

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複製配置

林 秀樹[†] 原 隆浩[†] 西尾 章治郎[†]

近年、無線通信機能を持つ移動体がパケットを中継して、暫定的にネットワークを構成するアドホックネットワークに関する研究がさかんに行われている。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するため、データの複製を配置することが有効である。本論文では、データ更新が不定期に発生するアドホックネットワークにおいて、複製を効果的に配置する方式を提案する。提案方式では、データ更新の発生間隔を確率的に考慮して複製を配置する。さらに、更新発生後の古い複製に対するアクセスが無駄な計算オーバーヘッドやロールバック処理の原因となることを考慮して、更新された確率の高い複製を無効化する。これにより、データアクセスの成功率を向上するだけでなく、古いキャッシュデータへのアクセス回数を削減できる。

Replica Allocation Considering Data Update Intervals in Ad Hoc Networks

HIDEKI HAYASHI,[†] TAKAHIRO HARA[†] and SHOJIRO NISHIO[†]

Recently, there has been increasing interest in ad hoc networks that are temporarily constructed of only mobile hosts. In ad hoc networks, since network division occurs frequently, it is effective to replicate data items. This paper proposes effective replica allocation methods in ad hoc networks where each data item is updated at inconstant intervals. The proposed methods allocate replicas based on probability density functions of the update intervals of data items. Also, they invalidate replicas that have been updated with high probability because accesses to old replicas impose extra computational overhead and rollbacks. As a result, the proposed methods not only improve data accessibility but also reduce the number of accesses to old replicas which have been updated.

1. はじめに

近年、無線通信技術の発展と計算機の小型化とともに、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環境が普及しつつある。特に、ルータ機能を持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^{4),21)}。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。たとえば、図1のアドホックネットワークにおいて、中央にいる2台の移動体間の無線リンクが切断されたとすると、左側にいる3台の移動体はデータ D_2 に、右側にいる3台の移動体はデータ D_1 にアクセスできなくなる。この

ような問題を解決する手法として、オリジナルデータを持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、相互接続し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^{18),20),23),25)~27)}。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、他のユーザの持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率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。

筆者らは文献11)において、移動体が限られたキャッシュ領域を持ち、データ更新が起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

[†] 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

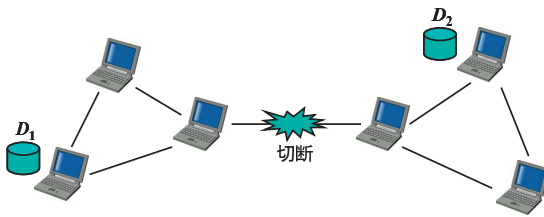


図 1 ネットワークの分断
Fig. 1 Network division.

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 12) 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、複製を持つ移動体は、そのオリジナルデータがいつ更新されるかをあらかじめ知ることはできない。そこで、本論文では、データ更新の発生間隔を確率的に計算して、アクセス成功率が高くなるように複製を配置する方式を提案する。また、不定期なデータ更新が発生する環境では、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、オリジナルを持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。そこで、提案方式では、データ更新の発生間隔を確率的に考慮して複製を破棄することで、必要に応じて、更新発生後の古い複製にアクセスする割合を削減する。

以下では、2 章で想定環境について述べる。3 章で本論文で提案する複製配置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。6 章で関連研究について述べる。最後に 7 章で本論文のまとめを述べる。

2. 想定環境

本論文では、データ更新が不定期に発生するアドホックネットワークにおいて、他の移動体の持つデータに対してアクセスする環境を想定する。なお本論文では、「データアクセス」という用語は、参照操作（リード）のみを指すものとし、更新操作（ライト）は含ま

ない。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持つデータの複製を作成する。ここで、複製の最適な配置を決定するためには、ネットワークの接続状態、各移動体の各データへのアクセス頻度、各データが次に更新される確率などを考慮し、システム全体のアクセス成功率が高く、古い複製へのアクセス回数が少なくなる複製配置を解析的に発見する必要がある。これは、非常に膨大な計算量になるため、各データの更新が発生する確率やネットワークの接続状態が時間的に変化するたびに再計算するのは、現実的に不可能である。そこで、本論文では、最適な複製配置を行うのではなく、複製を一定の周期（再配置周期）で再配置し、その時点でアクセス成功率が高く、古い複製へのアクセス回数が少なくなるような複製配置をヒューリスティックに決定する。

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）を持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持つ古い複製にアクセスした場合は失敗と見なす。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持つ場合、即座に成功する。なお本論文では、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持たず、複製のみを持つ場合、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、後にオリジナルを持つ移動体と相互接続した際に、成功が失敗かが決定する。これは、オリジナルを持つ移動体において、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、オリジナルを持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持っていない場合は、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、自身の持つオリジナルデータ以外に、データ C 個分のキャッシュ

領域を持ち、複製を作成する．

- 各移動体 M_i の各データ $D_j (j = 1, \dots, n)$ に対するアクセス頻度 p_{ij} は既知とし、時間的に変化しない．
- 各データ D_j は、そのオリジナルを持つ移動体によって更新される．更新発生の間隔は、確率密度関数 $f_j(t)$ によって定義され、これも既知とする．更新発生後、古いキャッシュデータは無効となる．実環境では、オリジナルを持つ移動体が更新発生のログを記録し、定期的に統計をとることで、 $f_j(t)$ を求めることができる．また、気象情報や株価情報のように更新が定期的に発生するデータなどでは、 $f_j(t)$ を正確に求めることが可能な場合もある．
- 各移動体 M_i は、ネットワーク内に存在する各データ D_j 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. 複製配置方式

本章では、まず、本論文で提案する複製配置方式について説明する．次に、提案方式におけるキャッシュ無効化について説明する．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

本論文では、文献 12) の周期的なデータ更新を考慮した複製配置方式を、不定期なデータ更新を考慮するように拡張した方式を提案する．

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．

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

ここで、 to_j は D_j の複製のタイムアウト時間 ($to_j \geq 0$)、 t_j は D_j が更新されてから経過した時間である． t_j は、現在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．

