

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

長尾 翼¹ 春本 要¹ 滝根 哲哉¹

概要：クラウド上でCDNを実現する際には、配信に必要なコストが利用するクラウド資源に大きく依存するため、利用するクラウド資源の選択手法が重要となる。しかしながら、既存の選択手法では、遅延時間に関する一種類のQoS要求しか考慮されておらず、また、クラウド資源の利用コストの設定にサーバの使用に伴う定額コストが含まれていない。CDNにおけるコンテンツの配信では、その種類やデータサイズなどによって、遅延時間や転送速度など異なるQoS要求を考慮すべきであり、また資源選択の手法を考える際には、より実際のクラウドサービスに近い利用コストの設定を用いるべきである。そこで本稿では、クラウド資源の利用コストの設定にサーバの使用に伴う定額コストを追加した上で、異なるQoS要求を考慮したクラウド資源の選択手法を提案する。さらに、その提案手法を0-1整数計画問題として定式化し、数値実験によって性能評価を行うことで提案手法の有用性を示す。

A Resource Seletion Method Considering Different QoS Requirements for Building CDN in the Cloud

TSUBASA NAGAO¹ KANAME HARUMOTO¹ TETSUYA TAKINE¹

1. はじめに

近年、仮想化技術の進展に伴い、計算資源、ストレージ資源、ネットワーク資源などの計算機資源を独自に導入し保有する代わりに、データセンタに設置されたクラウド資源をオンデマンドで利用する、クラウドコンピューティングと呼ばれる計算機資源の利用形態が一般化してきた。クラウドコンピューティングでは、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で提供されるクラウド資源をオンデマンドで利用することが可能であり、情報システム構築における初期投資コストの低減、さらにはランニングコストの低減が可能であることから、多くの組織でその活用が進められている。

一方、スマートフォンやタブレット端末等の普及や無線通信技術の進展によって、一般利用者は時間・場所を問わずにWebページや動画などのコンテンツにアクセスし利用できるようになったことから、インターネット上でコンテンツを提供するサービスの重要性がますます増大し

ている。このようなサービスの構築において、従来より、コンテンツ配信専用に構築されたCDN(Content Delivery Network)が活用されてきた。CDNでは、配信すべきコンテンツの複製を世界各地に設置されたエッジサーバに配置することによって、リクエスト元に近いエッジサーバからユーザへ配信を行うことを可能とし、ネットワークの負荷分散や転送遅延の低減を実現している。

このような背景のもとで、近年ではクラウドを利用してCDNを実現する手法が盛んに研究されている[1][2][3][4]。クラウドでは、計算資源、ストレージ資源、ネットワーク資源のそれぞれに対し、実際に使用した資源量によって課金が行われるが、コンテンツ提供者の立場から見ると、クラウドを利用したCDNの方が、従来のCDNでの定額課金よりもコンテンツ提供にかかるコストを低減できる可能性がある。また、クラウドを利用したCDNを商用サービスとして提供するプロバイダの立場から見ると、CDNサービス提供のために必要と考えられる資源を適切に選択することによって、従来のCDNよりも少ない資源でコンテンツ提供者が求めるサービスを提供できる可能性がある。

¹ 大阪大学 大学院工学研究科
Graduate School of Engineering, Osaka University

コンテンツ配信では、配信するコンテンツの種類やデータサイズなどによって重視される QoS 要求が異なる。それはユーザの満足度に強く影響する QoS 要求がコンテンツによって異なるためであり、例えば、一般的な Web ページの閲覧では遅延時間が重視されるが、動画等のストリーミング配信では遅延時間よりも転送速度が重視される。このことから、コンテンツ配信では、一般には、考慮すべき QoS 要求がコンテンツによって異なる。

しかし従来の研究では、資源選択の際に QoS 要求として遅延時間しか考慮されていなかった。そこで本研究では、クラウド上での CDN 実現において、地理的に分散して存在する多様なクラウド資源の中から、上記のような異なる QoS 要求を考慮しつつ、さらにはクラウド資源ごとに異なる利用料も考慮して、利用するクラウド資源を選択する手法を提案する。

本稿の構成は以下のとおりである。まず、2 章で本研究の想定環境と既存手法について述べる。次に 3 章では、異なる QoS 要求を考慮したクラウド資源の選択手法を提案する。さらに、4 章では既存手法と比較することで提案手法の性能評価を行う。最後に、5 章で本研究のまとめと今後の課題を述べる。

2. クラウド上での CDN

従来の CDN では、エッジサーバの配置はあらかじめ与えられており、その構成においてどのエッジサーバにコンテンツを配置すれば各コンテンツに対する QoS 要求を満足できるかという問題が研究されてきた。しかし、クラウドを利用した CDN では、地理的に分散して存在する多数のクラウド資源がエッジサーバ候補となり、そこから適切なクラウド資源を選択しエッジサーバとして利用する必要がある。その際、各コンテンツに対する QoS 要求を考慮した上で、クラウド資源の利用コストが最小となる選択を行う必要がある。

