アドホックネットワークにおける データ数と値を考慮した Top-k 検索

佐々木　勇和[†1]　原　隆浩[†1]　西尾　章治郎[†1]

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．しかし，この手法では，ネットワーク全体のデータ数，スコアの下限値，上限値が既知であるという，非現実的な環境を想定していた．そこで，本稿では，ネットワーク全体のデータ数，スコアの下限値，上限値は未知である現実的な環境を想定し，従来手法と同様に，検索結果の取得精度の維持しつつ，検索のためのトラヒックをさらに削減する手法を提案する．

A Top-k Query Method Considering the Number of Data Items and Their Scores in Mobile Ad Hoc Networks

Yuya SASAKI,[†1] Takahiro HARA[†1] and Shojiro NISHIO [†1]

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. However, we assumed an unreal environment that each mobile node knows the number of data items in the entire network, and the upper and the lower bound scores. In this paper, we assume a more realistic environment that each mobile node does not know the number of data items in the entire network, and the upper and the lower bound scores, and extend our previous method to adapt to such an environment.

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[3),4)]．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),5),6)]．

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．例えば図 1 において，左端の看護師が血圧の高い 3 人の被災者を検索する場合，各看護師が自身の管理情報から血圧の高い 3 人の被災者の情報を返信すると，必要以上の被災者の情報が返信されてしまう．一方，各看護師が最も血圧の高い被災者の情報のみを返信した場合，血圧が 3 番目に高い被災者 H の情報が返信されない．

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[8)] 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 5)，6) において，アドホックネットワークにおける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

†1 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
Dept. of Multimedia Eng., Graduate School of Information Science and Technology, Osaka University

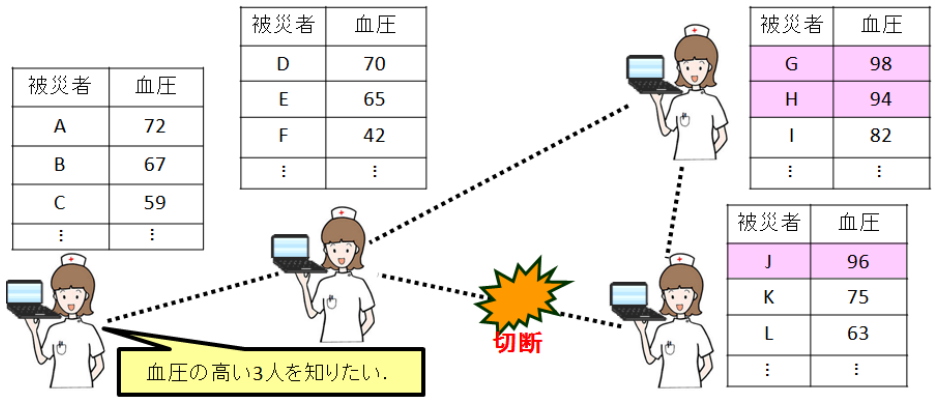


図 1 アドホックネットワークにおける Top-k 検索例
Fig. 1 Example of top-k query in a MANET.

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．

これらの手法（以降では，従来手法と呼ぶ）では，簡単化のために，ネットワーク全体のデータ数，スコアの上限値および下限値が既知であるという，非現実的な環境を想定していた．しかし，実環境では，各端末のもつデータ数はそれぞれ異なり，スコアの最小値，最大値は動的に変化することが一般的である．そこで，本稿では，ネットワーク全体のデータ数，スコアの上限値および下限値は未知であるという現実的な環境を想定して，これまでの提案手法を拡張する．拡張手法では，ネットワーク全体のデータ数を，ヒストグラム内のデータ数とネットワーク全体の端末数を用いて推定する．一方，スコアの上限値および下限値は，ヒストグラムを作成する際のヒストグラムの幅を決定するために必要である．そこで，拡張手法では，クエリが伝搬する経路上の端末が所持するデータのスコアの最小値および最大値を用いてヒストグラムの幅を決定する．そのため，各端末は，自身およびクエリを中継した端末が保持するデータのスコアの最小値および最大値の情報をクエリ伝搬時に添付して送信する．その際，スコアの最小値，最大値が更新されると，その度にヒストグラムを再形成する．

