

# IoTのためのFogコンピューティング基盤構築に関する研究

2013SE234 和田敦皓 2013SE209 竹内健人

指導教員：宮澤元

## 1 はじめに

すべてのモノをインターネットにつなげる Internet of things(IoT)の技術が広まってきている。IoTとは、情報機器だけではなく、世の中の様々なモノ(IoTデバイス)に通信機能を持たせてインターネットに接続するという概念のことを指す。IoTデバイスにセンサやアクチュエータを持たせることにより、身の回りの全てをインターネット経由で監視したり制御したりすることができる[2][6]。IoTデバイスのセンサで取得されたデータはクラウドコンピューティング(クラウド)上のIoTサービスによって収集され、その分析結果がIoTデバイスのアクチュエータの制御に用いられる。IoTサービスをクラウドを用いて提供することにより、高い信頼性と安定性を実現できる。

IoTサービスをクラウドで提供することによる問題点の一つはレイテンシである。一般にIoTデバイスはクラウドから地理的に離れた場所に設置される。クラウドとIoTデバイス間の距離が大きいくほど、通信レイテンシも大きくなる[3][5]ので、クラウドを用いて高いリアルタイム性が求められるようなIoTサービスを提供するのは困難である。

このようなIoTサービスを実現するために、IoTデバイスとクラウドの間にフォグと呼ばれる分散処理環境をIoTデバイスからネットワーク的に近いところに設置するという、フォグコンピューティングが提案されている[1][4][7]。IoTサービスをフォグ上で提供することにより、IoTサービスの通信遅延を大幅に抑え、10ms以内の応答を要求するリアルタイムIoTサービスを実現することができる。

一方、IoTサービスを利用するIoTデバイスは利用者と共に移動するので、実行環境の位置や状況が変化する。このような場合に、フォグとIoTデバイス間のレイテンシを考慮して、フォグを適切に利用するIoTサービスを実現することは非常に難しい。フォグにおけるIoTサービスの処理を、物理サーバ内で起動された仮想マシン(VM)で行い、必要に応じて適切な物理サーバにマイグレーションするフォグコンピューティングの枠組も提案されている[1]が、具体的に実装する上では様々な課題があると考えられる。

本研究では、IoTを考慮したフォグコンピューティング基盤を提案する。ユーザの要求やサービスの属性情報を用いてVM配置を最適化することができる。フォグコンピューティング実現に際しての課題を明確にするために、ネットワークレイテンシとIoTアプリケーションの応答時間の関係について調べる実験を行う。

## 2 研究の背景

本節では、IoTとクラウド/フォグコンピューティングの関係について延べ、構築すべきフォグコンピューティン

グ基盤の大まかなアーキテクチャを示す。

### 2.1 クラウドサービスとIoT

従来から、様々なITサービスがクラウドコンピューティングを利用して提供されてきた。サービスの利用者はPCやスマートフォンなどの手元のデバイスを通じて、これらのサービスを利用できる。

こうした情報機器を対象に提供されるクラウド上のITサービスの多くは、レイテンシに対する要求がそれほど高くないのに対し、IoTサービスでは高いリアルタイム性が要求されるものがある。例えば、車両の制御を行うようなIoTサービスでは、センサの情報を分析し、リアルタイムに制御を行わなければならない。

このような高いリアルタイム性を必要とするIoTサービスをクラウド上で実現すると通信レイテンシが大きすぎてリアルタイム制約を満たせない問題が発生してしまう。クラウド上で提供されるサービスの応答時間は、典型的には数十msから数百msであり、デバイスの制御を行うようなリアルタイム制約が厳しいアプリケーションの実現は難しい。

