

柔軟な経路表に基づく二次元平面上の構造化オーバーレイ

北條 真史^{1,a)} 長尾 洋也^{1,b)} 宮尾 武裕^{1,†1,c)} 首藤 一幸^{1,d)}

受付日 2014年5月14日, 採録日 2014年11月10日

概要: センサが構成する無線メッシュネットワークなどにおいてメッセージ配送やデータ収集を行うためには、ノード群でオーバーレイネットワークを構築することが有望である。地理的な近接性を考慮したルーティングや範囲問合せを行うためには、ノード位置に基づいたオーバーレイネットワークの構築が必要となる。ノード位置を ID として用いる場合、ルーティングの方式には、ノード位置すなわち ID に偏りがあっても経路長を短く抑えることが求められる。我々は、この要件を満たし、かつ、構造化オーバーレイの設計手法である柔軟な経路表 (FRT) の特長を備える構造化オーバーレイを提案する。柔軟な経路表の特長とは、経路表サイズの動的な設定や高い拡張性を指す。提案手法は、既存手法の 1 つである P2P ドロネーネットワークをトポロジとして採用し、ホップ数の推定に基づいて遠隔ノードとのショートカットリンクを形成することによって経路長を短く抑える。

キーワード: Peer-to-Peer, オーバレイネットワーク, 位置に基づくオーバーレイ

A Structured Overlay on a Two-dimensional Space Based on Flexible Routing Tables

MASASHI HOJO^{1,a)} HIROYA NAGAO^{1,b)} TAKEHIRO MIYAO^{1,†1,c)} KAZUYUKI SHUDO^{1,d)}

Received: May 14, 2014, Accepted: November 10, 2014

Abstract: It is a promising way to construct an overlay network with sensor nodes for wireless mesh networks to perform message delivery and data collection from the nodes. It is necessary for an overlay to reflect nodes' locations to perform geometric routing and geometric search, for example node IDs on the overlay are nodes' location. Furthermore, a route on the overlay should be small, in other words, involve small number of hops even if the distribution of node locations and IDs are biased. A structured overlay presented in this paper fulfills the requirements and provides features of Flexible Routing Tables (FRT), a method for designing routing algorithms for structured overlays. Features of FRT are the dynamic routing table size and high extensibility. The proposed overlay adopts P2P Delaunay Network as its topology, and the method keeps route length short by forming shortcut links to distant nodes based on the estimated number of hops.

Keywords: Peer-to-Peer, overlay network, location-based overlay

1. はじめに

構造化オーバーレイとは、実ネットワーク上のノード群で

形成するアプリケーション層の仮想ネットワークである。構造化オーバーレイではノードに ID を割り当て、この ID に基づき数学的・論理的な構造を持ったネットワークが構成される。構造化オーバーレイ上のメッセージ配送は、指定された目的 ID に向けて、各ノードが転送を繰り返すことで行われる。転送の際、各ノードは隣接ノードの ID とアドレスを載せた表、つまり経路表を参照し、転送先ノードを決定する。flooding のようなメッセージの拡散が起きず、ノードの増加に対してメッセージの転送回数は線形より低いオーダーでしか増えない、つまり、スケーラビリティに優

¹ 東京工業大学
Tokyo Institute of Technology, Meguro, Tokyo 152-8552, Japan

^{†1} 現在, GMO インターネット株式会社
Presently with GMO Internet, Inc.

^{a)} hojo.m.aa@m.titech.ac.jp

^{b)} hiroya.nagao@is.titech.ac.jp

^{c)} takehiro.miyao@is.titech.ac.jp

^{d)} shudo@is.titech.ac.jp

れる。

ここで、ノード群に対して実空間に沿った ID を割り当てることで、位置に基づいたメッセージ配送が可能となる。具体的には、センサ群で無線メッシュネットワークを構築し、実空間で近接するセンサどうしの通信でマルチホップ方式のメッセージ転送を行う場合などが該当する。実空間と ID 空間を対応させ、位置に基づいたメッセージ配送を実現するものとして、Z-net [1] や SONAR [2] などの二次元平面上の構造化オーバーレイがいくつか提案されている。これらの手法は、二次元平面上で距離の近いノードと通信経路を持ちつつ、メッセージの転送回数（経路長）を短くするために遠隔ノードとのショートカットリンクを形成する。ショートカットリンクは、実空間でノード位置に偏りがあることを想定し、ID の偏りがある場合も適切に経路長を削減するように形成される。このような設計により、ノード位置を ID として用いることで実空間の近接性を考慮したルーティングや地理的な範囲検索などを実現している。

これら既存の構造化オーバーレイは、大規模かつ不安定なネットワークで動作するように設計されている。そのため全ノードが載った経路表を最新の状態に管理する維持することは難しく、経路表に載せるエントリ数を制限している。つまり、経路表サイズを定めている。適切な経路表サイズはノードの参加・離脱の頻度や通信の品質などによって変化するものであるため、状況に応じて変更できることが望ましい。また、これらの手法は経路長の短いルーティングを実現するが、最短経路内のノード間で通信遅延が大きかったり、経由するノードの信頼性が低かったりする場合など、選択した最短経路が他よりも非効率である可能性がある。これは、ルーティングに際して ID のみに従って経路表構築を行っているために起こるものである。

