

推薦論文

アドホックネットワークにおける Top-k 検索のためのルーティング手法

天方 大地^{1,a)} 佐々木 勇和^{1,b)} 原 隆浩^{1,c)} 西尾 章治郎^{1,d)}

受付日 2012年12月17日, 採録日 2013年5月18日

概要：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、Top-k 検索を用いることが有効である。既存研究として、検索クエリをフラッディングし、ネットワーク内のデータの k 番目のスコアを推定することにより、検索結果に含まれないデータの返信を抑制する手法が提案されている。しかし、これらの手法はネットワーク内のすべての端末が検索クエリ、およびクエリ応答を送信するため、検索結果の取得に必要な端末によるメッセージ転送から、不要なトラフィックやメッセージ衝突が発生してしまう。本論文では、この問題を解決するため、アドホックネットワークにおける Top-k 検索のためのルーティング手法を提案する。提案手法では、各端末がスコアが大きいデータを取得するための検索クエリの送信先端末（宛先）を管理する経路表を作成する。また、検索を要求するデータを指定し、そのデータに対応する宛先に検索クエリをユニキャストすることで、検索結果の取得に必要な端末へのメッセージ転送を抑制する。さらに、ネットワークトポロジ変化にも着目し、トポロジ変化が激しい環境においても取得精度の低下を抑制することを目的として、提案手法の拡張手法も提案する。シミュレーション実験の結果から、提案手法は、小さいトラフィックで既存手法以上の取得精度を維持していることを確認した。

キーワード：Top-k 検索, ルーティング, モバイルアドホックネットワーク

A Top-k Query Routing Method in Mobile Ad Hoc Networks

DAICHI AMAGATA^{1,a)} YUYA SASAKI^{1,b)} TAKAHIRO HARA^{1,c)} SHOJIRO NISHIO^{1,d)}

Received: December 17, 2012, Accepted: May 18, 2013

Abstract: In mobile ad hoc networks, using a top-k query is effective to acquire only necessary data items. In a previous work, the methods which the query issuer floods a query message over the network and nodes that received the message prune data items which are not included in the top-k result by estimating k th score have been proposed. However, in these methods, unnecessary nodes to acquire the top-k result also transmit query and reply messages, which provokes unnecessary traffic and message collisions. In this paper, we propose a top-k query routing method in mobile ad hoc networks. Our proposed method utilizes a routing table which maintains node identifiers (*query addresses*) to forward a query message to acquire data items with high scores. Each node specifies ranks of data items to retrieve and sends a query message to corresponding query addresses. As a result, the proposed method can suppress unnecessary traffic while keeping high accuracy of query result. We also propose an extension of the proposed method, which aims to prevent the decrease of accuracy of query result in highly dynamic network. The simulation results show that the proposed methods achieve high accuracy of query result with small traffic compared with the existing method.

Keywords: top-k queries, routing, mobile ad hoc networks

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
Department of Multimedia Engineering Graduate School of
Information Science and Technology Osaka University, Suita,
Osaka 565-0871, Japan

a) amagata.daichi@ist.osaka-u.ac.jp

b) sasaki.yuya@ist.osaka-u.ac.jp

c) hara@ist.osaka-u.ac.jp

d) nishio@ist.osaka-u.ac.jp

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 [6], [11] 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。たとえば、緊急災害などが起こった場合、基地局などの損壊により、インターネットなどの通信基盤が使用できなくなる可能性がある。その際、図 1 のように、救急隊員の持つ端末でアドホックネットワークを形成し、情報共有することが考えられる。このとき、被害の最も大きいいくつかの被災所のデータを検索する場合などにおいて、Top- k 検索を行うことが有用である。

既存研究として、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 検索手法が提案されている [4], [12], [13]。これらの手法では、フラッディングによるデータ検索を行い、ネットワーク内のデータの k 番目のスコアを推定することにより、検索結果に含まれないデータの返信を可能な限り抑制している。しかし、これらの手法では検索結果の取得に必要な端末による検索クエリおよびクエリ応答の送信も多く行われてしまうという問題がある。この問題により、不要なトラヒックの増加や、メッセージ衝突などが生じ、検索結果の精度が低下することが考えられる。

そこで本論文では、この問題を解決するために、アドホックネットワークにおける Top- k 検索のための効率的なルーティング手法を提案する。提案手法では、経路表により、ネットワーク内のデータの順位、および検索クエリの送信先端末（宛先）を把握する。各端末は、要求するデータの順位を指定した検索クエリを、そのデータに対応する

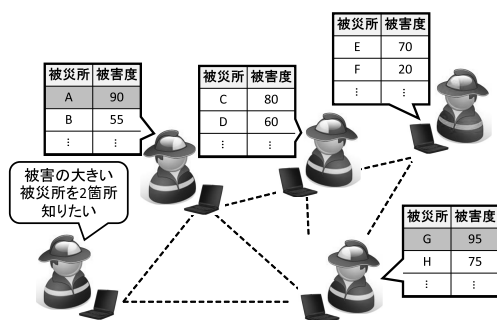


図 1 アドホックネットワークにおける Top- k 検索例

Fig. 1 Example of top- k query processing in a MANET.

宛先にユニキャストすることにより、検索結果の取得に必要な端末へのメッセージ転送を抑止する。経路表における宛先、またはクエリ応答の送信先とのリンク切断を検出した場合、リンク切断先への別経路を探索することにより、取得精度の低下を抑制する。

ここで、単一経路で検索クエリを転送する提案手法では、ネットワークポロジが頻繁に変化する場合、検索クエリやクエリ応答の転送率が低下することが考えられる。そのため、ネットワークポロジが頻繁に変化する環境においても取得精度の低下を抑制することを目的として、提案手法の拡張手法も提案する。拡張手法は、提案手法と検索クエリの転送方法、および経路探索方法が異なり、複数経路での検索クエリ転送を可能にする。シミュレーション実験の結果から、提案手法、およびその拡張手法は、既存手法よりもトラヒックを削減しつつ、検索結果の取得精度を維持していることを確認した。

以下では、2 章で関連研究について述べ、3 章で想定環境について説明する。また、4 章で提案手法について説明し、ネットワークポロジが頻繁に変化する環境における問題点について議論する。5 章では、4 章であげた問題点を解決する拡張手法について説明する。6 章でシミュレーション実験の結果を示し、最後に 7 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、データセンタリック型ルーティング、および、Top- k 検索に関する従来研究について紹介する。データセンタリック型ルーティング [3], [5], [7] は、ユーザが指定した属性のデータを検索するルーティングプロトコルであり、指定した条件での上位 k 個のデータを検索する Top- k 検索とアプローチが近いため、ここで紹介する。

文献 [5] では、センサネットワークにおいて、*directed diffusion* と呼ばれるデータセンタリック型ルーティング手法を提案している。この手法は、シンクと呼ばれる基地局端末が要求するデータの属性をネットワーク全体に伝搬させる。要求を満たすデータを保持している端末は、フラッディングを用いて基地局までデータを返信する。この手続きの間に、各センサ端末は、頻繁にデータが転送される経路や応答時間の短い経路の優先度を高くし、これらの経路をメッセージの転送用経路に設定しやすくする。これにより、データの転送を少ないトラヒックで実現できる。しかし、ネットワークポロジが動的に変化するアドホックネットワークでは、この手法は有効でない。

