \in \mathcal{Q},$$

$$\sum_{(\alpha, \beta) \in \mathcal{E}_\omega} x_{\alpha\beta}^{\rho\omega} - \sum_{(\gamma, \alpha) \in \mathcal{E}_\omega} x_{\gamma\alpha}^{\rho\omega} = \begin{cases} 1, & \alpha = C_0 \\ -1, & \alpha \in \mathcal{U} \\ 0, & \alpha \in \mathcal{C} \end{cases}$$

$$x_{\alpha\beta}^{\rho\omega} \leq y_{\alpha\beta}^\omega, \quad \forall (\alpha, \beta) \in \mathcal{E}_\omega$$

$$y_{\lambda\alpha}^\omega \leq z_\lambda, \quad \forall (\lambda, \alpha) \in \mathcal{E}_\omega$$

$$O_\lambda \leq z_\lambda H_\lambda, \quad \forall \lambda \in \mathcal{C}$$

$$QVD \leq QVDLimit$$

$$x_{\alpha\beta}^{\rho\omega} \in \{0, 1\}, \quad \forall (\alpha, \beta) \in \mathcal{E}_\omega$$

$$y_{\alpha\beta}^\omega \in \{0, 1\}, \quad \forall (\alpha, \beta) \in \mathcal{E}_\omega$$

$$z_\lambda \in \{0, 1, 2, \dots\}, \quad \forall \lambda \in \mathcal{C}$$

ただし、 O_λ , DC , QVD はそれぞれ式(1), 式(2), 式(3)で与えられる。

4. 性能評価

本章では、後述の評価モデルに従って問題を設定し、前章にて整数計画問題 P_2 として定式化した提案手法を CPLEX [6] を用いて解き、性能評価を行う。

4.1 評価モデル

本評価では、 40×40 の正方形の領域を想定し、この領域上に 16 人の仮想ユーザを等間隔に配置する。この仮想ユーザは自身の周囲の 10×10 の正方形の領域内のユーザ群を仮想的に集約したものであり、これをコンテンツをリクエストするユーザとみなす。個々のユーザではなく集約された仮想ユーザを用いるのは、ユーザ数が多数となると CPLEX による求解が困難になるためである。

この領域に、配信するコンテンツを格納しているオリジンサーバが 1 台、データセンタが複数存在する。オリジンサーバからは任意のデータセンタにコンテンツを配信可能であるが、ユーザへ直接配信を行うことは不可能とする。これは、通常オリジンサーバはクラウドの外に存在するためである。各データセンタからは任意のユーザへとコンテンツを配信可能である。

仮想サーバを提供するクラウドプロバイダには、CP1 と CP2 の 2 社が存在するものとする。これらはそれぞれ Amazon Web Services [7] と Microsoft Azure [8] をモデルとしており、その仮想サーバの利用コストには、両社の Web サイト上で公開されている仮想サーバの利用コストの表から、同スケール程度と思われるものの利用コストを用いた。使用コストについては両社ともに 1 時間当たりの使用コストであったため、720 倍して 1 か月当たりの使用コストとした。その他の 3 つのコストについては 1 GB 当たりのコストである。このようにして設定した両社の利用コストを表 2 に示す。CP1 の提供する仮想サーバは使用コストが低くダウンロードコストとストレージコストが高く設定されており、CP2 の提供する仮想サーバでは使用コストが高くダウンロードコストとストレージコストが低く設定されている。

配信するコンテンツは一種類とし、遅延時間に関する QoS 要求と、転送速度に関する QoS 要求の両方を持っているとする。遅延時間に関する QoS 要求については、 $B_1 = 10$ と定める。転送速度に関する QoS 要求については、配信を行うデータセンタにおいて、50Mbps の転送速度を必要であるとする。このとき、 H_1 は

$$\begin{aligned} H_1 &= 50 \times 10^6 \text{ bps} \times 60 \text{ s} \times 60 \text{ m} \times 24 \text{ h} \times 30 \text{ d} \times 0.125 \text{ B} \\ &= 16,200 \times 10^9 \text{ B} = 16,200 \text{ GB} \end{aligned}$$