そこで我々は、これらの問題点を解決した二次元平面上の構造化オーバーレイを提案する。提案手法はドロネー図をトポロジとする P2P ドロネーネットワーク [3] を利用することで、任意のノード間の通信経路を確保する。そのうえで、メッセージの転送回数（経路長）を小さくするために、P2P ドロネーネットワーク上で遠く離れた遠隔ノードを適切に選んで経路表を構築する。経路表構築の際、ノード ID の偏りに対応するために自ノードと遠隔ノード間の P2P オーバレイネットワーク上でのホップ数を推定し、これを指標として用いている。

既存手法の問題点を解決するために、提案手法は柔軟な経路表 (Flexible Routing Table, FRT) [4] に基づいて設計されている。FRT は構造化オーバーレイの設計方法論であり、これに基づいて設計された構造化オーバーレイは、経路長を小さく保つような経路表構築を行いつつ、既存手法では実現できなかった様々な柔軟性を持つ。FRT によりもたらされる柔軟性の例として、ルーティング性能の向上に関する指標を追加する拡張や、ネットワークの状況などに

よって経路表サイズを変更する機能などがあげられる。また、経路表サイズが大きい、またはネットワークに参加するノードが少ない状況下では、経路表にすべてのノードを載せることで、目的ノードに直接到達するシングルホップ動作を実現することができる。

本稿では提案アルゴリズムの詳細を述べ、提案手法が経路長の削減をすることをシミュレーションによって示す。また、提案手法の FRT 由来の拡張性に関する考察や、実地での利用可能性についての検討も行う。

2. 関連研究

本章では、まず二次元の ID 空間を用いる構造化オーバーレイのアルゴリズムをいくつか説明する。次に、二次元 ID 空間上の構造化オーバーレイのなかでも提案手法がトポロジとして採用した P2P ドロネーネットワークの詳細を述べる。最後に、我々が提案手法の経路表構築を設計するにあたって利用した、構造化オーバーレイにおけるルーティングアルゴリズムの設計手法である FRT を説明する。

2.1 二次元平面上の構造化オーバーレイ

2.1.1 Z-net

Z-net [1] は、二次元、あるいはそれ以上の多次元 ID 空間上にオーバーレイネットワークを構成する手法の 1 つである。この手法では、空間充填曲線の 1 つである Z 曲線を用いて多次元空間を一次元のビット列 (Z-address) に変換する Z-Ordering を用いる。各ノードは自身の Z-address を含む領域を担当し、ノード群は担当領域の Z-address をキーとする Skip graph [5] に基づいてネットワークを形成する。

Z-Ordering による一次元 Z-address への変換は空間上の近接性を反映するという特徴を持ち、空間上の範囲検索も Z-address の区間指定による検索で実現できる。しかしながら、空間上で近接する地点が Z-address において連続しない場所があり、また範囲検索時に Z-address における検索区間が複数に分断され、問合せが煩雑になる場合があることが問題である。

2.1.2 SONAR

SONAR [2] は N 次元トラス上の構造化オーバーレイである CAN [6] を拡張した手法である。CAN は各ノードが矩形領域を管理し、隣接する矩形領域の管理ノードと接続を持つ構造化オーバーレイである。新たなノードの参加時には、このノードの位置をあらかじめ担当していたノードが矩形領域を分割して参加ノードに割り当てる。SONAR では、CAN を構築する各ノードが遠隔ノードとのショートカットリンクを形成することで、効率の良いルーティングを実現する。ショートカットリンクは、CAN のネットワークで各次元に 2^i ホップ離れたノードと形成する。

SONAR は CAN に基づいたネットワーク構築を行うため、Z-net の問題点の 1 つであった範囲検索の複雑性を解

決している。しかしノードの参加・離脱が繰り返されることで管理領域の形状が複雑なものとなるという欠点があり、ショートカットリンクの管理コストも大きいといった問題点もある。

2.2 P2P ドロネーネットワーク

P2P ドロネーネットワーク [3] は、二次元平面上のオーバーレイネットワークを構築する手法の1つである。これは計算幾何分野で知られるドロネー図を利用した構造化オーバーレイであり、ノード間の通信経路をドロネー図の構成に従って決定する。ドロネー図は、空間上に散布された点(母点)を特定の条件によって辺で結ぶことによって構成されるものであり、P2P ドロネーネットワークでは各母点をノード、各辺を通信経路と見なしている。ドロネー図の性質により、P2P ドロネーネットワークは以下の特徴を持つ。

- ドロネー図は連結グラフであるため、任意のノードから任意のノードへの通信経路が必ず存在する。
- ノードの管理領域を割り当てる際にボロノイ分割を利用することができる。
- ユークリッド距離に基づく greedy routing により、任意のノードから任意の目的地の担当ノードに到達することができる。
- 範囲検索時は、検索範囲に含まれるノードが隣接するノードに問合せを伝播するという簡単な方式で、検索範囲内に問合せを行きわたらせることができる。
- ノードの参加・離脱が繰り返されても各ノードの管理領域は複雑になることがないため、経路表に不整合が生じにくい。