文献 [10] では、センサネットワークにおける Top- k 検

本論文の内容は 2012 年 7 月のマルチメディア、分散、協調とモバイル (DICOMO2012) シンポジウム 2012 にて報告され、マルチメディア通信と分散処理研究会主査により情報処理学会論文誌ジャーナルへの掲載が推薦された論文である。

索のためのルーティング手法を提案している．この手法では，端末の残余電力をもとにクラスタリングを行い，動的にクラスタヘッドを交換することにより，ネットワークの長寿命化を図っている．この手法は，メッセージ転送の効率化でなく，ネットワークの長寿命化を考慮している点で本研究と異なる．また，端末が移動するアドホックネットワークでは，クラスタを維持するコストが大きいため，この手法は有効でない．

Top-k 検索は，非構造 P2P ネットワーク [1], [9] やセンサネットワーク [8], [10], [14], [15] などの様々な分野においてさかんに研究が行われている．アドホックネットワーク，無線センサネットワーク，および P2P ネットワークは，各データが各端末に分散して保持されている点やマルチホップによる通信が必要な点が類似している．一方，有線の P2P ネットワークとは異なり，アドホックネットワークおよび無線センサネットワークでは，無線通信のためパケットロスを考慮する必要がある．さらに，アドホックネットワークでは，無線センサネットワークと異なり，端末が自由に移動する点や基地局端末が存在しない点も考慮しなければならない．このような特徴から，アドホックネットワークでは Top-k 検索の結果の精度を保証することが難しいため，この問題を解決するための研究がこれまでにほとんど行われていない．ここでは，アドホックネットワークにおける Top-k 検索を対象とする先行研究 [4], [12], [13] のうち，Top-k 検索の取得精度を最も高く維持できる 2 フェーズ検索手法 [13] を紹介する．

2 フェーズ検索手法は検索フェーズと収集フェーズに分けて Top-k 検索を行う．まず，検索フェーズにおいて，検索クエリ発行端末が検索クエリをフラッディングし，ネットワーク内のデータのスコアを取得し，取得したスコアの k 番目の値を閾値に設定する．その後，収集フェーズにおいて，設定した閾値を添付した検索クエリをフラッディングし，これを受信した各端末は閾値以上のスコアを持つデータのみ返信する．この手法では，閾値を設定することにより，検索結果に含まれないデータの返信を抑制することができる．しかし，すべての端末が 2 度，検索クエリ，およびクエリ応答を送信するため，不要なトラフィックが発生する．提案手法では，経路表を用いて，検索結果に含まれるデータの取得に必要な端末のみに検索クエリを送信する．これにより，トラフィックを削減しつつ，高い取得精度を維持できる．

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．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，各々がネットワーク内を自由に移動する．また，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．単純化のため，すべ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．Top-k 検索を行う端末は，検索条件，および要求データ数 k 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ことを目的とする．各端末は指定される最大の k の値 (k_{max}) を把握しているものとする．

4. 提案手法

本章では，経路表を用いて，検索結果の取得に必要な端末へのメッセージ転送を抑止する提案手法について説明する．まず，概要について述べ，その後，Top-k 検索，および経路表のメンテナンス方法について説明する．

4.1 概要

提案手法では，各端末は経路表を用いて Top-k 検索を行う．そのため，Top-k 検索を行う前に経路表を作成する必要がある．経路表は，データのスコア，およびデータの保持端末識別子を収集することにより作成する．経路表は，データのスコアの順位（順位表），順位表中のデータの保持端末の識別子（保持端末リスト），および順位表中のデータを取得するための検索クエリの送信先（宛先）により構成されている．検索クエリが発行端末は，各宛先に検索要求順位を添付した検索クエリを送信する．検索クエリを受信した各端末も自身の経路表に基づいて，受信した検索クエリを転送する．要求された順位のデータを保持している端末は，そのデータのみを返信する．また，Top-k 検索中にリンク切断を検出した場合，経路探索を行うことにより，経路表を修正する．さらに，データのスコアが更新された場合，順位の変更があるデータを持つ端末にスコア更新を通知する．

4.2 経路表の作成

本節では，経路表の作成方法について説明する．検索クエリ発行端末 M_p が，検索条件に一致するクエリに対する経路表を保持していない場合，経路表の作成を開始する．

M_p は経路表作成メッセージをフラッディングし，このメッセージを初めて受信した端末は，送信元端末を自身の親端末として記録する．また，スコアの高い R 個のデータのスコア，およびデータの保持端末識別子を親端末に送信する．ここで，スコア更新に対応するため， $R > k_{max}$ とする．さらに，返信メッセージを受信した端末は，スコアの高い R 個のデータのスコア，その保持端末識別子，および受信メッセージの送信元端末を経路表作成リストとして記録し，どのデータがどの端末から返信されてきたか把握する．この手順により， M_p は経路表を作成することがで

表 1 各端末の保持するデータのスコア

Table 1 Scores of data items held by each mobile node.

端末識別子	データのスコア
M_1	89, 88, 82, 78, 70, 66, 49, 30, 21
M_2	99, 81, 80, 77, 72, 71, 69, 55, 42
M_3	90, 79, 53, 41, 33, 29, 19, 12, 11
M_4	95, 87, 76, 61, 58, 50, 44, 37, 10
M_5	85, 83, 74, 67, 56, 23, 18, 15, 13
M_6	68, 62, 60, 59, 56, 40, 30, 25, 22

表 2 M_1 の経路表Table 2 M_1 's routing table.

順位	データのスコア	保持端末識別子	宛先
1	99	M_2	M_2
2	95	M_4	M_2
3	90	M_3	M_3
$\vdots$	$\vdots$	$\vdots$	$\vdots$
20	69	M_2	M_2

順位表 保持端末リスト

きる．経路表はスコアの降順によりソートされている．

M_p は順位表，および保持端末リストを添付したメッセージをフラッディングし，このメッセージを初めて受信した端末は，以下の手順で自身の経路表を作成する．

- (1) 受信した順位表，保持端末リストを経路表に格納する．
- (2) すべての宛先をメッセージの送信元端末とする．
- (3) 自身が保持端末であるデータの宛先を自身に更新する．
- (4) 経路表作成リスト中に含まれるデータと一致するデータの宛先を，経路表作成リストで記録した送信元端末に更新する．

さらに， k_{max} 位までのデータ保持端末は自身の識別子を添付したメッセージをフラッディングし，各データまでの最短経路を構築する．これにより，Top-k 検索中の検索クエリ，およびクエリ応答の送信の転送にかかるトラフィックを最小に抑えることができる．

以上の手続きにより，各端末は各データまでの最短経路による宛先を把握できる．表 1，および表 2 は各端末が持つデータのスコアと $R = 20$ の場合の端末 M_1 （ネットワークポロジは図 2 で示す）の保持する経路表を示す．この場合， M_1 は 1 位のデータを検索する際の検索クエリの宛先は M_2 であることを表している．

4.3 Top-k 検索

提案手法では，検索クエリ発行端末，および検索クエリを受信した各端末は，検索要求順位を指定した検索クエリを自身の経路表における各宛先にユニキャストする．また，各端末は，指定されたデータのみを返信する．

4.3.1 検索クエリの転送

検索クエリに含まれる要素は以下のとおりである．

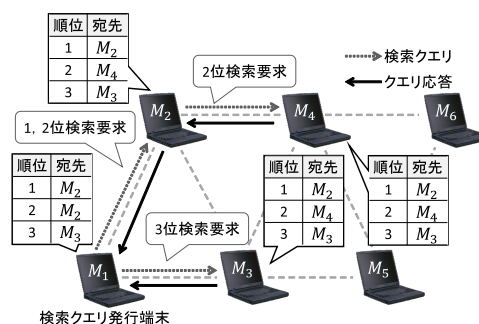


図 2 提案手法における Top-k 検索例

Fig. 2 Example of top-k query processing in the proposed method.

