

エッジコンピューティング環境における AP 実行環境の起動時間を考慮したリソース予約手法の提案

有菌 里奈† 横山 和俊†

高知工科大学情報学群†

1. はじめに

モバイルエッジコンピューティング(以下 MEC と略す)では, モバイル端末から要求される処理を, ネットワーク周縁に位置しているエッジサーバで処理をする. この環境では, モバイル端末の処理をデータセンタ(以降 DC と略す)まで通信を行わずに, 近くのエッジサーバ(以降 MEC サーバと呼ぶ)で処理をするため, リアルタイム処理が可能になる. しかし, MEC サーバは豊富なリソースを持たないため, MEC サーバへ割り当てる処理を効率的に選ぶ必要がある. 我々は, これまでにモバイル端末の混雑具合を利用したリソース割当て手法を提案している[1][2]. しかし, これらの手法では, モバイル端末の要求を処理するためのアプリケーション(以下 AP と略す)の実行環境の起動時間を考慮していない. そこで, 本稿では, AP の実行環境の起動時間を考慮した手法として, AP の実行環境が起動する時刻のモバイル端末の位置を求め, その位置に基づいてサーバのリソースを予約する手法を提案する.

2. 環境モデル

2.1. ネットワーク構成

想定するネットワーク構成を図1に示す. 構成は基地局, 集約局, 中間局, DCの4層構成としており, MECサーバはネットワークエッジである基地局に配置される.

2.2. MECサーバ

MECサーバは, 基地局を通して他のMECサーバやDCと通信を行う. また, 基地局の通信範囲は半径 r km以内とし, 基地局同士の通信を行う際は集約局を通じて行う必要がある. 異なる集約局配下にある基地局と通信を行う際は, 中間局を通じて対象となる基地局へと通信を行う.

2.3. モバイル端末

モバイル端末は, カーナビゲーションのように事前に移動経路がわかることを想定している.

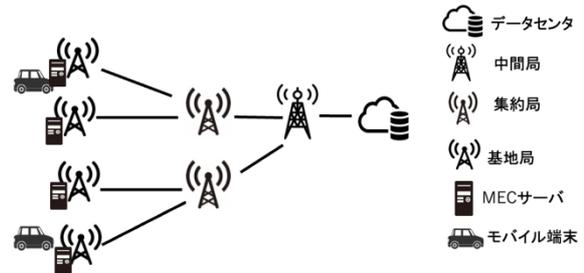


図1 ネットワーク構成

そのため, 移動時間・移動経路・移動速度・座標情報などを自身で保有しており, かつ, その情報をDCに送信している. また, 全てのモバイル端末は端末内だけで処理が不可能であるAPを使用しており, その処理を端末から物理的に最も近い位置にあるMECサーバへリソース割当てを依頼する.

2.4. 通信遅延

通信遅延の指標として, ホップ数を用いる. ホップ数は, 一つの転送・中継設備を通る毎に1ホップずつ加算される. 例えば, 同じ集約局内にある基地局同士が通信を行う場合, 基地局から集約局, 対象の基地局へ通信を行うため, ホップ数は3となる. また, DCへ通信を行う場合は, 基地局から集約局, 中間局を通じてDCへ通信を行うためホップ数は4となる. ホップ数が大きいほど通信遅延が発生しているとする.

2.5. リソース予約処理

リソース予約処理はDCで実現されており, 全てのMECサーバやモバイル端末の情報は, DCで管理されている.

3. 提案手法

3.1. 概要

本稿では, AP実行環境の起動時間を考慮したリソース割当て手法を提案する. 具体的には, モバイル端末がリソースを予約するMECサーバの選択方法を検討する.

提案手法では, 時刻 t の段階で, AP実行環境の起動時間(Δt とする)を加算した時刻($t + \Delta t$)におけるMECサーバの予約を行う. 図2の例を用い

て提案手法の概要の具体例を説明する。

図2はモバイル端末 m の時間毎に通過する経路を示している。モバイル端末 m は、 $M1 \cdot M2 \cdot M5 \cdot M11$ の順に基地局を通過する。例えば、現在の時刻を t 、APの実行環境の起動時間 Δt を1とした場合、時刻 $(t+1)$ では、モバイル端末 m が基地局 $M2$ の通信範囲に移動する。この時、 $M2$ の MEC サーバを割当てると時刻 $(t+1)$ でのホップ数は1となる。また、 $M2$ の MEC サーバがリソース不足、または起動時間に間に合わず予約できない場合、 $M5$ の MEC サーバを予約しておく、 $M5$ の通信範囲を通過する期間は、ホップ数1での通信が可能になる。

