

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移動予測を利用した リソース割当て手法

1225125 福永 昂輝 【分散処理 OS 研究室】

Resource allocation algorithm using mobile device movement prediction in edge computing environment

1225125 Koki Fukunaga 【Distributed System and Operating System Lab.】

1 はじめに

モバイルエッジコンピューティングにおいて一部のエッジサーバ (以下 MEC サーバと略す) に負荷が集中するとリソース不足が発生する可能性がある。この課題に過去研究では、モバイル端末と通信している MEC サーバの近傍にある MEC サーバを利用する集中アルゴリズムを提案した [1]。しかし、全てのモバイル端末のリソース割当てを一括管理するのは現実的ではない。本研究では、各 MEC サーバが割当ての決定を処理する分散アルゴリズムを提案し、有効性を評価する。

2 システムモデル

システムモデルを図 1 に示す。図 1 では、基地局とその基地局の情報を集約する集約局、集約局の情報をデータセンタへ送信する中間局が仲介となり、基地局間やデータセンタ (以下 DC と略す) との通信が実現される。以下、システムモデルについて説明する。

(1) MEC サーバ

図 1 の各基地局に MEC サーバを配置し、 m_i で表す。各 MEC サーバ m_i には割当て可能リソース量 R_i が与えられる。自身の R_i を超えない限り、モバイル端末のリソース要求を処理できる。

(2) モバイル端末

モバイル端末は道路ネットワーク上を走行する。モバイル端末を d_j で表し、各モバイル端末 d_j には通信している MEC サーバに要求するリソース量が設定されている。モバイル端末は道路ネットワーク上で、最も近い基地局と通信する。また、モバイル端末は走行履歴を持っており、直近 t 秒間の移動距離 s 、平均方位角 α がわかるとする。

(3) 周辺 MEC サーバ

各基地局の MEC サーバ m_i は、半径 $l.km$ 以内の周辺の MEC サーバ m_n の情報を得ることができる (以下周辺 MEC サーバ m_n と呼ぶ)。また、各 MEC サーバ m_i は通信しているモバイル端末の全ての要求リソース量 S_i がわかり、問い合わせる

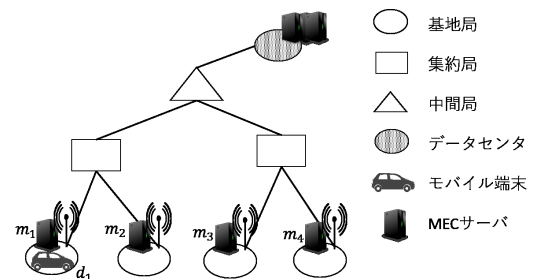


図 1 システムモデル

ことで半径 $l.km$ 以内の周辺 MEC サーバ m_n の S_n を知ることができる。また、MEC サーバと周辺 MEC サーバの距離は近いため、情報伝達にかかる遅延がないものとする。

(4) 割当て

割当ては一定時間ごとに各 MEC サーバ m_i が判断する。また MEC サーバ m_i が周辺 MEC サーバ m_n に対して割り当てたい場合は、対象の周辺 MEC サーバ m_n に対して割当て依頼を送信する。割当て依頼を受信した周辺 MEC サーバ m_n は自身の R_n を超えない場合は割り当てるが、超えた場合に割当て失敗として DC に割当て依頼を送信する。

(5) 遅延時間

システムモデルでの遅延時間をホップ数で表す。モバイル端末のリソース要求を割り当てる場合、様々な転送・中継設備を経由するほど通信遅延が発生する。本稿では、1つの転送・中継設備を経由することを 1 ホップとする。

3 リソース割当て手法

モバイル端末のリソース要求をどの MEC サーバに割り当てるべきかを考える。割当て手法は平均ホップ数を抑えることを目的とする。平均ホップ数はホップ数の総数に対して割当ての総回数を割ったものとする。以下では、評価する 3 つの手法について述べる。

(1) 最近傍割当て

各 MEC サーバ m_i は通信しているモバイル端末が自身に割当て可能か判断し、割当て不可の場合に物理的な距離が最も近い周辺 MEC サーバ m_n に対して割当て依頼を送信する。

(2) 最近傍・最小リソース量による割当て

各 MEC サーバ m_i は通信しているモバイル端末が自身に割当て可能か判断し、割当て不可の場合、周辺 MEC サーバ m_n で S_n が最小の MEC サーバに対して割当て依頼を送信する。

