

# アドホックネットワークにおける ヒストグラムと確率密度関数を用いた Top-k 検索手法

佐々木 勇和<sup>†</sup>      萩原 亮<sup>†</sup>      原 隆浩<sup>†</sup>      西尾 章治郎<sup>†</sup>

<sup>†</sup> 大阪大学情報科学研究科マルチメディア工学専攻 〒565-0871 大阪府吹田市山田丘 1-5

E-mail: †{sasaki.yuya,hagihara.ryou,hara,nishio}@ist.osaka-u.ac.jp

あらまし アドホックネットワークでは、膨大なデータの中から必要なデータのみを効率的に取得するため、端末が何らかの値（スコア）によって順序付けられたデータの上位  $k$  個のものを検索する Top-k 検索を用いることが有効である。筆者らはこれまでに、アドホックネットワークにおける効率的な Top-k 検索手法を提案している。本稿では、この手法を拡張し、検索結果の取得精度の維持しつつ、検索のためのトラフィックをさらに削減する手法を提案する。提案手法では、各端末がヒストグラムを作成し、ネットワーク全体のスコア分布および  $k$  番目のスコアを推定する。次に、推定した  $k$  番目のスコア以上のスコアをもつデータを返信することで、できる限り検索結果に入るデータのみを返信する。

キーワード アドホックネットワーク, Top-k 検索, スコア分布推定, ヒストグラム

## A Top-k Query Method using Histogram and Probability Density Function in Mobile Ad Hoc Networks

Yuya SASAKI<sup>†</sup>, Ryo HAGIHARA<sup>†</sup>, Takahiro HARA<sup>†</sup>, and Shojiro NISHIO<sup>†</sup>

<sup>†</sup> Department of Multimedia Engineering Graduate School of Information Science and Technology Osaka University 1-5, Yamadaoka, Suita, Osaka, 565-0871 Japan

E-mail: †{sasaki.yuya,hagihara.ryou,hara,nishio}@ist.osaka-u.ac.jp

**Abstract** In mobile ad hoc networks (MANETs), to acquire only necessary data items, it is effective that each mobile node retrieves data items using a top-k query, in which data items are ordered by the score of a particular attribute and the query-issuing mobile node acquires data items with  $k$  highest scores. In our previous work, we proposed a query processing method for top-k query for reducing traffic and also keeping high accuracy of the query result. In this paper, we extend this method to further improve the performance. In the extended method, each mobile node constructs a histogram from the scores of its holding data items and estimates the distribution of scores of all data items and  $k$ -th score in the entire network. This histogram is attached with query messages and replies so that other mobile nodes can estimate the score distribution and  $k$ -th score more accurately. When transmitting a reply, each mobile node sends back only data items whose scores that are larger than the estimated  $k$ -th score to reduce traffic as much as possible.

**Key words** Ad-hoc network, Top-k query, Score distribution estimation, Histogram

### 1. 序 論

近年、無線通信技術の発展と計算機の小型化や高性能化に伴い、ルータ機能をもつ端末のみで一時的な無線ネットワークを形成するアドホックネットワークへの関心が高まっている。アドホックネットワークにおけるデータ検索では、複数の端末が限られた通信帯域を共有するため、膨大なデータの中から必要なデータのみを効率的に取得する必要がある。特に各端末に限

られた資源を割り当てる場合や関連性の高い情報のみを収集する場合、検索条件とデータの属性値で決定する何らかの値（スコア）によって順序付けられたデータの上位  $k$  個のものを検索する Top-k 検索を用いることが有効である [2], [3]。

ここで、Top-k 検索を実現する単純な方法として、端末が検索クエリをネットワーク全体にフラッディングし、これを受信した端末が自身のもつデータの中からスコアの高いものを固定数返信する方法が考えられる。各端末の返信するデータの数が

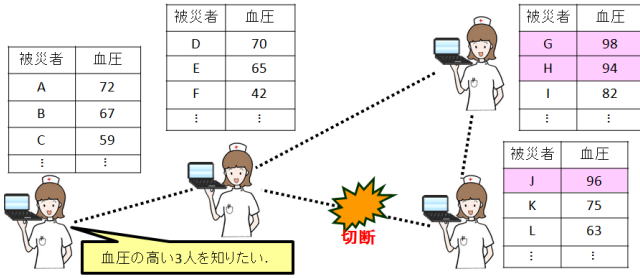


図 1 アドホックネットワークにおける Top-k 検索例

多い場合、検索クエリを発行した端末は、ネットワーク全体の上位  $k$  個のデータ（検索結果）を取得できる可能性が高い。しかし、検索結果に入らないデータまで返信されるため、不要なトラフィックが発生する。一方、各端末の返信データ数が少ない場合、検索結果に入らないデータが返信される可能性は低くなるが、検索結果に入るデータが返信されず、検索結果の取得精度が低下する。

