

アドホックネットワークにおけるデータ更新間隔を考慮した キャッシュ無効化について

林 秀樹[†] 原 隆浩[†] 西尾 章治郎[†]

[†] 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

〒 565-0871 大阪府吹田市山田丘 2-1

E-mail: {hideki,hara,nishio}@ist.osaka-u.ac.jp

アドホックネットワークでは、ネットワークの分断が頻繁に発生し、アクセス可能性が低下するため、データの複製を配置することが有効である。本稿では、データ更新が発生するアドホックネットワークにおいて、古いキャッシュデータを効果的に無効化する方式を提案する。提案方式では、各移動体がデータ更新の発生間隔を確率的に考慮することで、自身のキャッシュ領域にもつキャッシュデータを破棄し、古いキャッシュデータへのアクセス回数を削減する。これにより、更新後の古いデータに対する無駄なアクセスによって生じるロールバック処理の回数も削減できる。

On Cache Invalidation Considering Data Update Intervals in Ad Hoc Networks

Hideki HAYASHI[†]

Takahiro HARA[†]

Shojiro NISHIO[†]

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

In ad hoc networks, since network division occurs frequently, it is effective to replicate data items for improving data accessibility. In this paper, we propose a method which invalidates cached data items effectively in ad hoc networks where each data item is updated at inconsistent intervals. In this method, each mobile host discards cached data items which have been updated with high probability by considering their data update intervals. As a result, the proposed method reduces the number of accessing invalid cached data items which have been updated and the number of rollbacks caused by such invalid accesses.

1 はじめに

近年、無線通信技術の発展と計算機の小型化に伴い、携帯型計算機を持ち歩き、いつでもどこでもネットワークに接続することが可能な移動体計算機環境が普及しつつある。特に、ルータ機能をもつ移動体のみで一時的なネットワークを形成するアドホックネットワークに関する研究への関心が高まっている [1, 2, 10, 11, 12]。アドホックネットワークでは、移動体の移動によってネットワークが分断された場合に、分断された部分ネットワーク内のデータに対してアクセスできないため、データの利用率が低下してしまう (図 1)。この問題を解決する手法として、オリジナルデータをもつ移動体とは別の移動体に、データの複製を配置することが有効である。

従来のアドホックネットワークに関する研究の大

半は、IETF(Internet Engineering Task Force)を中心として、相互接続されている移動体間における通信性能を向上させるためのルーティングプロトコルに関するものである [7, 8, 9, 13, 14, 15, 16]。一方、アドホックネットワークにおけるアプリケーションには、ユーザ同士が直接コミュニケーションをとるものだけでなく、センサーネットワークでのデータ共有や、発掘調査などの協調作業において作業の効率化を図るために、他のユーザのもつデータにアクセスするものも多い。そのため、アドホックネットワークにおけるデータ利用性の向上を目的として、効率的に複製配置を行うことの重要性は高い。しかし、筆者らの知る限り、アドホックネットワークにおけるデータの複製配置に関する研究は、筆者ら以前には行われていない。

筆者らは文献 [3] において、移動体が限られたキ

キャッシュ領域をもち、データ更新が起こらないアドホックネットワークを想定して、データの複製を配置する方式を提案した。これらの提案方式は、データへのアクセス頻度とネットワークのトポロジを考慮した、ヒューリスティックな方式である。さらに文献 [4] において、データ更新が周期的に起こるアドホックネットワークを想定して、データ利用性向上のための複製配置方式を提案した。これらの提案方式では、各データへのアクセス頻度、各データの次の更新発生までの時間、およびネットワークの接続状態を考慮して複製の配置を決定する。

