

# エッジコンピューティングを考慮したクラウドゲーミングシステムに関する研究

2016SE072 角谷維 2016SE081 田中 渉太

指導教員：宮澤元

## 1 はじめに

クラウドコンピューティング(クラウド)技術を用いたクラウドゲーミングと呼ばれるコンピュータゲームの利用形態が注目されている。クラウドゲーミングでは、コンピュータゲームをクラウドサーバ上で実行することにより、ユーザがゲーム用の高性能なハードウェアを用意したり、コンピュータゲームを自分でインストールしたりする必要がなくなるといった利点がある一方で、ユーザとクラウドサーバ間の通信遅延や、多数のユーザが同時にゲームをプレイしてクラウドへの通信が集中することによってクラウドの計算能力や通信帯域に負荷がかかるという問題が発生する。

エッジコンピューティング技術をクラウドゲーミングに適用することでこうした問題を解決しようという提案もなされている [1]。しかし、一口にコンピュータゲームと言っても様々な形式のものがああり、これら全てについてエッジコンピューティング技術をどのように適用できるかが具体的に検討されているわけではない。

本研究の目的は、エッジコンピューティング技術を効果的に適用できるソフトウェア構成をゲームの種類別に具体的に示すことである。ゲームを構成するソフトウェアをエッジサーバとクラウドサーバに適切に配置することによって、通信遅延やクラウドへの負荷集中の問題を解決できる。

本稿では、2DRPG, 3D アクション, MMORPG の 3 種類のゲームについて、エッジサーバを利用するソフトウェア構成について具体的に示す。各構成における通信遅延をシミュレーションにより測定し、エッジサーバを利用しない構成と比較した場合の優位性を示す。

## 2 研究の背景

ここではクラウドゲーミングにおける課題と、その課題を解決する手段の一つであるエッジコンピューティング、エッジコンピューティングの有用性を確かめるツールの iFogSim と呼ばれるシミュレータについて示す。

### 2.1 クラウドゲーミングとその課題

クラウドゲーミングは、クラウド内でゲームサーバを所有し、ユーザがこのクラウドにアクセスをしてストリーミング形式でゲームを楽しむサービス形態のことである。クラウドゲーミングにおいて一番の課題が遅延である。ゲームというものは多種多様で様々なルールのもとユーザが楽しめるものとなっているが、リアルタイム性が求められる

ゲームにおいて遅延が生じてしまうと、ユーザ間での信頼性を失ってしまい、ゲームとしての楽しさも失われてしまう。また、ユーザが増えた際には情報量も増えていき、1 つあたりのクラウドサーバにおける負荷がとて大きくなってしまふ。このような場合、コストが大きくなってしまふのも課題の一つであると考えられる。そのほか、ストリーミング転送に用いる動画圧縮の手法にも課題がある。例えば、クラウドとデバイス間の解像度の違いに応じて転送データ量を最小化する必要がある。

クラウドゲーミングにエッジコンピューティングの技術を取り入れることで、目的である遅延や一極集中になってしまう課題を軽減することができる。クラウドゲーミングにエッジコンピューティングを適用しようとする先行研究ではゲーム映像のレンダリングをエッジに負担させることでクラウドの負担を軽減することを試みている [1]。また、コンピュータ処理の程度に着目した先行研究では、必要な処理が多いアプリケーションを動作させた場合には、クラウドやエッジを用いないローカルなコンピュータで動作させたほうが効率が良くなることが示されている [4]。

### 2.2 エッジコンピューティング

エッジコンピューティングとは、ユーザがソフトウェアを用いてクラウドにデータを送受信する際に、利用者からネットワーク的に近い場所にある計算リソースを用いることで、クラウドの遅延を低減する分散コンピューティングモデルである。

エッジと呼ばれる計算リソースを利用して、データセンタに送られるデータを事前に処理をすることによって、データ量を減らし、クラウドの負荷を軽減することができる。実際の利用形態としては、IoT デバイスにこの手法が用いられており、センサから与えられた生データを元にエッジが解析し、この解析されたデータをクラウド上に送ることで遅延を減らすことや一極集中になることを抑えることができる。

### 2.3 iFogSim

iFogSim は、エッジコンピューティングの環境をモデル化し、レイテンシ、ネットワークの輻輳、エネルギー消費量、運用コストなどに焦点を当てて、シミュレーションすることが可能なオープンソースキットである [2]。エッジデバイス、クラウドデータセンター、およびネットワークリンクを設定しシミュレーションをしてパフォーマンスメトリックを収集することが可能である。iFogSim は Cloudsim の連携プロジェクトであり、Cloudsim のレイ

