

## アドホックネットワーク上の スコア分布推定による Top-k 検索手法

佐々木 勇和<sup>†1</sup> 原 隆 浩<sup>†1</sup> 西尾 章 治郎<sup>†1</sup>

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。本論文では、アドホックネットワークにおいて、トラヒックの削減と検索結果の取得精度の低下を抑止を目的とする Top-k 検索手法を提案する。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  $k$  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持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。また、返信先の端末とのリンク切断を検出した端末は、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。

### A Top-k Query Processing Method by Predicting Score Distribution in Mobile Ad Hoc Networks

YUYA SASAKI,<sup>†1</sup> TAKAHIRO HARA<sup>†1</sup> and SHOJIRO NISHIO<sup>†1</sup>

In mobile ad hoc networks, 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 the  $k$  highest scores. In this paper, we propose a query processing method for top-k query for reducing traffic and also keeping high accuracy of the query result. In this method, each node constructs a histogram and estimates the  $k$ -th score in the entire network. When transmitting a reply, each mobile node sends back only data items whose scores are larger than the estimated  $k$ -th score to reduce traffic as much as possible. Moreover, when a mobile node detects the disconnection of a radio link, it sends the reply to another neighboring node to keep high accuracy of the query result.

### 1. はじめに

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ともない、ルータ機能を持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<sup>5),9)</sup>。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<sup>4)</sup>。

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持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多の場合、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、不要なトラヒックが発生する。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。また、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トポロジが動的に変化する。Top-k 検索の実行中に端末間のリンクが切断された場合、検索クエリを発行した端末が検索結果に含まれるデータを取得できず、検索結果の精度が低下する可能性がある。

そこで本論文では、アドホックネットワークにおいて、トラヒックの削減と検索結果の取得精度の低下を抑止を目的とする Top-k 検索手法を提案する。提案手法では、各端末が自身の持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持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。

以下では、2 章で関連研究を紹介する。3 章で本論文の想定環境を示し、4 章で提案手法

<sup>†1</sup> 大阪大学大学院情報科学研究科マルチメディア工学専攻

Department of Multimedia Engineering, Graduate School of Information Science and Technology, Osaka University

について述べる．5章でシミュレーション実験の結果を示す．最後に6章で本論文のまとめと今後の課題について述べる．

## 2. 関連研究

本章では，Top-k 検索に関する従来研究について紹介する．Top-k 検索は，さまざまな分野においてさかんに研究が行われているが，アドホックネットワークにおける Top-k 検索に関する先行研究はほとんどない．そこで，端末が自身や周辺端末情報のみを用いて自律分散的に動作するという観点でアドホックネットワークと類似している非構造型 Peer to Peer (P2P) ネットワークに着目し，Top-k 検索について紹介する．非構造型検索 P2P ネットワークでは，検索クエリは検索クエリ発行端末から隣接端末にブロードキャストされ，これを受信した端末からさらにその隣接端末に転送されるフラッディングという方法で処理される．

文献 1) では，検索クエリとクエリ応答を適切な隣接端末に送信することでトラフィックを削減する手法を提案している．またこの手法では，クエリ応答の返信時に隣接端末が存在しない端末は，検索クエリを発行した端末にクエリ応答を直接返信する．この研究では，検索クエリの転送にユニキャストを想定している点，任意の端末間で直接通信が可能なネットワークを想定している点で本研究と異なる．また文献 2) では，1 回の検索クエリの送信で 1 個のデータを取得し，この処理を  $k$  個の検索結果が取得できるまで繰り返す Top-k 検索手法を提案している．具体的には，最初の検索クエリによって，各端末から最上位 (1 番目) のデータのスコアを返送してもらい，その中で最もスコアの高いデータ (ネットワーク内で 1 番目のスコアを持つデータ) を取得する．次に，ネットワーク内で 1 番目のスコアを返送した端末に対して，2 番目のデータのスコアを返送してもらい，他の端末が返送した 1 番目のデータのスコアと比較することで，ネットワーク内の 2 番目のスコアを持つデータを取得する．この処理を  $k$  番目のスコアを持つデータを取得するまで繰り返す．これらの手法では，返信されるデータ数は少ないものの，検索結果の取得に時間がかかるため，ネットワークトポロジが動的に変化するアドホックネットワークでは有効ではない．

文献 6), 7) では，端末がクエリ応答の要求数  $k$  を含めた検索クエリをフラッディングし，これを受信した端末が自身の持つデータの中から上位  $k$  個のスコアを持つものを返信する手法を提案している．この研究では，検索結果に入らない不要なデータの返信によるトラフィックが多く，1 章で述べた問題点を解決できない．また文献 8) では，P2P コンテンツ検索システムにおける Top-k 検索手法として，検索クエリの受信時に各端末が上位  $k$  個のスコアを持つデータのみを返信する Fixed-k query 手法を提案している．この手法では，各端末が他端末からのクエリ応答を中継する際，上位  $k$  個に入る可能性のあるデータのみを返信することで，クエリ応答によるトラフィックをある程度削減できるが，基本的に各端末が  $k$  個のデータを返信するため，その効果は十分ではない．また Fixed-k query 手法を拡張し，各端末が，クエリ発行端末からのホップ数と自身の隣接端末数を考慮して，クエリ応答で返信するデータ数を決定する手法，およびすべての隣接端末からのクエリ応答を受信した後，返信すべき順位のスコアを持つデータのみを返信する手法を提案している．これらの手法は，クエリ応答によるトラフィックを削減できるが，各端末の持つデータやそのスコアに偏りがある環境では，上位  $k$  個のデータが返信されず，取得精度が低下する．文献 11) では，文献 8) の方式を拡張し，検索クエリの転送時に，自身までの転送経路上の端末の持つ上位  $k$  個のデータのスコアを転送する手法を提案している．この手法では，各端末の持つデータのスコアを転送することで，返信するデータ数を決定している点で本研究とは異なる．