一方、実環境では、データ更新が不定期に発生することが一般的である。このような環境では、移動体が更新発生後の無効な複製（キャッシュデータ）にアクセスする可能性がある。古い複製へのアクセスは、オリジナルをもつ移動体と再接続した際にロールバックされる。このような無駄なデータアクセスやロールバック処理は、消費電力が重要な問題となる移動体計算環境では好ましくない場合が多い。これまでに筆者らは、文献 [6] において、データ更新が不定期に起こるアドホックネットワークを想定して、移動体が無効化情報を放送することで、古い複製を効果的に無効化する方式を提案した。しかし、これらの提案方式では、無効化情報を受信するまで古い複製を無効化できないため、移動体のキャッシュ領域に古い複製が長時間保持される場合がある。そこで、本稿では、古い複製へのアクセス回数を削減するために、データ更新の発生間隔を確率的に計算し、各データにタイムアウト時間を設けることで、各移動体が自身のもつ複製を破棄する方式を提案する。

以下では、2章で想定環境について述べる。3章で本稿で提案する方式について述べる。4章で提案方式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。5章で提案方式に対する考察を行う。最後に6章で本稿のまとめを述べる。

2 想定環境

本稿では、データ更新が起こるアドホックネットワークにおいて、他の移動体のもつデータに対してアクセスする環境を想定する。各移動体は、自身のキャッシュ領域に、他の移動体がオリジナルとして

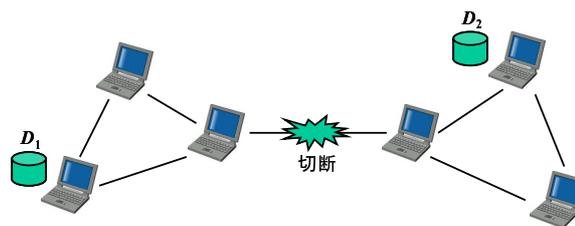


図 1: ネットワークの分断

もつデータの複製を作成し、複製を一定の周期（再配置周期）で再配置する。

移動体のデータアクセスは、オリジナルにアクセスした場合、もしくはオリジナルと同じタイムスタンプ（バージョン）をもつ複製にアクセスした場合にのみ成功し、オリジナルと異なるタイムスタンプをもつ古い複製にアクセスした場合は失敗とみなす。アクセス要求は、自身もしくは相互接続された移動体が、アクセス対象のオリジナルをもつ場合、即座に成功する。なお本稿では、相互接続された移動体とは、1ホップ以上の無線リンクで相互に通信可能な移動体の集合を指す。自身もしくは相互接続された移動体が、アクセス対象のオリジナルをもたず、複製のみをもつ場合、その複製に対して暫定的にアクセスする。暫定的なアクセスは、後にオリジナルをもつ移動体と相互接続した際に、成功か失敗かが決定する。これは、オリジナルをもつ移動体において、更新履歴を保持しておき、複製に暫定的にアクセスした移動体が、オリジナルをもつ移動体に接続した際に、アクセスを行った複製のタイムスタンプとアクセス時刻を知らせることで実現できる。暫定的なアクセスが失敗となった場合は、複製にアクセスする前の状態に戻るようロールバック処理を行う。一方、自身もしくは相互接続された移動体がアクセス対象のオリジナルや複製をもっていない場合は、アクセス要求は、即座に失敗する。

想定環境のその他の詳細を以下に示す。

- m 個の移動体（識別子： M_1, M_2, \dots, M_m ）が存在し、各々が自由に移動する。
- n 個のサイズの等しい異なるデータ（識別子： D_1, D_2, \dots, D_n ）が存在し、各々が特定の移動体にオリジナルデータとして保持されている。
- 各移動体 $M_i (i = 1, \dots, m)$ は、自身のもつオ

リジナルデータ以外に、データ C 個分のキャッシュ領域をもち、複製を作成する。

- 各移動体の各データに対するアクセス頻度は既知とし、時間的に変化しない。
- 各データは、そのオリジナルをもつ移動体によって更新される。更新発生後、古いキャッシュデータは無効となる。
- 各移動体は、ネットワーク内に存在する各データの最近の更新時刻（タイムスタンプ）を保持する。この情報を記録する表をタイムスタンプ表と呼ぶ。