ヤーが iFogSim 内に存在している。iFogSim は Cloudsim と連携をしてシミュレーションを行っている。

Cloudsim は様々なクラウド環境下において、クラウドコンピューティングインフラストラクチャやアプリケーションのモデル化、シミュレーションを可能にするためのフレームワークの一つである [3]。iFogSim はこの Cloudsim 内にあるシミュレーション機能と連携を取りながらエッジコンピューティングのモデリング及びシミュレーションを可能にしている。

図 1 に、iFogSim の GUI を用いたトポロジーエディタの実行画面を示す。このプログラムは、VR ゲームにおいてエッジを導入したトポロジーの例である。この図は EEG と呼ばれる脳波検知するデバイスを用いて二人で対戦する VR ゲームのトポロジーである。この図の中の proxy-server というデバイスがエッジに当たる。図の d1, m-0-0, m-0-1 までが一つの EEG デバイスに当たる。一人のプレイヤーに対して m-0-0, m-0-1 を装着し、s-0-0, s-0-1 で情報をまとめ、そのデータを proxy-server へと送る。受け取った情報をもとに proxy-server が判定などの計算を行い、この情報を cloud へと転送する。cloud は受け取った情報をもとに映像を描写し、各それぞれのユーザへと映像を届けている。この図の a-0-0, a-0-1 が一人のプレイヤーの映像出力の部分に当たる。

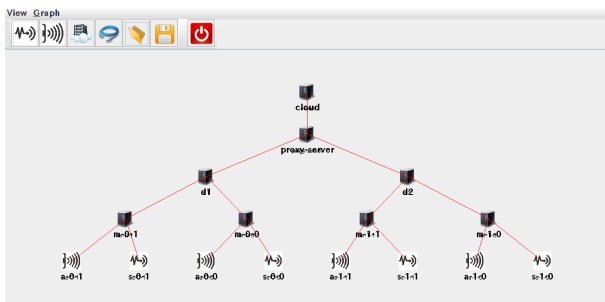


図 1 iFogSim を用いた VRgame におけるサンプルプログラム

### 2.3.1 iFogSim のクラス

iFogSim 内で用意されているクラスは以下のものがある。

- Fog デバイス
- センサー
- アクチュエータ
- タブル
- アプリケーション

これらを組み合わせることで、調べたい構成のシミュレーションをすることが可能になっており、iFogSim 内のアプリケーションを実行させることで GUI による構成の作成を簡単に行うことができる。

## 3 エッジゲーミング実現手法の検討

我々はエッジコンピューティング技術を効果的に適用したクラウドゲーミングをエッジゲーミングと呼ぶ。エッジゲーミングにおける適切なソフトウェア構成を示すために、ゲームの特徴に応じて既存のゲームをいくつかに分類し、分類ごとにソフトウェア構成を提案する。これは、ゲームの種類や構成によって必要なリソースが異なるので、適切なソフトウェア構成も異なるはずだと考えられるからである。例えば、2D のアクションゲームに 3D のハイエンドなゲームをするためのクラウドのリソースを与えても無駄になってしまい、コストも大きくなってしまう。

### 3.1 分類する際に考慮すべきゲームの特徴

本項では、クラウドゲーミングにおいてエッジを利用する場合に考慮すべきゲームの特徴を列挙していく。

#### 3.1.1 ゲームプログラム容量

エッジに配置できるプログラム容量には限りがあり、必要に応じてゲームプログラムを分割して動作させる必要がある。2D ゲームと 3D ゲームを比較すると、扱うデータのサイズが大きい分 3D ゲームのほうが容量が大きくなってしまふ。また、ゲーム内容がより充実したものになることで追加されるデータの数が増えていき容量の増加につながる。

#### 3.1.2 グラフィック処理

ゲームにおいてグラフィック処理は重要なセクションである。グラフィック処理に求められる性能は扱うゲームの種類によって変わってくる。例えば、2D ゲームと 3D ゲームでは 1 画面に表示されるオブジェクトの数や精度が違い、高精度な画像やモデルを使用していたり表示されるオブジェクトの数が多かったり画面の更新頻度が高かったりするとより高いグラフィック処理が求められる。

#### 3.1.3 リアルタイム性

クラウドゲーミングにおいて、どこまで遅延が許されるかということは考慮しなければならない重要な点である。コマンドをただ順番に入力するようなゲームではリアルタイム性があまり求められておらず、逆にシューティングゲームでは少しの遅延がゲームの品質に大きく左右する。

