

# エッジコンピューティングにおける CPU バウンドサービスへのアクセス時間の比較

2013SE085 小林和樹

指導教員：宮澤元

## 1 はじめに

家電製品や自動車など、これまでネットワークに接続されていなかった身の回りのあらゆるモノをインターネットに接続する Internet of Things(IoT) が普及しつつある。

多数の IoT デバイスが生成する大量のデータを処理するために、クラウドコンピューティング (クラウド) に代わってエッジコンピューティングが注目されている。エッジコンピューティングとは、利用者からネットワーク的に近距離にある計算リソース (エッジノード) をさまざまなデータ処理に活用する技術である。

エッジコンピューティングを用いて IoT デバイスのデータ処理を行う場合、エッジノードの多様性を考慮する必要がある。エッジコンピューティングでは、PC のように比較的高性能な計算ノードがエッジノードとして利用されることもあれば、ネットワークルータなどの組込機器がエッジノードとして用いられることもありうる。同じサービスインスタンスを利用する場合でも、そのサービスインスタンスが動作しているエッジノードによって処理性能が異なることがあるので、エッジコンピューティングを効率的に利用するためには、動作しているエッジノードに基づいて利用するサービスインスタンスを選択する必要がある。

本研究の目的は、エッジコンピューティングにおいて利用される計算ノードによって CPU バウンドサービスの処理性能が異なることを示すことである。研究課題は以下の 2 点である。

- CPU バウンドのサービスインスタンスが動作する計算ノードの違いによる処理時間の違いを確認する。
- エッジコンピューティングにおけるサービスインスタンスの選択手法について考察する。

本研究では、異なる種類の計算ノードを利用するコンテナオーケストレータを用いて、CPU バウンドのサービスインスタンスにアクセスする実験を行い、サービスへのアクセス時間を計測する。サービスインスタンスが動作する計算ノードの違いによる処理時間の違いについても調べる。実験結果に基づき、エッジコンピューティングにおけるサービスインスタンスの選択手法について考察する。

## 2 研究の背景

### 2.1 IoT

Internet of Things(IoT) と呼ばれるあらゆるモノをインターネットに接続する技術が普及している。IoT によりこれまではインターネットに接続されていなかった家電製

品や自動車などがインターネットに接続されるようになった。IoT によって、遠く離れたモノを操作したり、モノに搭載されたセンサを用いてデータを収集したり、モノの状態を調べたりすることができる。IoT で接続されるモノの多くは十分な計算リソースを持たないので、クラウドのような外部の計算リソースを利用して収集したデータの処理を行うことが普通である。

### 2.2 Kubernetes

Kubernetes[1] は、コンテナ化されているアプリケーションの運用や管理を行うコンテナオーケストレータである。Kubernetes の基本的なアーキテクチャを図 1 に示す。全体の制御を担当している Kubernetes Master にコンテナのスケジューラやコントロール、クラスタ全体の管理をしている etcd が存在している。アプリケーションコンテナはノードで動作し、ポッドと言う単位で管理されている。ポッドには、コンテナ、ネットワークおよび共有ストレージが含まれる。1 つのポッドに複数のコンテナが含まれることもある。ノードを管理するための Kubelet やネットワークを扱う Kube-proxy もノードで動作している。

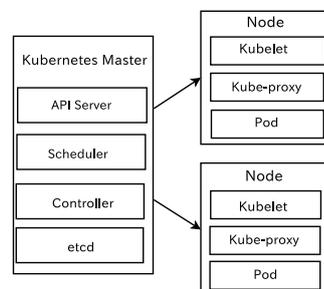


図 1 Kubernetes のアーキテクチャ

### 2.3 エッジコンピューティング

IoT によってクラウドに送られるデータが増えることでネットワークトラフィックやデータの処理負荷が増加しクラウドの動作が遅くなることがありうる。そこで、クラウドにデータを送る前に利用者にとってネットワーク的に近い場所にある計算リソースを用いて処理を行い、クラウドへの処理負荷やネットワークトラフィックを軽減するエッジコンピューティングが提案された。

エッジコンピューティングにおいてもクラウドと同様にコンテナによるアプリケーション実行環境を提供できるエッジコンピューティング基盤が必要である。これにより IoT から送られるデータ量やクラウドやエッジの計算ノード