提案方式では、文献 12) で提案した 3 つの複製配

置方式のアルゴリズムにおいて、PT 値を PTT 値に置き換えることにより、不定期なデータ更新に適應する．このように拡張した方式を E-SAF (Extended-Static Access Frequency) α 方式、E-DAFN (Ex-Dynamic Access Frequency and Neighborhood) α 方式、E-DCG (Ex-Dynamic Connectivity based Grouping) α 方式と呼ぶ．これらの方式では、PTT 値は、再配置周期ごとに各移動体で計算される．その際に、式 (1) に時刻に関するパラメータが含まれるため、同期的に計算するのが最も望ましい．本論文では、このような時刻同期がとれているものと想定する．同期がとれず非同期に計算が行われた場合、提案方式自体は動作するが、効果的に複製を配置できない可能性がある．特に、後述する E-DAFN α 方式と E-DCG α 方式では、自身が計算した PTT 値だけでなく、他の移動体が計算した PTT 値も考慮して複製を配置するため、非同期的な PTT 値の計算が原因で、アクセス成功率の低下や古い複製にアクセスする割合の増加を招く可能性がある．

3.1.1 E-SAF α 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、後に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．複製が無効になった後の動作は、下記の E-DAFN α 方式、E-DCG α 方式についても同様である．

3.1.2 E-DAFN α 方式

E-SAF α 方式では、同じアクセス特性を持つ移動体と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 α 方式では、E-SAF α 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．そのアルゴリズムは、次のようになる．

- (1) 再配置周期がくると、各移動体は、E-SAF α 方式を用いて、自身のキャッシュ領域に配置する複製を暫定的に決定する．
- (2) 各移動体は、自身の識別子を放送する．ネットワーク内に存在するすべての移動体が放送を終了すると、各移動体は、自身と相互接続している移動体の識別子を知る．
- (3) 識別子 M_i の添字 i が最も小さい移動体から、すべての隣接する移動体間で、次の操作を行う．隣接する移動体間でオリジナルデータや配置する複製に重複がある場合、一方がオリジナル

データの場合は複製を持つ移動体、両方とも複製の場合はそのデータの PTT 値が低い移動体において、配置する複製を変更する。この場合、隣接する移動体間において、複製配置の対象となっていないデータの中で、配置を変更する移動体の PTT 値が最も高いデータの複製を配置する。

3.1.3 E-DCG α 方式

E-DAFN α 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。そのアルゴリズムは次のようになる。

- (1) 再配置周期がくると、各移動体は自身の識別子を放送する。これにより、各移動体は、自身と相互接続している移動体の識別子を知る。
- (2) 識別子 M_i の添字 i が最も小さい移動体から、ネットワークの 2 連結成分を発見するアルゴリズムを実行し、発見された 2 連結成分を 1 つのグループとする (図 2)。そのため、任意の 1 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) グループ内に存在する各移動体の持つ PTT 値を各データごとに合計し、これをグループとしての PTT 値とする。
- (4) グループとしての PTT 値が高いデータから順に、グループ内での複製配置を決定する。ただし、グループ内の移動体のいずれかがオリジナルとして所持しているデータは、複製配置の対象としない。グループ内での複製の配置先は、キャッシュの空き領域がある移動体の中で、そのデータの PTT 値が最も高いものとする。
- (5) ネットワークに存在する全種類のデータの配置が決定した後、グループ内の移動体にキャッシュの空き領域がある場合は、空き領域がなくなるまで、グループとしての PTT 値が高いデータから順に、さらに複製の配置を決定する。グループ内での配置先は、そのデータを所持してなくて、キャッシュ領域に空きのある移動体の中で、そのデータの PTT 値が最も高いものとする。

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。

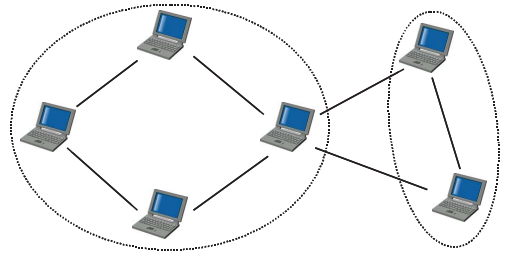


図 2 E-DCG α 方式におけるグループ
Fig. 2 Groups in the E-DCG α method.

データ D_j のオリジナルを持つ移動体が、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すべ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 の複製を持つ各移動体は、自身の持つ 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、その複製を持つ持たないにかかわらず、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。

先述のとおり、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、オリジナルはすで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、自身の持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。オリジナルを持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持つ複製の1つに暫定的にアクセスする。詳しい手順は、以下のとおりである。

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持つものがい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、もしくは複製を持つ場合、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持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、オリジナルを持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、オリジナルを持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、最も新しい複製を持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、後にオリジナルを持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

表 1 パラメータ設定
Table 1 Parameter configuration.

パラメータ	値
α	0.7 (0.05~1)
U_{avg}	50 (10~300)
C	10 (1~39)
v	1 (0.1~3)
R	8 (10)

- 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、すべての方向に等確率に、0から v (最大移動速度)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 R 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 C 個(キャッシュ領域のサイズ)の複製を作成し、再配置周期を10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、 $p_{ij} = 0.5 \times (1 + 0.001j)$ とする。各移動体は、自身の持つオリジナルデータを平均 U_{avg} (平均更新間隔)の指数分布に基づいた間隔で更新する。表1は、シミュレーションで用いるパラメータを示している。各パラメータは、基本的には定数値をとるが、いくつかのパラメータは、シミュレーション実験において表中の括弧内の範囲で変化する。

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。

また、文献(9)、(14)における更新データの配布先を

確率的に選択する方式を参考にして、複製の配置先を確率的に選択する下記的方式を、提案方式の比較対象として用いる。以下では、これを‘比較方式’と称する。

- (1) まず、各移動体の持つキャッシュ領域には制限があるため、各移動体の各データに対するアクセス頻度に比例させて、ネットワーク上に配置する各データの複製の数をあらかじめ決定する。シミュレーション実験では、各移動体 M_i の各データ D_j へのアクセス頻度が $p_{ij} = 0.5 \times (1 + 0.001j)$ であり、各データ間のアクセス頻度の差が小さいため、各データの複製の数は、すべて同じにした。
- (2) 次に、各データの複製の配置先を、各移動体のアクセス頻度に基づいて、確率的に選択する。シミュレーション実験では、各移動体の同一データに対するアクセス頻度が等しいため、複製の配置先はランダムに選択した。

この比較方式は、各移動体がネットワークポロジとデータの更新特性とは無関係に、複製の配置先を選択するため、本論文の提案方式において、これらを考慮したことによる効果も示すことができる。評価結果のグラフでは、この比較方式を‘Probability’として示す。

4.2 α の影響

まず、 α の提案方式に対する影響を調べた。その結果を、図 3、図 4 に示す。これらの図において、横軸は α を示す。縦軸は、図 3 ではアクセス成功率、図 4 では古い複製にアクセスした割合をそれぞれ示す。グラフ中の凡例は、用いている複製配置方式、およびキャッシュ無効化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。また、比較のため、複製の配置をまったく行わない場合を‘no replica’と表す。