#### 3.1.4 他ユーザとのインタラクション

オフラインでプレイするゲームは、ユーザの操作から映像の更新までがそのユーザー人で完結するが、オンラインマルチプレイのゲームに関しては、他ユーザが行った操作の結果と自分の操作の結果を同期して映像を更新する必要がある。

### 3.2 今回想定した構成

前項で述べた特徴をもとに、別々の特徴を持つ 3 タイプのゲームを想定し、エッジを用いる際の構成を示した。

### 3.2.1 2DRPG ゲーム

2DRPG ゲームはその他ゲームと比べた際に、容量が少なく遅延の影響を受けにくく、かつ映像のエンコードなどにかかる時間も比較的少ないゲームを想定する。また、ここではゲームプログラムはクラウドからエッジへインストールしてあるものとする。

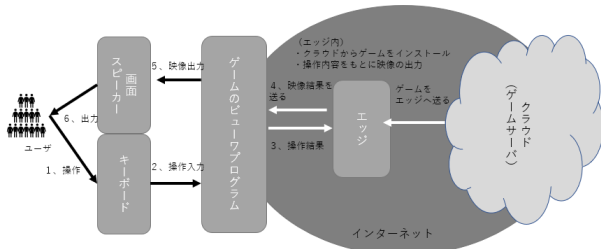


図2 2DRPG ゲームの概略図

図2は2DRPGゲームにおける概略図である。ユーザはキーボードなどのデバイスを用いて、操作を行う。操作された内容をビューワプログラムへ送る。操作結果をエッジへ送信する。エッジ内で操作された情報を元にゲームプログラムを動作させる。映像をレンダリングし、その結果をビューワプログラムへ送信する。映像内容の出力を行い、出力された映像を元にユーザはゲームをプレイすることができる。

### 3.2.2 3D アクションゲーム

3D アクションゲームは、前述した2DRPGゲームと比較した際に、容量が多く遅延の影響を受けやすい3Dのゲームを想定する。

2Dのゲームとは違い、高画質の映像のレンダリングをする必要があるため、映像のレンダリングを行う部分では、高いグラフィック演算処理をすることができると想定する。

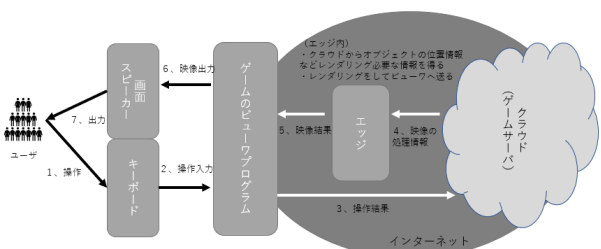


図3 3Dゲームにおける配置のイメージ

図3は3Dアクションゲームにおける概略図である。ユーザはキーボードなどのデバイスを用いて、操作を行う。操作された内容をビューワプログラムへ送る。操作結果をクラウドへ送信し、クラウド内で操作された情報を元にゲームプログラムを動作させる。クラウドで処理した

情報をエッジに送信し、エッジ内で映像のレンダリングを行う。レンダリングが終了した映像をビューワプログラムへ送信する。映像内容を画面やスピーカーなどに出力し、ユーザは出力された映像を元にゲームをプレイすることができる。

### 3.2.3 MMORPG

MMORPGは上記で示した3Dゲームに加え、ユーザが多数いるサーバに接続をしてその他ユーザと干渉もしながら遊ぶことのできるゲームを想定する。

このMMORPGがその他ゲームと違う点はデータの保存などの簡単な処理などは、エッジ内で行うのではなく、クラウドサーバ内で行う点にある。このように分散処理を行うことで、3Dアクションゲームと比較した際に効率が上がると想定している。

## 4 シミュレーション

3節で示したソフトウェア構成をもとにiFogSimを用いてシミュレーションを行う。シミュレーションは以下の3つをおこなった。なお、シミュレーションを行う際には、シングルクライアントを介してシステムにログインし、エッジサーバもしくはクラウドサーバにゲームプログラムがインストールされ、ゲームプログラムが実行されていると想定している。

### 4.1 エッジの有無によるネットワーク遅延の違い

センサーから受け取った情報などをエッジ内で処理するとクラウドとエッジ間のデータのやり取りが少なくなり、ネットワーク遅延も軽減できると考えられる。そこで、エッジの有用性を確かめるためのシミュレーションを行う。3Dアクションゲームについて、エッジを採用しない構成と採用する構成のそれぞれでシミュレーションを複数回行った。平均のネットワーク遅延を表1に示す。エッジを採用する構成の方がネットワーク遅延を軽減することができるが分かる。