### 2.2 フォグコンピューティング

クラウドにおけるレイテンシの問題を解決するために、フォグコンピューティングが提案されている[1][4][7]。フォグコンピューティングとは、IoTデバイスにネットワーク的に近い位置に設置されたフォグと呼ばれる分散処理環境を利用してIoTサービスを実現するコンピュータの利用形態を指す。IoTサービスのレイテンシを小さくすることができ、ネットワーク帯域の制約も回避しやすくなるという利点がある。具体的には、フォグでデータの集約やアプリケーションの実行を行いサービスを提供することで、レイテンシを数ms程度に収めることができる。

### 2.3 仮想マシンとフォグコンピューティング

フォグコンピューティングで低レイテンシのサービスを提供できるのは、サービスを提供するフォグがユーザからネットワーク的に近くに位置しているからである。従って、ユーザの地理的な位置が移動するような場合でも低レイテンシでサービスを提供するためには、ユーザの地理的な位置に関わらず、常にユーザからネットワーク的に十分近い距離にあるフォグでサービスを提供する必要がある。

このような形でサービスを提供するために、サービスをフォグで動作するVM上で提供する方法が考えられる。サービスを提供するVMは、ライブマイグレーション技術を利用することで、サービスを停止せずに、フォグ間を移動できるので、VMごとマイグレーションすることでユー

は低レイテンシで通信できるフォグ上のサービスを利用しつづけることが可能となる。

VM マイグレーションを利用したフォグコンピューティングの層状のアーキテクチャが提案されている [1]。図 1 は、その全体像を表す。最上位に位置するのは、デバイス



図 1 フォグコンピューティングの階層アーキテクチャ

内でフォグと通信するアプリケーションである。その下位に位置するモバイルデバイス層は、フォグの API によってアプリケーションのデータの位置の決定、もっとも近いフォグサーバの発見、アプリケーションの監視などを行う。VM の制御を行うフォグ層の機能は、マイグレーションのタイミング・位置の決定/実行、レイテンシなどによるサービス劣化のモニタリング、効率的なマイグレーションのためのフォグの情報の構築などである。その下位のクラウド層では、フォグ層を構築するためのサービス・インターフェースを提供する。

### 3 IoT を考慮したフォグコンピューティング基盤

2.3 節のようにフォグ上の複数の物理サーバを利用した VM の実行環境を実現するためには、フォグ上で動作する IaaS 基盤が必要となる。本節では、IoT アプリケーションを考慮してクラウドとフォグを一体的に運用する IaaS フォグコンピューティング基盤について述べる。

#### 3.1 システムの構成要素

本フォグコンピューティング基盤は以下の要素から構成される。

**コンピュータノード** クラウドやフォグでサービスを提供する物理ホストであり、この上で VM を動作させる。

**管理オブジェクト** コンピュータノード上の VM 管理を行うサーバソフトウェアである。ユーザであるデバイス、あるいは他の管理オブジェクトからの要求を受け、IoT サービスデータベースを参照しつつ、VM をどのコンピュータノードで動作させるかを決定し、各コンピュータノードに VM 起動を指示する。

**IoT サービスデータベース** VM 上で起動する IoT サービスを登録するデータベースで、各サービスの特性やサービスに対する要求などを記録している。

**デバイス** クラウド/フォグ上で動作するサービスを利用

するクライアントである。PC などの情報機器の他、IoT デバイスもこれに含まれる。

**ゲートウェイ** IoT デバイスをインターネットに接続する際、通信を中継する役割を持つ。単独ではインターネットに接続できない IoT デバイスを用いたり、複数の IoT デバイスをまとめてインターネットに接続する場合などに利用される。

#### 3.2 VM 起動ノードの決定

VM の起動ノードは、ユーザが送信した要求データと IoT サービスデータベースの情報を用いて管理オブジェクトが決定する。要求データとは IoT サービスを利用する際に IoT サービス (またはゲートウェイ) で生成されるもので、IoT サービスの特性や IoT サービスに対するユーザの指定が含まれる。IoT サービスデータベースには要求された IoT サービスの特性が記録されている。