このような特徴により、P2P ドロネーネットワークは二次元空間をノード群で管理する基盤ネットワークに適している。上記のように簡単な問合せの転送で実現する範囲検索は、Z-net の問題点であった範囲検索の複雑性を解消する。また、SONAR では空間の再分割が進むにつれて経路表エントリにも不整合が生じるため、一定時間ごとに確認・更新操作が必要だが、P2P ドロネーネットワークではノードの参加・離脱のたびにボロノイ分割による再分割を行い、経路表の不整合が起こることを防ぎ、経路表管理のコストを小さく保っている。これらの利点から、我々は提案手法のトポロジに P2P ドロネーネットワークを採用している。

図 1 は二次元平面上でのドロネー図とボロノイ分割の例である。ボロノイ分割とはドロネー図の各母点によるユークリッド距離に基づいた領域分割であり、任意の点が自身から最も近い母点に属するように分割されている。

2.2.1 自律分散生成アルゴリズム

二次元 ID 空間上のノード群が、P2P ドロネーネットワークを自律分散的に構築するアルゴリズムが提案されている [3]。この手法では、各ノードが自身の近傍のノードとの接続関係を自律的に計算することによって、全体で P2P

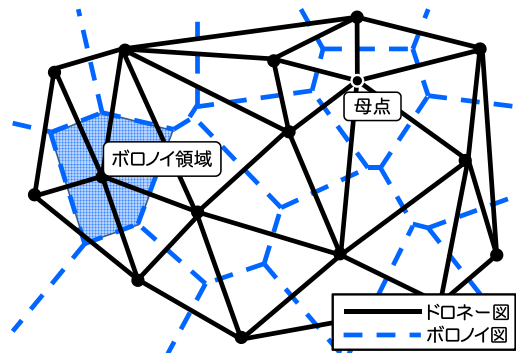


図 1 二次元平面上のドロネー図とボロノイ図

Fig. 1 Delaunay/Voronoi diagrams on a two-dimensional space.

ドロネーネットワークを構築することを可能としている。ノードの参加や離脱のネットワークに与える影響がそのノード近傍にとどまるため、ネットワークの再構築やノードの担当領域の再計算に関して高いスケーラビリティを期待できる。

2.2.2 遠隔接続経路

各ノードが自身の近傍ノードとの接続のみを持つ P2P ドロネーネットワークでは、遠隔ノードへのルーティング時に経路長が大きくなる。そこで、各ノードが遠隔接続経路 (Long Range Contacts, LRC) を構成し経路長を削減する手法が提案されている [7]。この手法では、各ノードが平面上の水平と垂直の 4 方向に関して、P2P ドロネーネットワークのホップ数に注目した LRC を形成する。ホップ数に注目した LRC は、ノードが一様に分布していない場合でも経路長の削減に効果的であり、ID に実空間の位置を用いる場合などでの利用が可能である。

2.3 柔軟な経路表 (FRT)

柔軟な経路表 (Flexible Routing Tables, FRT) は、構造化オーバーレイにおけるルーティングアルゴリズムの設計手法である。従来の構造化オーバーレイの多くがあらかじめ構築すべき経路表をノード ID の組合せとして厳密に定義し、これらの ID に最も近いノードを発見することで経路表構築を行っていたのに対して、FRT はまったく異なった経路表構築の方法を提案している。すなわち、経路表間の優劣を表す順序関係 $\leq_{ID}$ を定義し、これに従って経路表の改良を繰り返すというものである。構築すべき経路表を厳密に定義し、経路表に追加するノード候補を限定してしまうアルゴリズムに比べて、FRT に基づいて設計されたアルゴリズムは、2.3.1 項であげる様々な柔軟性を備える。

2.3.1 FRT に基づく設計の利点

FRT に基づいて設計されたアルゴリズムには、たとえば以下の利点があげられる。

- 経路表サイズ L の動的な調整が可能である。 L はノードの性能やネットワークの安定性などに応じて変更さ

れるものである。

- シングルホップ動作とマルチホップ動作を一貫して扱うアルゴリズム設計が可能となる。ネットワークに参加するノード数 N が $N \leq L$ を満たす際には経路表に全ノードを載せ、シングルホップ動作を実現する。 $N > L$ となる場合はマルチホップ動作となり、経路表に載せるノードを選択しながら性能向上を図る。たとえば、FRT-2-Chord [8] は、FRT に基づく設計により、一次元の環状 ID 空間でシングル/マルチホップ動作を一貫して扱うことのできる構造化オーバーレイである。
- 様々な指標を考慮するための拡張性を備える。例として、ノード ID に加えて通信遅延を考慮した Proximity-aware Flexible Routing Tables (PFRT) [9] や、グループ間通信の削減を考慮した Grouped Flexible Routing Tables (GFRT) [4] が提案されている。

2.3.2 経路表管理操作

FRT では、経路表を構築・改良していく際に次に述べる2つの操作をとる。

- エントリ学習操作 エントリ学習操作は、経路表にエントリを追加する操作である。この操作によって、新たに得たノード情報は選別されることなく経路表に追加される。追加するノード情報は様々な通信の結果から得るものであり、また、適当な ID に向けて能動的に探索することによって得ることも可能である。
- エントリ厳選操作 エントリ厳選操作は、経路表エントリ数があらかじめ設定されたサイズ L を超えたときに経路表エントリを1つ削除する操作である。2.3.3 項に述べる経路表間の順序関係に従って、最も優れた経路表となるように削除するエントリを決定する。