となる。配信するコンテンツのサイズを 100 GB、オリジンサーバがコンテンツの再配信を行う頻度 F を 1 とする。また、目的関数における係数 θ の値を 10^{-4} とする。

4.2 評価結果

前節の評価モデルに従い、性能評価を行った。

4.2.1 QVDLimit を変化させた評価

まず、許容可能な QVD の上限 $QVDLimit$ を変化させたときの、配信の際に必要なコスト DC について評価を行った。各

表 2 各クラウドプロバイダにおけるサーバの利用コストの設定値

	R_λ	P_λ	D_λ	S_λ
CP1	77.76	0	0.09	0.023
CP2	105.12	0	0.87	0.017

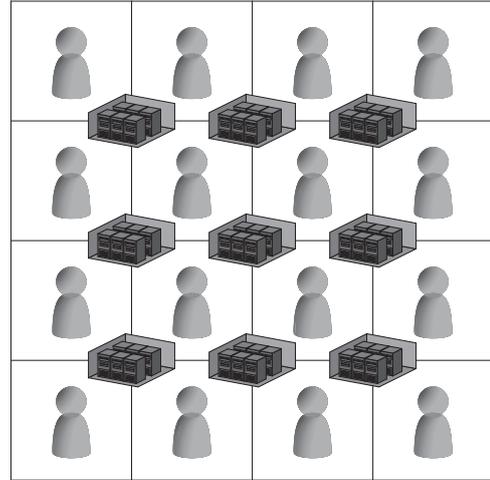


図 6 ケース 1 におけるデータセンタの配置

ユーザからのリクエストサイズを全て 3,000 GB とした上で、 $QVDLimit$ を 0 から 50 まで 5 刻みに変化させて、次の三種類のケースについて比較した。

- ケース 1 CP1 と CP2 の両方の仮想サーバが提供されるデータセンタを、図 6 に示す通りに等間隔に 9 か所配置したケース
- ケース 2 CP1 と CP2 の両方の仮想サーバが提供されるデータセンタを、想定領域内にランダムに 9 か所に配置したケース
- ケース 3 CP1, CP2 のそれぞれの仮想サーバのみが提供されるデータセンタを、想定領域内にランダムに 9 か所ずつ計 18 か所に配置したケース

ケース 2 とケース 3 において、データセンタをランダムに配置する際には、同じクラウドプロバイダの仮想サーバが提供されるデータセンタは、少なくとも距離 10 は離して配置した。このとき、各ケースにおいて、CP1 の仮想サーバのみをサーバ候補としたとき、CP2 の仮想サーバのみをサーバ候補としたとき、CP1 と CP2 の両方の仮想サーバをサーバ候補としたときの 3 つの場合における、 QVD と DC の値を散布図としてプロットした結果を、それぞれ図 7, 図 8, 図 9 に示す。なお、ケース 2 とケース 3 においては、ランダムにデータセンタを配置するため、同じ評価を 10 回ずつ行い、それら全ての結果を示している。

図 7 から、QoS 違反度の増加を許容するとコストを下げられることが分かる。比較した 3 つの場合のいずれにおいてもコストを下げる事ができているが、特に CP1 と CP2 の両方の仮想サーバをサーバ候補としたときにはコストが顕著に下がって