例えば統計・事務等の多くの計算リソースを使用するがレイテンシ制約が厳しくないサービスはクラウド上の VM で起動される。この場合、通常の IaaS クラウド基盤ソフトウェアと同様に、クラウドのコンピュータノードで VM を起動する。また、交通・医療等のリアルタイム性を求めるサービスはフォグ上の VM で起動する。この場合、リアルタイム性を確保するために、利用可能なフォグとクラウドの中でユーザから最もネットワーク経路が短い場所の管理オブジェクトに VM の起動を指示する。

図 2 に、フォグで VM を起動する例を示す。デバイス

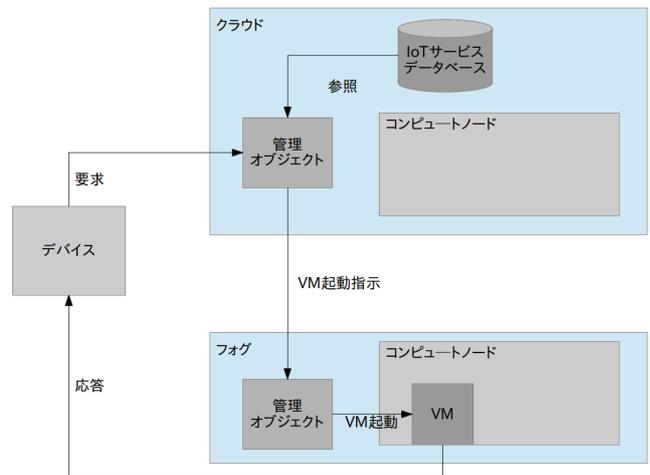


図 2 フォグにおける仮想マシンの起動

がクラウドに要求データを送信し、管理オブジェクトが受信すると管理オブジェクトは IoT サービスデータベースを参照し、クラウドかフォグのどちらで VM を起動するか決定する。フォグと判断され、クラウドの管理オブジェクトはフォグの管理オブジェクトに VM の起動を指示する。指示を受け取ったフォグの管理オブジェクトはそのフォグのコンピュータノードで VM を起動し、要求を出したデバイスに応答する。

### 3.3 システムの実装

OpenStack などの標準的な IaaS クラウド基盤ソフトウェアに以下の機能を追加することで提案したフォグコンピューティング基盤が実装できると考えられる。

**要求データの生成** ユーザーがサービスを利用する際に、デバイスまたはゲートウェイが要求データを生成し、サービスの属性 (リアルタイム性の重要度, 計算処理能力の大小など) を含ませる機能。VM の起動場所の判断材料の一つとしてクラウドで利用される。

**要求データの属性の読み取り** デバイスから受け取った要求データから、サービスの属性を読み取る機能。

**IoT サービスデータベース** IoT サービスとして提供可能なアプリケーションとその属性を登録した IoT サービスデータベースの作成。このデータベースとデバイスから送信された要求データから、サービスの属性が読み取られる。

**VM をクラウドかフォグのどちらで起動するか** クラウドの管理オブジェクトが要求データを受信した時、IoT サービスデータベースを参照し、その要求の属性を送られた要求データと合わせて読み取ることで、クラウドとフォグのどちらで VM を起動すべきかを決定する。

**最短経路の割り出し** サービスを受けるデバイスに最も近いフォグ (またはクラウド自身) を検索する機能。

**管理オブジェクト同士の通信** クラウドの管理オブジェクトがフォグの管理オブジェクトに VM 起動指示を送る機能。

## 4 Fog を用いた IoT アプリケーションにおける通信遅延の調査

本研究では、提案したフォグコンピューティング基盤実現上の課題を明確にするために、リアルタイム性を必要とする擬似的な IoT アプリケーション (以後、本アプリケーション) を作成した。本アプリケーションをクラウド上とフォグ上でそれぞれ動作させる場合を想定し、通信レイテンシをソフトウェア的に調整した環境上でレイテンシが本アプリケーションの動作に与える影響を調査した。