表1 エッジの有無によるネットワーク遅延の違い

エッジ無し	エッジ有り
2709.3 ms	128.6 ms

### 4.2 3D アクションゲームにおけるエッジの効果

3Dアクションゲームについて我々が提案したソフトウェア構成がネットワーク遅延の削減に有効であることを確かめるためにシミュレーションを行う。3Dアクションゲームについて、3.2.2節に示したネットワーク構成で実現した場合と、3.2.1節に示したような2Dロールプレイングゲームに適したネットワーク構成で実現した場合を想定して、それぞれの構成でシミュレーションを行う。それぞれの構成でネットワーク距離が等しくなるように設定した(表3, 4)。なお、3Dゲームのネットワーク構成はそ

他のゲームのソフトウェア構成と比較して特殊であり、iFogSim の GUI エディタでは直接作成できないので、文献 [5] を元に手作業で作成した。シミュレーションを複数回行い、平均のネットワーク遅延を表 2 に示す。3D アクションゲームは、それに適したネットワーク構成で実行した場合の方が、2DRPG に適した構成で実行するより低遅延になることが確かめられた。デバイス-エッジ間のネットワーク遅延を比較すると、エッジ内でデータを処理し、送信するデータ量に変化があるのでこのような結果が得られたと考えられる。

表 2 各ネットワーク構成で 3D アクションゲームを実行した時のネットワーク遅延の違い

ネットワーク構成	デバイス-エッジ間 [ms]	エッジ-クラウド間 [ms]
2DRPG	51.1	129.5
3D アクションゲーム	28.5	27.4

表 3 2DRPG におけるネットワーク構成

通信路	cpu length	network length
viewer edge	1000	100
edge cloud	1000	100
cloud edge	1000	100
edge viewer	1000	100

表 4 3D アクションゲームにおけるネットワーク構成

通信路	cpu length	network length
viewer cloud	1000	133
cloud edge	1000	133
edge viewer	1000	133

#### 4.3 MMORPG におけるエッジデバイス数の影響

表 5 エッジデバイスの数を変動させた際のネットワーク遅延

エッジの数 (個)	デバイス-エッジ間 [ms]	エッジとクラウド間 [ms]
1	3885.4	5125.5
2	2652.7	5196.9
5	215.0	4231.9
10	2443.3	4658.9

MMORPG を想定して、エッジデバイスの数を変動した処理を行い、かかったネットワーク遅延を示す。測定を行った結果を表 5 に示す。エッジとクラウド間を比較した際に、エッジデバイスの数を増やしていくとネットワーク遅延は減少していくが、1 つ目のシミュレーション結果のようにネットワークの遅延に差が大きく出るというわけではなかった。

## 5 おわりに

我々はクラウドゲーミングにエッジコンピューティングを適用した場合を想定し、3 種類のゲームについてエッ

ジを効果的に利用できるソフトウェア構成を提案した。シミュレーションによって、これらの構成がネットワーク遅延の削減に有用であることを示した。3D アクションゲームではエッジを利用したソフトウェア構成とすることで、遅延を低減することを確認した。また、クラウドゲーミングの品質を高めるために、エッジをたくさん投げればよいのではと考え、MMORPG などの大人数が接続した環境下で、エッジの数を増やし実験を行ったが、多くのエッジを投げたとしても低遅延に関する劇的な変化はあまり見られなかった。

今後の課題として、実際の環境で運用をしていく上では運用コストやビジネス性といった他の面についても検討を行う必要性がある。

## 参考文献

- [1] Xu Zhang, Hao Chen, Yangchao Zhao, Zhan Ma, Yiling Xu, Haojun Huang, Hao Yin, and Dapeng Oliver Wu, "Improving Cloud Gaming Experience through Mobile Edge Computing," in *IEEE Wireless Communications*, pp. 1–6, 2019.
- [2] Harshit Gupta, Amir Vahid Dastjerdi, Soumya K. Ghosh, Rajkumar Buyya, "iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments," in *Journal of Software: Practice and Experience*, Vol 47, Issue 9, pp. 1275–1296, 2017.
- [3] "CloudSim: A Framework For Modeling And Simulation Of Cloud Computing Infrastructures And Services," <http://www.cloudbus.org/cloudsim/>.
- [4] An Qin, Chengcheng Cai, Qin Wang, Yiyang Ni, Hongbo Zhu, "Game Theoretical Multi-User Computation Offloading for Mobile-Edge Cloud Computing," in *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 328–338, 2019.
- [5] Redowan Mahmud, Rajkumar Buyya, "Modelling and Simulation of Fog and Edge Computing Environments using iFogSim Toolkit," in *Fog and Edge Computing: Principles and Paradigms*, Chapter 17, 2018.