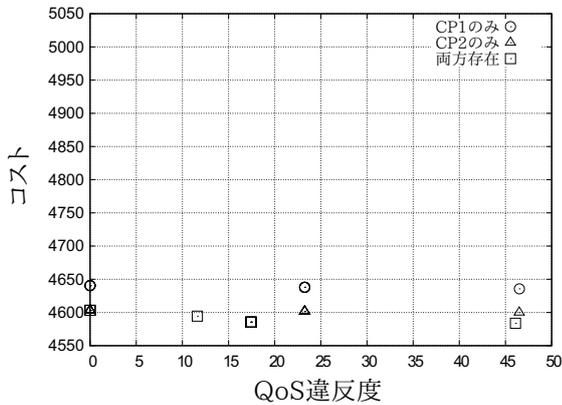


図 7 ケース 1 における結果

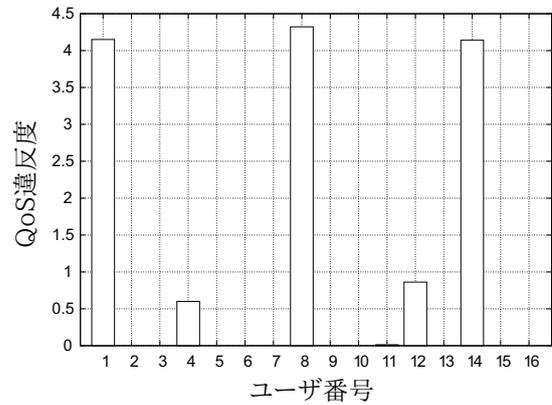


図 10 QoS 違反度の分布

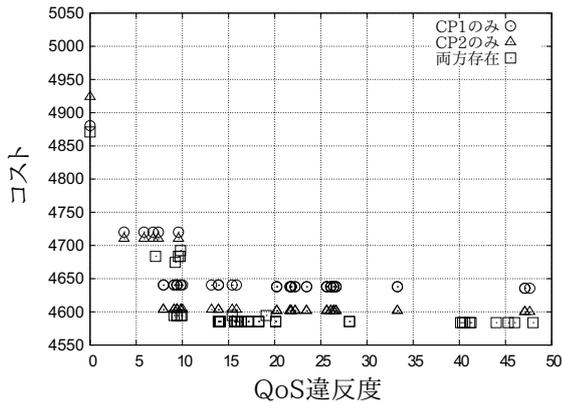


図 8 ケース 2 における結果

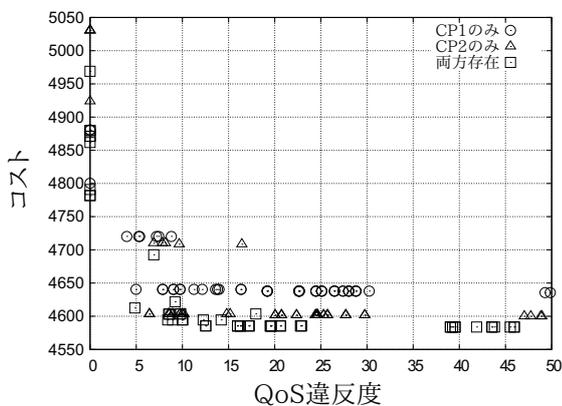


図 9 ケース 3 における結果

いる。これは、CP1 と CP2 の両方の仮想サーバをサーバ候補としたときには、ユーザからのリクエストサイズの総和が少ないデータセンターでは CP1 の仮想サーバを用い、多いデータセンターでは CP2 の仮想サーバを用いることができるためである。

図 8 においても、ケース 1 の結果と同じ傾向を見ることができる。ケース 1 では、全ユーザにおいて QoS 要求に違反せずに配信を受けることのできるデータセンターの存在が保証されており、特に想定領域の中心付近のユーザでは 4 か所のデータセンターから QoS 要求に違反することなく配信を受けることができる。しかし、データセンターをランダムに配置したケースでは、QVDLimit の値が小さいときには制約条件を満たして配信

することが不可能な場合も存在する。特にケース 2 では、データセンターが 9 か所にしか存在しないため、 $QVDLimit = 0$ のときには解が存在しないことも多く、存在した場合でもケース 1 と比較すると DC は割高なものとなっていることがわかる。 $QVDLimit$ を大きく設定し、 QVD が 10 を超えるような配信を許容すると、 DC はほぼ一定値をとるようになっている。例えば 1 ユーザにつき QoS 違反度が 1 増加することを許容するとし、図 8 における $QVD = 16$ 付近の DC を見ると、QoS 違反を許容しない場合と比較して 5% 以上のコストを削減できることがわかる。