ここで、検索対象となるデータはなんらかの分布に従っていることが考えられる。例えば、血圧は正規分布、ウェブサイトの訪問者数は Zipf 分布 [5] に従うことが知られている。そこで、これまでに筆者の研究グループは、文献 [3] において、アドホックネットワークにおけるトラフィックの削減と検索結果の取得精度の低下の抑止を実現する Top-k 検索手法を提案した。この手法では、各端末が自身のもつデータのスコアからヒストグラムを作成し、ネットワーク全体の  $k$  番目のスコアを推定する。このとき、 $k$  番目のスコアを正確に推定するために、各端末は検索クエリやクエリ応答にヒストグラムを添付して送信する。次に、各端末は推定した  $k$  番目のスコア以上のスコアをもつデータのみを返信することで、できる限り検索結果に入るデータのみを返信する。また、返信先の端末とのリンクが切断した端末は、他の隣接端末にクエリ応答を返信することで、検索結果の取得精度の低下を抑止する。

本稿では、検索性能のさらなる向上を目的とし、筆者らが文献 [3] で提案した手法を拡張する。拡張手法では、各端末はスコア分布の確率密度関数を用いて、スコア分布、および  $k$  番目のスコアの推定する。具体的には、データが特定の分布（正規分布など）に従っていると仮定し、その分布の確率密度関数のパラメータをヒストグラムから求め、さらにスコア分布のパーセント点を上側累積確率を用いて求める。これにより、文献 [3] の手法よりも、さらに正確に  $k$  番目のスコアを推定できる。また、推定の誤差を考慮して、 $k$  番目のスコアの補正を行う。具体的にはセーフティマージンを用いて  $k$  番目付近のデータの欠落を防ぎ、取得精度の低下を抑止する。

以下では、2. で提案手法について述べ、3. でシミュレーション実験の結果を示す。最後に 4. で本稿のまとめと今後の課題について述べる。

## 2. Top-k 検索手法

本章では、まず想定環境について述べ、各端末がヒストグラムを作成する手順について述べる。その後、本稿で提案する

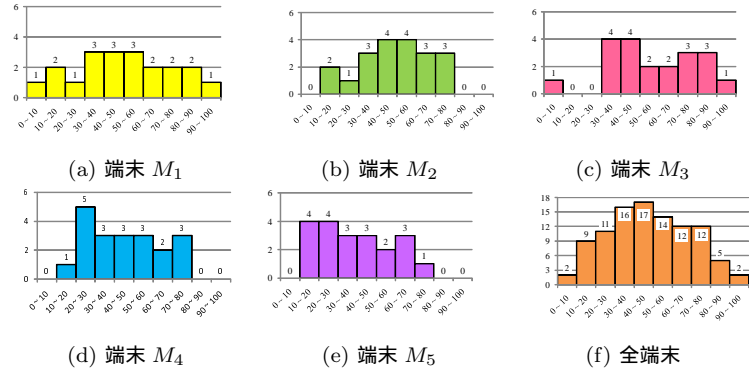


図 2 各端末の作成したヒストグラムとネットワーク全体のヒストグラム

Top-k 検索手法について説明する。

### 2.1 想定環境

本稿では、アドホックネットワークを構成する各端末が、自身と他の端末のもつデータに対して Top-k 検索を行う環境を想定する。Top-k 検索を行う端末は、検索条件を指定して検索クエリを発行し、ネットワーク内の上位  $k$  個のスコアをもつデータを取得する。

ネットワーク内には  $n$  個のデータが存在し、各々が特定の端末に保持されている。簡単化のため、全てのデータのサイズは等しく、複製は作成しないものとする。データのスコアは、検索条件とデータの属性値から決定し、何らかのスコアリング関数を用いて算出される。また、スコアは特定のスコア分布（例えば正規分布）に従っているものとする。

ネットワーク内には、 $m$  個の端末（識別子:  $M_1, M_2, \dots, M_m$ ）が存在し、各々が自由に移動する。各端末は、ネットワーク内のデータ数  $n$ 、ネットワーク内のデータのスコアが従う分布の種類、およびスコアの下限、上限を把握しているものとする。また、各端末のもつデータ数はネットワーク全体のデータ数 ( $n$ ) に比べて少ないため、各端末のもつデータのスコア分布は、ネットワーク全体のデータのスコア分布に従っているとは限らない。