文献 4) では，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維持を実現する Top-k 検索手法を提案している．この手法では，ネットワーク全体のデータのスコア分布を推測するのではなく，クエリが伝搬した経路上の端末が持つデータのスコアから， $k$  番目のスコアを，そのスコアを持つデータが確実に  $k$  個以上存在することを保証できるスコアとして推定する．具体的にこの手法では，クエリ経路上の端末が持ついくつかのデータのスコアを基準値として添付し，中継端末においてこれらの基準値が更新されていく．クエリの返送時には，この基準値のうち， $k$  番目のスコアに該当する値を閾値として，その値以上のスコアを持つデータのみをクエリ発行端末に返信していく．この手法では，上位  $k$  個に入らないことが確実なデータの返信を防ぐことはできるが，検索結果の絞り込みの効果はそれほど大きくない．一方，本論文の提案手法は，中継端末の持つデータのスコア分布からネットワーク全体の  $k$  番目のスコアを推定するため，絞り込みの効果が大きい．しかし，推定の精度が低い場合，必ずしも上位  $k$  個のデータを取得できない可能性がある．

文献 4) では，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維持を実現する Top-k 検索手法を提案している．この手法では，ネットワーク全体のデータのスコア分布を推測するのではなく，クエリが伝搬した経路上の端末が持つデータのスコアから， $k$  番目のスコアを，そのスコアを持つデータが確実に  $k$  個以上存在することを保証できるスコアとして推定する．具体的にこの手法では，クエリ経路上の端末が持ついくつかのデータのスコアを基準値として添付し，中継端末においてこれらの基準値が更新されていく．クエリの返送時には，この基準値のうち， $k$  番目のスコアに該当する値を閾値として，その値以上のスコアを持つデータのみをクエリ発行端末に返信していく．この手法では，上位  $k$  個に入らないことが確実なデータの返信を防ぐことはできるが，検索結果の絞り込みの効果はそれほど大きくない．一方，本論文の提案手法は，中継端末の持つデータのスコア分布からネットワーク全体の  $k$  番目のスコアを推定するため，絞り込みの効果が大きい．しかし，推定の精度が低い場合，必ずしも上位  $k$  個のデータを取得できない可能性がある．

## 3. 想定環境

本論文では，アドホックネットワークを構成する各端末が，自身と他の端末の持つデータに対して 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  $k$  個のスコアを持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，すべてのデータのサイズは等しいものとする．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（たとえば正規分布）に従っているものとする。データは、周期的に更新され、これにともない、データのスコアも更新される。

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。実験環境においてデータ数が既知な場合の例としては、災害地での被災者数の調査と診断がすでに完了しており、その後に関連情報を検索する状況などが考えられる。スコアの分布と上限値、下限値が既知な場合としては、たとえば、血圧などは、一般的に正規分布に従うことが知られており、常識的な範囲で下限値、上限値（たとえば、0 から 300）も既知であると考えられる。各端末の持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、各端末の持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。各端末は、データの複製を作成しないものとする。

各通信は、hop by hop により送信処理を行うものとする。具体的には、クエリの伝搬には、アプリケーション層でのフラッディングを用い、その処理の中で、自身の親端末、および他の隣接端末を把握する。クエリ応答にはアプリケーション層でのユニキャストを用いる。そのため、ネットワーク層において既存のルーティングプロトコルは用いない。

#### 4. 提案手法

本章では、トラフィックを削減しつつ、検索結果の取得精度の低下を抑止する Top-k 検索を提案する。以下では、まず提案する手法の概要について述べ、その後、各端末が作成するヒストグラムについて説明する。次に、検索クエリの転送時、クエリ応答の返信時、およびリンク切断の検出時におけるメッセージの処理手順について説明する。最後に、クエリ応答を返信する端末が、ヒストグラムを用いてスコア分布と  $k$  番目のスコアを推定する方法、および推定した  $k$  番目のスコアを補正する方法について説明する。

##### 4.1 ヒストグラムの作成

各端末は、検索条件から算出された自身の持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（個数）を数えたものであり、スコアの分布状況を表すのに用いられる。提案手法では、端末  $M_i$  の持つデータの中で、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、ネットワーク内のデータ

のスコアがとりうる値の範囲  $[MIN, MAX]$  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。

たとえば、ネットワーク内のデータのスコアが、平均 50、分散 20 の正規分布に従い、スコアのとりうる値の範囲が  $[0, 100)$ 、階級数  $C = 10$  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  $H_i$  ( $1 \leq i \leq 5$ ) およびネットワーク全体のヒストグラムを図 1 に示す。

ここで、各端末はクエリ伝搬時、クエリ応答時、および  $k$  番目のスコア推定時においてそれぞれ以下の 3 種類のヒストグラムを作成する。

- クエリヒストグラム：  
自身と、クエリ発行端末から自身までの経路上の（祖先）端末の保持するデータから構成されるヒストグラムで、クエリ伝搬時に添付される。
- 応答ヒストグラム：  
自身と自身の子孫端末が保持するデータから構成されるヒストグラムで、クエリ応答に添付される。
- 推定ヒストグラム：  
自身と、クエリ発行端末から自身までの経路上の（祖先）端末、および自身の子孫端末が保持するデータから構成されるヒストグラムで、スコア分布、および  $k$  番目のスコ

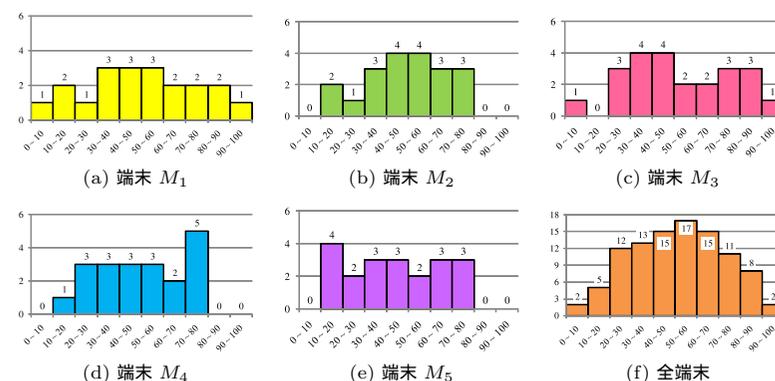


図 1 各端末の作成したヒストグラムとネットワーク全体のヒストグラム

Fig. 1 Histogram constructed by each node and that in the entire network.