しかし、 QVD が 16 程度であっても、それぞれのユーザの QoS 違反度が高々 1 程度となっているとは限らない。そこで、図 8 にプロットされている点のうち、CP1 と CP2 の両方の仮想サーバをサーバ候補としたときにおける、 $QVDLimit = 20$ のときの、ユーザごとの QoS 違反度の分布を図 10 に示す。これを見ると、1 ユーザ当たりの QoS 違反度の平均は 1 であっても、実際には半分以上のユーザは QoS 違反度が 0 となる一方、3 ユーザに QoS 違反度が集中してしまっていることがわかる。この 3 ユーザに関しては、遅延時間に関する QoS 要求で定められた配信時間の 1.5 倍近い配信時間がかかっており、サービスから離脱してしまう可能性がある。それゆえ、 DC が最適化されていても、事業の収入を考えた時に最適な結果になっているとは断言できない。

ケース 3 は、ケース 2 とほぼ同じ結果となっている。しかし、ケース 2 では CP1 と CP2 のどちらの仮想サーバから配信を受けても QoS 違反度は変わらないため、グラフにおいて CP1 のみの場合と CP2 のみの場合の QoS 違反度は同じとなる（○印と△印が縦に並ぶ）が、ケース 3 では CP1 と CP2 のデータセンターの配置が異なるため、QoS 違反度は CP1 のみの場合と CP2 のみの場合で同じにはならない点で、ケース 2 の結果と異なっている。

これらの図を用いると、あるコストを上限に CDN を運用したいと考えたときに QoS 違反度がどの程度になるのかといったことや、逆にある QoS 違反度以下で CDN を運用するためにはどの程度のコストが必要になるのかといったことがわかり、CDN 事業を展開する上での判断材料とすることが可能である。

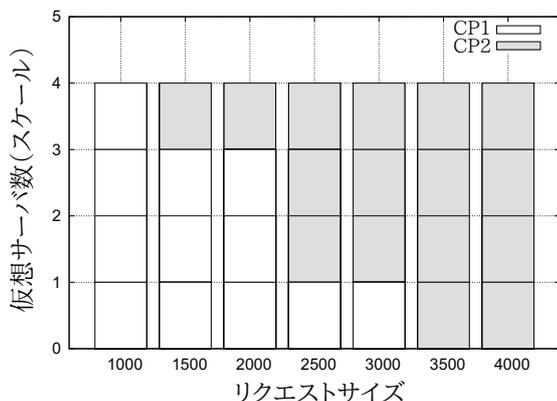


図 11 選択された仮想サーバ数 (QVDLimit=20 のとき)

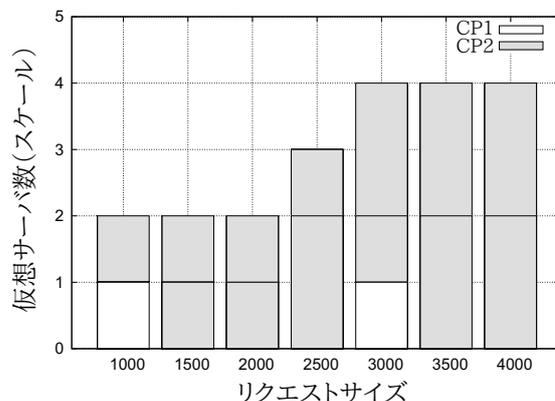


図 12 選択された仮想サーバ数 (QVDLimit=80 のとき)

4.2.2 リクエストサイズを変化させた評価

次に、選択された仮想サーバの特徴を示すために、ユーザからのリクエストサイズを変化させたときの、選択された仮想サーバ数について前節のケース 1 と同じ状況で評価を行った。各ユーザからのリクエストサイズを 1,000 GB から 4,000 GB まで 500 GB 刻みで変化させた上で、QVDLimit の値を 20 と 80 とした場合、選択された仮想サーバ数をそれぞれ図 11 と図 12 に示す。図 11 と図 12 において、縦軸は選択された仮想サーバ数をスケールで換算した量であり、複数つながっているものはつながっている分だけのスケールをもつ仮想サーバが選択されたことを示している。例えば、図 12 においてリクエストサイズが 3,000 GB の時には、CP1 のスケール 1 の仮想サーバが 1 台、CP2 のスケール 1 の仮想サーバが 1 台、CP2 のスケール 2 の仮想サーバが 1 台選択されたことを意味する。図 11 をみると、リクエストサイズが小さいときには CP1 の仮想サーバが選択され、リクエストサイズが大きくなるにつれて CP2 の仮想サーバが選択されるようになることがわかる。これは、リクエストサイズの増加に伴って、使用コストの低い仮想サーバを重視した選択から、ダウンロードコストの低い仮想サーバを重視した選択へと変化したことを示している。