### 2.2 ヒストグラムの作成

各端末は自身のもつデータのスコアを用いて、ヒストグラムを作成する。ヒストグラムは、階級と呼ばれる適当な大きさの区間ごとに、値がその区間に含まれるスコアの数を集めたものであり、スコアの分布状況を表すために用いられる。本稿では、端末  $M_i$  のもつデータの中で、そのスコアが階級  $c_j$  ( $1 \leq j \leq C$ ) に含まれる個数を数えたものを  $M_i$  のヒストグラム  $H_i$  とする。ただし、階級  $c_j$  は、ヒストグラムの幅の最大値と最小値の範囲  $[MIN, MAX)$  を、大きさの等しい  $C$  個の階級に分割したときの  $j$  番目の階級を示し、その範囲は  $[MIN + \frac{(j-1)(MAX-MIN)}{C}, MIN + \frac{j(MAX-MIN)}{C})$  となる。例えば図 2 は、ヒストグラムの幅の範囲が  $[0, 100)$ 、階級数  $C$  が 10 の場合に、ネットワーク内の端末  $M_1$  から  $M_5$  が作成したヒストグラムおよびネットワーク全体のヒストグラムを示す。

### 2.3 検索の手順

本節では提案手法における検索手順の概要を示す。なお、こ

の手順は、文献 [3] の手法と同様である。

### 2.3.1 検索クエリの転送

提案手法では、ヒストグラムに含まれるスコアの数が多いほど、そのスコア分布はネットワーク全体のスコア分布に近づくため、端末は  $k$  番目のスコアをより正確に推定できる。そこで各端末は、ヒストグラムを検索クエリに添付して送信し、検索クエリを中継する端末がヒストグラムを更新しながら転送する。以下では、検索クエリを発行した端末  $M_p$  と検索クエリを受信した端末  $M_q$  の動作について説明する。

(1) 端末  $M_p$  は検索条件、および要求データ数  $k$  を指定する。また、自身のもつデータのスコアを算出し、2.2 節の方法に従って、自身のヒストグラム  $H_p$  を作成する。

(2)  $M_p$  は自身の隣接端末に検索クエリを送信する。この検索クエリには、クエリ発行端末  $M_p$  の識別子、検索クエリの識別子、要求データ数  $k$ 、検索条件、経路端末リスト、およびクエリヒストグラムが含まれる。経路端末リストにはクエリ発行端末から自身までの経路上に存在する端末の識別子が含まれ、ここでは  $M_p$  のみとなる。クエリヒストグラムはこれらの経路上の端末のヒストグラムを統合したものであり、ここでは  $H_p$  となる。

(3) 検索クエリを受信した端末  $M_q$  は、それが初めて受信したものであれば、経路端末リストの末尾に格納されている端末を自身の親とし、経路端末リストに含まれる端末数から、クエリ発行端末から親までのホップ数を調べる。手順 (4) へ進む。

検索クエリが既に受信したものであれば、手順 (5) へ進む。

(4)  $M_q$  は、手順 (1) と同様に検索条件から自身のヒストグラム  $H_q$  を作成し、検索クエリに含まれるクエリヒストグラムに  $H_q$  を統合する。また、自身の識別子  $M_q$  を経路端末リストの末尾に追加する。 $M_q$  は、自身の隣接端末に検索クエリを送信し、手順 (3) に戻る。

(5) 検索クエリを再受信した端末  $M_q$  は、検索クエリに含まれる経路端末リストの末尾の端末を親でない隣接端末とし、経路端末リストに含まれる端末数から、その端末までのホップ数を記録する。また、経路端末リストの末尾から 2 番目の端末が自身の場合、経路端末リストの末尾の端末を自身の子とする。

検索クエリの転送では、各端末がクエリ発行端末から自身までの経路上に存在する端末のヒストグラムを統合したクエリヒストグラムを検索クエリに添付することで、ヒストグラムに含まれるスコア数を増加させる。また、各端末は経路端末リストにより、検索クエリ発行端末を根とする木構造における自身の親と子、クエリ発行端末から自身までの経路、および親以外の隣接端末のクエリ発行端末からのホップ数を把握できる。

図 3 を用いて、 $M_1$  が Top- $k$  検索を行った場合の検索クエリの転送例を説明する。吹出しは各端末が検索クエリに添付したクエリヒストグラムを示し、破線の矢印は端末間の親子関係を示す。検索クエリには、クエリ発行端末からの経路上の端末のヒストグラムを統合したものが添付されているが、統合の状態をわかりやすく示すため、端末  $M_1, M_2, M_3, M_4$ 、および  $M_5$  のヒストグラムに相当する部分をそれぞれ黄色、緑色、桃色、青色、および紫色で表す。なお、各端末のヒストグラムは