図 3 の結果から、複製を作成することで、アクセス成功率が大幅に向上することが分かる。また、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。これは、 α の値が大きくなると、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 α 方式、E-DAFN α 方式、E-SAF α 方式の順で

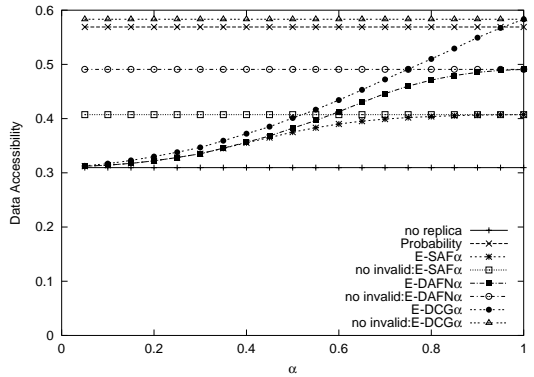


図 3 α とアクセス成功率
Fig. 3 α and data accessibility.

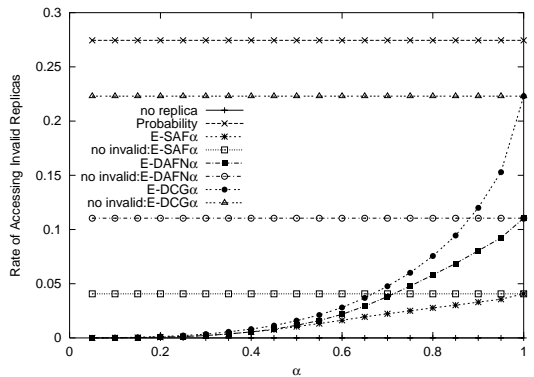


図 4 α と古い複製にアクセスした割合
Fig. 4 α and rate of accessing invalid replicas.

高いアクセス成功率を示すことが分かる。E-DCG α 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。また、比較方式は、複製の無効化を行わない場合の E-DCG α 方式に次いで、良い結果になっている。これは、比較方法では、ランダムに複製を配置するため、結果的に相互接続している移動体が多種類の複製を持つことになるからである。

図 4 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、すべ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分かる。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。一方、 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。複製を無効化

表 2 各データのアクセス成功率

Table 2 Data accessibility of each data item.

	E-SAF α	E-DAFN α	E-DCG α
D ₁	0.387 (-3.01)	0.437 (-2.02)	0.451 (-4.45)
D ₂	0.428 (7.27)	0.478 (7.17)	0.493 (4.45)
D ₃	0.428 (7.27)	0.478 (7.17)	0.492 (4.24)
D ₄	0.401 (0.5)	0.452 (1.35)	0.469 (-0.64)
D ₅	0.417 (4.51)	0.465 (4.26)	0.485 (2.76)
D ₆	0.414 (3.76)	0.461 (3.36)	0.48 (1.7)
D ₇	0.393 (-1.5)	0.446 (0)	0.465 (-1.49)
D ₈	0.385 (-3.51)	0.432 (-3.14)	0.453 (-4.03)
D ₉	0.438 (9.77)	0.489 (9.64)	0.51 (8.06)
D ₁₀	0.405 (1.5)	0.455 (2.02)	0.48 (1.7)
D ₁₁	0.371 (-7.02)	0.418 (-6.28)	0.443 (-6.15)
D ₁₂	0.391 (-2.01)	0.438 (-1.79)	0.462 (-2.12)
D ₁₃	0.397 (-0.5)	0.444 (-0.45)	0.469 (-0.64)
D ₁₄	0.39 (-2.26)	0.436 (-2.24)	0.462 (-2.12)
D ₁₅	0.383 (-4.01)	0.431 (-3.36)	0.455 (-3.61)
D ₁₆	0.374 (-6.27)	0.421 (-5.61)	0.444 (-5.94)
D ₁₇	0.397 (-0.5)	0.445 (-0.22)	0.471 (-0.22)
D ₁₈	0.403 (1)	0.451 (1.12)	0.475 (0.64)
D ₁₉	0.408 (2.26)	0.453 (1.57)	0.474 (0.43)
D ₂₀	0.39 (-2.26)	0.436 (-2.24)	0.462 (-2.12)
D ₂₁	0.422 (5.76)	0.47 (5.38)	0.493 (4.45)
D ₂₂	0.413 (3.51)	0.456 (2.24)	0.485 (2.76)
D ₂₃	0.417 (4.51)	0.463 (3.81)	0.493 (4.45)
D ₂₄	0.398 (-0.25)	0.449 (0.67)	0.478 (1.28)
D ₂₅	0.395 (-1)	0.441 (-1.12)	0.473 (0.22)
D ₂₆	0.398 (-0.25)	0.441 (-1.12)	0.473 (0.22)
D ₂₇	0.426 (6.77)	0.471 (5.61)	0.502 (6.36)
D ₂₈	0.416 (4.26)	0.46 (3.14)	0.487 (3.18)
D ₂₉	0.386 (-3.26)	0.435 (-2.47)	0.469 (-0.64)
D ₃₀	0.4 (0.25)	0.444 (-0.45)	0.477 (1.06)
D ₃₁	0.368 (-7.77)	0.411 (-7.85)	0.443 (-6.15)
D ₃₂	0.402 (0.75)	0.446 (0)	0.473 (0.22)
D ₃₃	0.426 (6.77)	0.471 (5.61)	0.502 (6.36)
D ₃₄	0.415 (4.01)	0.461 (3.36)	0.492 (4.24)
D ₃₅	0.38 (-4.76)	0.426 (-4.48)	0.461 (-2.34)
D ₃₆	0.393 (-1.5)	0.44 (-1.35)	0.473 (0.22)
D ₃₇	0.365 (-8.52)	0.414 (-7.17)	0.443 (-6.15)
D ₃₈	0.382 (-4.26)	0.427 (-4.26)	0.458 (-2.97)
D ₃₉	0.409 (2.51)	0.454 (1.79)	0.487 (3.18)
D ₄₀	0.36 (-9.77)	0.405 (-9.19)	0.438 (-7.21)
平均	0.399	0.446	0.472

表 3 各データの古い複製にアクセスした割合

Table 3 Rate of accessing invalid replicas of each data item.