アの推定に用いる．

## 4.2 メッセージの処理手順

### 4.2.1 検索クエリの転送時

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、そのスコア分布はネットワーク内のデータのスコア分布に近くなるため、端末はスコア分布と  $k$  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メッセージに添付して送信し、検索クエリメッセージ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  $M_p$  と検索クエリメッセージを受信した端末  $M_q$  の動作について説明する．

- (1) 端末  $M_p$  は検索条件、および検索での要求データ数  $k$  を指定する．また、検索条件から自身の持つデータのスコアを算出し、4.1 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) 端末  $M_p$  は、すべての隣接する端末に検索クエリメッセージを送信する．このメッセージには、検索クエリ発行端末  $M_p$  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  $QH$  が含まれる．ここで、経路端末リストは、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、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る．この場合、経路端末リストには  $M_p$  のみが含まれ、クエリヒストグラム  $QH$  は  $M_p$  のヒストグラム  $H_p$  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、経路端末リストを記録する．手順 (4) へ進む．  
受信した検索クエリメッセージが、すでに受信したものであれば、手順 (5) へ進む．
- (4) 端末  $M_q$  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、検索クエリメッセージに含まれるクエリヒストグラム  $QH$  に  $H_q$  を統合する．また、自身の識別子  $M_q$  を経路端末リストに追加する． $M_q$  は、自身の隣接端末に検索クエリメッセージを送信し、手順 (3) に戻る．
- (5) 検索クエリメッセージを再受信した端末  $M_q$  は、そのメッセージの経路端末リストに  $M_q$  を含んでいない場合、そのリスト末尾の端末を隣接端末とし、また、経路端末リストに含まれる端末数から 1 減らした数をクエリ発行端末からその端末までのホップ数として、隣接端末リストに記録する．また、経路端末リストの末尾から 2 番

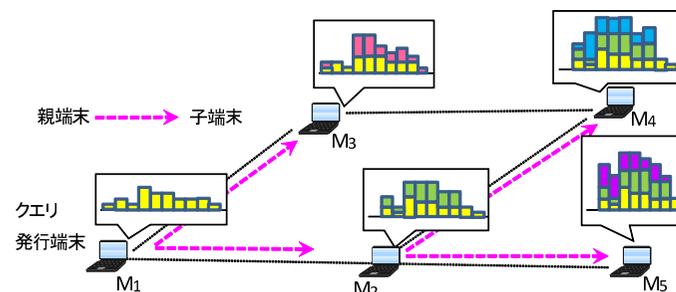


図 2 検索クエリの転送例  
Fig.2 Query message transmission.

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メッセージに添付することで、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を把握でき、隣接端末リストにより、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．さらに、クエリ伝搬の処理の中で、自身の親端末、子端末、および隣接端末の情報、クエリ伝搬経路上の端末が持つデータのスコアから構成されるクエリヒストグラムを把握している．これらの情報に加えて、3 章で述べたように、各端末は、全端末数、スコアが従っているスコア分布の種類を知っている．

図 2 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．図において、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、端末間をつないでいる破線は無線リンク、つまり隣接関係を示している．また、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、統合の状態を分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 1 で表されるものとする．

### 4.2.2 クエリ応答の返信時

4.2.1 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、スコア分布や  $k$  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メッセージにも添付して送信する．また、不要なデータの返信を抑止するため、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データを検出した場合、そのデータを返信しない。以下では、4.2.1 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、自身の送信した検索クエリメッセージに添付したクエリヒストグラムを用いて、スコア分布と  $k$  番目のスコアの推定 (4.3 節)、および  $k$  番目のスコアの補正 (4.4 節) を行う。その後  $M_r$  は、クエリ応答メッセージを自身の親に送信する。このメッセージには、検索クエリ発行端末  $M_p$  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  $RH$  が含まれる。ここで、返信リストは自身の持つデータの中で、推定した  $k$  番目のスコア以上のスコアとそのスコアを持つデータを示す。また、応答ヒストグラムは、自身および自身の子孫である端末のヒストグラムを統合したものであり、この場合、 $M_r$  のヒストグラム  $H_r$  となる。
- (2) すべての子からクエリ応答メッセージを受信するか、検索クエリメッセージを送信してから一定時間経過した端末  $M_s$  は、受信したすべてのクエリ応答メッセージに含まれる応答ヒストグラム、および検索クエリメッセージ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、および端末  $M_s$  のすべての子孫端末のヒストグラムを統合したものである。端末  $M_s$  は、この統合したヒストグラムを用いて、スコア分布と  $k$  番目の推定 (4.3 節)、および  $k$  番目のスコアの補正 (4.4 節) を行う。手順 (3) へ進む。
- (3) 端末  $M_s$  は、クエリ応答メッセージを作成し、自身の親に送信する。ここで、返信リストは、受信したすべてのクエリ応答メッセージに含まれる返信リストおよび自身の持つデータの中で、推定した  $k$  番目のスコア以上のスコアとそのスコアを持つデータが含まれる。ただし、該当するデータ数が要求データ数  $k$  より大きい場合、上位  $k$  個のスコアとそのスコアを持つデータのみが含まれる。また、応答ヒストグラムは、自身のヒストグラム  $H_s$  と受信したすべてのクエリ応答メッセージに含まれる応答ヒストグラムを統合したものが含まれる。
- (4) クエリ応答の返信後、別の子からクエリ応答メッセージを受信した端末  $M_s$  は、手順 (2) と同様にヒストグラムを統合し、再びスコア分布と  $k$  番目のスコアの推定 (4.3 節)、および  $k$  番目のスコアの補正 (4.4 節) を行う。その後端末  $M_s$  は、手順 (3) と同様にクエリ応答メッセージを作成し、自身の親に送信する。ただし、データの重複を防ぐため、すでに返信したデータとそのスコアは返信リストから削除する。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持つデータのみを返信

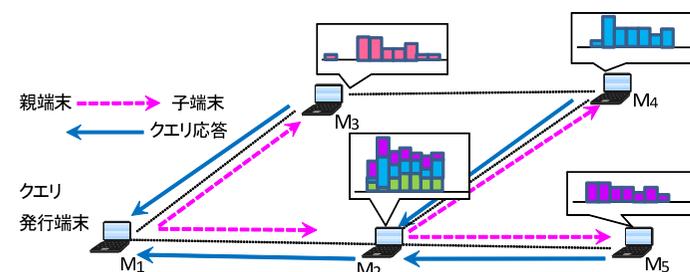


図 3 クエリ応答の返信例  
Fig. 3 Reply transmission.