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および再形成する手順について述べる．その後，本稿で提案する 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．また，データは発生，消滅するため，ネットワーク全体のデータ数 n は動的に変化する．簡単化のため，全てのデータのサイズは等しく，各端末は複製を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：$M_1, M_2, ..., M_m$）が存在し，各々が自由に移動する．各端末は，ネットワーク内の端末数 m，ネットワーク内のデータのスコアが従う分布の種類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数えたものであり，スコアの分布状況を表すために用いられる．従来手法では，スコアの下限値，上限値が既知であると想定していため，ヒストグラムの幅を予め静的に決定していた．しかし，本稿では，スコアの下限値，上限値は未知であると想定するため，各端末が所持するスコアの情報に基づいて，その最小値および最大値からヒストグラムの幅を動的に決定する．ここで，データは発生，消滅を繰り返すため，ある時点でネットワーク内に存在する全データのスコアの最大値および最小値が，必ずしもスコアの下限値，上限値と一致するとは限らない．例えスコアの下限値および上限値が既知であったとしても，それらを用いてヒストグラムの幅を決定するとヒストグラムの幅を不必要に大きくしてしまうことが考えられる．そのため，本研究のように実際に存在するスコアの最小値および最大値を用いてヒスト

グラムの幅とした方が，スコア分布をより正確に把握することができるものと考えられる．

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，ヒストグラム内に存在するスコアの最小値，最大値 $[MIN, MAX)$ 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．

2.3 ヒストグラムの再形成

本稿の想定環境では，スコアの最小値，最大値が更新される度にヒストグラムの幅が変化する．そのため，ヒストグラムを再形成する必要がある．ここで，ヒストグラム内には各データの具体的なスコアの情報は含まれていないため，正確にヒストグラムを再形成することは不可能である．そこで，近似的に，j 番目の階級に存在する全てのスコアを $MIN + \frac{(2j-1)(MAX-MIN)}{2C}$ として，更新されたスコアの最小値および最大値を用いてヒストグラムを作成する．つまり，階級 c_j に属する全てのスコアは，階級 c_j の範囲の中央の値がそのデータのスコアであるとして，ヒストグラムを再形成する．

2.4 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，この手法は，経路上の端末がもつデータの最小値および最大値をクエリに添付する点と，ヒストグラムを再形成する点を除いて，従来手法と同様である．

2.4.1 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，スコアの最小値，最大値，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，自身のもつデータのスコアの最小値，最大値と検索クエリに含まれているスコアの最小値，最大値を比較する．最小値，最大値が更新される場合，2.3 節の方法に従って，検索クエリに含まれるクエリヒストグラムを再形成する．さらに手順 (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，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．さらに，スコアの最大値，最小値の情報を添付することで，経路上の端末がもつデータのスコアの最大値，最小値を把握できる．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 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．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．

2.4.2 クエリ応答の返信

2.4.1 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

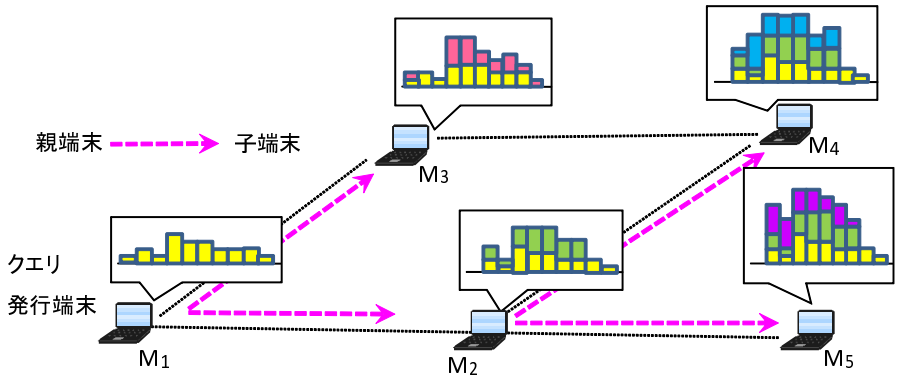


図 2 検索クエリの転送例
Fig. 2 Query message transmission.