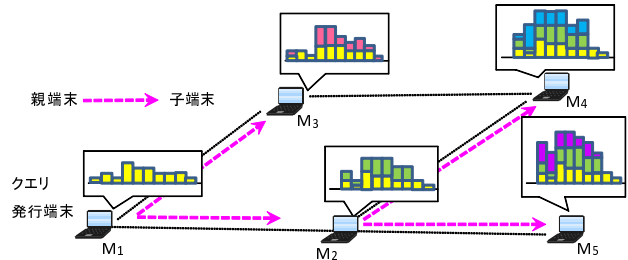


図 3 検索クエリの転送例

図 2 で表されるものとする。

### 2.3.2 クエリ応答の返信

2.3.1 項で述べたように、提案手法では、ヒストグラムに含まれるスコア数が多いほど、 $k$  番目のスコアをより正確に推定できるため、各端末はヒストグラムをクエリ応答にも添付して送信する。また、クエリ応答を中継する端末は、受信した情報から返信の必要がないと判断されるデータは返信しない。以下では、2.3.1 項において、端末  $M_p$  が検索クエリを転送した後、各端末がクエリ応答を返信する動作について説明する。

(1) 自身の子がない端末  $M_r$  は、送信した検索クエリに添付したクエリヒストグラムを用いて、 $k$  番目のスコアの推定 (2.4 項) を行う。その後  $M_r$  は、クエリ応答を自身の親に送信する。このクエリ応答には、クエリ発行端末  $M_p$  の識別子、検索クエリの識別子、返信リスト、および応答ヒストグラムが含まれる。ここで返信リストには、 $M_r$  のもつデータの中で推定した  $k$  番目のスコア以上のスコアをもつデータとそのスコアが含まれる。また、応答ヒストグラムは、自身および  $M_r$  の全ての子孫端末のヒストグラムを統合したものが含まれ、ここでは  $H_r$  となる。

(2) 全ての子からクエリ応答を受信するか、自身の検索クエリを送信してから一定時間経過した端末  $M_s$  は、受信した全てのクエリ応答に含まれる応答ヒストグラムと、自身が検索クエリに添付したクエリヒストグラムを統合する。このヒストグラムは、クエリ発行端末から  $M_s$  までの経路上の端末、および  $M_s$  の全ての子孫端末のヒストグラムを統合したものとなる。 $M_s$  は、このヒストグラム  $H'$  を用いて、 $k$  番目のスコアの推定 (2.4 項) を行う。

(3)  $M_s$  は、クエリ応答を作成し、自身の親に送信する。ここで、返信リストは、受信した全てのクエリ応答に含まれる返信リストおよび  $M_s$  のもつデータの中で、推定した  $k$  番目のスコア以上のスコアをもつデータとそのスコアが含まれる。ただし、該当するデータの数及要求データ数  $k$  より大きい場合、上位  $k$  個のスコアとそのスコアをもつデータのみが含まれる。また、応答ヒストグラムは、ヒストグラム  $H_s$  と受信した全ての応答ヒストグラムを統合したものとなる。

クエリ応答の返信では、推定した  $k$  番目のスコア以上のスコアをもつデータのみを返信する。また、中継端末で検索結果に入らないと判断できたデータを返信しない。さらに、自身および自身の子孫端末のヒストグラムを統合した応答ヒストグラムをクエリ応答に添付することで、クエリ発行端末に近い端末

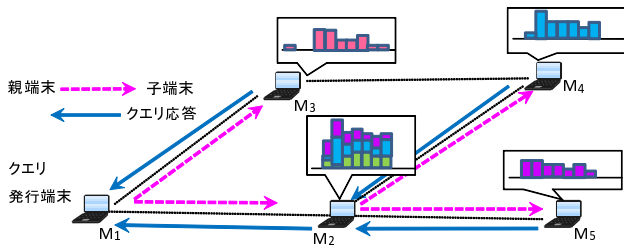


図 4 クエリ応答の返信例

ほど、ヒストグラムに含まれるスコア数が増加し、 $k$  番目のスコアをより正確に推定できる。その結果、検索結果に入らないデータの返信を抑制できるため、取得精度を低下させずにトラフィックを削減できる。

図 3 および図 4 を用いて、クエリ応答の返信例を説明する。図 4 において、吹出しは各端末がクエリ応答に添付した応答ヒストグラムを示し、矢印はクエリ応答の流れを示す。例えば、 $M_4$  は、図 3 に示すクエリヒストグラム ( $M_1$  [黄色],  $M_2$  [緑色],  $M_4$  [青色] のヒストグラムを統合したもの) を用いて、 $k$  番目のスコアを推定し、応答ヒストグラム ( $M_4$  [青色] のヒストグラム) と推定した  $k$  番目のスコア以上のスコアをもつデータを、親である  $M_2$  に返信する。