3 キャッシュ無効化方式

本章では、まず、各データにタイムアウト時間を設けてキャッシュを無効化する提案方式に適した複製配置方式について説明する。次に、提案方式について述べる。さらに、提案方式に適したデータアクセス方法について述べる。

3.1 複製配置方式

文献 [5] では、データ更新が不定期に発生する環境を想定した複製配置方式を提案している。本稿では、文献 [5] の提案方式をデータのタイムアウト時間を考慮するように拡張した方式を、複製配置方式として用いる。以下では、この方式について説明する。

まず、データ D_j の各複製に対して、PTT 値と呼ぶ評価値を次のように定義する。

$$p_{ij} \cdot \int_0^{to_j} f_j(t + t_j) \cdot t dt \quad (1)$$

ここで、 p_{ij} を移動体 M_i のデータ D_j へのアクセス頻度、 to_j を D_j の複製のタイムアウト時間 ($to_j \geq 0$)、 t_j を D_j が更新されてから経過した時間、 $f_j(t)$ を D_j の更新発生間隔の確率密度関数とする。 t_j は、複製の再配置時刻と各移動体が保持している D_j のタイムスタンプの差から求められる。なお、 D_j がすでにタイムアウトしているとき ($to_j < 0$)、PTT 値は 0 と定義する。

PTT 値は、移動体がデータ D_j の複製を破棄するまでにアクセスが成功する平均回数を表している。

したがって、再配置周期ごとに各移動体が PTT 値を計算し、PTT 値の高いデータの複製を配置することで、アクセス成功率の向上を図りながら、古い複製へのアクセス回数を削減できる。 to_j の決定法については、次節で詳しく説明する。

本稿では、PTT 値を用いた下記の複製配置方式に基づいて、複製の再配置を行う。

1. E-SAF⁺ 方式

各移動体において、自身のキャッシュ領域の許す限り、PTT 値の高いデータから順に複製を配置する。データのタイムアウト時間が経過して複製が無効になった場合、その後、そのオリジナルデータもしくは有効な複製にアクセスしたときに再びその複製を配置する。この動作は、下記の E-DAFN⁺ 方式、E-DCG⁺ 方式についても同様である。

2. E-DAFN⁺ 方式

E-SAF⁺ 方式では、同じアクセス特性をもつ移動体が同じ複製を配置してしまうため、複製の重複が多く、ネットワーク全体のアクセス成功率が低い。そこで、E-DAFN⁺ 方式では、E-SAF⁺ 方式で各移動体に複製を暫定的に配置した後、隣接する移動体間で複製の重複を解消する。

3. E-DCG⁺ 方式

E-DAFN⁺ 方式よりも、さらに広範囲で複製を共有するため、再配置周期ごとに安定度の高い移動体のグループを作成し、グループ内で複製を共有する。具体的には、ネットワークの 2 連結成分を一つのグループとして、グループ内の移動体の PTT 値が高いデータの複製を配置する。2 連結成分をグループにしているため、任意の一つの移動体がネットワークから離脱してもグループは分断されない。

3.2 キャッシュ無効化

提案方式では、データ更新が発生するアドホックネットワークにおいて、更新の発生間隔を確率的に考慮して、各データにタイムアウト時間を設ける。これにより、各移動体が、前回の更新から長時間経

過した複製を破棄するため、古い複製へのアクセス回数を削減できる。

データ D_j のオリジナルをもつ移動体が、直前の更新から時間 τ_j 以内に次の更新を行う確率は、次式で表せる。

$$\int_0^{\tau_j} f_j(t) dt \quad (2)$$

提案方式では、ネットワーク内に存在する全てのデータに一定の閾値 α を設定する。式 (2) で求めた確率が α 以上になるとき、データ D_j の複製をもつ各移動体は、自身のもつ D_j の複製をキャッシュから削除する。より具体的には、各移動体が D_j の複製を配置するとき、もしくはタイムスタンプ情報を記録するときに、式 (2) を用いて D_j が更新された確率が α になる時間 τ_j を求め、これを D_j のタイムアウト時間 to_j とする。複製配置の際の PTT 値は、式 (1) と to_j から求めることができる。