FRT における経路表構築では、経路表エントリが最大サイズ L を超えるまでは学習操作が繰り返される。 L を超えるようになった際には学習操作とともに厳選操作が実行され、経路表の改良が行われる。

2.3.3 経路表間の順序関係 $\leq_{ID}$

経路表間の順序関係 $\leq_{ID}$ とは、経路表に含まれるノード ID の組合せに注目し、その優劣を判定するために定義するものである。 $\leq_{ID}$ を用いることによって、2つの経路表 E と F について $E \leq_{ID} F$ が得られたとき、 E は F よりも優れた経路表であると判断できる。FRT に基づく構造化オーバーレイの設計者は、採用するオーバーレイネットワークのトポロジや距離などの指標から適切に経路表間の順序関係 $\leq_{ID}$ を定義することで、2.3.1 項で述べた FRT の利点を保ちながら構造化オーバーレイを設計することができる。

FRT に基づくアルゴリズムは、このような順序関係に従って経路表の改良が繰り返されることでルーティング効率を向上させていく。

2.3.4 FRT に基づく構造化オーバーレイ

FRT に基づく構造化オーバーレイとして、これまでに一次元

表 1 経路表の構成要素

Table 1 Parts of a routing table.

構成要素	サイズ
ドロネー隣接ノードセット D	期待値 6 以下
ショートカットリンクリスト S	L (変更可能)

の環を ID 空間とする FRT-Chord [4] や FRT-2-Chord [8], FRT-Chord# [10] などが提案されてきた。これらのうち FRT-Chord# はノード ID の偏りに対応した構造化オーバーレイであり、Chord の用いる ID 空間である環上でノード間のホップ数を推定することによって経路表構築を行っていると考えられる。提案手法では FRT-Chord# の動作を参考に、二次元平面でホップ数を推定する方法を考案している。

3. 提案手法

我々は、既存手法にはない柔軟性を備えた、二次元平面上の構造化オーバーレイを提案する。提案手法は、P2P ドロネーネットワークによって任意のノード間の到達性保証と各ノードの管理領域の決定を行ったうえで、FRT に基づいた経路表構築を行い、ノード ID に偏りがある場合でも適切に経路長を削減する。

経路表構築の際、ノード間の ID 距離を指標として用いるとノード ID の偏りがある場合に適切に経路長を削減することができない。そのため提案手法は、経路表構築に用いる指標としてノード間のホップ数を採用する。ノード間のホップ数は、ID 距離と異なり2ノードの ID からただちに計算することができないため、後述する手法で間接的な指標によって推定する。各ノードはこの推定によって自身の経路表エントリの組合せを評価し、FRT に基づいて経路表の改良を行う。

FRT に基づいた設計により、提案手法は経路長を短く保ちつつ、P2P ドロネーネットワークなどの既存手法にはない様々な指標を導入する拡張性を持つ。たとえば位置情報を用いる際に現れる特有の状況として、直接通信を行えるノードに距離の制約がある場合が考えられる。このような状況下では、経路表に載せるノードを自ノードからの距離で制限することで対応することができ、この制約に従いつつ適切な経路表構築を行うことができる。

3.1 経路表構築

提案手法の構築する経路表は表 1 で示す構造をとる。以下でこれらの構築方法について述べる。

3.1.1 ドロネー隣接ノードセット

提案手法は P2P ドロネーネットワークをトポロジに採用しているため、このネットワーク上で通信経路を持つノードをドロネー隣接ノードセット D で管理する。ドロネー図の特性により母点の次数は期待値が6以下となるので、

この経路表のサイズに関しても同様に期待値が6以下となる。ドロネー隣接ノードの発見および管理は、2.2.1 項で述べた P2P ドロネーネットワークの自律分散生成アルゴリズム [3] に基づいている。

3.1.2 ショートカットリンクリスト

P2P ドロネーネットワーク上の通信経路のみを用いたルーティングでは、遠く離れたノード間で経路長が長くなってしまうため、これを削減するために保持する遠隔ノードの情報をショートカットリンクリスト S で管理する。

ショートカットリンクを形成するノードを選択する際、経路長を削減するために何らかの指標を導入する必要がある。ノードが一様に分布する状況では、ノード間の距離を指標とすることで適切に経路長を削減できるが、ノード分布に偏りがある状況において、この方式では経路長が長くなる。これは、自ノードからある距離だけ離れたノードを選択してショートカットリンクを形成するという方針により、ショートカットリンクがノード密度の大きいところに形成されなかったり、ノード密度の小さいところに必要以上に多く形成されてしまったりすることが原因である。これに対し、ノード分布に偏りがあっても経路長を削減するために、オーバーレイネットワークにおけるホップ数を指標とする方法がある。ホップ数は、距離と異なりノード分布の偏りに対して普遍的指標である。あるホップ数離れたノードを選ぶという方針により、ノードの密度に対応してショートカットリンクを形成することができる。このような事実から、提案手法ではノード分布の偏りにかかわらず経路長を削減するための指標として、P2P ドロネーネットワークにおけるホップ数を導入する。