本アプリケーションは、仮想的な機械の温度を監視し、温度が一定範囲に収まるように制御するものである。機械の温度が基準値に近づいた場合、早急に対応しなければ、機械が停止し、作業が途切れてしまうので、ミリ秒単位のごまめな制御が必要となるという状況を想定している。こういった動作をクラウドとフォグでそれぞれ実行した場合の温度変化と応答時間を計測する。

本アプリケーションは、2つの要素から構成される。一つは、サーバ側で実行するもので、デバイスからのデータ送信を待つ。クラウドやフォグ上で動作する VM で実行されるサービスを想定している。クライアントである温

度センサからの接続を待ち、接続され次第、送られるデータの温度を分析する。設定された基準値以上の温度が送られていた場合、温度を下げる命令をクライアント側に送信する。もう一つはデバイス側のプログラムで、温度データを擬似的に生成し、それをサーバに送る、サーバから受け取った命令に応じて温度を調節するといった役割を持つ。0.5 秒ごとに温度データを生成し、上記のサーバ側のプログラムに送信する。サーバからデータが返されたら、その命令に応じて温度を調節する。また、このプログラムで、データ送信からサーバからデータを受け取るまでの時間を測ることで、サーバの応答時間を測定する。

## 5 実験

4 節のアプリケーションをフォグとクラウドそれぞれを想定した環境で実行して、本アプリケーションで制御する仮想機械の温度変化と、本アプリケーションの応答時間を測定する実験を行った。

### 5.1 実験環境

実際にデバイスとクラウドとの地理的な距離を作り出すことは困難なので、Linux の tc コマンドを使用して、IoT デバイスとサーバとの間に通信レイテンシを発生させた。

表 1 に、サーバ側と IoT デバイス側となる PC の実行環境を示す。

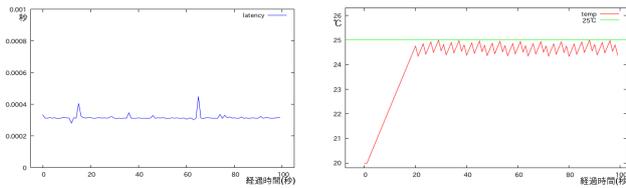
表 1 サーバと IoT デバイス側 PC の仕様

項目名	サーバ	IoT デバイス側
CPU	Intel(R) Core(TM) i5-3320M	
コア数	4	
クロック周波数	2.60GHz	
メモリ	3.8GiB	
ネットワーク	1000Base-T ギガビットイーサネット	
OS	Ubuntu 16.04.1 LTS	Ubuntu 12.04.2 LTS

### 5.2 実験結果

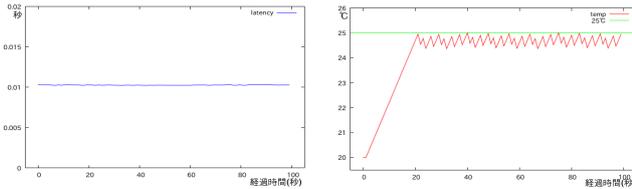
温度データの初期値を 20 , 目標値を 25 に設定した。発生させた通信レイテンシは、非常に近いフォグとの通信を想定した 0ms, 平均的な距離のフォグとの通信を想定した 10ms, 国内クラウドとの通信を想定した 50ms, 国外クラウドとの通信を想定した 100ms となっている。実験結果を図 3 から図 6 に示す。各グラフの横軸は実験開始からの経過時間で、縦軸が応答時間と温度を示す。応答時間はいずれの場合でもほぼ一定で、通信レイテンシが反映されている。温度変化についても通信レイテンシが影響しており、通信レイテンシが大きい環境では目標温度を上回り、制御が失敗したケースがあることがわかる。実験結果のまとめを表 2 に示す。クラウドを想定した環境では要求を満たすことができない事例が存在することが確認できた。IoT をよりリアルタイム性を必要とするバーチャル