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，2.4.1 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) 自身の子がいない端末 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（2.6 節）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，スコアの最小値，最大値，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 と表す．

(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スコアの最小値，最大値を比較する．スコアの最小値，最大値が更新された場合，ヒストグラムを再形成する．さらに，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（2.6 節）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，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．また，中継端末は検索結果に入らないと判断できるデータは返信しない．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，スコアの最小値，最大値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ほど，ヒストグラムに含まれるスコア数が増加し，ネットワーク内のデータのスコアの最小値，最大値，および 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止できるため，取得精度を低下させずにトラヒックを削減できる．

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，矢印はクエリ応答の流れを示す．例えば，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．

2.4.3 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．

2.5 データ数の推定

ヒストグラム内のスコア数 n_{hist}，経路上の端末数 m_{path}，およびネットワーク全体の端末数 m を用いて，ネットワーク全体のデータ数を推定する．具体的には，各端末は，以下の式を用いてデータ数を推定する．

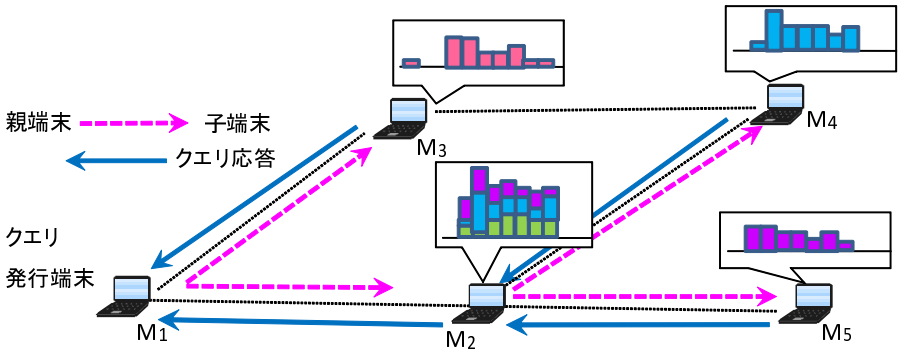


図 3 クエリ応答の返信例
Fig. 3 Reply transmission.

$$n_{est} = m \cdot \frac{n_{hist}}{m_{path}}. \tag{1}$$

つまり，経路上の全端末が同じ数のデータを保持していると仮定して，ネットワーク全体のデータ数を推定する．

例えば，ヒストグラム内のスコア数 $n_{hist} = 400$，経路上の端末 $m_{path} = 4$，ネットワーク全体の端末数 $m = 5$ の場合，ネットワーク全体のデータ数 n は，500 と推定される．

2.6 k 番目のスコアの推定

文献 5)，6) で提案した従来手法では，各端末はクエリ応答を返信する時，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，ヒストグラム内のスコア数を用いて推定する方法（文献 5)），および確率密度関数を用いて推定する方法（文献 6)）を説明する．

2.6.1 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．

$$\sum_{l=a}^{C} \frac{n_{est}}{n_{EH}} \cdot e_l \geq k, \tag{2}$$

$$n_{EH} = \sum_{l=1}^{C} e_l. \tag{3}$$

ここで，n_{est} は推定された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．

$$S_k = MIN + \frac{a(MAX - MIN)}{C}. \tag{4}$$

ここで，MIN と MAX は，ヒストグラムに含まれるスコア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．