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である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メッセージに添付することで、クエリ発行端末に近い端末ほど、推定に用いる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。これらにより、各端末は検索結果に入るデータを返信しつつ、検索結果に入らないデータの返信を抑制でき、取得精度を低下させずにトラフィックを削減できる。

図 2 および図 3 を用いて、クエリ応答の返信例を説明する。図 3 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、端末間の破線は図 2 と同様に無線リンク (隣接関係) を示す。また、実線の矢印はクエリ応答の流れを示す。たとえば、 $M_4$  は、図 2 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持つデータを、親である  $M_2$  に返信する。

#### 4.2.3 リンク切断の検出時

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ト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、クエリ応答メッセージ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、リンク切断を検出した端末は、他の隣接端末にクエリ応答を送信する。以下では、親とのリンク切断を検出した端末  $M_t$  が、他の隣接端末にクエリ応答を返信する動作について説明する。

- (1) リンク切断時に親端末と切断した端末  $M_t$  は、親以外の隣接端末にクエリ応答メッセージを転送する。ここで、自身の隣接端末リストから、転送先の隣接端末を決定す

る．複数の端末と隣接している場合，トラヒックや遅延の増分をできる限り小さくするため，隣接端末のうちクエリ発行端末からのホップ数が最小のものに転送する．どの端末とも隣接していない場合，クエリ応答メッセージを破棄する．

- (2) 自身の子以外の端末からクエリ応答メッセージを受信した端末  $M_u$  は，端末  $M_t$  がクエリ発行端末から自身までの経路上に存在しない場合，受信したメッセージを子からのメッセージと同様に扱い，4.2.2 項の手順に従って親に返信する．一方，端末  $M_t$  がクエリ発行端末から自身までの経路上に存在する場合（端末  $M_t$  が端末  $M_u$  の親端末である場合も含む）， $M_u$  が親に返信したメッセージは端末  $M_t$  まで返信されてしまう．そこで，この場合，端末  $M_u$  は，受信したクエリ応答メッセージを親以外のいずれかの隣接端末に送信する．親以外の端末と隣接していない場合，クエリ応答メッセージを破棄する．

リンク切断の検出では，端末が親以外の隣接端末にクエリ応答を返信することで，検索結果をできる限りクエリ発行端末まで返信し，取得精度の低下を抑制できる．ただし，切断を検出した端末が孤立している場合や親からメッセージを受信した端末が他の端末と隣接していない場合，クエリ応答がクエリ発行端末まで返信されないため，取得精度が低下してしまう．

図 3 を用いて，リンク切断時の処理例を説明する．たとえば， $M_2 - M_4$  間のリンクが切断した場合， $M_4$  はクエリ発行端末からのホップ数が最小の  $M_3$  にクエリ応答メッセージを送信する．一方， $M_1 - M_2$  間のリンクが切断した場合，端末  $M_2$  は  $M_4$  にクエリ応答メッセージを送信する．これを受信した  $M_4$  は， $M_2$  がクエリ発行端末から自身までの経路上に存在するため，親以外の隣接端末（ここでは  $M_3$ ）にクエリ応答メッセージを送信する．

### 4.3 スコア分布と $k$ 番目のスコアの推定

提案手法では，各端末はクエリ応答を返信するとき，クエリ発行端末から自身までの経路上の端末，および自身の子孫の端末のヒストグラムを統合した推定用ヒストグラム  $EH$  を用いて，ネットワーク内のデータのスコア分布および  $k$  番目のスコアを推定する．以下ではまず，ヒストグラム内のスコア数を用いて推定する方法を説明する．次に，確率密度関数を用いて推定する方法を説明する．

#### 4.3.1 ヒストグラム内のスコア数による推定

ヒストグラム内のスコア数による推定では，各端末は推定用ヒストグラム  $EH$  がネットワーク内のデータのスコア分布と等しいと考え，スコア分布と  $k$  番目のスコアを推定する．具体的には，推定を行う端末は，以下の式を満たす最大の自然数  $a (\leq C)$  を求める．

$$\sum_{l=a}^C \frac{n}{n_{EH}} \cdot e_l \geq k, \tag{1}$$

$$n_{EH} = \sum_{l=1}^C e_l. \tag{2}$$

ここで， $n$  はネットワーク内のデータ数， $k$  は要求データ数， $C$  は推定用ヒストグラム  $EH$  の階級数， $e_j (1 \leq j \leq C)$  は階級  $c_j$  に含まれるスコアの数を示す．つまり， $n_{EH}$  は推定用ヒストグラムに含まれるスコア数を示し，式 (1) は推定用ヒストグラム内のスコア数とネットワーク内のデータ数の比に基づいて，推定用ヒストグラムの各階級に含まれるスコア数を増やしたヒストグラムを，ネットワーク内のデータのスコア分布として推定していることを示す．このとき  $a$  は，階級  $c_a$  以上の階級に含まれるスコア数の合計が  $k$  以上となる最大の整数を示す．次に端末は，以下の式を用いて， $k$  番目のスコア  $S_k$  を求める．

$$S_k = MIN + \frac{a(MAX - MIN)}{C}. \tag{3}$$

ここで， $MIN$  と  $MAX$  は，スコアのとりうる値の最小値と最大値を示し， $S_k$  は階級  $c_a$  の下限値を示す．

ヒストグラム内のスコア数による推定を行った場合， $k$  番目のスコアとして，該当するヒストグラムの階級の下限值を用いるため，実際の  $k$  番目のスコアより低く推定する傾向がある．そのため，返信データ数が  $k$  個より多くなり，必要なデータの取りこぼしは少なくなる．しかし，その反面，返信データの増加により，トラヒックが大きくなってしまいう傾向がある．

図 4 (a) のヒストグラムを用いて，ヒストグラム内のスコア数による推定を行う例を説明

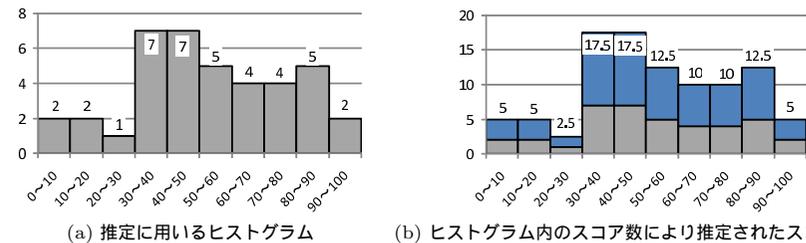


図 4 ヒストグラムを用いたスコア分布推定  
Fig. 4 Score distribution estimation by using histogram.