2.1 想定環境

本研究ではクラウド資源をエッジサーバ候補とみなし、多数存在するエッジ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利用するエッジサーバを選択する手法を提案する。以下ではエッジサーバを単にサーバと呼び、配信に必要な総コストを合計コストと呼ぶ。まず、想定環境を図 1 に示す。提供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。各サーバ候補を使用する際には、サーバの使用に伴う定額コスト、格納されたコンテンツのストレージ量に伴う従量コスト、サーバに流入するトラフィック量に伴う従量コスト、サーバから流出するトラフィック量に伴う従量コストが生じ、

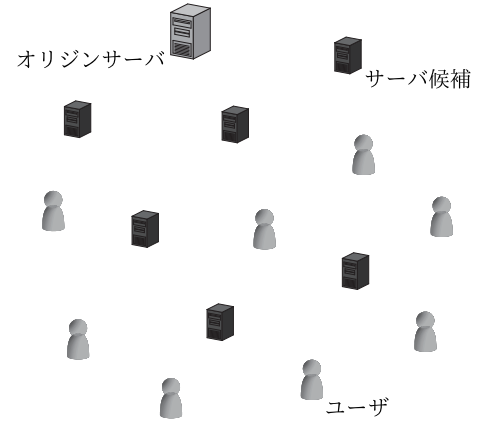


図 1 本研究の想定環境

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ぶ。また、これら 4 つのコストを総称して利用コストと呼ぶ。ここで、多様なクラウド資源を活用することを考慮し、利用コストはサーバ候補ごとに異なるものとする。

2.2 既存手法

文献 [1] では、前節の環境において各コンテンツに対する QoS 要求を満たしつつ合計コストを最小化する問題を 0-1 整数計画問題として以下のように定式化している。まず、オリジンサーバを C_0 とし、サーバ候補の集合を $C = \{C_1, C_2, \dots, C_n\}$ 、ユーザの集合を $U = \{U_1, U_2, \dots, U_m\}$ と表す。サーバ候補とユーザを合わせてノードと呼び、ノードの集合を $\mathcal{N} = \{C_0\} \cup C \cup U$ と表す。またノード u からノード v へコンテンツを転送・配信する際に、他のノードを中継することなく直接配信可能であるならば、 (u, v) をエッジと呼び、エッジの集合を $\mathcal{E}$ と表す。すなわちノード v がサーバ候補である場合には、 (u, v) がエッジであることはノード u とノード v が接続関係にあることを表し、ノード v がユーザである場合には、 (u, v) がエッジであることはノード u からノード v へと QoS 要求を満たした配信が可能であることを表す。文献 [1] では、遅延時間に関する QoS 要求を考慮する際、サーバとユーザの間の遅延時間はその距離に比例すると仮定しているため、ノード v がユーザである場合に、 (u, v) がエッジであることはノード u とノード v の距離が一定値以下であることを表す。なお一般に、 (u, v) がエッジであっても、 (v, u) がエッジであるとは限らない。

さらに、サーバ候補 C_u におけるストレージコストを S_u 、アップロードコストを P_u 、ダウンロードコストを D_u とする。文献 [1] では、サーバの使用コストは考慮されていない。また、コンテンツが古くなることを防ぐために、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。コンテンツの配信サイズを W とし、コンテンツに対するユーザ U_k からのリクエストサイズを w_k とする。このと

き、ノード u からノード v にコンテンツを配信する際にかかるコスト V_{uv} は以下のように表される。

$$V_{uv} = \begin{cases} (S_v + P_v F + D_u F)W, & v \in \mathcal{C} \\ w_v D_u, & v \in \mathcal{U} \end{cases}$$

さらに、 C_0 から U_k への配信の際にエッジ (u, v) を使用するか否かを二値変数 x_{uv}^k で表す。すなわち、 x_{uv}^k は以下で定義される。

$$x_{uv}^k = \begin{cases} 1, & U_k \text{ への配信で } (u, v) \text{ を使用する} \\ 0, & U_k \text{ への配信で } (u, v) \text{ を使用しない} \end{cases}$$

この x_{uv}^k に対し、すべてのエッジ (u, v) に対して

$$x_{uv}^k \leq y_{uv}, \forall k \in \{1, 2, \dots, m\}$$

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

$$y_{uv} = \begin{cases} 1, & (u, v) \text{ を使用する} \\ 0, & (u, v) \text{ を使用しない} \end{cases}$$

すなわち、 y_{uv} は全ユーザへの配信を考えたときにエッジ (u, v) を使用するか否かを表す変数である。この y_{uv} を用いると、全エッジについて前述の V_{uv} と掛け合わせて総和をとることで、合計コストを計算することができる。これを目的関数とすることによって、文献 [1] ではこの問題を以下の 0-1 整数計画問題として定式化している。

$$P_1: \text{最小化} \quad \sum_{(u,v) \in \mathcal{E}} y_{uv} V_{uv}$$

制約条件 $\forall k \in \{1, 2, \dots, m\},$

$$\sum_{u \in \mathcal{N}} x_{uw}^k - \sum_{v \in \mathcal{N}} x_{vu}^k = \begin{cases} 1, & u = C_0 \\ -1, & u = U_k \\ 0, & \text{その他} \end{cases}$$

$$x_{uv}^k \leq y_{uv}, \forall (u, v) \in \mathcal{E}$$

この制約条件式は、オリジンサーバから流出する配信パスと各ユーザへ流入する配信パスが1つだけ存在し、各サーバ候補では流入する配信パスと流出する配信パスの数がユーザごとに等しいことを表しており、これはオリジンサーバから各ユーザへの配信パスがユーザごとにただ1つ存在することを保証するものとなっている。