各移動体は自身のキャッシュ領域を監視し、無駄なアクセスを削減するために、タイムアウト時間を経過した複製をキャッシュ領域から破棄する。このとき、破棄した複製を配置していたキャッシュ領域は、空けたままにしておく。その後、破棄した複製のオリジナル、または複製にアクセスしたとき、そのデータの複製をキャッシュ領域に再び配置する。

それと同時に、ある複製のタイムアウト時間が経過したら、その複製をもつもたないに関わらず、そのタイムスタンプ情報をタイムスタンプ表から破棄する。その後、破棄した複製のオリジナル、または複製にアクセスしたときに、自身のもつタイムスタンプ表にそのタイムスタンプ情報を記録する。

ここで、閾値 α を小さく設定すると、更新された確率の低い複製を破棄するため、更新された古い複製にアクセスする回数は削減できるが、実際は更新されていない複製が破棄される可能性も高くなる。そのため、アクセス成功回数も減少してしまう。したがって、 α の値は、システム特性や要求事項に応じて、慎重に決定しなければならない。

3.3 データアクセス

本節では、提案方式を用いる際のデータアクセス方法について説明する。

先述の通り、アクセス要求をした移動体がオリジナルにアクセスした場合は、即座にアクセスが成

功する。しかし、複製に暫定的にアクセスした場合は、オリジナルは既に更新されている可能性があるため、アクセスが失敗となる場合がある。

そこで、アクセス対象のデータが、自身のもつオリジナルデータではない場合、ネットワーク内にアクセス要求を放送する。オリジナルをもつ移動体と相互接続していないとき、自身を含め、相互接続している移動体のもつ複製の一つに暫定的にアクセスする。詳しい手順は、以下の通りである。

まず、アクセス要求を行う移動体は、相互接続している移動体の中で、アクセス対象のデータをもつものがあるのかを調べるために、次のような情報を含むデータ問合せパケットを放送する。

- 自身の識別子
- アクセス対象のデータの識別子

次に、データ問合せパケットを受信した移動体は、自身がアクセス対象のオリジナル、もしくは複製をもつ場合、アクセス要求を行った移動体に、その旨を伝える返信パケット（データ問合せ返信パケット）を送信する。ここで、相互接続している各移動体が異なるタイムスタンプ表をもつ可能性があるため、データ問合せ返信パケットにタイムスタンプ情報を付加する必要がある。したがって、データ問合せ返信パケットは、次のような情報を含む。

- 自身の識別子
- アクセス対象のデータの識別子
- オリジナルもしくは複製かを表すフラグ
- データ（複製）のタイムスタンプ

その後、データ問合せをした移動体は、オリジナルをもつ移動体からデータ問合せ返信パケットを受信すると、その移動体にデータ要求パケットを送信し、オリジナルにアクセスする。一方、オリジナルをもつ移動体と相互接続していない場合は、データ問合せに対して返信があった移動体の中で、最も新しい複製をもつ移動体にデータ要求パケットを送信する。その後、その複製に暫定的にアクセスする。

暫定的なアクセスを行った移動体は、後にオリジナルをもつ移動体と相互接続した際に、暫定的なアクセスが成功か失敗かを調べるため、次のような情報をアクセス履歴として保持する。

- 複製にアクセスした時刻
- アクセスした複製のデータ識別子
- アクセスした複製のタイムスタンプ（バージョン）

4 性能評価

本章では、提案方式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。

4.1 シミュレーション環境

50×50の2次元平面上に、40個の移動体 ($M = M_1 \dots M_{40}$) が存在する。各移動体は、全ての方向に等確率に、0から1の範囲でランダムに決定した速度で移動する。各移動体の無線通信範囲は、半径8の円とする。ネットワーク内には、40種類のデータ ($D = D_1 \dots D_{40}$) が存在し、 D_j は M_j にオリジナルデータとして保持されている。各移動体は、最大10個の複製を作成し、再配置周期を40として、3.1節で述べた複製配置方式を用いて、複製を再配置する。各移動体 M_i の各データ D_j へのアクセス頻度は、 $p_{ij} = 0.5 \times (1 + 0.001i)$ とする。データの更新は、平均更新間隔 U_{avg} の指数分布に基づいた間隔で発生する。