2.6.2 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．

$$Q(x) = \int_{x}^{\infty} f(t)dt = \frac{k}{n_{est}}. \tag{5}$$

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．

2.7 k 番目のスコアの補正

提案手法では，各端末は，2.6 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．

この問題を解決するために筆者らは，文献 6) において，セーフティマージンを用いてスコアの補正を行う方法を提案した．この方法では，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．しかし，本研究では，スコアの下限値および上限値が未知であると想定しているため，セーフティマージンが有効に機能しない可能性がある．具体的には，スコアの最小値と最大値の幅が小さい場合，セーフティマージンの適用により，返信データ数が増加し過ぎることが考えられる．一方，最小値と最大値の幅が大きい場合，セーフティマージンを用いても，返信データ数が変化しないといったことが起こりうる．

そこで，本稿では，返信データ数に応じて，セーフティマージンを決定する．具体的には，以下の式を用いて返信データ数の増加率 I を決定する．

$$I = \alpha \cdot (1 - \frac{n_{EH}}{n_{est}}) \tag{6}$$

ここで，α 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．

その後，各端末は，ヒストグラムから $k \cdot (1 + I)$ 番目のスコアを求め，そのスコア以上の所持するデータを返信する．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，返信データ数の増加率 I は大きくなり，推定する $k \cdot (1 + I)$ 番目のスコアは小さくなる．

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[7] を用いた．

3.1 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（$M_1, \ldots, M_{50}$）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] のサイズのデータ存在し，各端末が 30 から 200 個のデータを保持するものとする．具体的には，シミュレーション開始時に各端末は 80 から 120 個のデータを保持しており，シミュレーション開始後，2400 秒までは，各端末のデータ数が 300 秒ごとに 0 から 5 個の範囲でランダムに増加し，2400 秒から 4800 秒まで 300 秒毎に 0 から 5 個の範囲でランダムにデータが消滅するものとした．さらに，4800 秒経過後は，300 秒毎に 0 から 5 個の範囲でランダムにデータが発生するものとした．また，3.4 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，3000 [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，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（2.6.1 項）と確率密度関数により推定する場合（2.6.2 項）を用い，ヒストグラムの階級数 $C = 20$，マージン係数 $\alpha = 1.0$ とした．ここで，比較手法として，従来手法である文献 5) と文献 6) の手法，つまりスコアの上限値（320）と下限値（80）を既知とする手法を用いた．さらに，各端末が自身のもつデータの中からスコアの高い $R = k\ , k/2,\ k/50$ 個のデータを固定数返信する手法も比較手法とした．例えば，$R = k/50$ の場合は，全端末で k 個のデータを返信する．

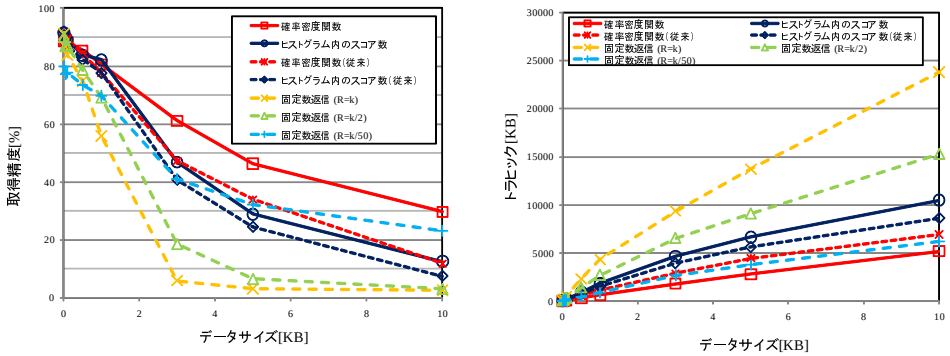
本実験では，要求データ数 k は基本的に 100 とし，3.3 節，3.4 節では 1〜300 の間で変化させた．またデータサイズ d は基本的に 1 [KB] とし，3.2 節では 0.01〜10 [KB] の間で変化させた．以上のシミュレーション環境において，各端末の初期位置をランダムに決定し，7,200[秒] を経過させたときの以下の評価値を調べた．

- 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．
- トラヒック: 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）をトラヒックとする．