3. 提案手法

前章で述べた既存手法には、次の二つの問題点が存在する。第一の問題点は、サーバの利用コストの設定が適切でないことである。実際のクラウドサービスでは、サーバの使用に伴う定額コスト、すなわちサーバの使用コストが存在する [5]。しかしながら、既存手法の利用コストの設定

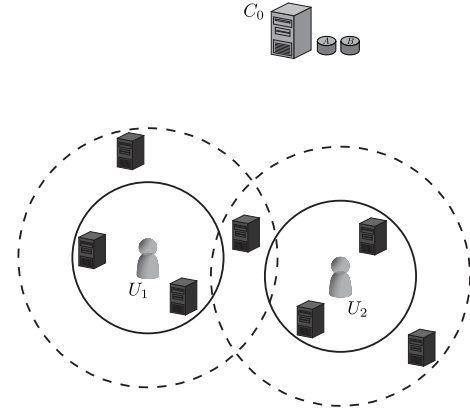


図 2 配信環境の例

にはこれが含まれていない。また、第二の問題点は、既存手法では遅延時間に関する一種類の QoS 要求しか考慮していないことである。もし、サーバの使用コストが無ければ、遅延時間に関する複数の QoS 要求に対して、それぞれの QoS 要求ごとに個別に問題 P_1 を解き、利用するサーバをコンテンツごとに定めればよい。以下では、これを既存手法と呼ぶ。しかし、サーバの使用コストが存在する場合は、すべての QoS 要求を同時に考慮しなければコストの最小化は図れない。また、遅延時間以外の QoS 要求、例えば、転送速度などを考慮する方法は議論されていない。本章では、これらの問題点を考慮した手法を提案する。

3.1 複数の QoS 要求を考慮した手法

まず、サーバの使用コストが存在し、かつ、複数の遅延時間に関する QoS 要求が存在する場合について考える。

例として、図 2 の状況において、オリジンサーバ C_0 からユーザ U_1 と U_2 へ、種類 A と種類 B の二種類のコンテンツを配信することを考える。どちらのコンテンツに対しても遅延時間に関する QoS 要求が存在し、A のコンテンツの方が B のコンテンツよりも許容される遅延時間が短いとする。図中に示した円は、遅延時間の指標としての距離の制約を表しており、各ユーザは、実線の円の内側にあるサーバからコンテンツの配信を受ける場合には A のコンテンツに対する遅延時間に関する QoS 要求が満たされ、破線の円の内側にあるサーバからコンテンツの配信を受ける場合には B のコンテンツに対する遅延時間に関する QoS 要求が満たされるとする。これらの QoS 要求を考慮した上で、図中の 7 つのサーバ候補のうち、どのサーバ候補を選択して配信を行うと合計コストを最小化できるかを考える。

このとき、二種類のコンテンツは、図 3 に示されるように同じサーバから配信することも、図 4 に示されるように異なるサーバから配信することも可能である。この配信に用いるサーバを、既存手法ではストレージコスト、アップロードコストとダウンロードコストのみを考慮して選択を行うが、提案手法ではさらにサーバの使用コストも考慮