- *Query issuer's ID*: 検索クエリの発行端末の識別子
- *Query ID*: 検索クエリの識別子
- *Sender node's ID*: 検索クエリ送信端末の識別子
- *Search rank list*: 検索要求順位のリスト

検索クエリ発行端末は，経路表における k 位までの各宛先に，それぞれ検索要求順位を指定した検索クエリを送信する．また， k 位までに同じ宛先が複数存在する場合，search rank list に検索要求順位をまとめることで，不要に検索クエリの送信回数を増やさない．検索クエリを初めて受信した端末は，その送信元端末を親端末として記録し，検索クエリ発行端末と同様の操作により検索クエリを転送する．検索要求順位として i 位を指定した検索クエリを受信した端末は，自身が i 位のデータを持っていない場合，自身の経路表における i 位の宛先に検索クエリを転送する．これにより，検索結果の取得に必要な端末のみに検索クエリを転送することができる．

4.3.2 クエリ応答の転送

提案手法では，経路表における k 位のスコア以上のスコアを持つデータを返信する．これにより，検索結果に含まれないデータの返信による不要なトラフィックを削減できる．クエリ応答に含まれる要素は以下のとおりである．

- *Query ID*: 検索クエリの識別子
- *Sender node's ID*: クエリ応答送信端末の識別子
- *Data list*: データとそのスコアのリスト

検索クエリを転送する必要がない端末，つまり，検索を要求されたデータをすべて持っている場合，親端末にクエリ応答を送信する．クエリ応答中の Data list には，要求されたデータのみ添付する．これにより，検索結果の取得に必要なのない端末への検索クエリの転送を抑制し，不要なデータの返信による無駄なトラフィックが発生しない．検索クエリの中継端末は，自身が要求した順位のデータをすべて受信するか，検索クエリを送信してから一定時間経過した場合，クエリ応答を親端末に送信する．さらに，クエリ応答を受信した端末は，自身の経路表とクエリ応答の送信元端末を参照する．このとき，データの宛先とクエリ応答の送信元端末が異なる場合，そのデータの宛先をクエリ応

答の送信元端末に更新する。

図 2 を用いて、端末 M_1 が $k = 3$ として Top-k 検索を行う例を説明する。検索クエリを発行する M_1 は自身の経路表を参照し、要求するデータに対する宛先（1 位および 2 位： M_2 , 3 位： M_3 ）に検索クエリをユニキャストする。検索クエリを受信した端末も M_1 と同様に経路表を参照し、検索クエリを送信する。また、要求されたデータを検索クエリの送信元端末に送信する。これにより、 M_1 はネットワーク内の上位 3 個のデータを取得できる。

4.4 経路表のメンテナンス

提案手法では、リンク切断やデータのスコアの更新が起きた場合においても検索結果の取得精度を維持するため、経路表を修正する必要がある。そこで本節では、経路表の修正方法について説明する。

4.4.1 リンク切断への対応

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。リンク切断が生じ、検索クエリを経路表における宛先に送信できない場合、およびクエリ応答を親端末に送信できない場合、Top-k 検索の取得精度が低下する。そのため、リンク切断を検出した端末は、経路探索メッセージを送信することにより、宛先、または親端末までの別経路を探索する。

経路探索メッセージの処理は、検索クエリ転送時、およびクエリ応答返信時において同様である。リンク切断を検出した端末は TTL を設定した経路探索メッセージをブロードキャストする。ここで、TTL はホップ数により転送範囲を指定する値である。経路探索メッセージを受信した端末は、自身がリンク切断先の端末、または検索クエリの発行端末（クエリ応答返信時のみ）の場合、経路探索応答メッセージを返信する。それ以外の端末の場合、TTL を 1 減らし、その値が 1 以上であるならば経路探索メッセージをブロードキャストする。経路探索応答メッセージを受信した端末は、経路表におけるリンク切断先に該当する宛先を経路探索応答メッセージの送信元端末に更新する。

この処理により、リンク切断が起きた場合においても別経路を探索し、別経路の発見に成功した場合、検索クエリ、およびクエリ応答の転送を継続でき、取得精度を維持できる。

4.4.2 スコア更新への対応

データのスコアが更新された場合、データの順位が変わる可能性がある。提案手法では、データのスコアを更新した端末は、自身の経路表を修正し、データの順位が変わったデータを持つ端末にスコアの更新を通知する。データ D_i のスコアが更新された際、経路表を修正しなければならない場合が 3 通り（データの順位が変化する、新たに経路表に挿入する、経路表から削除する）ある。各端末は、Algorithm 1 に従って経路表を修正する。

Algorithm 1 経路表修正アルゴリズム

```

1: if  $D_i$  moves to different rank in the routing table then
2:   Resort the recorded information in descending order by score
3: else if  $D_i$  is newly included in the routing table then
4:   Insert the information on  $D_i$ 
5:   Move down the rank of each data item whose score is smaller  $D_i$ 
6: else if  $D_i$  is excluded from the routing table then
7:   Delete the information on  $D_i$ 
8:   Raise the rank of each data item whose score is smaller than  $D_i$ 
9: end if

```

データのスコアを更新した端末は、順位が変わったデータの宛先にスコア更新メッセージを送信し、このメッセージを受信した端末は、経路表を修正する。また、順位が変わったデータの宛先は、メッセージの送信元端末に更新する。その後、順位が変わったデータを持つ端末まで、データのスコアを更新した端末と同様の方法でスコア更新メッセージを転送する。

これにより、経路表の修正がすぐに必要な端末（順位が変わったデータを持つ端末）にのみスコア更新を通知できる。また、スコア更新メッセージを受信しなかった端末は、Top-k 検索中のクエリ応答の受信の際に経路表を修正できる。そのため、データのスコア更新通知にかかるトラフィックを削減しつつ、経路表の精度、および取得精度を維持できる。

4.5 提案手法の問題点

提案手法は、検索結果に含まれるデータの取得に必要な端末にのみ検索クエリを送信することにより、トラフィック削減を実現している。しかし、ネットワークトポロジの変化が激しい環境では、以下の要因により、取得精度が低下してしまうことが考えられる。