初期位置として各移動体をランダムに配置し、シミュレーションの単位時間ごとに、各移動体の各データに対するアクセス頻度に基づいてアクセス要求を発行させる。シミュレーション実験では、100,000単位時間を経過させたときの下記の評価値について調べる。

- アクセス成功率

シミュレーション時間内に発生したアクセス要求の総数に対するアクセス成功回数の割合。
- 古い複製にアクセスした割合

シミュレーション時間内に発生したアクセス要求の総数に対する、更新発生後の無効なキャッシュデータに対して行われた暫定アクセスの総数の割合。

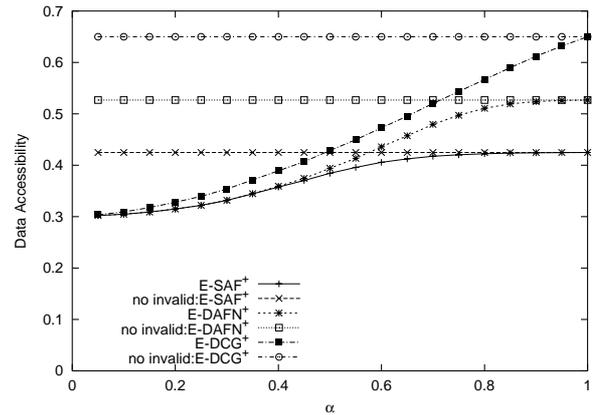


図 2: α とアクセス成功率

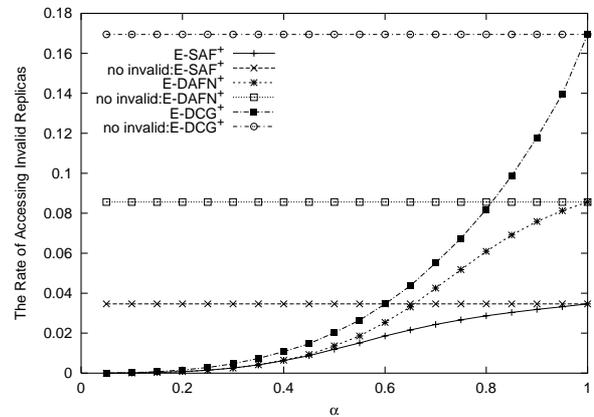


図 3: α と古い複製にアクセスした割合

4.2 α の影響

まず、 U_{avg} を 100 に固定して、 α の値を 0.05 から 1 まで変化させたときの提案方式の性能を調べた。シミュレーション結果を、図 2、図 3 に示す。これらの図において、横軸は α を示す。縦軸は、図 2 ではアクセス成功率、図 3 では古い複製にアクセスした割合をそれぞれ示す。また、グラフ中の凡例は、用いている複製配置方式、および提案方式の使用の有無を表している。複製配置方式のみを表記しているものは、複製の無効化を行う場合を表し、複製配置方式と ‘no invalid’ を併記しているものは、複製の無効化を行わない場合を表す。

図 2 の結果から、複製の無効化を行う場合、 α の値が大きくなると各複製配置方式のアクセス成功率が高くなることわかる。これは、 α の値が大き

なると、各データに関するタイムアウト時間が長くなるため、ネットワーク内にアクセス可能な複製が多く残るからである。一方、複製を無効化しない場合、 α の値に依存せず、各複製配置方式のアクセス成功率は一定の値になる。複製の無効化を行う場合で、複製配置方式について比較すると、E-DCG⁺方式、E-DAFN⁺方式、E-SAF⁺方式の順で高いアクセス成功率を示すことがわかる。E-DCG方式では、グループ内で複製を共有するため、相互接続している移動体間で複製の重複が解消され、多種類のデータにアクセスできている。