	E-SAF α	E-DAFN α	E-DCG α
D ₁	0.026 (18.18)	0.045 (18.42)	0.048 (0)
D ₂	0.024 (9.09)	0.041 (7.89)	0.045 (-6.25)
D ₃	0.026 (18.18)	0.045 (18.42)	0.05 (4.17)
D ₄	0.025 (13.64)	0.041 (7.89)	0.045 (-6.25)
D ₅	0.024 (9.09)	0.04 (5.26)	0.048 (0)
D ₆	0.022 (0)	0.036 (-5.26)	0.041 (-14.58)
D ₇	0.025 (13.64)	0.043 (13.16)	0.052 (8.33)
D ₈	0.023 (4.55)	0.041 (7.89)	0.049 (2.08)
D ₉	0.022 (0)	0.037 (-2.63)	0.044 (-8.33)
D ₁₀	0.024 (9.09)	0.041 (7.89)	0.048 (0)
D ₁₁	0.021 (-4.55)	0.038 (0)	0.047 (-2.08)
D ₁₂	0.023 (4.55)	0.042 (10.53)	0.053 (10.42)
D ₁₃	0.021 (-4.55)	0.038 (0)	0.047 (-2.08)
D ₁₄	0.022 (0)	0.037 (-2.63)	0.047 (-4.17)
D ₁₅	0.022 (0)	0.038 (0)	0.047 (-2.08)
D ₁₆	0.023 (4.55)	0.039 (2.63)	0.047 (-2.08)
D ₁₇	0.02 (-9.09)	0.039 (2.63)	0.048 (0)
D ₁₈	0.023 (4.55)	0.039 (2.63)	0.047 (-2.08)
D ₁₉	0.023 (4.55)	0.037 (-2.63)	0.045 (-6.25)
D ₂₀	0.021 (-4.55)	0.037 (-2.63)	0.046 (-4.17)
D ₂₁	0.024 (9.09)	0.039 (2.63)	0.047 (-2.08)
D ₂₂	0.023 (4.55)	0.039 (2.63)	0.049 (2.08)
D ₂₃	0.021 (-4.55)	0.037 (-2.63)	0.046 (-4.17)
D ₂₄	0.02 (-9.09)	0.035 (-7.89)	0.047 (-2.08)
D ₂₅	0.023 (4.55)	0.038 (0)	0.05 (4.17)
D ₂₆	0.021 (-4.55)	0.035 (-7.89)	0.046 (-4.17)
D ₂₇	0.023 (4.55)	0.036 (-5.26)	0.048 (0)
D ₂₈	0.023 (4.55)	0.037 (-2.63)	0.047 (-2.08)
D ₂₉	0.021 (-4.55)	0.037 (-2.63)	0.049 (2.08)
D ₃₀	0.019 (-13.64)	0.032 (-15.79)	0.043 (-10.42)
D ₃₁	0.021 (-4.55)	0.036 (-5.26)	0.05 (4.17)
D ₃₂	0.021 (-4.55)	0.036 (-5.26)	0.047 (-2.08)
D ₃₃	0.024 (9.09)	0.04 (5.26)	0.051 (6.25)
D ₃₄	0.023 (4.55)	0.037 (-2.63)	0.051 (6.25)
D ₃₅	0.022 (0)	0.036 (-5.26)	0.048 (0)
D ₃₆	0.022 (0)	0.037 (-2.63)	0.049 (2.08)
D ₃₇	0.019 (-13.64)	0.036 (-5.26)	0.047 (-2.08)
D ₃₈	0.021 (-4.55)	0.038 (0)	0.05 (4.17)
D ₃₉	0.022 (0)	0.037 (-2.63)	0.05 (4.17)
D ₄₀	0.02 (-9.09)	0.035 (-7.89)	0.048 (0)
平均	0.022	0.038	0.048

する場合、複製配置方式について比較すると、つねに E-DCG α 方式が高い値を示し、E-DAFN α 方式、E-SAF α 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。また、比較方式は、どの提案方式よりも、高い値を示す。これは、各データのアクセス頻度のみを考慮して複製を配置するため、各移動体が更新発生後の無効な複製を多く配置してしまうからである。

本論文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、本論文の想定環境よりも、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大きくなる。比較方式は、アクセス成功率に関して、提案方式の中で最もアクセス成功率が高い E-DCG α 方式に次いで良い結果を示すが、古い複製にアクセスした割合に関しては、どの提案方式よりも高い値を示す。したがって、提案方式は、比較方式に比べて、古い複製にアクセスした割合を大幅に削減できる。特に、E-DCG α 方式は、すべての場合でアクセス成功率および古い複製にアクセスした割合の両方で、比較方式

よりも高い性能を示している。これは、後述する他のすべての評価結果についても同じことがいえる。

ここで、各複製配置方式における各データのアクセス成功率と古い複製にアクセスした割合を調べた結果を、表 2 と表 3 にそれぞれ示す。表中の括弧内の値は、平均からの増減の割合 (%) を表示したものである。

表 2 の結果より、すべての方式において、各データのアクセス成功率は、平均値から 10% 未満の増減のばらつきになることが分かる。ここで、統計上は厳密な意味はないが、目安として、各方式における各データのアクセス成功率のばらつきを平均を計算した。その結果、E-SAF α 方式で 3.78%、E-DAFN α 方式で 3.4%、E-DCG α 方式で 3.03% となり、E-DCG α 方式が最もばらつきが少ない。これは、E-SAF α 方式では、他の移動体の持つデータを考慮せずに、配置する複製を決定するため、相互接続している移動体間で同じような複製を配置してしまい、特定のデータのみへのアクセスが成功しやすい状況になるからである。一方、E-DCG 方式は、相互接続している移動体で複製の重複を解消するため、多種類のデータにアクセスでき、ばらつきが小さくなっている。

表 3 の結果より、すべての方式において、各データの古い複製にアクセスする割合は、平均から 20%

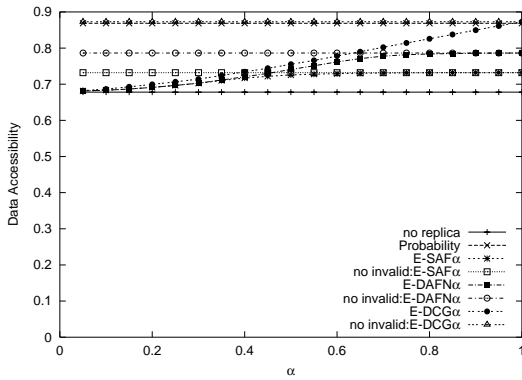


図 5 α とアクセス成功率 ($R = 10$)
Fig. 5 α and data accessibility ($R = 10$).