自律分散環境において、任意のノード間のホップ数を計測することは通信コストが大きい。そのため提案手法では、自ノードと経路表エントリ間のホップ数の推定を用いて経路表構築を行う。この推定は、各経路表エントリが自ノードからホップ数に関してどれほど離れているかという点に注目して行っていく。各エントリが自ノードからホップ数に関してどれほど離れているかを推測するために、経路表の中で各エントリをホップ数順に並べることを考える。そのために、まず経路表内でのグラフ距離 $h_s(e)$ を定義する。

定義 3.1 (経路表内でのグラフ距離 $h_s(e)$) 経路表内でのグラフ距離 $h_s(e)$ とは、自ノード s と経路表エントリによって構成したドロネー図上での、 s, e 間の最短経路における辺数である。

次にこれを用いて経路表エントリの順位 $r_s(e)$ を次のように定義する。

定義 3.2 (経路表エントリの順位 $r_s(e)$)

$$r_s(e) = \#\{e^* \in D \cup S | h_s(e^*) < h_s(e)\} \quad (1)$$

ここで定義した経路表エントリ e の順位 $r_s(e)$ は、経路表の全エントリを $h_s(e)$ に関する昇順に並べたときの順位と

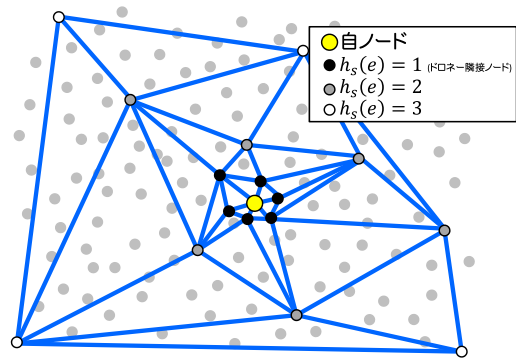


図 2 二次元平面上の経路表エントリ

Fig. 2 Routing table entries on a two-dimensional space.

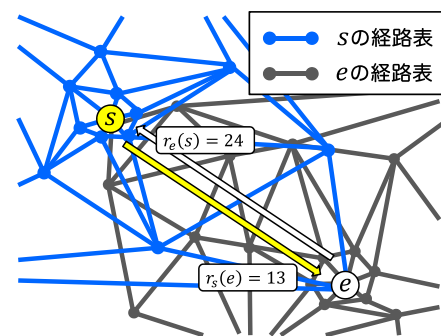


図 3 経路表エントリの順位差

Fig. 3 A difference in orders of a routing table entry.

見なすことができる。

図 2 は経路表エントリでドロネー図を構成した例である。このとき $h_s(e) = 1$ となる 6 つのノードは D に載るドロネー隣接ノードであり、 $r_s(e) = 1$ である。 $h_s(e) = 2$ となる 6 つの経路表エントリに関しては $r_s(e) = 7$ 、 $h_s(e) = 3$ となる 4 つのエントリは $r_s(e) = 13$ と計算される。順位が大きいエントリは、P2P ドロネーネットワーク上で自ノードからのホップ数が大きいと考えることができる。提案手法はこの順位に従ってショートカットリンクを形成するノードを選んで経路表を構築する。

3.2 経路表間の順序関係 $\leq_{ID}$

提案手法では、経路表間の順序関係 $\leq_{ID}$ を定義するために、自ノード s とエントリ e に関して計算した経路表エントリの順位差 $f(e)$ を用いている。

定義 3.3 (経路表エントリの順位差 $f(e)$)

$$f(e) = |r_s(e) - r_e(s)| \quad (2)$$

$f(e)$ は、 s の経路表内での e の位置と e の経路表内での s の位置の誤差を表すものである。この誤差は、両ノードが独立して自律的に経路表構築を行っているため、 s の経路表に基づいて推測したホップ数と e の経路表に基づくそれが異なることによって生じる。

たとえば図 3 のような状況を考える。これは自ノード s と経路表エントリ e の順位差に関する概念図である。図

の状況では、経路表内でのグラフ距離に関して $h_s(e) = 3$, $h_e(s) = 4$, 経路表内の順位に関して $r_s(e) = 13$, $r_e(s) = 24$ となっている. このとき, $f(e) = |r_s(e) - r_e(s)| = 11$ と計算され, s - e 間のホップ数の推測が両ノードで異なっており, s の推測するホップ数が e のそれよりも小さくなっているということが分かる. 自ノード s は, このように $f(e)$ を計算することで, 適切にホップ数を推測してショートカットリンクを形成しているかを e との間で確認することができる. もしも図 3 のように s と e の間でホップ数の推測に誤差が出ている場合には, e へのショートカットリンクは適切でないと判断し, 経路表の改良が繰り返される過程で削除する.

以上の考え方から, 提案手法は各エントリの $f(e)$ が 0 に近いほど優れた経路表であると定義する. このとき経路表間の順序関係 $\leq_{ID}$ は, 経路表 E 内のエントリ e に関して $f(e)$ を降順に並べた列 $f(E) = \{f(e_i)\}$ を用いて次のように定義される.