● 単一経路による検索クエリの転送

提案手法では、検索要求順位を添付した検索クエリを各宛先に送信する。そのため、各データまでの検索クエリの転送経路は 1 本となる。しかし、アドホックネットワークでは端末が自由に移動し、ネットワークトポロジが動的に変化するため、単一経路による検索クエリの転送は信頼性に欠ける。

● 非効率な経路探索

提案手法では、検索クエリ転送時にリンク切断が起きた場合、経路探索を行うことにより別経路を発見している。この手続きはメッセージの往復が必要であり、検索クエリを再転送するまで遅延が生じる。この遅延により、新たなリンク切断が生じる可能性があり、検索クエリの転送率を低下させる。

以上の要因から、ネットワークトポロジが頻繁に変わる場合、提案手法では経路表に最新のネットワークトポロジを反映できず、検索結果の取得精度が低下することが考

られる。そのため、よりネットワークポロジの変化に対応できるルーティング手法が必要である。

5. 拡張手法

本章では、本論文における提案手法の検索方法を拡張した手法について説明する。初めに拡張のアイデアについて述べ、その後、Top-k 検索、および経路表のメンテナンス方法について説明する。

5.1 拡張のアイデア

拡張手法は提案手法と同様の経路表を用いるが、複数経路による転送を行うことで、検索クエリの転送率の向上を目指す。具体的には、各端末が経路表における k 位までのすべての宛先に検索クエリを送信することにより、複数経路による検索クエリの転送を実現する。しかし、宛先の数が多い場合、ユニキャストによる転送では送信トラヒック、および遅延が増加してしまう。そのため、拡張手法では経路表における k 位までの宛先に、マルチキャストにより検索クエリを送信する。さらに、受信、および傍受した検索クエリから、転送が不要な宛先を決定し、不要な検索クエリの転送を抑制しつつ、複数経路による検索クエリの転送を実現する。

ここで、4.2 節で説明した最短経路の構築は、単一経路による検索クエリの転送を行う提案手法では有効である。しかし、最短経路となる宛先は端末によって異なるため、拡張手法では不要な転送経路を作成してしまう。そのため、拡張手法では最短経路の構築は行わない。

5.2 Top-k 検索

拡張手法では、経路表、および受信した検索クエリに添付されている情報から、自身の送信する検索クエリの転送先を決定する。また、検索クエリを受信した端末は、経路表における k 位のスコア以上のスコアを持つデータのみを返信する。本節では、Top-k 検索の手順について詳しく説明する。

5.2.1 検索クエリの転送

検索クエリに含まれる要素は以下のとおりである。

- *Query issuer's ID* : 検索クエリの発行端末の識別子
- *Query ID* : 検索クエリの識別子
- *Number of requested data items* : 要求データ数, k
- *Query condition* : 検索条件
- *QueryPath* : 検索クエリの転送経路リスト (検索クエリ発行端末から検索クエリ送信元端末までの端末識別子のリスト)
- *Query address list* (QL_s) : 検索クエリの送信元端末が送信した宛先端末の識別子のリスト
- *Parent's query address list* (QL_p) : 検索クエリの送信元端末の親が送信した宛先端末の識別子のリスト

Algorithm 2 検索クエリ転送アルゴリズム

```

1: /* 検索クエリの受信 */
2: if  $M_q$  receives the query for the first time then
3:   parent  $\leftarrow$  sender node
4:   for  $i = 0$  to  $k$  do
5:      $M_q$ 's  $QL_s \leftarrow$  RoutingTable.QueryAddress[i]
6:   end for
7:    $M_q$ 's  $QL_s \leftarrow M_q$ 's  $QL_s - \text{Query}.QL_s - \text{Query}.QL_p - \text{Query}.RouteList$ 
8:   if  $M_q$ 's  $QL_s = \emptyset$  then
9:     Send reply message to the parent
10:  else
11:    Set query timer
12:  end if
13: else
14:    $NL_{query} \leftarrow NL_{query} + \text{sender node}$ 
15:    $M_q$ 's  $QL_s \leftarrow M_q$ 's  $QL_s - \text{sender node}$ 
16: end if
17: /* 検索クエリの送信 */
18: if query timer expired then
19:   if  $M_q$ 's  $QL_s \neq \emptyset$  then
20:      $M_q$ 's  $QL_p \leftarrow \text{received } QL_s$ 
21:     Send query message
22:   else
23:     Send reply message to the parent
24:   end if
25: end if

```

検索クエリの発行端末 M_p は、検索条件を指定し、それに一致するクエリに対する経路表を保持しているか確認する。保持していない場合、4.2 節の処理を行う。経路表をすでに保持している場合、または経路表を作成後、 M_p は要求データ数を指定し、経路表における k 位までのデータの宛先を QL_s に格納する。その後、検索クエリを送信する。このメッセージにおいて、 QL_p は $\emptyset$ であり、検索クエリの発行端末識別子は M_p である。また、*QueryPath* には M_p が格納されている。

検索クエリを受信した端末 M_q は、Algorithm 2 に従って検索クエリの転送を行う。ここで、Algorithm 2 における RoutingTable.QueryAddress[i] は、 M_q の経路表の i 位の宛先を示す。また、 NL_{query} は検索クエリの送信元端末、およびその端末の検索クエリ発行端末からのホップ数のリストである。初めて検索クエリを受信した端末は、検索クエリの送信元端末に ACK を送信する。ACK を受信した端末は、その送信元端末を子端末に追加する。

検索クエリを受信した端末は、上位 k 個のすべてのデータを検索しようとする (Algorithm 2 第 4~6 行)。これにより、検索クエリの転送経路を複数にできる。さらに、各端末は受信した検索クエリに含まれる QL_s , QL_p , *QueryPath* から、すでに検索クエリが転送されている端末を把握できる (Algorithm 2 第 7 行) ため、これらを宛先から除外できる。これにより、不要な転送を抑制しつつ、提案手法に比べて検索クエリの転送率を向上させることができる。

5.2.2 クエリ応答の返信

拡張手法におけるクエリ応答の返信は、基本的に提案手法と同様である。自身の検索クエリにおける QL_s が $\emptyset$ である端末 M_r は、クエリ応答の返信を開始する。クエリ応答の返信は Algorithm 3 に従って行う。

Algorithm 3 クエリ応答返信アルゴリズム

```

1: /* クエリ応答の受信 */
2: Compare query address (corresponding to the received data
   items) to the sender node
3: if query address  $\neq$  sender node then
4:   if query address =  $M_r$ 's child then
5:      $M_r$ 's children  $\leftarrow M_r$ 's children - the query address
6:   end if
7:   query address  $\leftarrow$  sender node
8: end if
9: if  $M_r$  receives a data item which is not recorded in  $M_r$ 's
   routing table then
10:  Update its own routing table (as described in Section
     4.4.2)
11: end if
12: /* クエリ応答の送信 */
13: if  $M_r$  receives a reply message from all children || pre-
   determined time limit has elapsed then
14:  Add all received data items and  $M_r$ 's data items whose
     scores are within the  $k$ th rank to  $M_r$ 's reply message
15:  Send reply message to its parent
16: end if
17: if  $M_r$  detects a link disconnection between itself and its par-
   ent then
18:  if  $M_r$ 's  $NL_{query} \neq \emptyset$  then
19:    parent  $\leftarrow$  neighbor node with the minimum hop count
       to the query-issuing node from  $NL_{query}$ 
20:     $NL_{query} \leftarrow NL_{query} - \text{parent}$ 
21:    Send reply message to its parent
22:  else
23:    Discards reply message
24:  end if
25: end if