図3の結果から、複製の無効化を行う場合、 α の値が非常に小さいとき、全ての複製配置方式において、ほとんど古い複製へアクセスしないことがわかる。これは、 α の値が小さくなると、データのタイムアウト時間が短くなるため、ネットワーク内に存在する複製が短時間で無効化されるからである。また、 α の値が大きくなると、更新後の古い複製へアクセスする割合が高くなる。これは、ネットワーク内に有効な複製が多く残り、アクセス成功率が高くなる反面、更新発生後の古い複製も多く残ってしまう。一方、複製を無効化する場合で、複製配置方式について比較すると、常にE-DCG⁺方式が高い値を示し、E-DAFN⁺方式、E-SAF⁺方式と続いている。これは、前述と同様の理由によるものである。

以上の結果より、本稿のシミュレーション環境では、 α の値を大きくした場合のアクセス成功率の増分と古い複製にアクセスした割合の増分は、前者の方がかなり大きい。しかし、古い複製へのアクセスは、無駄な計算やロールバック処理により、余分な電力消費や計算オーバーヘッドが生じる。そのため、複製の無効化を行う意義は大きい。また、本稿の想定環境よりも、データ更新頻度が高い場合には、古い複製にアクセスした割合の増分はより大きくなる。

4.3 平均更新間隔の影響

次に、 α の値を0.7に固定して U_{avg} を10から400まで変化させたときの提案方式の性能を調べた。シミュレーション結果を、図4、図5に示す。これらの図において、横軸は平均更新間隔を示す。縦軸は、図4ではアクセス成功率、図5では古い

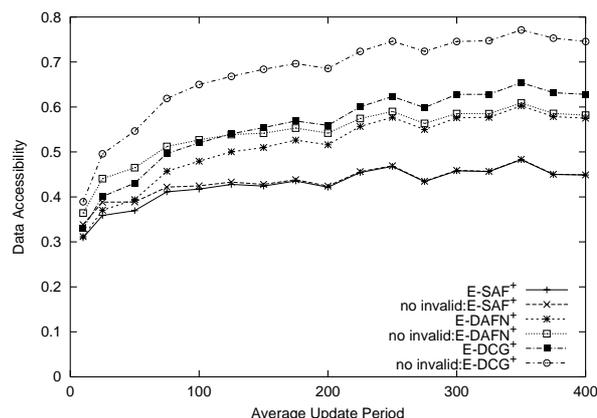


図4: 平均更新間隔とアクセス成功率

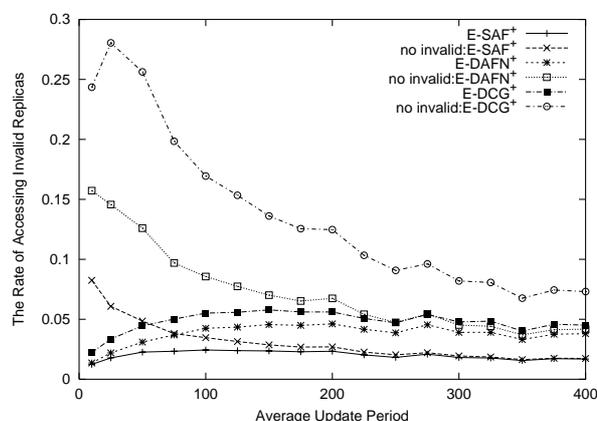


図5: 平均更新間隔と古い複製にアクセスした割合

複製にアクセスした割合をそれぞれ示す。

図4の結果から、平均更新間隔が大きくなると、全ての場合においてアクセス成功率が高くなる。これは、平均更新間隔が大きくなると、各移動体のキャッシュ領域に保持している複製の有効な時間が長くなるからである。また、各複製配置方式において、複製の無効化を行う場合と行わない場合を比較すると、後者の方がアクセス成功率が高い。これは、タイムアウト時間を越えた複製を無効化することにより、実際にはまだ更新が発生していない複製を破棄してしまう場合があるためである。