また、図 12 をみると、リクエストサイズの増加に伴い、選択された仮想サーバの数が増加していることがわかる。これはスケール 1 の仮想サーバ 1 台につき H_λ までしかリクエストを処理できないことによるものである。このとき、合計 H_λ を超えるサイズのリクエストを処理する際に、2 カ所のデータセンタでスケール 1 の仮想サーバを 2 台利用するよりも、1 カ所のデータセンタでスケール 2 の仮想サーバを 1 台利用することで、ストレージコストを削減できる。すなわち、スケーラビリティを活用した配信サーバ群の選択を行うことができるといえる。ただし、1 カ所のデータセンタからの配信の際には QVD の増加を避けられないため、QVDLimit を小さく設定した図 11 の状況においては、仮想サーバのスケーラビリティは活用されていない。

5. おわりに

本稿では、クラウド上での CDN 事業展開において、各地に

存在するデータセンタ内で提供される仮想サーバのスケーラビリティを活用して、コンテンツ配信に用いるサーバ群を選択する手法を提案した。また、コンテンツに対する QoS 要求に対する違反度を定義し、提案した手法を配信コストと QoS 違反度のトレードオフを考慮できるような整数計画問題として定式化した。さらに、現実のクラウドプロバイダのコスト設定をモデルとした数値実験を行うことで、多少の QoS 違反度の増加を許容するだけで 5% 程度のコストを削減可能であることを示した。また、CP1 と CP2 の仮想サーバの料金設定の特徴に基づき、どのようなときにそれぞれの仮想サーバが選択され、またどのようなときに仮想サーバのスケーラビリティが活用されるかを示した。今後の課題としては、ユーザのリクエストの時間変化にも対応可能な手法の提案などがあげられる。

文 献

- [1] Chen, F., Guo, K., Lin, J., et al.: Intra-cloud Lightning: Building CDNs in the Cloud, *Proc. IEEE INFOCOM 2012*, pp.433–441.
- [2] Wang, M., Jayaraman, P.P., Ranjan, R., et al.: An Overview of Cloud Based Content Delivery Networks: Research Dimensions and State-of-the-Art, *Transactions on Large-Scale Data- and Knowledge-Centered Systems XX*, pp.131–158 (2015).
- [3] Wang, F., Liu, J., and Chen, M.: CALMS: Cloud-Assisted Live Media Streaming for Globalized Demands with Time/Region Diversities, *Proc. IEEE INFOCOM 2012*, pp.199–207.
- [4] Salahuddin, A.M., Elbiaze, H., Ajib, W., et al.: Social Network Analysis Inspired Content Placement with QoS in Cloud-based Content Delivery Networks, *Proc. IEEE GLOBECOM 2015*.
- [5] 長尾翼, 春本要, 滝根哲哉, “クラウド上での CDN 実現における異なる QoS 要求を考慮した資源選択手法,” 情報処理学会マルチメディア, 分散, 協調とモバイル (DICOMO2016) シンポジウム論文集, 5D-4, pp.1016–1023, 2016 年 7 月.
- [6] IBM: IBM ILOG CPLEX Optimization Studio, IBM (オンライン), 入手先 (<http://www-03.ibm.com/software/products/ja/ibmilogcplexoptistud>) (参照 2016-05-11).
- [7] Amazon Web Services: 料金 - Amazon EC2 | AWS, Amazon Web Services (オンライン), 入手先 (<https://aws.amazon.com/jp/ec2/pricing/>) (参照 2016-05-11).
- [8] Microsoft Azure: 価格の概要 - Azure 価格の仕組み | Microsoft Azure, Microsoft Azure (オンライン), 入手先 (<https://azure.microsoft.com/ja-jp/pricing/>) (参照 2017-01-01).