定義 3.4 (提案手法の経路表間順序関係 $\leq_{ID}$)

$$E \leq_{ID} F \Leftrightarrow f(E) \leq_{dic} f(F) \quad (3)$$

ここで $\leq_{dic}$ は, 2 つの降順列の間で次のように定義される辞書式順序関係である.

定義 3.5 (辞書式順序関係 $\leq_{dic}$)

$$\begin{aligned} \{a_i\} <_{dic} \{b_i\} &\Leftrightarrow a_k < b_k \quad (k = \min\{i \mid a_i \neq b_i\}) \\ \{a_i\} =_{dic} \{b_i\} &\Leftrightarrow a_i = b_i \\ \{a_i\} \leq_{dic} \{b_i\} &\Leftrightarrow (a_i <_{dic} b_i) \vee \{a_i\} =_{dic} \{b_i\} \end{aligned} \quad (4)$$

3.3 FRT に基づく経路表の改良

提案手法のショートカットリンクリストは FRT に基づいて構築する設計となっているため, 2.3.2 項で述べた 2 つの操作を繰り返すことで改良されていく.

3.3.1 エントリ学習操作

提案手法の経路表では, 通信の結果から自動的に得られるノードの ID と物理ネットワークのアドレスのほかに, $r_e(s)$ の値も管理する. そのため, 種々の通信の結果から学習操作を行う場合, $r_e(s)$ を把握するための通信をとまなう. 適当な ID に向けて探索を行い, その ID を担当するノードの情報を要求するという方式では, あらかじめ $r_e(s)$ も含めた情報を要求することでこの通信が不要となる.

3.3.2 エントリ厳選操作

エントリ厳選操作では, 3.2 節で定義した経路表間の順序関係 $\leq_{ID}$ に従って, 経路表から以下を満たすエントリ e^* を削除する.

$$S \setminus \{e^*\} \leq_{ID} S \setminus \{e\}, \quad e \in S \quad (5)$$

4. 評価

提案手法のルーティング性能を検証するために, オー

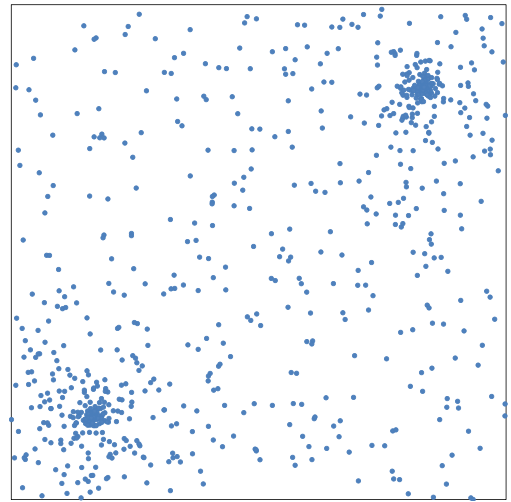


図 4 Zipf 分布 ($\alpha = 0.7$) を用いた際のノード配置
Fig. 4 Node's layout at Zipf distribution ($\alpha = 0.7$).

バレイネットワーク構築ツールキットである Overlay Weaver [11], [12] 上にこれを実装し, シミュレーションにより評価実験を行った. 実験環境は以下のとおりである.

- Overlay Weaver 0.10.4
- OS 64 bit Windows 7
- CPU Intel Core i7-3612QM 2.10 GHz
- メモリ 8.00 GB
- Java SE 7 Update 21

ノード ID の偏りに対する挙動を確かめるため, ノード群への ID の割当てとして次のものを用意した.

- (1) 全ノードを平面上に一樣に分布させたもの
- (2) 全ノードのうち 2 割を平面上に一樣に分布させ, 残りの 8 割のノードを平面に 2 地点を集中点とし, これらの周りに距離が Zipf 分布 ($\alpha = 0.7, 1.0$) に従うように残りの 8 割のノードを分布させたもの

なお, Zipf 分布は分布関数 $f(x; \alpha) = Cx^{-\alpha}$ で定義される分布である. 用意した 3 種類の分布は, 一樣分布, Zipf 分布 ($\alpha = 0.7$), Zipf 分布 ($\alpha = 1.0$) の順に偏りが大きくなる. 図 4 に, Zipf 分布 ($\alpha = 0.7$) の例を示す.

本実験のシナリオは以下である. なお, 各ノードは能動的に探索を行う方式で得たノード情報のみを用いてエントリ学習操作を行う.

- (1) 全ノードをネットワークに参加させ, P2P ドロネーネットワークの構築を完了する.
- (2) 各ノードが, 探索先 ID がノード ID と同様の分布に従う探索を繰り返し, ショートカットリンクリストを構築する.
- (3) 各ノードが指定回数の探索を終えた時点で, 全ノードについて探索時の経路長を測定する.