図5の結果から、複製の無効化を行う場合、平均更新間隔が非常に小さいとき、全ての複製配置方式において、古い複製にアクセスした割合は低くなっている。これは、複製が短時間で無効になるため、複製配置時には、オリジナルをもつ移動体と相互接

続している場合にのみ、その複製が作成されるからである。つまり、オリジナルと同じバージョンの複製しか作成されないため、無駄なアクセスの回数が非常に低くなる。また、複製の無効化を行わない場合、全ての複製配置方式において、平均更新間隔が大きくなると、古い複製にアクセスした割合が減少する傾向にある。これは、平均更新間隔が大きくなると、各移動体のもつ複製が長時間有効になるからである。一方、各複製配置方式において、複製の無効化を行う場合と行わない場合の差を比較すると、平均更新間隔が小さいときには、無効化による効果が非常に大きいことがわかる。平均更新間隔が大きくなるにつれ、複製が長時間有効となるため、その効果が小さくなる。

5 考察

本章では、提案方式に関して、いくつかの観点から考察を行う。

5.1 閾値 α の影響

提案方式の性能は、各データに設定されている閾値 α に大きく依存する。 α の値が大きい場合、キャッシュ領域に複製を長時間保持できるため、ネットワーク内にアクセス可能な複製が多く残り、全体としてアクセス成功率が向上する。一方、 α の値が小さい場合、アクセス成功率は低下するが、複製を配置しても短い時間で複製を破棄するため、ネットワーク内に更新発生後の古い複製が少なくなり、古い複製へアクセスする割合が小さくなる。したがって、実環境で提案方式を適用する場合、データの更新頻度や各移動体の計算能力、バッテリー容量などのシステム特性に応じて α の値を慎重に決定する必要がある。

また、提案方式では、ネットワーク内に存在する全てのデータに対して同じ α の値を設定するものとしている。しかし、各データの更新発生の特性や、オリジナルをもつ移動体の性能などに応じて、各データの α の値を変更することが好ましい場合もある。このような拡張については、今後検討する予定である。

5.2 データの更新間隔

提案方式は、各データの更新発生間隔の確率密度関数 $f_j(t)$ を用いて複製の配置や古い複製の無効化を行う。実環境では、オリジナルをもつ移動体が、データの更新間隔を記録し、 $f_j(t)$ を周期的に求めることが考えられる。求めた $f_j(t)$ は、オリジナルをもつ移動体によって、他の移動体に知らされる。しかし、オリジナルをもつ移動体と相互接続していない移動体は、最新の $f_j(t)$ を知るができない。そのため、これまで保持していた古い $f_j(t)$ の情報を用いて複製の配置や無効化を行わなければならない。

また、データの更新間隔が全く特性をもたずに発生する場合、オリジナルをもつ移動体でさえ $f_j(t)$ を求めることが困難である。このように、実環境では、正確な $f_j(t)$ を得られない状況が多く存在するものと考えられる。提案方式の性能は、 $f_j(t)$ の精度に大きく依存するため、正確な $f_j(t)$ が得られない場合の対処は、今後の重要な課題の一つである。

5.3 データアクセスの成功、失敗

本稿では、移動体がアクセス要求を行うとき、オリジナルデータ、もしくはオリジナルと同じタイムスタンプをもつ複製にアクセスしなければ、失敗とみなされる。しかし、実環境では遺跡発掘調査での調査状況データのように、データが多少古くても、ある程度の有効性をもつ場合がある。このようなアプリケーションでは、 α をある程度低めの値に調節して、アクセス成功率を向上させるアプローチも考えられる。

6 まとめ

本稿では、データ更新が発生するアドホックネットワークにおいて、次に更新が発生するまでの時間を確率的に考慮して、各移動体が自身のキャッシュ領域にもつ古い複製を破棄することで、古い複製にアクセスする回数を削減する方式を提案した。また、このような環境に適應するデータアクセス方法について述べた。