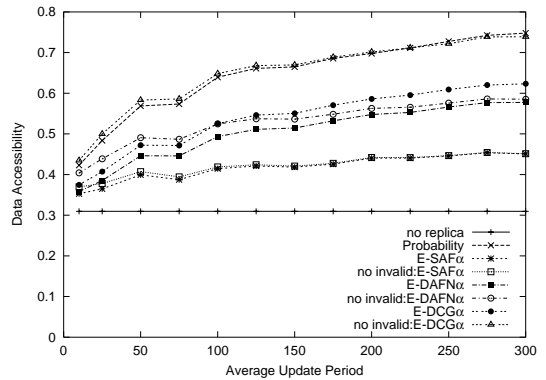


図 7 平均更新間隔とアクセス成功率
Fig. 7 Average update period and data accessibility.

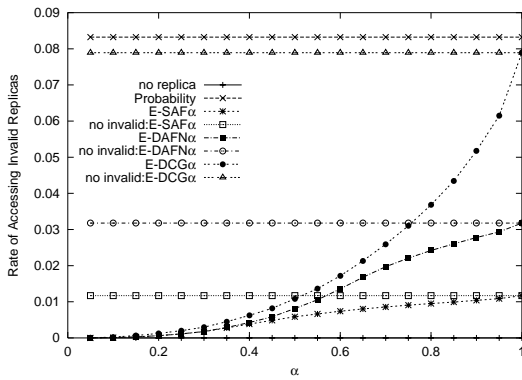


図 6 α と古い複製にアクセスした割合 ($R = 10$)
Fig. 6 α and rate of accessing invalid replicas ($R = 10$).

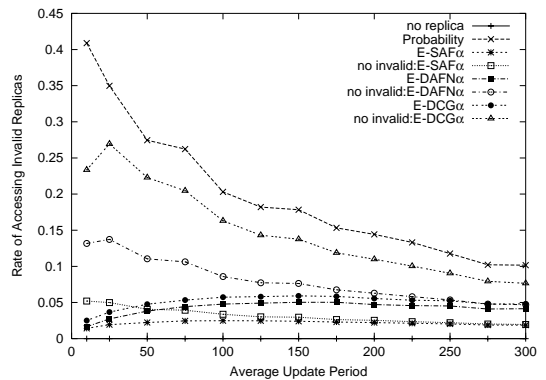


図 8 平均更新間隔と古い複製にアクセスした割合
Fig. 8 Average update period and rate of accessing invalid replicas.

未満の増減のばらつきになることが分かる．また、各方式において、各データの古い複製にアクセスした割合のばらつきの平均は、E-SAF α 方式で 6.25%、E-DAFN α 方式で 5.33%、E-DCG α 方式で 3.75% となり、E-DCG α 方式が最もばらつきが少ない．これは、上述の理由により、E-SAF α 方式、E-DAFN α 方式の順で複製の重複が大きいため、特定のデータに対するアクセスが成功する反面、それらが更新された後の古い複製にもアクセスしてしまうからである．

以上の結果より、すべての方式において、アクセス成功率や古い複製にアクセスした割合に関して、データごとで大きなばらつきが発生しないことが分かる．

また、各移動体の無線通信範囲を半径 10 の円とし、相互接続する移動体数が多く、アクセス成功率が十分に高い環境についても同様に評価した．その結果を図 5 と図 6 に示す．

図 5 および図 6 の結果は、図 3 および図 4 の結果と同じ特性を示している．したがって、アクセス成功

率が十分に高い環境でも、閾値 α の調整によって、アクセス成功率や古い複製にアクセスした割合を調整できる．

4.3 平均更新間隔の影響

次に、平均更新間隔の提案方式に対する影響を調べた．その結果を、図 7、図 8 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 7 ではアクセス成功率、図 8 では古い複製にアクセスした割合をそれぞれ示す．

図 7 の結果から、平均更新間隔が大きくなると、すべての場合においてアクセス成功率が高くなるのが分かる．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超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 8 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、すべ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、オリジナルを持つ移動体と相互接続している場合のみ、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、すべ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、各移動体の持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分かる。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。

4.4 キャッシュ領域の影響

次に、キャッシュ領域の提案方式に対する影響を調べた。その結果を、図 9、図 10 に示す。これらの図において、横軸はキャッシュ領域のサイズを示す。縦軸は、図 9 ではアクセス成功率、図 10 では古い複製にアクセスした割合をそれぞれ示す。

図 9 の結果から、すべての複製配置方式において、キャッシュ領域が大きくなると、アクセス成功率が高くなる事が分かる。提案方式では、E-DCG α 方式、E-DAFN α 方式、E-SAF α 方式の順にアクセス成功率が高いことが分かる。この理由は、図 3 の結果に対して述べたものと同様である。E-DCG α 方式は、アクセス成功率が緩やかに高くなり、ある値を超えると、ほぼ一定になる。これは、ある値を超えると、相互接続している移動体の持つデータの種類の数がほぼ一定になるからである。一方、E-SAF α 方式では、線形的に高くなる。これは、相互接続している移動体の持つ複製の重複が多いため、移動体自身の持つ複製の種類の数に大きく影響するからである。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くなる事が分かる。

図 10 の結果から、すべての複製配置方式において、キャッシュ領域が大きくなると、古い複製にアクセスした割合が高くなっている。これは、キャッシュ領域が大きくなると、有効な複製を多く配置する反面、更新発生後の無効な複製も多く配置してしまうからで

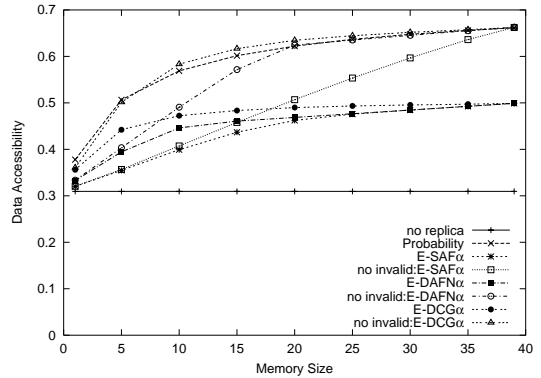


図 9 キャッシュ領域とアクセス成功率
Fig. 9 Memory size and data accessibility.

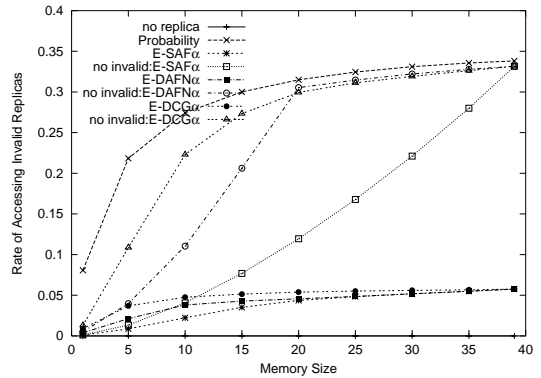


図 10 キャッシュ領域と古い複製にアクセスした割合
Fig. 10 Memory size and rate of accessing invalid replicas.