4.1 他アルゴリズムとの比較

4.1.1 比較対象のアルゴリズム

提案手法のほかに、比較対象として2種類のアロリズムを用意した。いずれも提案手法と同様に学習操作と厳選操作をもって経路表構築を行うが、その厳選操作の方式が異なる。

ランダムに経路表構築を行う方式

この方式は厳選操作の際に指標をまったく用いず、ランダムに選んだエントリを削除するものである（以後、ランダム手法と呼ぶ）。

距離を指標とするアルゴリズム

この方式は、ノードが二次元平面上に一樣に分布する仮定の下で、ノード間の距離を指標として経路表構築を行うものである（以後、距離ベース手法と呼ぶ）。

距離ベース手法では、優れた経路表を定義するにあたって自ノード s と各エントリ e のユークリッド距離 $d(s, e)$ に注目する。エントリを自ノードからの距離で昇順に並べた経路表 $E = \{e_i\}$ に関して、エントリの分布が次の確率密度に従うものを優れた経路表と定義する。

$$f(e_i) = C \frac{1}{d(s, e_i)} \quad (6)$$

ここで C は確率の総和を1にするための定数である。各エントリが自ノードからの距離の逆数に比例する確率で経路表に存在することにより、距離ベース手法は、ネットワーク参加ノード数が N 、経路表サイズが L のとき、経路長を $O(\log^2 N / \log L)$ とすることができる。この性質は Small-World 現象に関する研究 [13] を参考にしている。

距離ベース手法は、ノードが一樣に分布する状況下で効率の良いルーティングを行う方式であるため、ID に偏りがある場合、経路長を短く保つことが難しいと考えられる。

4.1.2 実験結果

提案手法と、これのほかに用意したランダム手法、距離ベース手法の3つの手法についてそれぞれ同様の実験を行い、各ノードの探索が100回となったところで平均経路長および99パーセンタイルを計測した。実験時のパラメータはノード数を $N = 1000$ 、経路表サイズの最大値を $L = 20$ とした。

図5、図6、図7がこの実験の結果である。まず、いかなる指標も用いないランダム手法と指標を用いる他の2手法の違いに注目する。いずれの結果もランダム手法は経路長が極端に長くなっており、FRTに基づいて経路表構築を行う他の2手法は経路長が短くなっている。このことから、FRTに基づく経路表構築が経路長削減に有効であることが分かる。

次に提案手法と距離ベース手法の2つに注目し、ID分布の偏りに対する挙動について考察する。距離ベース手法は偏りが大きくなるにつれて平均経路長が長くなっているのに対して、提案手法はノードの偏りが大きくなっても平

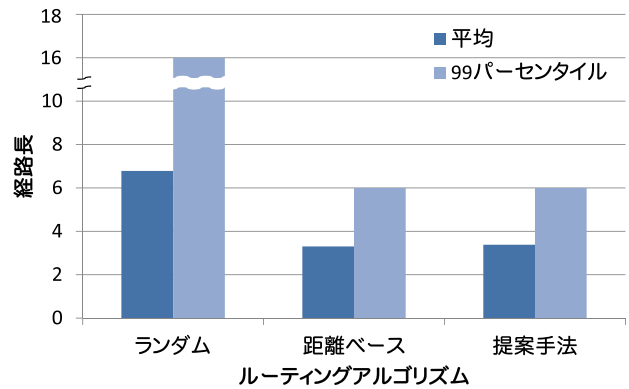


図5 一樣分布における経路長

Fig. 5 Route lengths at uniform distribution.

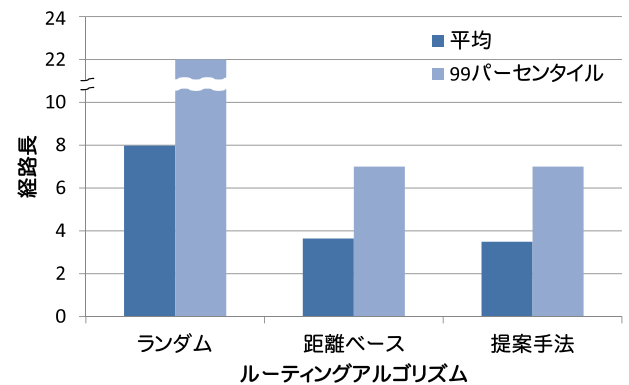


図6 Zipf 分布 ($\alpha = 0.7$) における経路長

Fig. 6 Route lengths at Zipf distribution ($\alpha = 0.7$).

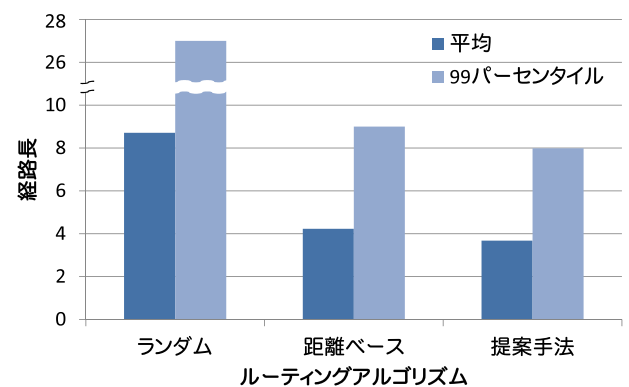


図7 Zipf 分布 ($\alpha = 1.0$) における経路長

Fig. 7 Route lengths at Zipf distribution ($\alpha = 1.0$).