リアリティやロボティクスまで拡大する場合，フォグコンピューティングは欠かせないものであると考えられる．



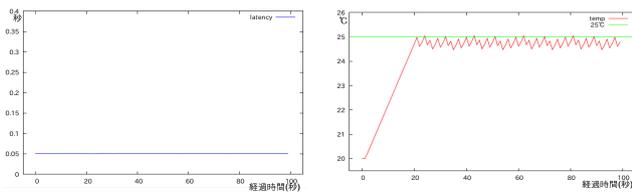
(a) 応答時間 (b) 温度変化

図3 非常に近いフォグ



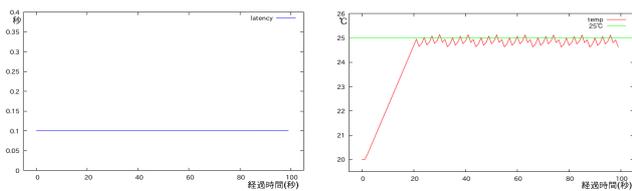
(a) 応答時間 (b) 温度変化

図4 平均的な距離のフォグ



(a) 応答時間 (b) 温度変化

図5 国内のクラウド



(a) 応答時間 (b) 温度変化

図6 国外のクラウド

## 6 おわりに

本研究では，IoT を考慮してクラウドとフォグを一体的に管理するフォグコンピューティング基盤を提案した．具体的な IoT アプリケーションを想定し，通信レイテンシとアプリケーションの応答時間の関係を調べるために，クラウドやフォグの VM 上で動作することを想定した擬似的な IoT アプリケーションを作成し，それをを用いた実験を行った．その結果，レイテンシが国内の一般的なクラウドとの通信にかかる程度のものであれば，高いリアルタイム性を必要とするアプリケーションが正常に動作しない事例が存

表2 アプリケーションの実行結果

遅延	応答時間	温度制御
0ms	0.3ms	制御成功
10ms	10ms	制御成功
50ms	50ms	制御失敗
100ms	100ms	制御失敗

在することがわかった．今後は提案アーキテクチャを既存の IaaS 基盤ソフトウェアを拡張する形で実装していく．

## 参考文献

- [1] Luiz F. Bittencourt, Marcio Morate Lopes, Ioan Petri and Omer F. Rana: *Towards Virtual Machine Migration in Fog Computing*. 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (2015).
- [2] 二木 真明: “INTERNET OF THINGS IoT におけるクラウドの役割とセキュリティ上の課題”, <https://www.altairsecurity.com/documents/Internet-of-Things.pdf> (2016年12月28日アクセス)
- [3] 田中 裕之, 高橋 紀之, 川村 龍太郎: “IoT 時代を拓くエッジコンピューティングの研究開発”, <http://www.ntt.co.jp/journal/1508/files/jn201508059.pdf> (2016年12月28日アクセス)
- [4] 松岡 功: “フォグコンピューティングは定着するか”, <http://japan.zdnet.com/article/35084645/> (2016年12月28日アクセス)
- [5] Grace Walker: “クラウドコンピューティングの基礎-従来とは異なる方法でコンピューター・リソースを提供する”, <https://www.ibm.com/developerworks/jp/cloud/library/cl-cloudintro/cl-cloudintro-pdf.pdf> (2016年1月6日アクセス)
- [6] 河村 雅人, 大塚 紘史, 小林 佑輔, 小山 武士, 宮崎 智也, 石黒 佑樹, 小島 康平: “絵で見て分かる IoT/センサの仕組みと活用”, 株式会社翔泳社 2015.
- [7] OpenFog コンソーシアムアーキテクチャワーキンググループ: “OpenFog アーキテクチャの概要”, <https://openfog.jp/wp-content/uploads/OpenFog-Architecture-Overview-WP-2-2016-Japanese-Version.pdf> (2016年12月28日アクセス)