### 2.3.3 リンク切断の検出時の処理

アドホックネットワークでは、端末の移動によりネットワークポロジが動的に変化する。ここで、親とのリンクが切断された端末は、クエリ応答を返信できないため、検索結果の取得精度が低下する。そこで、親とのリンクが切断した端末は、他の隣接端末にクエリ応答を送信し、別経路を探索することで、データをクエリを発行した端末までできる限り返信する。

### 2.4 $k$ 番目のスコアの推定

文献 [3] の手法では、各端末で作成されたヒストグラムがネットワーク全体のスコア分布と同じと仮定して  $k$  番目のスコアを推定しており、実際のネットワーク全体のスコア分布を考慮していなかった。そのため、スコア分布に大きな偏りがある分布 (Zipf 分布など) では推定の誤差が大きくなってしまふ。そこで、本稿の提案手法では、ネットワーク全体のスコア分布の種類が既知であるという想定に基づき、確率密度関数を用いて、スコア分布、および  $k$  番目のスコアを推定する。具体的には、推定を行う端末は、推定用ヒストグラムの各階級の中央値が、その階級に含まれるスコア数ずつ存在すると仮定し、ネットワーク内のデータのスコア分布の確率密度関数  $f(x)$  におけるパラメータを求める。次に端末は、確率密度関数  $f(x)$  の上側累積確率  $Q(x)$  が、以下の式を満たす  $x$  を求める。

$$Q(x) = \int_x^{\infty} f(t)dt = \frac{k}{n}. \quad (1)$$

ここで、 $x$  は上側累積確率が  $Q(x)$  となるパーセント点を示す。式 (1) により、 $x$  は、上側累積確率がネットワーク内のデータに対する上位  $k$  個のデータの割合となるスコアを表すため、これを  $k$  番目のスコア  $S_k$  と推定する。

例えば、ネットワーク内のデータのスコア分布が正規分布に

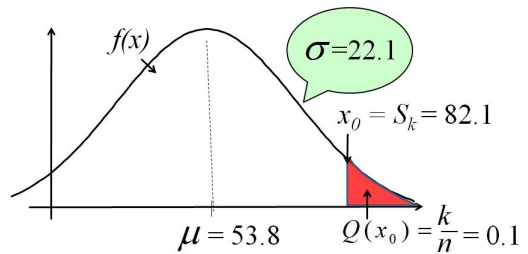


図 5 確率密度関数により推定されたスコア分布

従う場合、確率密度関数  $f(x)$  は、次式ようになる。

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (2)$$

図 5 は、この推定法によって推定されたスコア分布と  $k$  番目のスコアを示している。

### 2.5 $k$ 番目のスコアの補正

提案手法では、各端末は、2.4 節の方法に従って推定した  $k$  番目のスコア以上のスコアをもつデータを返信するため、推定した  $k$  番目のスコアが実際と異なる場合、性能が低下してしまう。例えば、端末が  $k$  番目のスコアを実際より高く推定した場合、実際の  $k$  番目のスコアから推定した  $k$  番目のスコアまでのスコアをもつデータは、検索結果に入るにも関わらず返信されないため、取得精度が低下する。

そこで提案手法では、セーフティマージンを用いてスコアの補正を行う。具体的には、推定した  $k$  番目のスコアをセーフティマージンの大きさだけ減算した値に補正する。ここで、ネットワーク内のデータ数に対して推定用ヒストグラムに含まれるスコア数が少ない場合、推定した  $k$  番目のスコアが実際と異なる可能性、および実際のスコアとの差が大きい可能性が高い。そのため、セーフティマージンは、ネットワーク内のデータ数が多く、ヒストグラム内に含まれるスコア数が少ないほど、大きくなるように設定する。具体的には、各端末は、以下の式を用いてセーフティマージンの大きさ  $M$  を決定する。

$$M = \alpha \cdot \frac{n}{n_{EH}}. \quad (3)$$

ここで  $\alpha$  は、事前に設定される正の定数であり、セーフティマージンに対する重み係数を示す。次に端末は、以下の式を用いて、 $k$  番目のスコアを補正する。

$$S_k' = S_k - M. \quad (4)$$

ここで、 $S_k$  は 2.4 節で推定した  $k$  番目のスコア、 $S_k'$  は補正後の  $k$  番目のスコアを表す。つまり、ネットワーク内のデータ数が多く、ヒストグラム内のスコア数が少ないほど、セーフティマージン  $M$  は大きくなり、 $k$  番目の推定スコアはその分低くなる。