ドの処理負荷などを考慮して、クラウドとエッジの計算リソースを効率的に利用できる。

### 3 エッジコンピューティング基盤における計算ノードの処理性能

エッジコンピューティングで利用される計算ノードの処理性能は均一ではない。利用者にとってネットワーク的に近くにある計算リソースを用いて処理を行うので、利用者ごとに利用する計算ノードが異なる。計算ノードごとに利用できる計算リソースも異なるので処理性能も異なる。そこで、計算ノードの処理性能を考慮して処理を行う計算ノードを選ぶ必要がある。この際、どのような処理を行うかによっても選ぶべきノードが変わる可能性がある。例えば、計算負荷が大きな処理を行う場合と入出力負荷が大きな処理を行う場合では求められる計算リソースが異なるので、処理を行うべきノードも異なる。

## 4 実験

計算ノードの CPU 性能の違いが利用者ノードからのアクセス時間に与える影響を調べるために実験を行う。CPU バウンドのサービスを Kubernetes クラスタで動作させ、利用者ノードからサービスを利用する際のアクセス時間を計測する。サービスが動作する計算ノードの違いによるサービスアクセス時間の違いを調べる。

### 4.1 実験の詳細

サービスとして、http でアクセスすると円周率を 2000 桁計算して結果を返すプログラムを利用した。計算ノードにラベルを付け、サービスを実行する Pod が配置される計算ノードを明示できるように設定した。利用者ノードからサービスへのアクセスには curl コマンドを利用した。サービスアクセス時間は、シェルの time コマンドを使って計測した。同じ内容の実験を 10 回ずつ行い、平均値を結果として用いた。

### 4.2 実験環境

マスターノードと 2 台のクラウドノードに PC を、2 台のエッジノードに NVIDIA Jetson Nano を用いてそれぞれ 1000Base-T Ethernet で接続した環境を用いる。実験環境の詳細を表 1 で示す。

表 1 実験環境の詳細

計算ノード	PC	Jetson Nano
OS	Ubuntu20.04 LTS	Ubuntu18.04 LTS
カーネル	5.4.0-65-generic	4.9.140
CPU	Intel Core i7-7700K	NVIDIA Tegra X1
メモリ	32GiB	4GiB

表 2 サービスアクセス時間 (秒)

ホスト名	クラウド		エッジ	
	ノード 1	ノード 2	ノード 1	ノード 2
平均値	1.21	1.20	4.99	4.99
標準偏差	0.004	0.005	0.010	0.016

### 4.3 実験結果

実験結果を表 2 に示す。クラウドノードとエッジノードの平均値で比べるとおよそ 3.8 秒の差があり、クラウドノードが約 4.1 倍速い結果となった。クラウドノードとエッジノードの標準偏差の値を比べると、エッジノードへのアクセス時間の方がクラウドノードへのアクセス時間よりややばらつきが大きい。

### 4.4 考察

エッジコンピューティングでサービス選択をする際に、サービスが動作するノードの CPU 性能の違いを考慮する必要がある。また実際のエッジコンピューティング環境では、エッジノードの方がクラウドノードより利用者ノードにネットワーク的に近い場所に存在するので利用者ノードからの通信レイテンシが小さい可能性が高い。計算ノードとの通信レイテンシの違いによりサービスアクセス時間も影響を受けるので、エッジノードの性能だけでなく通信レイテンシを考慮することでエッジコンピューティングにおけるサービス選択を改善できると考える。

## 5 おわりに

本研究では、エッジコンピューティングにおける計算ノードによる CPU バウンドサービスへのアクセス時間の違いについて実験を行った。クラウドノードとエッジノードのアクセス時間におよそ 4.1 倍の差があったことから CPU の性能が高ければアクセスにかかる時間が短くなることがわかった。このことから、エッジコンピューティングにおけるサービス選択に際して、サービスの動作ノードの CPU 性能の違いを考慮する必要があると考えられる。

今回の実験環境と異なり、本来のエッジコンピューティング環境では利用者ノードとアクセスするノード間の通信レイテンシがノードごとに異なるので、アクセスするノードに応じてアクセス時間も変わる。サービス選択にあたってはアクセスするノード性能の他に通信レイテンシも考慮する必要がある。

今後は、通信レイテンシを設定できる環境で実験を行い、エッジコンピューティング基盤におけるサービス選択のための研究を行う。

### 参考文献

- [1] Kubernetes の概要: <https://kubernetes.io/ja/docs/concepts/overview/> (2021/2/12 アクセス)。