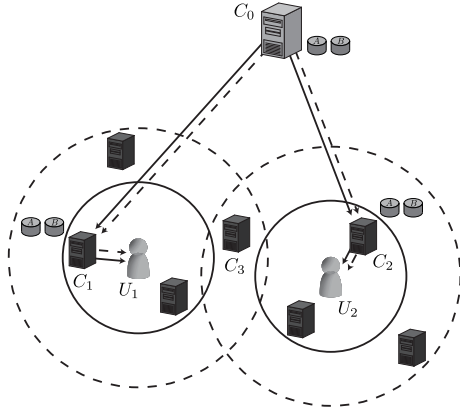


図 3 二種類のコンテンツが同じサーバから配信される例

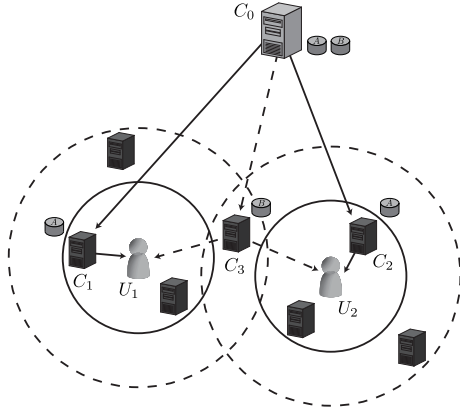


図 4 二種類のコンテンツが異なるサーバから配信される例

して選択を行うため、合計コストを低減することが可能となる。

以上の考え方にに基づき、提案手法を次のように定式化する。 L 種類のコンテンツが存在し、これらのコンテンツは遅延時間に関する異なる QoS 要求のみをもつとする。コンテンツの種類の集合を $\mathcal{L}$ 、QoS 要求の集合を $\mathcal{Q}$ とすると、 $\mathcal{L} = \{1, 2, \dots, L\}$ 、 $\mathcal{Q} = \{Q_1, Q_2, \dots, Q_L\}$ である。このとき、2.2 節の記号・変数を Q_l を考慮して再定義する必要がある。エッジの集合 $\mathcal{E}$ は、QoS 要求ごとに異なるため、 Q_l を考慮した場合のエッジの集合を $\mathcal{E}_l$ と表す。QoS 要求ごとにエッジの集合が異なると、 x_{uv}^k, y_{uv} も QoS 要求ごとに定義されることとなり、それらをそれぞれ x_{uv}^{kl}, y_{uv}^l と表す。また、QoS 要求の異なるコンテンツはサイズも異なると考えられることから、 Q_l が要求されるコンテンツの配信サイズを W_l 、 Q_l が要求されるコンテンツに対するユーザ U_k からのリクエストサイズを w_k^l と表す。ここで、オリジンサーバがコンテンツの再配信を行う頻度 F はコンテンツの種類に依存しないと仮定すると、 Q_l を考慮した場合のノード u からノード v にコンテンツを配信する際にかかるコスト V_{uv}^l は以下のようになる。

$$V_{uv}^l = \begin{cases} (S_v + P_v F + D_u F) W_l, & v \in \mathcal{C} \\ w_v^l D_u, & v \in \mathcal{U} \end{cases}$$

さらに、前述の問題点の 1 つであったサーバの使用コストを考慮するため、サーバ候補 C_u の使用コストを R_u とし、二値変数 z_u を以下で定義する。

$$z_u = \begin{cases} 1, & \text{サーバ候補 } C_u \text{ を使用する場合} \\ 0, & \text{サーバ候補 } C_u \text{ を使用しない場合} \end{cases}$$

この z_u を用いると、全サーバ候補について R_u と掛け合わせて総和をとることで、配信の際に必要なサーバの総使用コストを計算することができる。すなわち、以上の記号を用いて提案手法を定式化すると以下のようになる。

$$P_2: \text{最小化} \quad \sum_{l \in \mathcal{L}} \left(\sum_{(u,v) \in \mathcal{E}_l} V_{uv}^l y_{uv}^l \right) + \sum_{u \in \mathcal{C}} R_u z_u$$

$$\text{制約条件} \quad \forall k \in \{1, 2, \dots, m\}, \forall l \in \mathcal{L},$$

$$\sum_{(u,w) \in \mathcal{E}_l} x_{uw}^{kl} - \sum_{(v,u) \in \mathcal{E}_l} x_{vu}^{kl} = \begin{cases} 1, & u = C_0 \\ -1, & u = U_k \\ 0, & \text{その他} \end{cases}$$

$$x_{uv}^{kl} \leq y_{uv}^l, \forall (u,v) \in \mathcal{E}_l$$

$$y_{uv}^l \leq z_u, \forall (u,v) \in \mathcal{E}_l$$

3.2 異なる QoS 要求を考慮した手法

次に、上記の遅延時間に関する複数の QoS 要求に加え、遅延時間とは異なる QoS 要求も考慮した手法を提案する。ここでは、遅延時間とは異なる QoS 要求としてコンテンツの転送速度を考える。動画などのコンテンツにおいては、スムーズな視聴が期待されることから転送速度が重視されるため、これを考慮することで既存手法よりも高いユーザの満足度を得ることができる。

直接的に転送速度に関する QoS 要求を定式化するのは難しいため、各サーバの負荷はユーザからの総リクエストサイズで表されるとし、その総リクエストサイズがある一定値を超えた場合に負荷増大によって転送速度が低下するものとする。すなわち、あるサーバへの総リクエストサイズが一定値以下であるとき、そのサーバから配信されるコンテンツは転送速度に関する QoS 要求が満たされることを意味する。

具体的な例として、図 5 の状況において、オリジンサーバ C_0 からユーザ $U_1, U_2, U_3, U_4, U_5, U_6$ へコンテンツを配信することを考える。簡単のために遅延時間は一種類の場合を考えることとし、図中の各サーバ候補の周りの破線で書かれた円はその円の内側のユーザへの配信であれば遅延時間に関する QoS 要求を満たすことを意味する。

ここで C_1 の方が C_2 よりもダウンロードコストが低いサーバ候補である場合を考える。転送速度を考慮しない場合には、図 6 に示されるような結果となり、 C_1 は負荷増大によって転送速度が低下してしまう。その結果