する．この図は， $M_3$  が 4.2.1 項のメッセージ処理において作成した推定用ヒストグラムを示す．式 (2) より  $n_{EH} = 40$  となるため， $n = 100$  とすると， $M_3$  はヒストグラムの各階級に含まれるデータ数を 2.5 倍したものを，ネットワーク内のデータのスコア分布と推定する (図 4(b))． $k = 10$ ， $c = 10$  とすると， $a = 9$  において，

$$\sum_{l=9}^{10} \frac{100}{40} \cdot e_l = \frac{100}{40} (2 + 5) \geq 10.$$

と  $k$  を超えるため， $M_3$  は  $k$  番目のスコア  $S_k$  を階級  $c_9$  の下限値 80 と推定する．

#### 4.3.2 確率密度関数による推定

確率密度関数による推定では，各端末は，ネットワーク内のデータのスコア分布の種類を把握していることを利用して，スコア分布と  $k$  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．

$$Q(x) = \int_x^{\infty} f(t) dt = \frac{k}{n}. \quad (4)$$

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 (4) 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．

確率密度関数による推定を行った場合， $k$  番目のスコアの推定値がヒストグラムの各階級の下限值，上限値に依存せず，より厳密に推定することができる．そのため，返信データ数はネットワーク全体で  $k$  個程度になり，スコア数を用いる手法よりトラフィックが小さくなる．しかし，収集したデータのスコアに大きな偏りがあると， $k$  番目のスコアの推定誤差が大きくなるため，スコアを誤って高く推定した場合は， $k$  番目の付近のスコアを持つデータを取りこぼしてしまう可能性がある．

図 4(a) のヒストグラムを用いて，確率密度関数による推定を行う例を説明する．ネットワーク内のデータのスコア分布が正規分布に従う場合，確率密度関数  $f(x)$  は，

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (5)$$

となる．ヒストグラムより平均  $\mu$  と標準偏差  $\sigma$  は，それぞれ  $\mu = 53.8$ ， $\sigma = 22.1$  となるた

め， $n = 100$ ， $k = 10$  として，式 (4) を満たす  $x (= S_k)$  を求めると， $S_k = 82.1$  となる．

#### 4.4 $k$ 番目のスコアの補正

提案手法では，各端末は，4.3 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持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．たとえば，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持つデータは，検索結果に入るにもかか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．

$$M = \alpha \cdot \frac{n}{n_{EH}}. \quad (6)$$

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．

$$S_k' = S_k - M. \quad (7)$$

ここで， $S_k$  は 4.3 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．

## 5. 評価実験

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 Qualnet<sup>10)</sup> を用いた．

#### 5.1 シミュレーション環境

600 [m] × 600 [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<sup>3)</sup> 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．クエリ伝搬時は，各端末はトランスポート層に

において、UDP でブロードキャストを行い、隣接端末にクエリを転送する。クエリ返信時は、クエリ伝搬時に把握した親端末に、TCP でユニキャストを行う。また、MAC 層以下では、802.11 に従い、RTS/CTS により送信制御を行う。パケットが損失した場合、再送の回数は 3 回までとする。これは、再送処理の繰返しにより、無線回線の混雑を防ぐためである。ネットワーク内には、サイズ  $d$  [KB] 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、5.4 節を除いてネットワーク全体のスコア分布は正規分布に従うものとし、スコアの幅は [80, 320) とした。

各端末は、3,000 [秒] の間隔で Top-k 検索クエリを発行する。Top-k 検索手法には、提案手法において、スコア分布をヒストグラム内のスコア数により推定する場合 (4.3.1 項) と確率密度関数により推定する場合 (4.3.2 項) を用い、 $C = 20$  とした。ここで、文献 4) で提案されている手法において、基準値数が 20 の場合を、比較手法として用いた。さらに、各端末が自身の持つデータの中からスコアの高い  $R = k, k/2, k/50$  個のデータを固定数返信する手法も、比較手法として用いた。たとえば、 $R = k/50$  の場合は、全端末で  $k$  個のデータを返信する。なお、準備実験の結果、提案手法において、ヒストグラム内のスコア数によりスコア分布を推定する方法では、もともと  $k$  番目のスコアを低めに推定するため、スコア補正 (4.4 節) の効果がなかった。そのため、この手法では、スコア補正を行わないものとした。一方、確率密度関数を用いる手法では、 $\alpha = 0.75$  とした。

本実験では、データサイズ  $d$  は基本的に 1 [KB] とし、5.2 節では 0.01 ~ 10 [KB] の間で変化させた。ここで、本論文では、高画質な画像などサイズが大きなデータを扱うようなアプリケーションではなく、たとえば、災害地において、被災者の健康状況を診断し、名前、性別、年齢、症状、場所や低解像度の写真など、比較的小さなデータサイズのタプルを蓄積するようなアプリケーションを想定している。このようなアプリケーションでは、最大で 10KB という環境設定は適当と考える。また要求データ数  $k$  は基本的に 100 とし、5.3 節、5.4 節では 1 ~ 300 の間で変化させた。