```

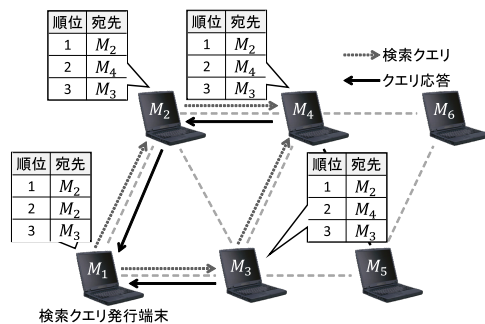


図 3 拡張手法における Top-k 検索例

Fig. 3 Example of a behavior of top-k query processing in the extended method.

各端末は、クエリ応答の受信により経路表を修正 (Algorithm 3 第 2~11 行) し、経路表の精度を維持する。また、親端末とのリンク切断を検出した場合、 NL_{query} から検索クエリ発行端末までのホップ数が最小の隣接端末を選択し、その端末にクエリ応答を送信することにより、取得精度の低下を抑制する (Algorithm 3 第 2~25 行)。提案手法では、クエリ応答返信時にリンク切断を検出した場合、経路探索を行い、親端末までの別経路を発見する必要がある。しかし、複数経路による検索クエリの転送を行う拡張手法では、隣接端末を把握している場合、経路探索を行わず、隣接端末にクエリ応答を再送する。そのため、経路探索にかかるトラヒック、および遅延を抑制でき、ネットワークトポロジの変化に対応しやすい。

図 3 を用いて端末 M_1 が $k = 3$ の Top-k 検索を行う例を説明する。検索クエリを発行する端末 M_1 は、自身の経

路表を参照し、 k 位までの宛先となる端末 (M_2 , および M_3) に検索クエリを送信する。この検索クエリを受信した M_2 , M_3 も同様に、自身の送信する検索クエリの QL_s を経路表から抽出する。この場合、 M_2 の送信する検索クエリの QL_s は M_3 と M_4 , および M_3 の送信する検索クエリの QL_s は M_2 と M_4 である。このとき、自身が受信した検索クエリから M_2 は M_3 を、 M_3 は M_2 を QL_s から除外できる。 M_2 および M_3 は、 M_4 に検索クエリを送信する。 M_4 は、 M_2 および M_3 から検索クエリを受信する。ここで、 M_2 からの検索クエリを先に受信した場合、 M_4 は M_2 を親とし、 M_3 を隣接端末として NL_{query} に記録する。 QL_s が $\emptyset$ の M_4 , および子端末のいない M_3 は、クエリ応答を親に送信する。これにより、 M_1 は上位 3 個のデータを取得できる。この例から分かるように、提案手法では、不要にマルチキャストメンバを増やすことなく検索クエリを送信しており、さらに M_4 までの複数経路を実現している。その結果、不要なトラヒックを抑えつつ、上位 k 個のデータを持つ端末までの検索クエリの転送率を向上させることができる。

5.3 経路表のメンテナンス

本節では、拡張手法における経路表の修正について説明する。

5.3.1 リンク切断への対応

検索クエリ転送時に経路表における宛先とのリンク切断を検出した場合、経路探索メッセージ ($RREQ$) を送信することにより、別経路を発見する。拡張手法では、 $RREQ$ に自身の送信する検索クエリを添付する。これにより、経路探索応答メッセージ ($RREP$) 受信後に、検索クエリを再転送する必要がない。その結果、経路探索の遅延を可能な限り抑制しつつ、検索クエリの転送率を向上させることができる。

端末 M_s が行う経路探索は Algorithm 4 に従う。経路探索のアルゴリズムは基本的に提案手法と同様である。しかし、拡張手法では複数経路により検索クエリを転送しているため、 $RREQ$ を受信した端末がリンク切断先の端末でない場合においても、リンク切断先の端末が、自身の隣接端末や子端末であるならば $RREP$ を返信できる。つまり、リンク切断先の端末がすでに検索クエリを受信済みであるか把握している (Algorithm 4 第 9~10 行)。これにより、 $RREQ$ の不要な転送を抑制し、経路探索にかかるトラヒック、および遅延を抑制できる。

5.3.2 スコア更新への対応

提案手法では、データのスコアが更新された際、順位の変化したデータの宛先にスコア更新メッセージを送信している。しかし、この方法では検索クエリを転送する際、経路表を更新した端末を経由する必要があるため、順位の変更されたデータまでの経路が遠回りとなり、不要なトラ

Algorithm 4 経路探索アルゴリズム

```

1: /* RREQ の受信 */
2: if  $M_s$  receives RREQ for the first time then
3:   Store the sender node
4:   if destination node is itself then
5:     Send RREP to the stored sender node
6:   if  $M_s$  has not received the query message then
7:     Perform the procedure of Algorithm 2
8:   end if
9:   else if destination node is included in  $NL_{query}$  or  $M_s$ 's
     children then
10:    Send RREP to the stored sender node
11:   else if RREQ.TTL > 0 then
12:     TTL  $\leftarrow$  TTL - 1
13:     Broadcast RREQ
14:   end if
15: end if
16: /* RREP の受信 */
17: Update the query addresses of corresponding (destination
   node's) ranks to the sender node
18: if  $M_s$  is the route request node then
19:   if all destination nodes have already received the query
     message then
20:     Send reply message to its parent
21:   end if
22: else
23:   Send RREP to the stored sender node
24: end if

```

ヒックが発生してしまう。そこで拡張手法では、スコア更新メッセージを、経路表における k_{max} 位までの宛先に送信する。スコア更新メッセージを受信した端末も同様の操作を行い、検索クエリ転送時と同様に不要な転送先は送信先から除外する。さらに、このメッセージを傍受することにより、遠回り経路が構築されることを抑制する。経路表の修正方法については提案手法と同様に、Algorithm 1 に従う。

6. シミュレーション評価

本章では、提案手法、およびその拡張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^{*1}を用いた。

6.1 シミュレーション環境

800 [m]×500 [m] の 2 次元平面状の領域に 100 台の端末 ($M_1, M_2, \dots, M_{100}$) が存在する。各端末はランダムウェイポイント [2] に従い、 v [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、128 [Byte] のサイズのデータが 2,000 個存在するものとし、各端末はそれぞれ 20 個のデータを持つものとした。データのスコアは正規分布に従い、とりうる値は [1, 999] の範囲に含まれる整数とした。ここで、提案手法、および拡張手法において、経路表に保持する順位 R を 100 とし、検索クエリ転送時やクエリ応答返信時にリンク切断を検出した際の経路

^{*1} Scalable Network Technologists: Creators of QualNet Network Simulator Software,
<URL: <http://www.scalable-networks.com/>>.