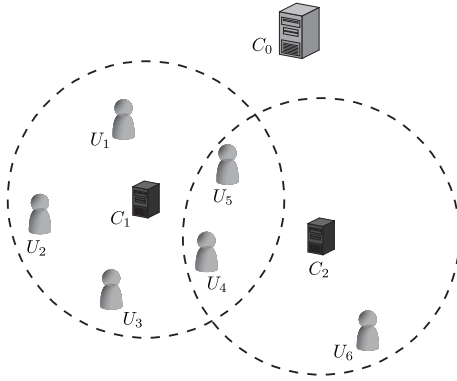


図 5 配信環境の例

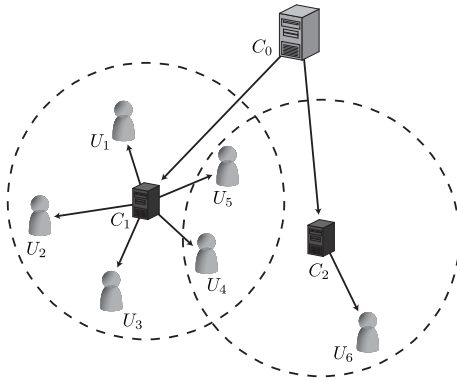


図 6 既存手法を用いた場合

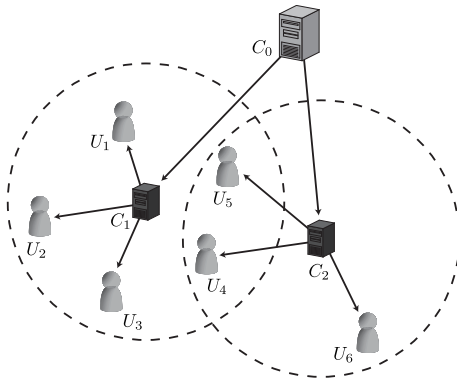


図 7 提案手法を用いた場合

U_1, U_2, U_3, U_4, U_5 は転送速度に関する QoS 要求が満たされないことになる。しかしながら、提案手法では C_1 の負荷を考慮して U_4, U_5 を C_2 からの配信とすることによって、図 7 に示されるように、 C_1 の負荷を分散し転送速度に関する QoS 要求を満たすことが可能となる。ただし、転送速度を考慮していない手法と比較すると合計コストは増加する。

以上の考え方にに基づき、提案手法を次のように定式化する。 L 種類のコンテンツが存在し、これらのコンテンツは、遅延時間に関する QoS 要求のみをもつもの、転送速度に関する QoS 要求のみをもつもの、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、の 3 つに分類されるとする。これらのうち、転送速度に関す

る QoS 要求をもつコンテンツが S 種類であるとする。このとき、全コンテンツの種類を集を $\mathcal{L}$ 、転送速度に関する QoS 要求をもつコンテンツの種類を集を $\mathcal{S}$ ($\mathcal{S} \subseteq \mathcal{L}$)、QoS 要求の種類を集を $\mathcal{Q}$ とすると $\mathcal{L} = \{1, 2, \dots, L\}$, $\mathcal{S} = \{1, 2, \dots, S\}$, $\mathcal{Q} = \{Q_1, Q_2, \dots, Q_L\}$ となる。

Q_l が要求されるコンテンツ l の配信について、二値変数 z_u^l を以下で定義する。

$$z_u^l = \begin{cases} 1, & \text{サーバ候補 } C_u \text{ を } l \text{ の配信に使用する場合} \\ 0, & \text{サーバ候補 } C_u \text{ を } l \text{ の配信に使用しない場合} \end{cases}$$

また、サーバ候補 C_u に対するユーザからの総リクエストサイズ O_u は以下のように表すことができる。

$$O_u = \sum_{k \in \mathcal{U}, l \in \mathcal{L}} w_k^l y_{uk}^l$$

サーバ候補 C_u のコンテンツ s についての負荷限界を B_u^s とすると、サーバ候補 C_u が転送速度に関する QoS 要求をもつコンテンツ s を配信する際に、 $O_u \leq B_u^s$ が成り立っているならば、そのサーバから配信されるコンテンツ s は転送速度に関する QoS 要求が満たされていることを意味する。その他の記号・変数は、3.1 節で使用したものをそのまま用いる。以上の記号を用いて、提案手法を定式化すると以下ようになる。

$$\begin{aligned} P_3: \text{最小化} \quad & \sum_{l \in \mathcal{L}} \left(\sum_{(u,v) \in \mathcal{E}_l} V_{uv}^l y_{uv}^l \right) + \sum_{u \in \mathcal{C}} R_u z_u \\ \text{制約条件} \quad & \forall k \in \{1, 2, \dots, m\}, \forall l \in \mathcal{L}, \\ & \sum_{(u,w) \in \mathcal{E}_l} x_{uw}^{kl} - \sum_{(v,u) \in \mathcal{E}_l} x_{vu}^{kl} = \begin{cases} 1, & u = C_0 \\ -1, & u = U_k \\ 0, & \text{その他} \end{cases} \\ & x_{uv}^{kl} \leq y_{uv}^l, \forall (u,v) \in \mathcal{E}_l \\ & y_{uv}^l \leq z_u^l, \forall (u,v) \in \mathcal{E}_l \\ & z_u^l \leq z_u, \forall u \in \mathcal{C}, \\ & z_u^s O_u \leq B_u^s, \forall u \in \mathcal{C}, \forall s \in \mathcal{S} \end{aligned}$$