Top-k 検索クエリには制限時間を設けて、各端末は  $(60 - (1.2 \times \text{クエリ発行端末からのホップ数}))$  [秒] の制限時間に従って返信を待つものとした。したがってクエリ発行端末は、60 秒経過後、すべての端末から返信がない場合、そこで検索を終了する。ここで、予備実験の結果より、制限時間を長くしていくと、基本的に取得精度が向上し、約 50 秒で定常状態になることが分かっている。この制限時間を 50 秒以上に設定しても、取得精度にほとんど差はでなかった。そのため、本実験では、少し余裕を持たせて、クエリ返信を待つ制限時間を 60 秒と設定した。

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。

- 検索結果の取得精度  
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。
- トラフィック  
シミュレーション時間内に Top-k 検索によって送信された検索クエリメッセージ、およびクエリ応答メッセージの top-k 検索ごとの平均バイト数。

## 5.2 データサイズの影響

データサイズ  $d$  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 5 に示す。図中の各グラフにおいて、「確率密度関数」、「ヒストグラム内のスコア数」は、それぞれ提案手法においてスコア分布を確率密度関数を用いて推定する場合、およびヒストグラム内のスコア数により推定する場合を示す。また、「文献 4) の手法」は、文献 4) で提案されている手法、「固定数返信 ( $R = x$ )」は、各端末が自身の持つデータの中からスコアの高い  $R (= x)$  個のデータを返信する手法を示す。これらのグラフにおいて、横軸はデータサイズ  $d$  を表している。縦軸は、図 5 (a) では取得精度、図 5 (b) ではトラフィックを表している。

この結果から、データサイズが大きくなると、いずれの手法も、取得精度は低くなること分かる。これは、各端末が返信するクエリ応答メッセージのサイズが大きくなり、メッセージを損失する機会が増加するからである。データサイズが小さい範囲では、提案手法のヒスト

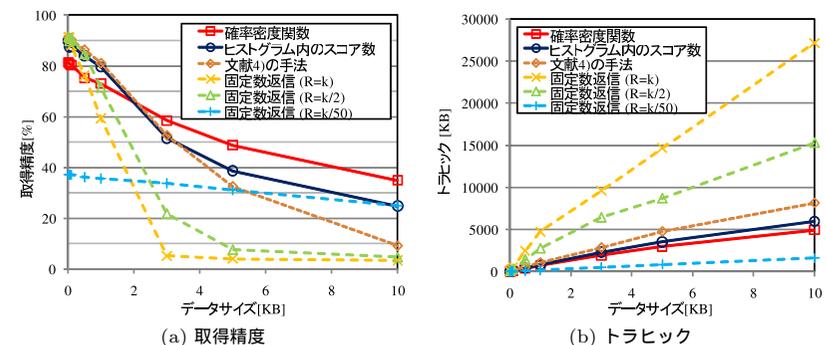


図 5 正規分布におけるデータサイズ  $d$  の影響

Fig. 5 Effects of data size (Normal distribution).

グラム内のスコア数を用いる場合の取得精度は、文献 4) の手法や固定数返信 ( $R = 100$ ) の場合とほぼ等しくなっている。これは、データサイズが十分小さいため、いずれの手法においても、トラフィック増加によるパケットロスがほとんど発生していないためである。ヒストグラム内のスコア数を用いる場合、4.3.1 項で述べたように、 $k$  番目のスコアを低めに推定するという特徴から、推定誤差による取得精度の低下はほとんど認められない。

一方、確率密度関数を用いた場合では、データサイズが小さい範囲において、取得精度が提案手法のヒストグラム内のスコア数を用いる方法、文献 4) の手法、および固定数返信  $R = 100$ , 50 より低い。この手法では、4.3.2 項で述べたように、返信データ数を  $k$  個程度に抑えるため、スコアを誤って高く推定したとき、 $k$  番目のスコア付近を持つデータを返信時に取りこぼしやすくなる。特に、データサイズが小さい場合、パケットロスによる影響がほとんどないため、推定誤差の影響が取得精度に大きく影響し、取得すべきデータが返信されず、取得精度が低下してしまう。つまり、確率密度関数を用いる手法では、データサイズがそれほど大きくない環境において、取得精度低下の最大の要因は、推定誤差によるデータの取りこぼしとなる。

データサイズが大きい範囲では、提案手法の確率密度関数を用いる場合の取得精度が高い。これは、提案手法でヒストグラム内のスコア数を用いる場合や文献 4) の手法、固定数返信 ( $R = 100$ ) の場合、返信するデータ数が多いため、トラフィックが大きくなり、パケットロスによりメッセージが損失する機会が多いからである。提案手法の確率密度関数を用いた場合では、返信データ数が抑制されているため、他の手法よりもトラフィックが小さく、取得精度が高くなる。一方、ヒストグラム内のスコア数を用いる場合、推定した  $k$  番目のスコアを小さく推定する傾向があるため、確率密度を用いた場合よりもトラフィックが大きくなる。つまり、この手法では、取得精度低下の最大の要因は、トラフィック増大によるパケットロスとなる。しかし、文献 4) の手法や固定数返信 ( $R = k/2, k$ ) の場合と比べると、トラフィックは抑えられている。

文献 4) の手法は、クエリに基準値のみを添付するため、クエリサイズはそれほど増加しないが、上位  $k$  個以上のデータを収集することが確実となる安全な推測を行うため、返信データの絞り込みが十分ではなく、トラフィックが大きくなる。データサイズが小さい範囲では、パケットロスの影響をほとんど受けないため、取得精度は高く、トラフィックの絶対量もそれほど大きくならない。しかし、データサイズが大きい場合、絞り込みが不十分のため、トラフィックが大きくなるだけでなく、パケットロスにより取得精度も低くなる。一方、提案手法では、経路上の端末が持つデータのスコア情報から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ため、文献 4) の手法よりも返信データの絞り込めることが特徴である。そのため、データサイズが大きい場合には、トラフィックの低減効果により文献 4) の手法よりもトラフィック、取得精度ともに良い性能を示す。一方、返信データサイズが小さい場合には、推定誤差により、特に確率密度関数を用いて推定する手法において、文献 4) の手法よりも取得精度が低くなる。