表 3 パラメータ設定

Table 3 Configuration of parameters.

パラメータ	意味	値
k	要求データ数	30 (1~50)
v	速度	0.5 (0~3) [m/秒]
I_q	クエリ発行周期	3 (2~10) [秒]
I_s	スコアの更新周期	90 (15~600) [秒]

探索の TTL は 2 とした。

Top-k 検索手法として、提案手法、拡張手法、および文献 [13] で提案された 2 フェーズ検索手法を用いた。

本実験のパラメータを表 3 に示す。左側の値を基本的には用い、その影響を調べる際に、括弧内の値で変化させる。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、 I_q [秒] ごとにランダムに選ばれた端末が Top-k 検索を行うものとした。シミュレーション開始から 600 [秒] 経過した際の以下の評価値を調べた。

- 取得精度

順位付き検索結果の性能を測る MAP (Mean Average Precision) の値を取得精度とする。MAP は、各クエリの平均精度 AP (Average Precision) を平均化したものである。AP および MAP は以下の式で求める。

$$AP_i = \frac{1}{k} \sum_{j=1}^k \frac{x}{j} \cdot e \quad (1)$$

$$MAP = \frac{1}{querynum} \sum_{i=1}^{querynum} AP_i \quad (2)$$

AP_i は i 番目のクエリの平均精度である。 x は、取得した解の上位 j 個のうち正解集合の j 位以内である解の個数である。 $querynum$ はクエリの発行数を示す。また、 e は以下のように定義される。

$$e = \begin{cases} 1 & (j \text{ 番目の解が正解集合に含まれる}) \\ 0 & (j \text{ 番目の解が正解集合に含まれない}) \end{cases} \quad (3)$$

したがって、MAP は、より上位のデータを取得できているほど、高い値となる。

- トラヒック

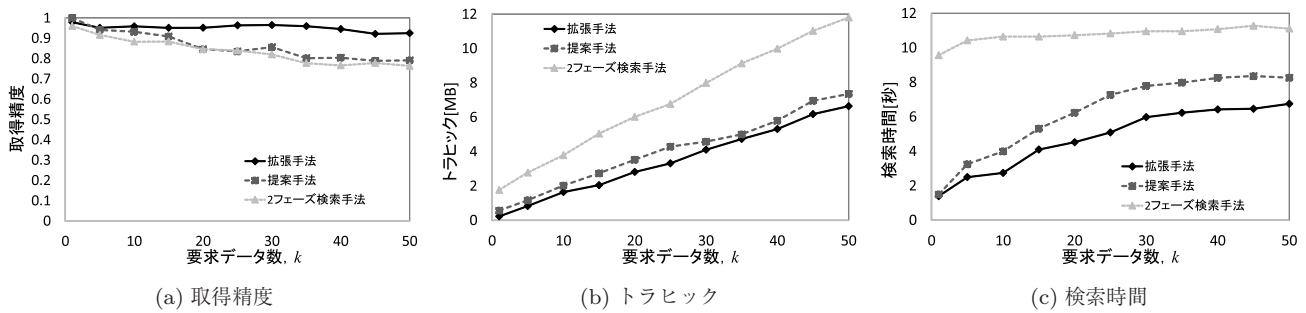
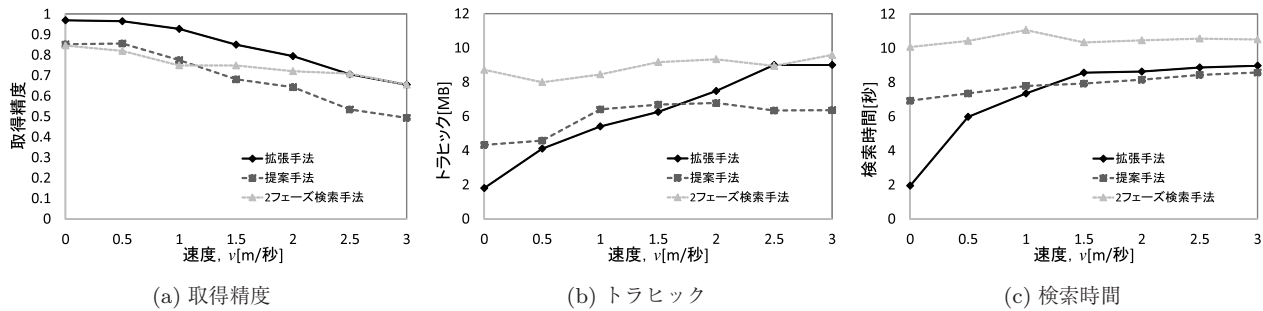
発行したすべての検索クエリを処理するために送信されたすべてのメッセージの総バイトをトラヒックとする。

- 検索時間

全検索クエリに対する、検索クエリ発行端末が検索クエリを発行してから検索結果を取得するまでの平均時間を検索時間とする。

6.2 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各手法の性能を図 4 に示す。これらの図において、横軸は要求データ数 k を表し、縦軸は図 4(a) では取得精度、図 4(b) ではトラヒッ

図 4 要求データ数 k の影響Fig. 4 Impacts of k .図 5 速度 v の影響Fig. 5 Impacts of v .

ク、および図 4(c) では検索時間を表す。

図 4(a) の結果より、提案手法、および拡張手法は 2 フェーズ検索手法よりも取得精度を高く維持できていることが分かる。さらに、拡張手法は提案手法よりも高い取得精度を達成している。これは、複数経路による検索クエリの転送、および効率的な経路探索により、検索クエリ、およびクエリ応答の転送率が提案手法よりも高いためである。ここで、2 フェーズ検索手法は、検索フェーズ時のパケットロスの影響により、指定した閾値が正確でない場合がある。そのため、検索結果に含まれないデータの返信による無駄なトラフィックが生じ、収集フェーズ時にパケットロスが生じやすく、取得精度が低下する。

図 4(b) の結果より、 k が大きくなるとすべての手法においてトラフィックが大きくなっている。これは、 k が大きくなると返信データ数が多くなり、クエリ応答のメッセージサイズが大きくなるためである。提案手法では、各宛先にユニキャストにより検索クエリを送信している。一方、拡張手法では、マルチキャストにより、すべての宛先に同時に検索クエリを転送している。拡張手法は複数経路を構築しているが、このマルチキャストの効果により、検索クエリの送信トラフィックを小さく抑えている。さらに拡張手法では、クエリ応答返信時にリンク切断が生じた場合、経路探索をせずに隣接端末に再送できる場合が多いため、経路探索にかかるトラフィックを抑制することができている。ここで 2 フェーズ検索手法は、フラッディングによる検索クエリの転送、および、検索結果に含まれないデータの返信が生じるため、他の 2 つの手法よりもトラフィックが大きい。