なお、制約条件の最後の式は、

$$\sum_{k \in \mathcal{U}, l \in \mathcal{L}} w_k^l y_{uk}^l \leq z_u^s B_u^s + (1 - z_u^s) \sum_{k \in \mathcal{U}, l \in \mathcal{L}} w_k^l$$

と書き換えることで、線形化可能である。

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 0-1 整数計画問題として定式化した提案手法を CPLEX [6] を用いて解き、既存手法を用いて解いた場合と比較して性能評価を行う。

表 1 4.2.1 節における各パラメータの設定値

パラメータ	設定
サーバ候補 C_j の使用コスト (R_j)	[5000, 25000] 上の一様分布に従う
サーバ候補 C_j のストレージコスト (S_j)	[5, 25] 上の一様分布に従う
サーバ候補 C_j のアップロードコスト (P_j)	[5, 25] 上の一様分布に従う
サーバ候補 C_j のダウンロードコスト (D_j)	[5, 25] 上の一様分布に従う
オリジンサーバがコンテンツの再配信を行う頻度 (F)	1
Q_1 におけるサーバとユーザ間の距離の制約	10
Q_2 におけるサーバとユーザ間の距離の制約	20
Q_1 が要求されるコンテンツの配信サイズ (W_1)	20
Q_2 が要求されるコンテンツの配信サイズ (W_2)	40
Q_1 が要求されるコンテンツに対するユーザ U_k からのリクエストサイズ (w_k^1)	[5, 10] 上の一様分布に従う
Q_2 が要求されるコンテンツに対するユーザ U_k からのリクエストサイズ (w_k^2)	[5, 10] 上の一様分布に従う

4.1 評価モデル

本評価では、 70×70 の正方形の領域を想定し、この領域上に 49 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。

この領域内に、サーバ候補をランダムに配置する。オリジンサーバと各サーバ候補の間、ならびに各サーバ候補の間では、コンテンツを直接配信可能であるとする。また、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。

各コンテンツに対する QoS 要求は、遅延時間に関して 2 種類の QoS 要求 Q_1 と Q_2 が存在するとし、 Q_1 においてはサーバとユーザの間の距離の制約を 10、 Q_2 においては 20 と設定する。このとき、配置したサーバ候補の位置によっては、遅延時間に関する QoS 要求を満たすことのできる距離にサーバ候補が存在しないユーザが発生することがある。この場合、そのようなユーザに対してはどのサーバ候補から配信を行っても QoS 要求を満たすことができないため、すべてのサーバ候補のうち最も合計コストの低くなるサーバ候補から配信を行うものとする。このようなユーザのリクエストを QoS 違反と呼び、QoS 違反となったリクエストサイズの総和を全ユーザの総リクエストサイズで割ったものを QoS 違反率と定義する。

4.2 評価結果

本節では、前節の評価モデルに従い、複数の QoS 要求を考慮した手法の評価結果、および、異なる QoS 要求を考慮した手法の評価結果について述べる。

4.2.1 複数の QoS 要求を考慮した手法の評価

まず、複数の QoS 要求を考慮した手法について、想定領域内に配置するサーバ候補数を 1 台から 49 台まで変化させたときの合計コストと QoS 違反率についての評価を行っ

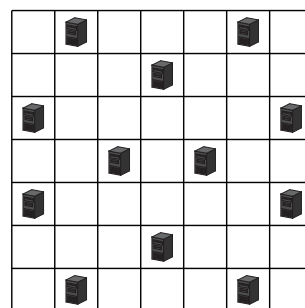


図 8 最初の 12 台のサーバ候補の配置

た。サーバ候補数を変化させる際には、サーバ候補の配置を毎回変更するのではなく、サーバ候補を領域内に追加していくことで評価を行う。ただし、最初に配置する 12 台のサーバ候補の位置はランダムではなく毎回固定位置とし、図 8 に示す通りであるとする。これら 12 台が全て使用されると QoS 違反となるリクエストが発生することなく全ユーザにコンテンツを配信することが可能となるため、主に、13 台以上のサーバ候補が存在する場合に興味がある。

評価を行う際のパラメータは、表 1 に示すように設定した。各サーバ候補の利用コストの価格差は、最大でも 5 倍である。このときの、サーバ候補数に対する合計コストと QoS 違反率の変化を図 9 に示す。

図において、破線が既存手法の、実線が提案手法の合計コストの変化であり、点線が QoS 違反率の変化である。図では、QoS 違反率が 0 % となるまで、すなわちサーバ候補数が 12 台となるまでは、合計コストは単調増加となっているが、一般には、合計コストは単調増加とはならず、増加傾向をとるものの一部減少する場合も存在する。これは、QoS 違反となったリクエストをもつユーザを、すべてのサーバ候補のうち最も合計コストの低くなるサーバ候補から配信を受けるものとしているためである。

QoS 違反率が 0 % となった以降は、提案手法の合計コストは単調減少となる。これはサーバ候補数が増えることに伴い、複数の QoS 要求を考慮することによる適切な選択の効果が表れているためである。しかしながら、既存手