ある。また、ほとんどの場合で、E-DCG α 方式、E-DAFN α 方式、E-SAF α 方式の順に高い値を示している。これは、図 9 と同様の傾向である。また、キャッシュ領域が大きい場合、複製の無効化を行う場合と行わない場合を比べると、後者の方が非常に大きいことが分かる。これは、キャッシュ領域が大きくなるにつれて、ネットワーク上に無効な複製が多く残ってしまうからである。

4.5 最大移動速度の影響

最後に、最大移動速度の提案方式に対する影響を調べた。その結果を、図 11、図 12 に示す。これらの図において、横軸は最大移動速度を示す。縦軸は、図 11 ではアクセス成功率、図 12 では古い複製にアクセスした割合をそれぞれ示す。

図 11 の結果から、すべての複製配置方式において、最大移動速度が大きくなると、アクセス成功率が高くなる事が分かる。これは、最大移動速度が大きくなると、各移動体がネットワーク内に存在する多くの移

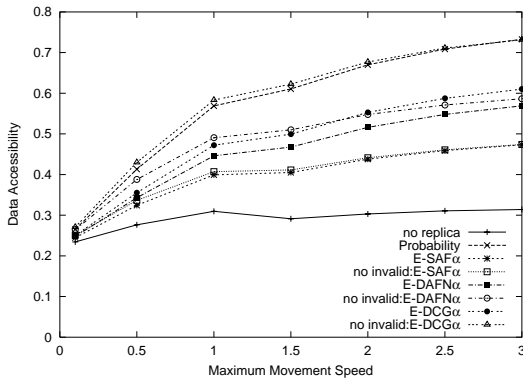


図 11 最大移動速度とアクセス成功率

Fig. 11 Maximum movement speed and data accessibility.

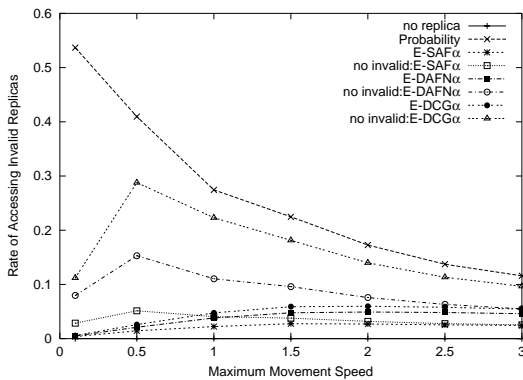


図 12 最大移動速度と古い複製にアクセスした割合

Fig. 12 Maximum movement speed and rate of accessing invalid replicas.

動体と接続する機会が増えるため、より新しいデータが伝播するからである。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、後者の方が高い値を示している。

図 12 の結果から、すべての複製配置方式において、最大移動速度が非常に小さい場合、古い複製にアクセスする割合が低いことが分かる。このような場合、ほとんどネットワークポロジが変化しないため、複製が再配置されても、相互接続している移動体の持つ複製が変わらない。そのため、アクセス対象のデータに暫定的なアクセスすらできず、即座に失敗するケースが多いからである。最大移動速度が大きくなると、すべての複製配置方式において古い複製にアクセスする割合が小さくなる。これは、最大移動速度が大きくなると、より新しいデータが伝播するからである。また、複製の無効化を行う場合と行わない場合を比べると、後者の方が高い値を示している。

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すべ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長や、オリジナルを持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、オリジナルを持つ移動体が、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、オリジナルを持つ移動体によって、他の移動体に知らされる。しかし、オリジナルを持つ移動体と相互接続していない移動体は、最新の $f_j(t)$ を知る事ができない。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まったく特性を持たずに発生する場合、オリジナルを持つ移動体でさえ $f_j(t)$ 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 1 つである。

5.3 データアクセスの成功、失敗

本論文では、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持つ複製にアクセスしなければ、失敗と見なされる。しかし、アプリケーションによっては、ユー

が必ずしも最新の情報にアクセスする必要がなく、多少古い情報にアクセスしても、そのユーザにとって有効な情報が得られる場合もある。たとえば、複数のユーザによる大規模なデータ収集作業の効率化を図るため、あるユーザが、他ユーザの収集状況に関する情報にアクセスする場合を考える。このとき、アクセスしたユーザは、最新の情報を得られず、多少古い情報を得た場合でも、他ユーザのこれまでのデータ収集状況から、ある程度の現在の収集状況を予測できる。その後、アクセスしたユーザが、他ユーザと異なる作業を行うことで、全体の作業の効率化を図れる。このようなアプリケーションでは、 α をある程度高めの値に調節して、アクセス成功率を向上させるアプローチも考えられる。

また、文献 24) では、接続性の高い移動体どうしてクラスタを形成し、クラスタ内では一貫性を厳密に保持し、クラスタ間では非一貫性を制限付で許容することで、トランザクションスループットを向上している。本研究でも、このようなアプローチを利用して、提案方式を拡張することにより、結果的にアクセス成功率の向上と一貫性保持のバランスをとることができる。

6. 関連研究

分散データベースの分野では、データの利用率を向上させるため、これまでにデータの複製に関する研究が行われてきた^{6),29),31),33)}。これらの研究は、データの複製を作成することによって、データの利用率を向上させる点で本研究のアプローチと類似している。しかし、固定ネットワーク上の分散データベースシステムでは、障害が頻繁に発生しないため、少数の複製の作成で十分な場合が多いのに対し、本研究で想定するアドホックネットワークでは、頻繁にネットワークの分断が生じるため、これらを考慮して複製を配置する点でまったく異なる。また、信頼性の低い固定ネットワーク上の複製管理方式もいくつか提案されている^{9),14)}。文献 9) では、固定ネットワーク上に存在するサイトに配置されたデータベースの複製の一貫性を保つために、更新データを配布する方法を提案している。一方、文献 14) では、ネットワーク内に存在するすべてのプロセスにメッセージを伝播させることを目的として、ゴシップ・プロトコルに基づく Pbcast を提案している。文献 9), 14) は、データ更新を考慮した複製の分散配置について議論している点で、本研究と類似している。しかし、文献 9), 14) の提案方式では、更新データを配布するのに対して、本論文の提案方式では、各データの更新間隔を考慮して移動体自らが古