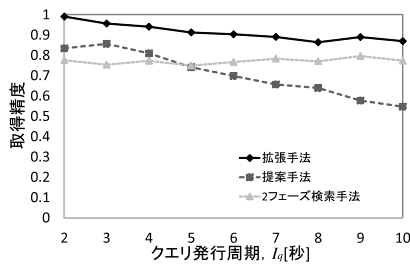
図 4(c) の結果より、提案手法、および拡張手法は検索時間が 2 フェーズ検索手法よりも短い。提案手法、および拡張手法は 2 フェーズ検索手法に比べて子端末数が少なく、子端末からのクエリ応答を待つ時間が短い。また、提案手法では単一経路による検索クエリの転送を行っているため、リンク切断の機会が多く、経路探索による遅延が大きい。これらの理由から、拡張手法の検索時間が最短となる。

6.3 移動速度 v の影響

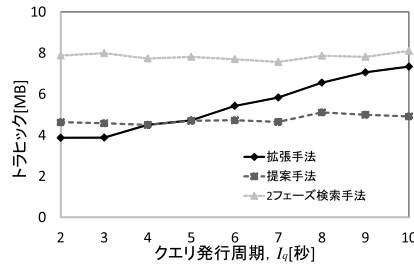
移動速度 v を変化させたときの各手法の性能を図 5 に示す。これらの図において、横軸は移動速度 v を表し、縦軸は図 5(a) では取得精度、図 5(b) ではトラフィック、および図 5(c) では検索時間を表す。

図 5(a) の結果より、拡張手法は提案手法より取得精度を高く維持していることが分かる。拡張手法では複数経路による検索クエリの転送を行っているため、提案手法に比べて検索クエリの転送率が高い。さらに、拡張手法では経路探索時の RREQ に検索クエリの情報を添付しているため、検索クエリの転送遅延を可能な限り抑制している。これにより、ネットワークポロジの変化により対応できるため、取得精度を高く維持できている。

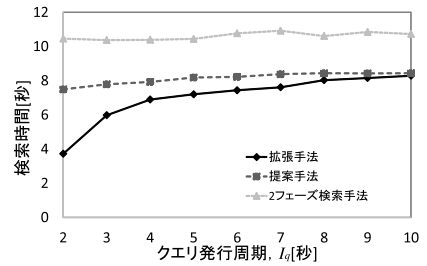
図 5(b) の結果より、拡張手法では v が大きくなると、トラフィックが大きくなることが分かる。 v が大きくなると、リンク切断の機会が多くなり、拡張手法、および提案手法では多くの端末が経路探索を行う。 v が大きい場合、拡張手法は提案手法よりもトラフィックが大きい。これは、経路探索時の RREQ に検索クエリの情報を添付しているため、



(a) 取得精度

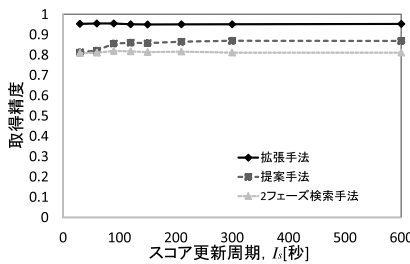


(b) トラフィック

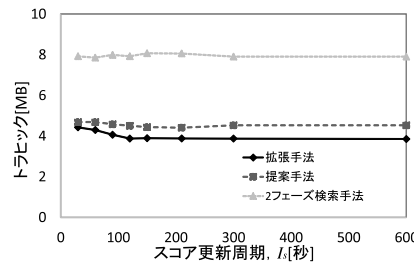


(c) 検索時間

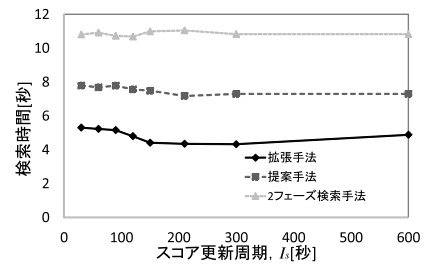
図 6 クエリ発行周期 I_q の影響

Fig. 6 Impacts of I_q .


(a) 取得精度



(b) トラフィック



(c) 検索時間

図 7 スコア更新周期 I_s の影響

Fig. 7 Impacts of I_s .

そのサイズが大きくなることが一因である。さらに、提案手法では図 5(a) から分かるように、取得精度が低く、拡張手法に比べてクエリ応答の返信数が少ない。その結果、提案手法の方がトラフィックが小さくなっている。

図 5(c) の結果より、拡張手法、および提案手法は v が大きくなると検索時間が長くなることが分かる。 v が大きくなると、リンク切断が多くなり、中継端末でクエリ応答が破棄される機会が多くなる。ここで 2 フェーズ検索手法は、多くの端末が最大待ち時間までクエリ応答を待ち、さらに 2 往復による検索を行うため、提案手法、および提案手法よりも検索時間が長くなる。

6.4 クエリ発行周期 I_q の影響

クエリ発行周期 I_q を変化させたときの各手法の性能を図 6 に示す。これらの図において、横軸はクエリ発行周期 I_q を表し、縦軸は図 6(a) では取得精度、図 6(b) ではトラフィック、および図 6(c) では検索時間を表す。

図 6(a) の結果より、提案手法は I_q が大きくなると取得精度が低下していることに対し、拡張手法は最も高い取得精度を維持していることが分かる。 I_q が大きくなると、Top-k 検索が行われていない間に端末が移動し、経路表における宛先端末とのリンク切断が生じる場合が多くなる。提案手法、および拡張手法では、多くの端末が経路探索を行うが、拡張手法では経路探索と同時に検索クエリの転送も行っているため、検索クエリの転送率が高い。しかし、提案手法では経路探索終了後に検索クエリの転送を行うため、経路探索に失敗した場合に検索クエリを転送でき

ない。2 フェーズ検索手法は、クエリを発行するごとにフラッシングを行うため、基本的に I_q に依存しない。

図 6(b) の結果より、拡張手法では I_q が大きくなると、トラフィックが大きくなるが、提案手法はトラフィックが小さいことが分かる。これは、6.3 節で述べた理由と同様である。

図 6(c) の結果より、提案手法、および拡張手法は I_q が大きくなると検索時間が長くなることが分かる。提案手法で検索時間が長くなるのは、 I_q が大きくなると、経路探索を行う機会が増加するためである。拡張手法では、経路探索にともなうトラフィックの増加により、パケットロスが発生しやすくなるため、子端末からのクエリ応答を待つ時間が長くなる。

6.5 スコア更新周期 I_s の影響

スコア更新周期 I_s を変化させたときの各手法の性能を図 7 に示す。これらの図において、横軸はスコア更新周期 I_s を表し、縦軸は図 7(a) では取得精度、図 7(b) ではトラフィック、および図 7(c) では検索時間を表す。

図 7(a) の結果より、拡張手法では I_s が短い場合でも取得精度を高く維持できていることが分かる。これは、拡張手法におけるスコア更新メッセージは、経路表の k_{max} 位までの宛先に送信することから、経路表の精度を維持できるためである。5.3.2 項でも説明したように、提案手法におけるスコア更新メッセージの転送方法では、検索クエリの転送経路を遠回りのものにしてしまう。そのため、検索クエリ、およびクエリ応答の転送ホップ数が大きくなり、