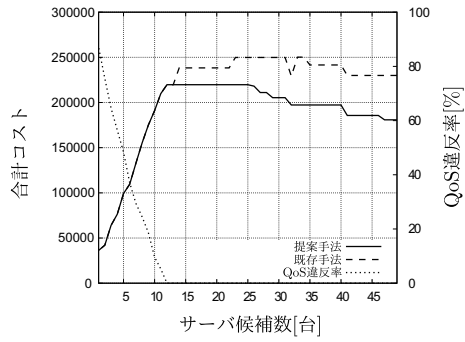


図 9 合計コストと QoS 違反率の変化 ($R_j \in [5000, 25000]$)

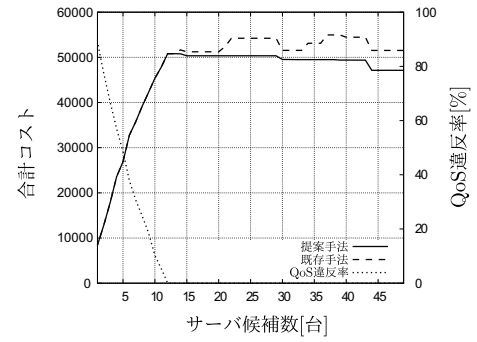


図 11 合計コストと QoS 違反率の変化 ($R_j \in [500, 2500]$)

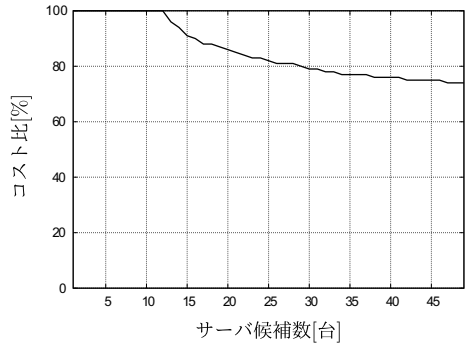


図 10 コスト比の変化 ($R_j \in [5000, 25000]$)

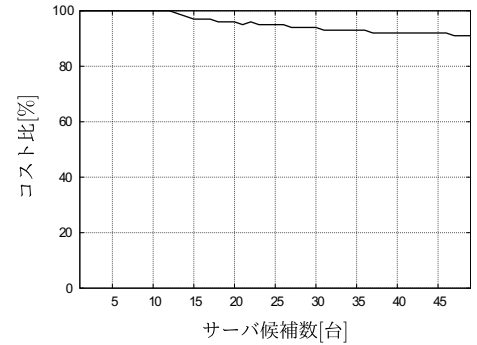


図 12 コスト比の変化 ($R_j \in [500, 2500]$)

法の合計コストは単調減少なものとはならない。これは、既存手法ではサーバの使用料を考慮せずに利用するサーバ候補の選択を行っていることによる。

さらに、同じ評価をサーバ候補の配置や各コスト、ユーザからのリクエストサイズを変えて 100 回行い、既存手法と提案手法の合計コストの平均を得た。このときの、既存手法のコストの平均に対する提案手法の合計コストの平均の割合（コスト比）を図 10 に示す。この図から、サーバ候補数が多くなるにつれて提案手法はより効果的となることがわかる。特に、サーバ候補数がユーザ数と同数程度の場合、このパラメータ設定では既存手法に対して 3/4 程度の合計コストとなる。

次に、各サーバ候補の使用コストの設定値を、表 1 の 1/10 の $[500, 2500]$ 上の一様分布に従うものとした場合について評価を行った。このときの、サーバ候補数に対する合計コストと QoS 違反率の変化を図 11 に示す。QoS 違反率が 0 % となった以降の合計コストの変化を図 9 と比較すると、サーバ候補数の増加に伴う提案手法の合計コストの下がり幅が少なく、また、既存手法の合計コストとの差も小さくなっている。同様に、同じ評価を 100 回行ったときのコスト比を図 12 に示す。図 10 と比較すると、下がり幅が半分程度になっていることがわかる。このことから、提案手法は、各サーバ候補の使用コストの設定値が大きいほど効果的である。

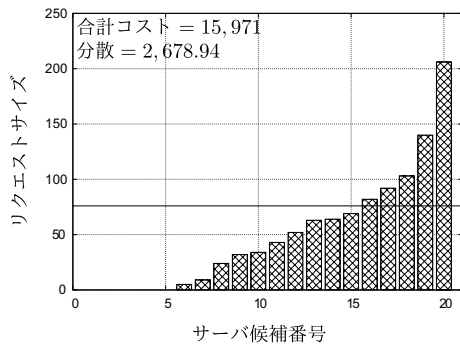
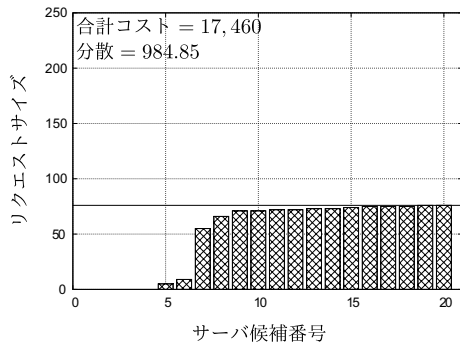
4.2.2 異なる QoS 要求を考慮した手法の評価