いと予想される複製を無効化するため、アプローチが異なる。しかしながら、各移動体の持つ古い複製を最新の状態に保つことを目的として、文献 9), 14) のような更新データの配布を検討していくことは重要であり、今後の課題である。一方、ネットワークの状況を考慮した動的な複製配置方式もいくつか提案されている^{22),32)}。文献 22) では、クライアントがネットワークの性能を監視し、性能が低下すると、あらかじめ決められた端末に複製を配置して、ネットワークの負荷を分散させる方式を提案している。また、文献 32) では、通信トラフィックを削減するために、各ノードが受信したオブジェクトに対するリードとライトの発生比を考慮して、そのオブジェクトの複製を配置するか否かを定期的に決定する方式を提案している。この方式は、ネットワーク全体でリードの方が多い環境では、広範囲に複製が配置されるが、ライトの方が多い環境では、狭い範囲に複製が配置される。文献 22), 32) の方式は、ネットワーク内の状況を把握して、動的に複製を配置する点は、本論文の提案方式と同様のアプローチである。しかし、ネットワークが分断されない環境で複製配置を行う点で、本論文の提案方式とは根本的に異なる。

Web キャッシングの分野では、データアクセスの応答時間の短縮と Web サーバの負荷の軽減を目的とし、クライアントを含む小規模な組織内に存在する複数のプロキシサーバ間で協調的にインターネットオブジェクトをキャッシュする研究が行われてきた^{5),8),16),34)}。本研究では、相互接続している複数の移動体が複製を共有するため、これらの従来研究と類似点がある。さらに、文献 8), 16) では、Web サーバの持つオリジナルデータの更新を考慮し、プロキシサーバの持つキャッシュデータの有効期限を設定する。これは、本研究において、オリジナルデータの更新間隔を考慮し、各移動体の持つ複製にタイムアウト時間を設定する点に類似している。しかし、これらの研究では、固定ネットワーク上の安定なプロキシ間通信を想定しているため、キャッシュデータの配置先までは考慮していないのに対し、本研究では、頻繁にネットワークの接続と切断が生じる環境を想定し、複製の配置先まで考慮している点異なる。

モバイル環境におけるデータベースシステムにおいて、無線通信コストの削減を主な目的とし、データベースの複製を作成する手法もいくつか提案されている^{3),7),15),17),24)}。これらは、移動体が、固定ネットワーク上のデータベースサーバの保持するデータベースに対して、無線通信を用いて読み書きの操作を行う。

また、断線時の作業効率の低下を防ぐために、各移動体のキャッシュ領域にデータベースの複製を作成し、断線中はその複製に対してデータベース操作を行う。したがって、各移動体に複製を配置する点で本研究のアプローチと類似している。しかし、これらは、固定ネットワーク上に存在するサイトへの1ホップの無線通信を想定しているのに対し、本研究では、アドホックネットワークでのマルチホップな通信を想定している点で異なる。

サーバが放送チャネルを用いて繰り返しデータを放送するプッシュ型情報システムの研究分野では、データアクセスの応答時間を短縮するために、クライアントでのキャッシュ方式がいくつか提案されている^{1),2),10),30)}。これらは、主にクライアントが移動体であることを想定し、クライアントのキャッシュの置換えを、クライアントの各データアイテムに対するアクセス頻度、各データアイテムの放送頻度、各データアイテムが次に放送されるまでの残り時間などのパラメータに基づいて行う。これらは、移動体がデータをキャッシュする点で本研究のアプローチに類似している。しかし、複数の移動体間において、互いのキャッシュデータの共有を想定していない点で、本研究とは異なる。

アドホックネットワークに関する研究分野においても、データの利用率の向上を目的としたデータの複製配置に関する研究がいくつか報告されている^{19),28)}。文献 19) では、サーバの役割を担う移動体に複製を配置するだけでなく、データ更新を想定した複製の一貫性保持のために、分散データベースにおけるクォラム・システムに基づいた方式を提案している。サーバの役割を担う移動体は、あらかじめネットワーク上に存在するすべての移動体の中から複数個選択され、他の移動体からのデータ更新やデータ問合せの処理を行う。この方式は、移動体に複製を配置する点で、本研究の提案方式と類似している。しかし、サーバの役割を担う移動体にのみ複製を配置する点や、その移動体のキャッシュ領域に制限のない点で、本研究の提案方式と異なる。一方、本研究の提案方式では、ネットワーク上に存在するすべての移動体が限られたキャッシュ領域に複製を配置するため、大容量のキャッシュ領域を持つ移動体が存在しない環境でも、データの利用率を有効に向上できる。また、文献 28) では、同一データに対して複数の移動体がデータ更新を行うアドホックネットワークにおいて、新たな複製の一貫性を定義し、その一貫性を保持するために、更新データを配布する方式を提案している。この方式は、アドホックネットワーク上に存在するすべての移動体に複製を

配置する点で、本研究の提案方式と類似している。しかし、複製の一貫性保持のために、更新データを配布する点で、本研究の提案方式と異なる。本研究の提案方式では、各移動体が各データの更新間隔を考慮して、古いと予想される複製を移動体自らが無効化するため、文献 28) のように、更新データの配布に要する通信トラヒックが発生しない。したがって、本研究の提案方式は、文献 28) の方式と比較して、無線帯域が比較的狭い環境や移動体のバッテリーが乏しい環境でも適用できる。しかし、先述のように、更新データの配布は、複製を最新に保つために有効であるため、今後はこの方式のアプローチを導入することも検討している。

7. おわりに

本論文では、データ更新が不定期に発生するアドホックネットワークにおいて、アクセス成功率を向上させることを目的として、更新発生間隔を確率的に考慮した複製配置方式を提案した。また、この方式では、必要に応じて、古い複製にアクセスする割合を削減するために、複製の無効化を行う。さらに本論文では、提案方式に適した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、提案方式は、データの更新発生間隔とネットワークポロジを考慮することにより、効果的にアクセス成功率を向上させることができることを確認した。また、各データに閾値 α を設定して、複製を無効化することで、必要に応じて古い複製にアクセスする割合を削減できることを確認した。

これまでに筆者らは、文献 13) 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ている。今後は、文献 13) 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」、および文部科学省 21 世紀 COE プログラム(研究拠点形成費補助金)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) Acharya, S., Alonso, R., Franklin, M. and Zdonik, S.: Broadcast disks: Data management for asymmetric communication environments,