3.2 データサイズ d の影響

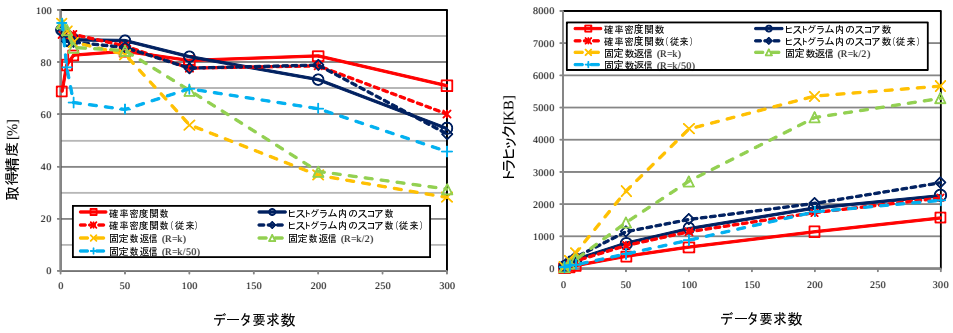
データサイズ d を変化させたときの提案手法の性能を調べた．その結果を図 4 に示す．これらのグラフにおいて，横軸はデータサイズ d を表している．縦軸は，図 4 (a) では検索結果の取得精度，図 4 (b) ではトラヒックをそれぞれ表す．なお，文献 5) および文献 6) の従来手法は,「(従来)」と表記している．

この結果から，提案手法において確率密度関数を用いた場合の取得精度が，他の手法より高いことがわかる．これは，返信データ数が抑えられているため，トラヒックが小さく，パケットロスの発生が少ないためである．一方，提案手法においてヒストグラム内のスコア



(a) 取得精度　　(b) トラヒック

図 4　データサイズ d の影響（正規分布）
Fig. 4　Effects of datasize (normal distribution).



(a) 取得精度　　(b) トラヒック

図 5　要求データ数 k の影響（正規分布）
Fig. 5　Effects of k (normal distribution).

数を用いた場合，d が小さいときの取得精度は高い．しかし，d が大きくなると，固定返信（$R = 2$）より取得精度が低くなる．これは，絞り込みの精度が甘いため，返信データ数が多く，パケットロスの発生が多いためである．提案手法は，従来手法に比べて，トラヒックが小さく，取得精度が高いことがわかる．これは，提案手法では，クエリ発行時点のスコアの最大値および最小値を用いるため，ヒストグラムの幅を従来手法よりも正確に推定できているためである．固定数返信（$R = k, k/2$）は，返信データ数が多いために，d が大きくなると，著しく取得精度が低くなる．

3.3　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 5 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 5 (a) では検索結果の取得精度，図 5 (b) ではトラヒックをそれぞれ表す．

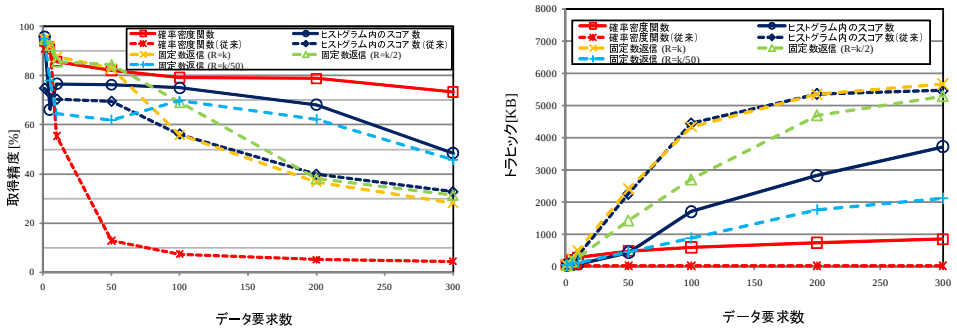
この結果から，提案手法において確率密度関数を用いた場合，k が非常に小さいときに取得精度が低い．これは，推定の誤差のために，上位 k 個に入るデータを取りこぼしていることを表している．k が大きくなると，推定の誤差の影響が少なくなるため，取得精度は高くなる．一方，提案手法において，ヒストグラム内のスコア数を用いた場合は，k が小さい場合の取得精度は高い．k が大きくなると，トラヒックが大きくなり，パケットロスが多く発生してしまうため，取得精度が低くなる．従来手法において確率密度を用いた場合，k が非常に小さい場合でも取得精度が高い．これは，k 番目のスコアを実際より高く推定してしまった場合でも，従来手法のセーフティマージンによって上位 k 個に入るデータを取得できているためである．一方，本稿の提案手法におけるセーフティマージンでは，k が小さい場合，有効に機能していないことがわかる．これは，k が小さいと，返信データ数の増加率 I が大きくなっても，推定される $k(1 + I)$ 番目のスコアがほとんど変わらないためである．k が大きい場合は，返信データ数が多くなり過ぎないため，従来手法より取得精度が高いことがわかる．