## 3. 評価結果

本章では、提案手法の性能評価のために行ったシミュレーション実験の結果を示す。本実験では、ネットワークシミュレータ Qualnet4.0 [4] を用いた。



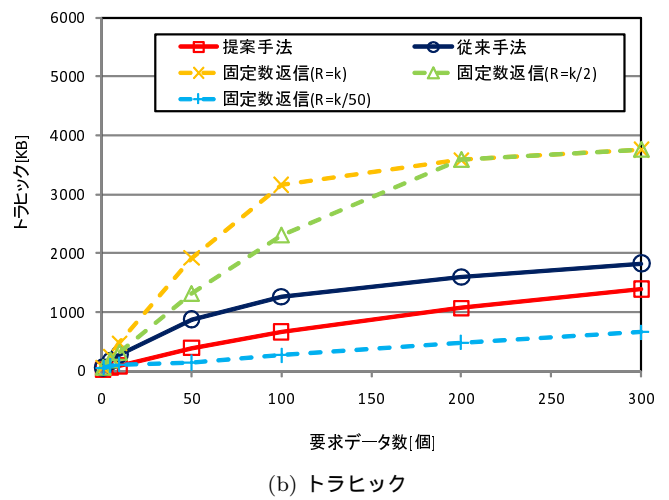
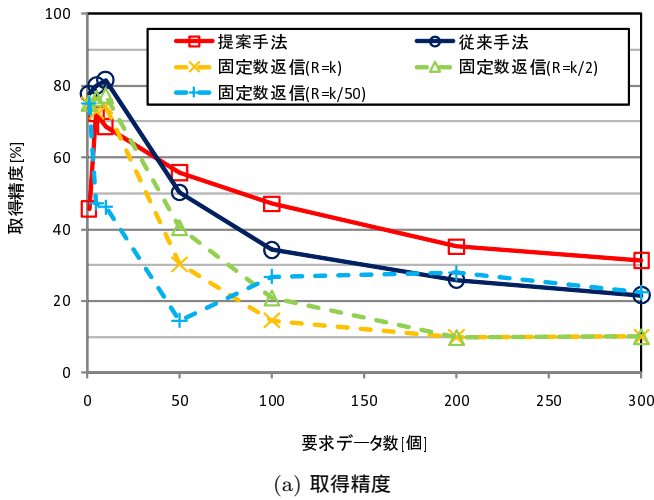


図 6 要求データ数  $k$  の影響

表 1 パラメータ設定

パラメータ	意味	値
$\alpha$	マージン係数	0.75 (0~1)
$k$	要求データ数	100 (1~300)
$d$	データサイズ	1 (0.01~10)

### 3.1 シミュレーション環境

1,000[m]×1,000[m] の 2 次元平面上の領域に 50 台の端末 ( $M_1, \dots, M_{50}$ ) が存在する。各端末はランダムウェイポイント [1] に従い、0.5 [m/秒] の速度で移動する。停止時間は 60[秒] とした。各端末は、IEEE802.11b を使用し、伝送速度 11 [Mbps]、通信伝搬距離が 100 [m] 程度となる送信電力でデータを送信する。ネットワーク内には、 $d$  [KB] のサイズのデータが 5,000 個存在し、各端末はそれぞれ 100 個のデータを保持するものとした。また、ネットワーク全体のスコア分布は正規分布に従うとし、スコアの幅は [80, 320) とした。

各端末は、1000 から 2000 [秒] の間隔で Top- $k$  検索クエリを発行する。ここで、比較手法として、文献 [3] の手法 (従来手法と称する) および、各端末が自身のもつデータの中からスコアの高い  $R = k$ 、 $k/50$  個のデータを固定数返信する手法を用いた。ここで、 $R = k/50$  の場合は、全端末で  $k$  個のデータを返信する。

表 1 に本実験で用いたパラメータを示す。各パラメータは基本的には定数値をとるが、そのパラメータの影響を調べる際には括弧内の範囲で変化させた。以上のシミュレーション環境において、各端末の初期位置をランダムに決定し、7,200[秒] を経過させたときの以下の評価値を調べた。

- 平均取得精度: 上位  $k$  個のデータの中で、検索クエリの発行後 60 [秒] の間に取得できたデータの数の割合を取得精度とする。平均取得精度は、シミュレーション時間内に発行された全クエリに対する取得精度の平均である。
- トラフィック: シミュレーション時間内に発行された全クエリに対する、送信された検索クエリおよびクエリ応答の平均データ量 (1 回分) をトラフィックとする。