(3) 走行履歴による割当て

モバイル端末の直近 t 秒間を予測時間とし、平均方位角 α 方向に移動距離 s ほど動かした座標を推定移動場所 G とする。次の順序で各 MEC サーバ m_i が一定時間ごとに割当てを決定する。

- (a) 直近 t 秒間のモバイル端末の走行履歴を参照し、推定移動場所 G を算出
- (b) 推定移動場所 G と周辺 MEC サーバ m_n との物理的な距離 D_n を算出
- (c) 評価関数 $D_n + f(R_n, S_n)$ の値が最も小さい MEC サーバ m_n に対して割当て依頼を送信

関数 $f(R_n, S_n)$ は、 $S_n < R_n$ の時、0 を出力し、 $R_n < S_n$ の時、500 を出力する関数。この関数によってペナルティを与え、割当て不可を抑える。この評価関数によってモバイル端末の推定移動場所から最も近い MEC サーバに割り当てることができ、移動に合わせた割当てが可能になる。

4 評価

4.1 評価内容

OpenCellID で岡山駅周辺の KDDI の基地局配置を取得し、各基地局に MEC サーバを配置する。また OpenStreetMap で岡山駅周辺の道路ネットワークを取得し、交通シミュレータ SUMO で取得した道路ネットワーク上を走行する車両を生成し、交通シミュレーションを実行した。本研究では、交通シミュレータ SUMO で走らせた車両を取得し、モバイル端末として扱った。表 1 にシミュレータでの実験パラメータを示す。評価項目とし

表 1 実験パラメータ

シミュレーション時間	1000s
基地局の通信範囲	500m
モバイル端末の要求リソース	50-500 のランダム
割当て可能リソース量	2500
割当て時間	10s ごと
予測時間	5s

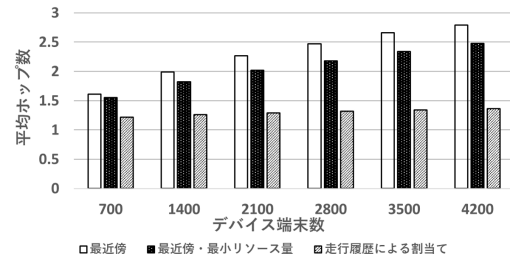


図 2 平均ホップ数

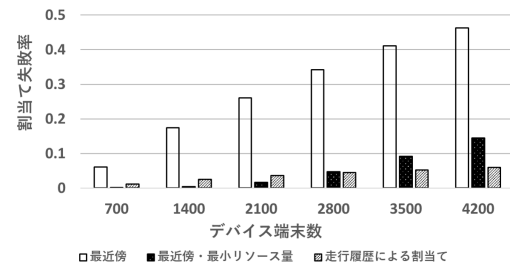


図 3 割当て失敗率

ては、平均ホップ数と割当て失敗率とする。割当て失敗率は、割当て失敗回数を全ての割当て回数で割ったものとする。

4.2 評価結果

図 2 に各割当て手法の平均ホップ数の結果を、図 3 に割当て失敗率を示す。平均ホップ数の結果では、走行履歴による割当てが最も良かった。これは、移動を考慮しているため、割当てが決定し、次の割当て開始時までの間、1 ホップになることが多いからである。割当て失敗率の結果では、モバイル端末数が 2800 を超えると最近傍・最小リソース量の結果が走行履歴による割当てよりも悪くなる。この理由は、MEC サーバの割当て可能リソース量をモバイル端末のリソース要求が超える状態が増えると、割当て可能リソース量が最も小さい MEC サーバに周辺 MEC サーバが割当て依頼を多く送信するため、割当て失敗が増加するからである。走行履歴による割当ては、全体的に割当て失敗が発生している。この理由は、多くのモバイル端末の移動方向が同じ場合は、同じ周辺 MEC サーバに割当て依頼を送るため、割当て失敗が一定数発生するからと考えられる。

5 おわりに

本稿では、モバイル端末のリソース要求を分散アルゴリズムで割当てを決定する手法の提案とともにシミュレーションし、提案手法の有効性を示した。

参考文献

[1] 大崎康平, 福永昂輝, 横山和俊, "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(2017).