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 $[0, 8000)$ とした．

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 を表す．縦軸は，図 6 (a) では検索結果の取得精度，図 6 (b) ではトラヒックをそれぞれ表す．

この結果から，提案手法において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わかる．一方，ヒストグラム内のスコア数を用いた場合は，Zipf 分布の場合，スコアに大きな偏りがあるため，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が多い．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．特に，$k = 5$ の場合において精度が非常に低くなっている．これは，スコアが非常に高いデータと上位 k 個に入るスコアをもつデータがヒストグラム内にある場合，ス



(a) 取得精度　　(b) トラヒック

図 6　要求データ数 k の影響（Zipf 分布）
Fig. 6　Effects of k (Zipf distribution).

コアが非常に高いデータの入っている階級のスコアの下限値を k 番目のスコアと推定してしまい，他の階級のデータが返信されないためである．Zipf 分布において，従来手法の精度がよくないのは，スコアの幅が非常に広く，推定の誤差が大きくなるためである．特に，従来手法において確率密度関数を用いた場合，取得精度が非常に低い．これは，k 番目のスコアを高く推定しており，上位 k 個に入るデータが返信されていないためである．

4. 結　　論

本稿では，ネットワーク全体のデータ数，スコアの下限値および上限値が未知である現実的な環境を想定し，検索結果の取得精度の維持しつつ，検索のためのトラヒックを削減する Top-k 検索手法を提案した．提案手法では，ネットワーク全体のデータ数を推定し，さらにクエリが転送される経路上の端末がもつデータのスコアの最小値，最大値を用いて，スコア分布のヒストグラムを作成し，k 番目のスコアを推定する．

シミュレーション実験の結果から，提案手法がトラヒックの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．ただし，k が小さいときに提案手法のセーフティマージンが有効に機能していないことが明らかになったため，今後，改善の余地がある．

本研究の提案手法では，ネットワーク分断が発生した場合，分断したネットワーク内の端末がもつデータにアクセスできないため，Top-k 検索の結果の取得精度が 100% とはならない．今後は，データの複製を配置し，ネットワーク分断が発生した場合でも，Top-k 検索の結果の取得精度を維持することを検討している．

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) 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) 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) Johnson.D.B, “Routing in Ad Hoc Networks of Mobile Hosts,”Proc. IEEE WMCSA’94, pp.158–163, 1994.
4) Perkins. C. E, and Ooyer. E .M, “Ad-hoc On-Demand Distance Vector Routing,”Proc. IEEE WMCSA’99, pp.90–100, 1999.
5) 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.
6) 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と確率密度関数を用いた Top-k 検索手法,” 第 2 回データ工学と情報マネジメントに関するフォーラム (DEIM2010), Feb. 2010.
7) Scalable Networks: makers of QualNet and EXata, the only multi-core enabled network simulation and emulation software.: http://www.scalable-networks.com/.
8) G.K.Zipf, “Human behavior and the principle of least effort,” *Addison-Wesley*, 1949.