- Proc. ACM SIGMOD'95*, pp.199–210 (1995).
- 2) Acharya, S., Franklin, M. and Zdonik, S.: Prefetching from broadcast disks, *Proc. IEEE ICDE'96*, pp.276–285 (1996).
 - 3) Barbara, D. and Imielinski, T.: Sleepers and workholic: Caching strategies in mobile environments, *Proc. ACM SIGMOD'94*, pp.1–12 (1994).
 - 4) Broch, J., Maltz, D.A., Johnson, D.B., Hu, Y.C. and Jetcheva, J.: A performance comparison of multi-hop wireless ad hoc network routing protocols, *Proc. Mobicom'98*, pp.85–97 (1998).
 - 5) Bowman, C.M., Danzig, P.B., Hardy, D.R., Manber, U. and Schwartz, M.F.: The Harvest information discovery and access system, *Computer Networks and ISDN Systems*, Vol.28, No.1–2, pp.119–125 (1995).
 - 6) Carey, M.J. and Livny, M.: Distributed concurrency control performance: A study of algorithm, distribution, and replication, *Proc. VLDB'88*, pp.13–25 (1988).
 - 7) Cay, J., Tan, K.L. and Ooi, B.C.: On incremental cache coherency schemes in mobile computing environments, *Proc. IEEE ICDE'97*, pp.114–123 (1997).
 - 8) Chankhunthod, A., Danzig, P., Neerdaels, C., Schwartz, M.F. and Worrel, K.J.: A hierarchical Internet object cache, *Proc. USENIX technical conference'96*, pp.153–163 (1996).
 - 9) Demers, A.J., Greene, D.H., Hauser, C., Irish, W. and Larson, J.: Epidemic algorithms for replicated database maintenance, *ACM Proc. Symposium on Principles of Distributed Computing (PODC'87)*, pp.1–12 (1987).
 - 10) Grassi, V.: Prefetching policies for energy saving and latency reduction in a wireless broadcast data delivery system, *Proc. ACM MSWIS 2000*, pp.77–84 (2000).
 - 11) 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).
 - 12) 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).
 - 13) 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).
 - 14) Hayden, M. and Birman, K.: Probabilistic broadcast, Cornell University Technical Report, TR96-1606 (1996).
 - 15) Huang, Y., Sistla, P. and Wolfson, O.: Data replication for mobile computer, *Proc. ACM SIGMOD'94*, pp.13–24 (1994).
 - 16) Iyer, S., Rowstron, A. and Druschel, P.: Squirrel: A decentralized peer-to-peer web cache, *Proc. ACM PODC 2002*, pp.213–222 (2002).
 - 17) Jing, J., Elmagarmid, A., Helal, A. and Alonso, R.: Bit-sequences: An adaptive cache invalidation method in mobile client/server environments, *Mobile Networks and Applications*, Vol.2, No.2, pp.115–127 (1997).
 - 18) Johnson, D. B.: Routing in ad hoc networks of mobile hosts, *Proc. IEEE Workshop on Mobile Computing Systems and Applications*, pp.158–163 (1994).
 - 19) Karumanchi, G., Muralidharan, S. and Prakash, R.: Information dissemination in partitionable mobile ad hoc networks, *Proc. Symposium on Reliable Distributed Systems (SRDS'99)*, pp.4–13 (1999).
 - 20) Ko, Y.B. and Vaidya, N.H.: Location-aided routing (LAR) in mobile ad hoc networks, *Proc. Mobicom'98*, pp.66–75 (1998).
 - 21) Ng, M.J. and Lu, I.T.: 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).
 - 22) Noble, B., Fleis, B., Kim, M. and Zajkowski, J.: A case for fluid replication, *Proc. Network Storage Symposium (Netstore'99)* (1999).
 - 23) Park, V.D. and Corson, M.S.: A highly distributed routing algorithm for mobile wireless networks, *Proc. Infocom'97*, pp.1405–1413 (1997).
 - 24) Pitoura, E. and Bhargava, B.: Data consistency in intermittently connected distributed systems, *IEEE Trans. Knowledge and Data Engineering*, Vol.11, No.6, pp.896–915 (1999).
 - 25) Pearlman, M.R. and Haas, Z.J.: Determining the optimal configuration for the zone routing protocol, *IEEE Journal on Selected Areas in Communications*, Vol.17, No.8, pp.1395–1414 (1999).
 - 26) Perkins, C.E. and Bhagwat, P.: Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, *Proc. ACM SIGCOMM'94*, pp.234–244 (1994).
 - 27) Perkins, C.E. and Royer, E.M.: Ad hoc on demand distance vector routing, *Proc. IEEE WMCSA'99*, pp.90–100 (1999).
 - 28) Rothermel, K., Becker, C. and Hahner, J.: Consistent update diffusion in mobile

ad hoc networks, Technical Report 2002/04, Computer Science Department, University of Stuttgart (2002).

- 29) Stonebraker, M.: Concurrency control and consistency in multiple copies of data in distributed INGRES, *IEEE Trans. Softw. Eng.*, Vol.3, No.3, pp.188–194 (1979).
- 30) Su, C.J. and Tassiulas, L.: Joint broadcast scheduling and user's cache management for efficient information delivery, *Proc. Mobicom '98* pp.33–42 (1998).
- 31) Thomas, R.H.: A majority consensus approach to concurrency control for multiple copy databases, *ACM Trans.Database Systems*, Vol.4, No.2, pp.180–209 (1979).
- 32) Wolfson, O., Jajodia, S. and Huang, Y.: An adaptive data replication algorithm, *ACM Trans. Database Systems*, Vol.22, No.2, pp.255–314 (1997).
- 33) Wolfson, O. and Milo, A.: The multicast policy and its relationship to replicated data placement, *ACM Trans. Database Systems*, Vol.16, No.1, pp.181–205 (1991).
- 34) Wolman, A., Voelker, G.M., Sharma, N., Cardwell, N., Karlin, A.R. and Levy, H.M.: On the scale and performance of cooperative web proxy caching, *Proc. ACM SOSP '99*, pp.16–31 (1999).

(平成 15 年 3 月 26 日受付)

(平成 15 年 9 月 5 日採録)



林 秀樹

2002 年大阪大学工学部電子情報エネルギー工学科卒業。現在、同大学院情報科学研究科博士前期課程在学中。移動体計算環境におけるデータベースに興味を持つ。日本データ

ベース学会会員。



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 年より同大学院情報科学研究科マルチメディア工学専攻助手となり、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。データベースシステム、分散処理に興味を持つ。IEEE、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 年より同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を併任。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、ACM Trans. on Internet Technology, Data & Knowledge Engineering, Data Mining and Knowledge Discovery, The VLDB Journal 等の論文誌編集委員。情報処理学会フェローを含め、ACM, IEEE 等 9 学会の会員。