### 3.2 要求データ数 $k$ の影響

要求データ数  $k$  を変化させたときの提案手法の性能を調べた。その結果を図 6 に示す。これらのグラフにおいて、横軸は要求データ数  $k$  を表している。縦軸は、図 6 (a) では検索結果の取得精度、図 6 (b) ではトラフィックをそれぞれ表す。

この結果から、本稿の提案手法の方が検索結果の絞り込みの影響が大きく、トラフィックが小さいことがわかる。そのため、 $k$  が増加して、返信データ数が多くなっても、パケットロスが少ないため、取得精度が比較手法よりも高い。しかし、 $k$  が非常に小さい場合、 $k$  番目のスコアを大きく推定する傾向があり、取得精度が低くなってしまふ。 $k = 1$  の場合にそれが顕著に表れている。

### 3.3 マージン係数 $\alpha$ の影響

マージン係数  $\alpha$  を変化させたときの提案手法の性能を調べた。その結果を図 7 に示す。これらのグラフにおいて、横軸はマージン係数  $\alpha$  を表す。縦軸は、図 7 (a) では検索結果の取得精度、図 7 (b) ではトラフィックをそれぞれ表す。

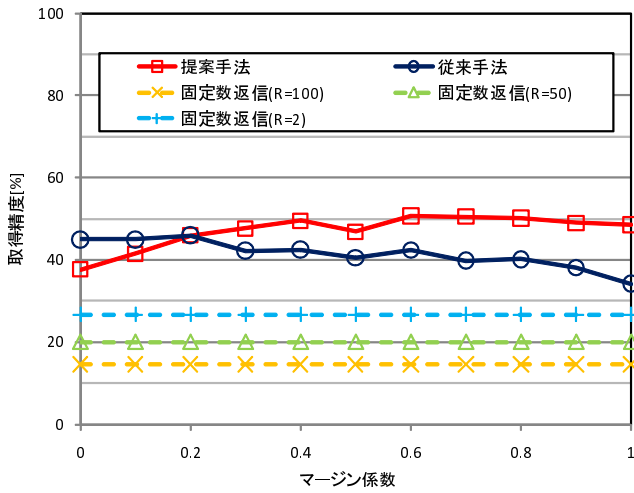
この結果から、マージン係数  $\alpha$  を大きくすると、返信データ数が多くなるため、トラフィックも大きくなる。従来手法、提案手法ともに、初めは  $\alpha$  を大きくすると取得精度が増加する。さらに  $\alpha$  を大きくすると、返信データ数が多くなりすぎ、パケットロスが発生するため取得精度が低下する。提案手法の方が従来手法より、取得精度の低下が始まるのが遅いのは、従来手法の方が  $k$  番目のスコアを初めから低く推定しているためである。

### 3.4 Zipf 分布の影響

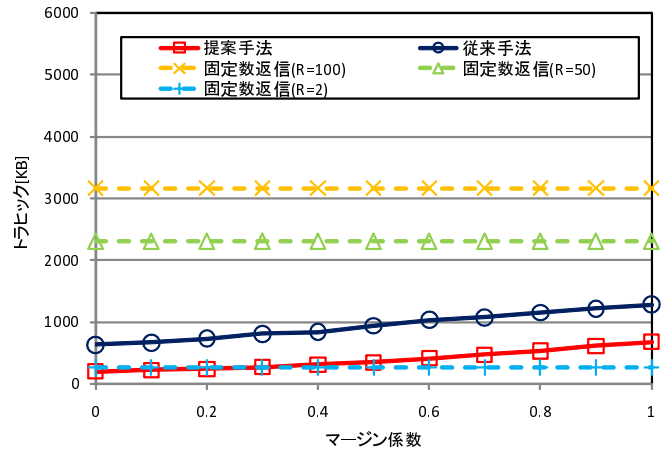
次に、データのスコアが正規分布ではなく Zipf 分布に従う場合のデータ要求数  $k$  の影響を調べた。スコアの幅は [0, 2000) とした。

その結果を図 8 に示す。これらのグラフにおいて、横軸は要求データ数  $k$  を表す。縦軸は、図 8 (a) では検索結果の取得精度、図 8 (b) ではトラフィックをそれぞれ表す。

この結果から、提案手法の方がトラフィックが小さく抑えられ、取得精度も高いことがわかる。一方、従来手法では、推定の際に  $k$  番目のスコアを高すぎたり低すぎたり推定してしまうこと