ここで、実験結果において、取得精度が最大で 90%程度となるのは、ネットワーク分断が発生しており、すべての端末にアクセスできていないためである。本シミュレーション環境の領域サイズと通信範囲においては、平均で全体の約 95%の端末が相互接続しており、5%程度の端末が分断されている。また、実験環境において、ネットワーク全体で 3,000 [KB] 程度のトラフィックが発生すると、パケットロスが発生しやすくなる。本実験では、各端末 1 台が異なるタイミングでクエリを発行しているため、1 つのクエリに対する返信が 3,000 [KB] 程度になると、パケットロスが頻繁に発生する。そのため、各手法において、図 5 (b) でトラフィックが 3,000 [KB] となるデータサイズの付近で、図 5 (a) の取得精度が急激に低下している。

### 5.3 データ要求数 $k$ 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) では取得精度、図 6 (b) ではトラフィックを表している。

全体として、データ要求数  $k$  が大きくなると、各手法においてトラフィックが増加するため、固定数返信 ( $R = k/50$ ) の場合を除いて、パケットロスの影響により取得精度が低下し

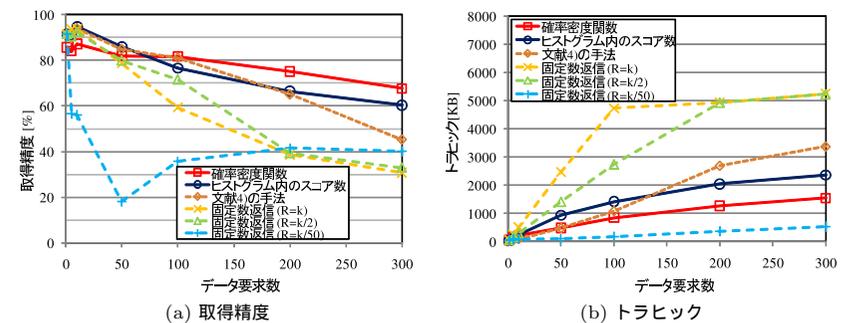


図 6 正規分布における要求データ数  $k$  の影響

Fig. 6 Effects of  $k$  (Normal distribution).

ている．提案手法は，5.2 節の結果と同様に，返信データサイズが大きい環境では，文献 4) の手法や固定数返信 ( $R = k, k/2$ ) の場合と比較して，返信データ数の絞り込みの効果により，トラヒックが低減して，取得精度が高くなっている．

提案手法において，ヒストグラム内のスコア数による推定を行った場合， $k$  が小さい範囲では， $k$  番目のスコアを実際より小さく推定する傾向により，データの取りこぼしはほとんど生じないため，高い取得精度を示す．しかし， $k$  が大きい範囲では， $k$  番目のスコアを低く推定することにより，パケットロスの影響で，取得精度が低くなる．

確率密度関数による推定を行った場合， $k$  が小さい範囲において，取得精度が低くなっている．この手法では，スコアを誤って高く推定したとき， $k$  番目のスコア付近を持つデータを返信時に取りこぼしやすいため，特に， $k$  が非常に小さく返信データサイズが小さい場合に，他の手法よりも取得精度が低下してしまう．一方， $k$  が大きい範囲では，推定誤差よりも，絞り込みによるトラヒック低減の効果が大きいいため，他の手法よりも取得精度が高くなる．

ここで，固定数返信 ( $R = k/50$ ) の場合において，要求データ数  $k = 50$  の場合において，取得精度が低いのは，各端末が上位 50 個に入るデータを複数持つ場合においても，1 つのデータしか返信しないためである．

#### 5.4 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<sup>12)</sup> 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 7 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 7(a) では検索結果の取得精度，図 7(b)

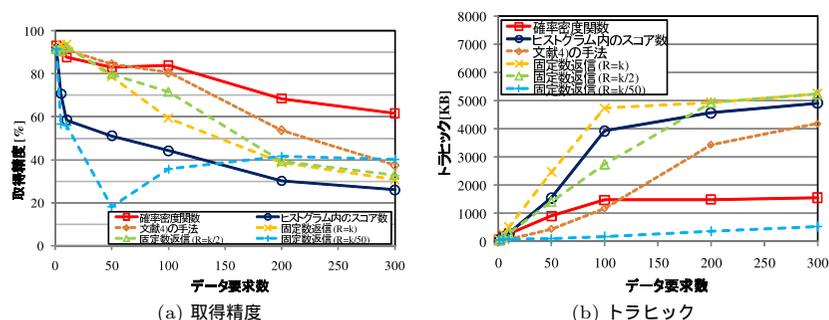


図 7 Zipf 分布における要求データ数  $k$  の影響  
Fig. 7 Effects of  $k$  (Zipf distribution).

ではトラヒックをそれぞれ表す．

この結果から，確率密度関数を用いた場合，トラヒックが小さく抑えられ，取得精度も高いことが分かる．一方，ヒストグラム内のスコア数を用いた場合は，推定の際に  $k$  番目のスコアを高くしたり低すぎたり推定してしまうことが多い．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，トラヒックが大きく，かつ取得精度も低くなっている．文献 4) の手法では，データ要求数  $k$  が大きくなるにつれて，絞り込みが甘くなるため，トラヒックが大きくなり，取得精度も下がる．固定数返信の場合，分布により大きな差異が発生しないため，正規分布とほぼ同じ結果になっている．提案手法において確率密度関数を用いる場合，比較手法よりもトラヒックが小さく，取得精度が非常に高くなっている．一方，ヒストグラム内のスコア数を用いる場合，固定数返信 ( $R = k$ ) の結果とほぼ同様になっており，精度が低いことが分かる．

#### 6. おわりに

本論文では，アドホックネットワークにおいて，トラヒ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，各端末が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ことで，ネットワーク全体のデータのスコア分布と  $k$  番目のスコアを推定する． $k$  番目のスコアを推定するアプローチとして，ヒストグラム内のスコア数を用いる手法と，確率密度関数を用いる手法の 2 つの方法を提案した．各端末は推定した  $k$  番目のスコア以上のスコアを持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．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転送することで，取得精度の低下を抑止する．