次に、3.2 節において提案した、異なる QoS 要求を考慮

した手法の評価を行った。サーバ候補数を 20 台に固定し、すべてのサーバ候補を領域内にランダムに配置した。また、全コンテンツについて遅延時間に関する QoS 要求と転送速度に関する QoS 要求の両方を考慮し、転送速度が低下する総リクエストサイズ B_u^s は、全サーバで同一の値 76 とした。その他のパラメータの設定値は表 2 の通りである。この条件のもとで 3.1 節で提案した手法 P_2 と比較し、各サーバ候補へのリクエストサイズと合計コストについて評価した。得られた各サーバへのリクエストサイズとその合計、さらに合計コストを図 13 と図 14 に示す。図におけるサーバ候補番号は、各サーバ候補へのリクエストサイズを昇順に並べかえたものである。図 13 と図 14 を見ると、転送速度を考慮しない場合はサーバへの要求量が 200 を超えているサーバがあるが、転送速度を考慮した場合はすべてのサーバ候補へのリクエストサイズを設定した一定値以下に抑えられていることがわかる。転送速度を考慮した場合には合計コストが 9.3 % 程度増加したが、分散の値を 37 % 程度に抑えることができた。合計コストが増加した原因は、ダウンロードコストの低いサーバの使用が制限されたことによって、サーバからユーザへのコンテンツ配信にかかるコストが増加したためである。また、図から選択されたサーバ数が増加したことがわかるが、それによってサーバの使用コストの合計とサーバからサーバへのコンテンツ配信にかかるコストが増加したことも合計コストの増加した原因である。

表 2 4.2.2 節における各パラメータの設定値

パラメータ	設定
サーバ候補 C_j の使用コスト (R_j)	[150, 250] 上の一様分布に従う
サーバ候補 C_j のストレージコスト (S_j)	[5, 15] 上の一様分布に従う
サーバ候補 C_j のアップロードコスト (P_j)	[5, 15] 上の一様分布に従う
サーバ候補 C_j のダウンロードコスト (D_j)	[5, 15] 上の一様分布に従う
オリジンサーバがコンテンツの再配信を行う頻度 (F)	1
Q_1 におけるサーバとユーザ間の距離の制約	10
Q_2 におけるサーバとユーザ間の距離の制約	20
Q_1 が要求されるコンテンツの配信サイズ (W_1)	5
Q_2 が要求されるコンテンツの配信サイズ (W_2)	10
Q_1 が要求されるコンテンツに対するユーザ U_k からのリクエストサイズ (w_k^1)	[5, 15] 上の一様分布に従う
Q_2 が要求されるコンテンツに対するユーザ U_k からのリクエストサイズ (w_k^2)	[5, 15] 上の一様分布に従う

図 13 各サーバ候補へのリクエストサイズ, その合計コストと分散 (P_2 を用いた場合)図 14 各サーバ候補へのリクエストサイズ, その合計コストと分散 (P_3 を用いた場合)

5. おわりに

本論文では、クラウド上での CDN 実現において、クラウド資源の利用コストにサーバの使用コストを加えたうえで、複数の QoS 要求を考慮したクラウド資源の選択手法を提案した。また、異なる QoS 要求を考慮した資源の選択手法の提案も行った。複数の QoS 要求を考慮した資源選択手法については、サーバの使用料を考慮せず単一の QoS 要求のみを考慮して資源選択を行った結果を重ね合わせる手法に対して、ユーザ数と同数程度のサーバ候補数の場合は約 3/4 程度のコストとなることを示した。その際、サーバの使用料が大きいほうが提案手法は有効であることも示し

た。さらに、コンテンツの配信の遅延時間だけでなくコンテンツの転送速度も考慮に入れた手法では、サーバへの総リクエストサイズを一定値以下に抑えることはできたが、コストが 9.3 % 程度増加することを示した。

今後の課題としては、次のようなことが考えられる。まず、今回はコンテンツに対する各ユーザからのリクエストサイズが既知である場合について考えたが、実際にはそれはリクエストされるまでわからない。そこで、ユーザからのリクエストサイズが未知であり、リクエストされて初めてわかる場合におけるクラウド資源の選択手法を考える必要がある。さらに、今回のような整数計画問題として定式化して解く手法は、ノード数が増加すると計算時間が爆発的に増加してしまい、多数のサーバやユーザの存在する大規模な問題においては現実的な運用が不可能である。したがって、大規模な問題においても最適解に近い結果を得ることのできる発見的な手法を提案する必要がある。

参考文献

- [1] Chen, F., Guo, K., Lin, J., et al.: Intra-cloud Lightning: Building CDNs in the Cloud, *Proc. IEEE INFOCOM*, pp.433–441 (2012).
- [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*, pp.199–207 (2012).
- [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] 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).
- [6] IBM: IBM ILOG CPLEX Optimization Studio, IBM (オンライン), 入手先 (<http://www-03.ibm.com/software/products/ja/ibmilogcpleoptistud>) (参照 2016-05-11).