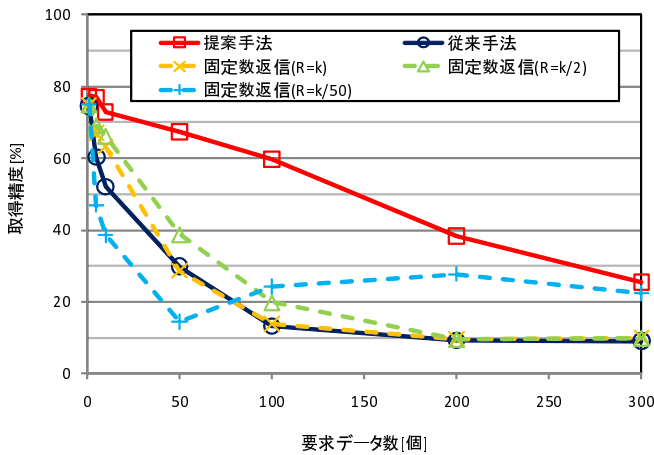


(a) 取得精度

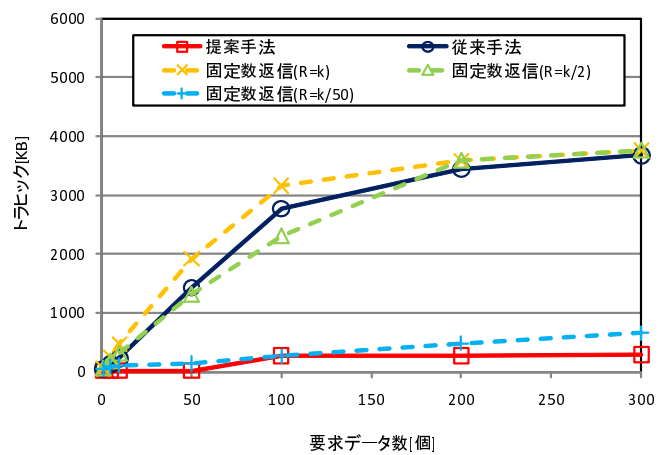


(b) トラフィック

図 7 マージン係数  $\alpha$  の影響



(a) 取得精度



(b) トラフィック

図 8 Zipf 分布における要求データ数  $k$  の影響

が多い。これは、Zipf 分布の場合、スコアに大きな偏りがあるためである。その結果、トラフィックが大きく、かつ取得精度も低くなっている。 $k$  が大きくなった場合、提案手法の取得精度が低くなっているのは、スコアを高く推定してしまう傾向があるためである。そのため、トラフィックも小さくなっている。

#### 4. 結論

本稿では、アドホックネットワークにおいて、トラフィックの削減と検索結果の取得精度の低下の抑止を目的とする Top-k 検索のためのメッセージ処理手法を提案した。提案手法では、ネットワーク全体のスコア分布を確率密度関数を用いて推定することにより、文献 [3] の手法よりも正確なスコア分布の推定が可能になる。

シミュレーション実験の結果から、提案手法は従来手法よりもトラフィックを削減しつつ、取得精度の低下を抑止できることを確認した。

本研究では、簡単化のため、ネットワーク内のデータ数、およびスコアのとりうる値の範囲が既知と想定した。しかし実環

境では、これらは時々刻々と変化すると考えられるため、各端末が予め把握することは困難である。そこで今後は、各端末が取得したデータやそのスコアのログを記録し、定期的に統計をとるなど、上記の値や分布の種類を動的に推定する必要がある。

#### 謝辞

本研究の一部は、(財) 近畿移動無線センター・モバイルワイヤレス助成金、文部科学省科学研究費補助金・基盤研究 S(21220002) および総務省委託研究「ユビキタスサービスプラットフォーム技術の研究開発」の研究助成によるものである。ここに記して謝意を表す。

#### 文献

- [1] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing (WCMC)*, 2(5): 483–502, 2002.
- [2] R. Hagihara, M. Shinohara, T. Hara, and S. Nishio, "A message processing method for top-k query for traffic reduction in ad hoc networks," *Proc. Int. Conf. on Mobile Data Management*, pp.11-20, 2009.

- [3] 佐々木勇和, 萩原亮, 原隆浩, 篠原昌子, 西尾章治郎, “アドホックネットワークにおけるヒストグラムを用いた Top-k 検索手法,” マルチメディア通信と分散処理ワークショップ (DPSWS), pp.13-18, Oct. 2009.
- [4] Scalable Networks: makers of QualNet and EXata, the only multi-core enabled network simulation and emulation software.: <http://www.scalable-networks.com/>.
- [5] G.K. Zipf, “Human behavior and the principle of least effort,” *Addison-Wesley*, 1949.