パケットロスが起きやすくなることも取得精度を低下させる要因となる。

図 7(b), および図 7(c) の結果より, 拡張手法では I_s が小さい場合, トラヒックが大きく, 検索時間が長くなることが分かる。これは, スコア更新メッセージを経路表の k_{max} 位までの宛先に送信することから, 検索クエリとのメッセージ衝突が生じやすく, 再送の影響でトラヒック, および, 遅延が増加するためである。

7. おわりに

本論文では, アドホックネットワークにおいて, 検索結果の取得に必要な端末への検索クエリの転送を抑止する Top-k 検索のためのルーティング手法を提案した。提案手法では, 各端末が上位データを取得するための検索クエリの宛先を管理した経路表を保持する。Top-k 検索の際は, 要求するデータの順位を添付した検索クエリを, そのデータに対応する宛先に送信する。検索クエリを受信した端末も同様の操作を行うことにより, 検索結果の取得に必要な端末のみで Top-k 検索を行える。さらに, 提案手法における単一経路での検索クエリの転送方法が, ネットワークトポロジの変化が大きい場合に Top-k 検索の取得精度の低下を招くことを考慮して, 複数経路で検索クエリを転送する拡張手法も提案した。拡張手法では, 経路表の k 位までの宛先に検索クエリをマルチキャストし, これを受信した端末も同様の処理を行いつつ, 受信した検索クエリに添付されている情報から不要な転送先を除外する。拡張手法により, 提案手法とは異なり, 検索クエリを複数経路で転送できるため, 検索クエリの転送率を向上させることができる。

提案手法, および拡張手法は, 経路表における k 位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により, 検索結果に含まれないデータの返信による不要なトラヒックを削減する。さらに, 検索クエリやクエリ応答の送信の際にリンク切断を検出した場合, 経路探索を行うことにより, 転送のための別経路を発見する。シミュレーション実験の結果から, 提案手法, および拡張手法は, 既存手法よりも小さいトラヒックで検索結果の取得精度を維持していることを確認した。

ネットワーク内で端末密度が小さい場合, 提案手法では経路探索が成功する確率は低下する。そこで, データの複製を保持することにより, 検索クエリの転送範囲を抑制できると考える。今後は, データの複製を作成する環境を想定し, そのような環境に適応できるように提案手法を拡張する予定である。

謝辞 本研究の一部は, 文部科学省研究費補助金・基盤研究 S (21220002), および基盤研究 B (24300037) 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, E., Pacitti, E. and Valduriez, P.: Reducing network traffic in unstructured P2P systems using top-k queries, *Distributed and Parallel Databases*, Vol.19, No.23, pp.67–86 (2006).
- [2] Camp, T., Boleng, J. and Davies, V.: A survey of mobility models for ad hoc network research, *Wireless Communications and Mobile Computing*, Vol.2, No.5, pp.483–502 (2002).
- [3] Cecílio, J., Costa, J. and Furtado, P.: Survey on data routing in wireless sensor networks, *Wireless Sensor Network Technologies for the Information Explosion Era*, Vol.278, pp.3–46 (2010).
- [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 (2009).
- [5] Intanagonwatt, C., Govindan, R. and Estrin, D.: Directed diffusion for wireless sensor networking, *IEEE/ACM Trans. Networking*, Vol.11, No.1, pp.2–16 (2003).
- [6] Johnson, B.D.: Routing in ad hoc networks of Mobile Hosts, *Proc. IEEE WMCSA*, pp.158–163 (1994).
- [7] Li, X., Xu, J. and Wu, J.: HR-SDBF: An approach to data-centric routing in WSNs, *Int. Journal. High Performance Computing and Networking*, Vol.6, No.3, pp.181–196 (2010).
- [8] Liu, X., Xu, J. and Lee, W.-C.: A cross pruning framework for top-k data collection in wireless sensor network, *Proc. Int. Conf. on Mobile Data Management*, pp.157–166 (2010).
- [9] Michel, S., Peter, T. and Weikum, G.: KLEE: A framework for distributed top-k query algorithms, *Proc. Int. Conf. on VLDB*, pp.637–648 (2005).
- [10] Mo, S., Chen, H. and Lie, Y.: Clustering-based routing for top-k querying in wireless sensor networks, *EURASIP Journal on Wireless Communications and Networking*, Vol.2011, No.1, pp.1–13 (2011).
- [11] Perkins, E.C. and Ooyer, M.E.: Ad-hoc on-demand distance vector routing, *Proc. IEEE WMCSA*, pp.90–100 (1999).
- [12] Sasaki, Y., Hara, T. and Nishio, S.: A top-k query method by estimating score distribution in mobile ad hoc networks, *Proc. Int. Conf. on Advanced Information Networking and Applications Workshops*, pp.944–949 (2010).
- [13] Sasaki, Y., Hara, T. and Nishio, S.: Two-phase top-k query processing in mobile ad hoc networks, *Proc. Int. Conf. on Network-Based Information Systems*, pp.42–49 (2011).
- [14] Wu, M., Xu, J., Tang, X. and Lee, W.-C.: Top-k monitoring in wireless sensor networks, *IEEE Trans. Knowledge and Data Engineering*, Vol.19, No.7, pp.962–976 (2007).
- [15] Ye, M., Lee, W.-C. and Lee, D.-L.: Distributed processing of probabilistic top-k queries in wireless sensor networks, *IEEE Trans. Knowledge and Data Engineering*, Vol.25, No.1, pp.76–91 (2011).

推薦文

アドホックネットワークでのデータスコアに基づく Top-k 検索のため, 各端末が経路表により検索クエリの転送先を把握し, 上位 k 個のデータ取得に必要な端末のみに

検索クエリを送信することで、トラフィックを削減しつつ、取得精度の低下を抑制する新しいルーティング方式を提案している。シミュレーションを通じ提案手法による性能改善が明確に示されており、学術的に高い貢献が認められる。よって、本研究会からの推薦に値する。

(マルチメディア通信と分散処理研究会主査 勝本道哲)



天方 大地 (学生会員)

2012 年大阪大学工学部電子情報工学科卒業。現在、同大学大学院情報科学研究科博士前期課程在学中。分散環境におけるクエリプロセッシング技術に興味を持つ。日本データベース学会の学生会員。



佐々木 勇和 (学生会員)

2009 年大阪大学工学部電子情報エネルギー工学科卒業。2011 年同大学大学院情報科学研究科博士前期課程修了。現在、大学院情報科学研究科博士後期課程在学中。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。モバイルコンピューティング、ネットワーク環境におけるデータ管理技術に関する研究に従事。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 (フェロー)

1975 年京都大学工学部数理工学科卒業。1980 年同大学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を経て、1992 年大阪大学工学部教授、2002 年同大学大学院情報科学研究科教授となり、現在に至る。その間、大阪大学サイバーメディアセンター長、大学院情報科学研究科長、理事・副学長を歴任。データベースシステムにおけるデータおよび知識管理に関する研究に従事し、紫綬褒章、立石賞功績賞等を授与される。日本学術会議会員。本会では理事を歴任し、論文賞、功績賞を受賞。IEEE, 電子情報通信学会フェロー。