つまり、モバイル端末 m が MEC サーバと通信するホップ数を小さくするためには、極力通過予定である基地局の MEC サーバを予約する方が良い。

3.2. 予約アルゴリズム

予約アルゴリズムを以下に説明する。ここで、時刻 t での予約サーバとは、時刻 t にリソース割当ての予約を行っている MEC サーバのことである。また、割当てサーバとは、モバイル端末が現時点でリソースを割当てられている MEC サーバのことである。

<ステップ1>予約サーバの候補の選出

[ステップ1-1]時刻 $(t + \Delta t)$ のモバイル端末の位置を求める。時刻 t と同じ集約局中の基地局の通信範囲内であればステップ1-2、異なる場合は1-5へ。

[ステップ1-2]時刻 $(t + \Delta t)$ から集約局が変わるまでの時刻の間で、モバイル端末が通過する基地局の MEC サーバを求め、予約サーバの候補とする。

[ステップ1-3]予約サーバの候補のうち、現在地から最も近い基地局の MEC サーバを最優先とし、候補としているサーバが割当て可能であるか判定を行う順番を決定する。

[ステップ1-4]ステップ1-3までで予約サーバの候補となるサーバを選出できなかった場合、同じ集約局中にある全ての MEC サーバを予約サーバの候補として選出する。この時、現在の割当てサーバが現在の時刻 t から予約を行いたい時刻 $(t + \Delta t)$ までで割当て可能な状態にある場合、これを優先的に予約サーバの候補として選出する。

[ステップ1-5]時刻 $(t + \Delta t)$ から集約局が変わるまでの時刻の間で通過する MEC サーバを求め、予約サーバの候補とする。

[ステップ1-6]予約サーバの候補のうち、現在地から最も近い基地局の MEC サーバを、候補となっているサーバの中でも優先度を高くし、割当て可能か判定を行う順番を決定する。

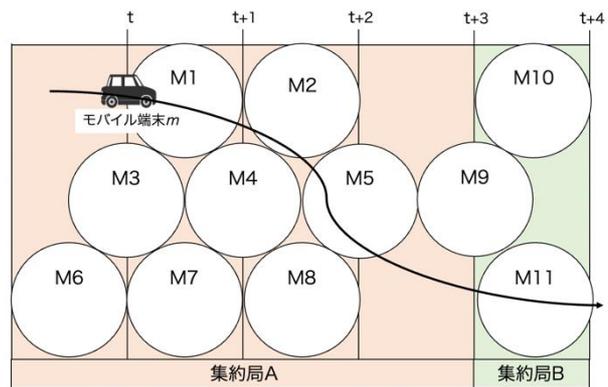


図2 モバイル端末が時刻毎に通過する経路

[ステップ1-7]ステップ1-6で予約サーバの候補が決定しなかった場合、時刻 $(t + \Delta t)$ での集約局内の MEC サーバを対象に広げ、予約可能な MEC サーバを時刻 $(t + \Delta t)$ での予約サーバの候補とする。

<ステップ2>モバイル端末の割当て優先順序

[ステップ2-1]ステップ1で選出した予約サーバの候補となっている MEC サーバの数を、モバイル端末毎に求める。

[ステップ2-2]候補となるサーバの数が少ない順に、モバイル端末の割当て優先順序を決定する。

<ステップ3>予約サーバの決定

予約可能なサーバ群である候補サーバのリソースより、予約可能か判定する。予約が可能であれば、その MEC サーバを時刻 $(t + \Delta t)$ での予約サーバとし、APの実行環境を起動する。

<ステップ4>割当てサーバの切り替え

モバイル端末が MEC サーバを予約している時刻であれば、現在の割当てサーバを予約サーバへ切り替える。

4. おわりに

モバイル端末の移動経路を利用して、事前に MEC サーバへリソース割当ての予約を行う手法を提案した。今後は、シミュレーションによる提案手法の評価を行う。

参考文献

- [1]大崎, 福永, 横山, "Cloudlet 環境における移動経路計画を用いたリソース割当て手法の検討", 情報処理学会研究報告, Vol.2017-DPS-173, No.8, pp.1-8(2018).
- [2]福永, 横山, "モバイルエッジコンピューティング環境におけるモバイル端末の走行履歴を用いたリソース割当て手法の検討", 第82回全国大会講演論文集, 第3分冊, pp147-148(2020).