性能評価のために行ったシミュレーション実験の

結果から，各データに設定される閾値 α によって，アクセス成功率と古い複製へアクセスした割合が大きく影響を受けることを確認した．

今後は，文献 [6] で提案した無効化情報を放送する方式も併用する環境において，提案方式の性能評価を行う予定である．また，移動体に設定する閾値 α を，状況に応じて変更できるように，提案方式を拡張する予定である．

謝辞

本研究の一部は，文部科学省科学技術振興調整費「モバイル環境向 P2P 型情報共有基盤の確立」，文部科学省 21 世紀 COE プログラム（研究拠点形成費補助金），および日本学術振興会若手研究 (B)(13780330) 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] D.J. Baker, J. Wieselthier, and A. Ephremides, “A distributed algorithm for scheduling the activation of links in a self-organizing, mobile, radio network,” Proc. IEEE ICC’82, pp. 2F6.1–2F6.5, 1982.
- [2] J. Broch, D.A. Maltz, D.B. Johnson, Y.C. Hu, and J. Jetcheva, “A performance comparison of multi-hop wireless ad hoc network routing protocols,” Proc. Mobicom’98, pp. 85–97, 1998.
- [3] 原 隆浩, “アドホックネットワークにおけるデータ利用性向上のための複製配置,” 電子情報通信学会和文論文誌 B, vol. J84–B, no. 3, pp. 632–642, 2001.
- [4] 原 隆浩, “アドホックネットワークにおける周期的なデータ更新を考慮した複製配置方式,” 電子情報通信学会和文論文誌 B, vol. J84–B, no. 7, pp. 1391–1395, 2001.
- [5] 原 隆浩, “アドホックネットワークにおける不定期データ更新を考慮した複製配置,” 情報科学技術フォーラム (FIT 2002) 論文集, vol. 2, pp. 27–28, 2002.
- [6] 林 秀樹, 原 隆浩, 西尾 章治郎, “アドホックネットワークにおける不定期データ更新を考慮したキャッシュ無効化について,” 情報処理学会マルチメディア通信と分散処理ワークショップ論文集, pp. 219–224, 2002.
- [7] M. Jiang, J. Li, and Y.C. Tay, “Cluster based routing protocol(CBRP),” Internet Draft, draft-ietf-manet-cbrp-spec-01.txt, 1999.
- [8] D.B. Johnson, “Routing in ad hoc networks of mobile hosts,” Proc. IEEE Workshop on Mobile Computing Systems and Applications, pp. 158–163, 1994.
- [9] Y.B. Ko, N.H. Vaidya, “Location-aided routing (LAR) in mobile ad hoc networks,” Proc. Mobicom’98, pp. 66–75, 1998.
- [10] S. Lee and C. Kim, “Neighbor supporting ad hoc multicast routing protocol,” Proc. MOBIHOC’2000, pp. 37–44, 2000.
- [11] S. Lee, W. Su, J. Hsu, M. Gerla and R. Bagrodia, “A performance comparison study of ad hoc wireless multicast protocols,” Proc. INFOCOM’2000, pp. 565–574, 2000.
- [12] M.J. Ng and I.T. Lu, “A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks,” IEEE Journal on Selected Areas in Communications, vol. 17, no. 8, pp. 1415–1425, 1999.
- [13] M.R. Pearlman and Z.J. Haas, “Determining the optimal configuration for the zone routing protocol,” IEEE Journal on Selected Areas in Communications, vol. 17, no. 8, pp. 1395–1414, 1999.
- [14] C.E. Perkins and P. Bhagwat, “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers,” Proc. ACM SIGCOMM’94, pp. 234–244, 1994.
- [15] C.E. Perkins and E.M. Royer, “Ad hoc on demand distance vector routing,” Proc. IEEE Workshop on Mobile Computing Systems and Applications, pp. 90–100, 1999.
- [16] E.M. Royer and C.E. Perkins, “Multicast operation of the ad-hoc on-demand distance vector routing protocol,” Proc. Mobicom’99, pp. 207–218, 1999.