シミュレーション実験の結果から，返信データサイズが大きい環境において，提案手法は，文献 4) の手法に比べて，トラヒックを削減しつつ，取得精度の低下を抑止できることを確認した．提案手法において， $k$  番目のスコアを推定する 2 つの手法を比較すると，ヒストグラム内のスコア数による推定を行った場合，返信データサイズが大きいとき，トラヒックが増大し，パケットロスにより取得精度が低下することが分かった．また，スコア分布に大きな偏りがある場合 (Zipf 分布など)，推定の精度が著しく低下することが分かった．返信データサイズが小さいときは，トラヒックの増大の影響が小さいため， $k$  番目のスコアを低めに推定する効果から，取得精度を高く維持できることを確認した．一方，提案手法において，確率密度関数による推定を行った場合，返信データサイズが小さいとき，推定誤差の

影響により、取得精度が低下した。返信データサイズが大きいときは、他の手法と比べて、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、データのスコアが従う分布の種類、およびスコアの上限值、下限値の範囲が既知と想定した。しかし実環境では、これらの想定が適切ではない状況も多くある。特に、ネットワーク内のデータ数やスコアの最大値、最小値は時間的に変動する機会が多いため、各端末があらかじめ把握することは困難である。そこで今後は、各端末がクエリ伝搬時に中継する情報や過去に取得したデータの情報から、これらの値を推定する手法について検討する予定である。また、提案手法では、Top-k 検索をメッセージ往復で実行する方法を採用している。これは、Top-k 検索を検索フェーズとデータ収集フェーズに分けて 2 段階で実行すると、実行遅延が大きくなることを考慮したものである。しかし、1 往復で実行するアプローチと 2 段階で実行するアプローチには、実行遅延とトラフィック、取得精度にトレードオフの関係があるものと予測される。このトレードオフの調査について、今後検討する予定である。

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、および文部科学省科学研究費補助金・基盤研究 S (21220002)、特定領域研究 (18049050) の研究助成によるものである。ここに記して謝意を表す。

### 参 考 文 献

- 1) Akbarinia, R., Pacitti, E. and Valduriez, P.: Reducing network traffic in unstructured P2P systems using top-k queries, *Distributed and Parallel Databases*, Vol.19, No.2-3, pp.67-86 (2006).
- 2) Balke, W.-T., Nejd, W., Siberski, W. and Thaden, U.: Progressive distributed top-k retrieval in peer-to-peer networks, *Proc. Int. Conf. on Data Engineering*, pp.174-185 (Apr. 2005).
- 3) Camp, T., Boleng, J. and Davies, V.: A survey of mobility models for ad hoc network research, *Wireless Communications and Mobile Computing (WCMC)*, Vol.2, No.5, pp.483-502 (2002).
- 4) Hagihara, R., Shinohara, M., Hara, T. and Nishio, S.: 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 (May 2009).
- 5) Johnson, D.B.: Routing in Ad Hoc Networks of Mobile Hosts, *Proc. IEEE WM-CSA '94*, pp.158-163 (1994).
- 6) Kalnis, P., Ng, W.S., Ooi, B.C. and Tan, K.-L.: Answering similarity queries in peer-to-peer networks, *Proc. Int. World Wide Web Conf.*, pp.482-483 (May 2004).

- 7) Kalnis, P., Ng, W.S., Ooi, B.C. and Tan, K.-L.: Answering similarity queries in peer-to-peer networks, *Information Systems*, Vol.31, No.1, pp.57-72 (2006).
- 8) 松波秀和, 寺田 努, 西尾章治郎: P2P 型コンテンツ検索システムのための効率的な Top-k 検索手法, *情報処理学会論文誌*, Vol.47, No.9, pp.2850-2859 (2006).
- 9) Perkins, C.E. and Ooyer, E.M.: Ad-hoc On-Demand Distance Vector Routing, *Proc. IEEE WMCSA '99*, pp.90-100 (1999).
- 10) Scalable Networks: Makers of QualNet and EXata, the only multi-core enabled network simulation and emulation software (online), available from <http://www.scalable-networks.com/>
- 11) 寺田 努, 松波秀和, 西尾章治郎: P2P 型コンテンツ検索システムにおけるコンテンツ分布を考慮した Top-k 検索処理手法, *情報処理学会論文誌: データベース*, Vol.48, No.SIG14 (TOD 35), pp.1-10 (2007).
- 12) Zipf, G.K.: *Human behavior and the principle of least effort*, Addison-Wesley (1949).

(平成 22 年 4 月 7 日受付)

(平成 22 年 10 月 4 日採録)



佐々木 勇和

2009 年大阪大学工学部電子情報エネルギー工学科卒業。現在、同大学大学院情報科学研究科博士前期課程在学中。モバイル環境における検索技術に興味を持つ。日本データベース学会の学生会員。



原 隆浩 (正会員)

1995 年大阪大学工学部情報システム工学科卒業。1997 年同大学院工学研究科博士前期課程修了。同年同大学院工学研究科博士後期課程中退後、同大学院工学研究科情報システム工学専攻助手、2002 年同大学院情報科学研究科マルチメディア工学専攻助手、2004 年より同大学院情報科学研究科マルチメディア工学専攻准教授となり、現在に至る。工学博士。1996 年本学会山下記念研究賞受賞。2000 年電気通信普及財団テレコムシステム技術賞受賞。2003 年本学会研究開発奨励賞受賞。2008 年、2009 年本学会論文賞受賞。データベースシステム、分散処理に興味を持つ。IEEE, ACM, 電子情報通信学会, 日本データベース学会の各会員。



西尾章治郎（正会員）

1975年京都大学工学部数理工学科卒業．1980年同大学院工学研究科博士後期課程修了．工学博士．京都大学工学部助手，大阪大学基礎工学部および情報処理教育センター助教授，大阪大学大学院工学研究科情報システム工学専攻教授を経て，2002年より大阪大学大学院情報科学研究科マルチメディア工学専攻教授となり，現在に至る．2000年より大阪大学サイバーメディアセンター長，2003年より大阪大学大学院情報科学研究科長，その後2007年より大阪大学理事・副学長に就任．この間，カナダ・ウォータールー大学，ビクトリア大学客員．データベース，マルチメディアシステムの研究に従事．現在，Data & Knowledge Engineering等の論文誌編集委員．本会理事を歴任．本会論文賞を受賞．電子情報通信学会フェローを含め，ACM，IEEE等8学会の各会員．

---