均経路長の増加は微小である。一樣分布に対して Zipf 分布 ($\alpha = 1.0$) での平均経路長の増加は、距離ベース手法が約28%、提案手法が約9%であった。99パーセンタイルに関しては、一樣分布と Zipf 分布 ($\alpha = 0.7$) のものでは距離ベースと提案手法で差が出なかったが、偏りの大きい Zipf 分布 ($\alpha = 1.0$) のもので提案手法のほうが小さい値となった。

これらの結果から、提案手法がノードIDの偏りに対応していることが分かる。また、一樣分布では提案手法、距離ベース手法ともに同等の結果である。距離ベース手法が一

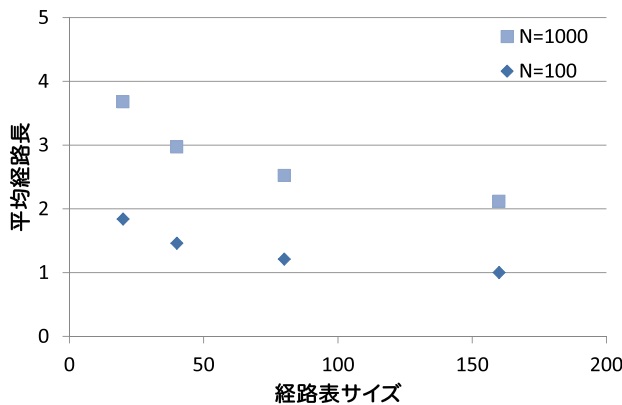


図 8 経路表サイズと平均経路長

Fig. 8 Routing table sizes and route lengths.

様分布の際に有効なアルゴリズムであることから、提案手法は一様分布においても経路長を削減することが分かる。

4.2 経路表サイズの変更

提案手法の FRT 由来の特徴の 1 つに経路表サイズの変更がある。我々は経路表サイズを段階的に変更して実験を行い、これにともなう平均経路長の変化を観察した。ノード ID の分布は $\alpha = 1.0$ の Zipf 分布のものをを用い、実験時のパラメータはノード数を $N = 100, 1000$ の 2 種類、経路表サイズの最大値を $L = 20, 40, 80, 160$ の 4 段階とした。いずれの実験も、各ノードの探索回数が 400 回となったところで経路長を計測した。

図 8 がこの実験の結果である。2 種類のノード数のいずれの結果も、経路表サイズが大きくなるにつれて、平均経路長が短くなっていることが分かる。

FRT に基づくアルゴリズムの特徴の 1 つに、ネットワークに参加する全ノード数よりも経路表サイズが大きいとき、経路表にすべてのノードを載せ、シングルホップ動作を実現するというものがある。実験結果の $N = 100, L = 160$ のものに注目すると、平均経路長は 1.002 であった。これはほとんどの探索がシングルホップによって実現されていることを表す。この実験結果から、提案手法が FRT 由来の特徴の 1 つである、シングルホップ動作とマルチホップ動作の一貫した管理が実現されていることが分かった。

4.3 経路表の改良回数

4.1 節および 4.2 節の実験で、 $N = 1000, L = 20$ のものについて探索回数 100 回と 400 回のそれぞれについて結果を得たが、これらに平均経路長の変化はほとんど見られなかった。このことから、探索回数 100 回の時点で十分に改良がなされた経路表が構築できていることが推測できる。

4.2 節におけるその他のパラメータにおける結果は、いずれも探索回数 400 回よりも多くの探索を行っても平均経路長はほとんど短くならなかった。そのため、いずれのパラメータでも探索回数 400 回あるいはより少ない回数で、

十分に改良がなされた経路表が得られると考えられる。

一般のノード数と経路表サイズにおいて、経路表の改良に必要な探索回数に関する調査は今後の課題である。

5. ノードの離脱

本稿の実験ではノードがネットワークから離脱する状況を扱っていない。以下、ドロネー隣接ノードセットとショートカットリンクリストのそれぞれについて、ノードの離脱を扱う方法を考察する。

ドロネー隣接ノードセットは、これを管理する P2P ドロネーネットワークの手法により、ノードの離脱時にネットワーク全体の到達性を保ったまま適切に再構築できる。しかし、短時間に多くのノードが離脱したり、空間上局所的にいくつかのノードが離脱したりする場合には、ネットワークが分断される可能性がある。分断耐性を向上させるためには、ドロネー隣接ノードセットに 2, 3 ホップ離れたノードを保持しておく手法が考えられる。

ショートカットリンクリストは、通信できなくなったノードを発見し次第削除することで正しく改良が進められる。通信できなくなったノードを検知するためには、各エントリに対して定期的に生存確認をする必要があるが、各エントリの $r_e(s)$ を最新のものに更新するためのメッセージ通信と合わせて行うことができる。

6. まとめと今後の課題

我々はノード ID に偏りがある状況に対応する、二次元平面上の構造化オーバーレイを提案した。提案手法はノード位置を ID として用いることができるため、ノード群が実空間の地理に基づいた情報管理を行う場合などの応用がある。また、提案手法により構築したオーバーレイネットワーク上では、実空間の近接性に対応した経路選択や地理的な範囲問合せを実現することも容易である。

提案手法は FRT に基づいた構造化オーバーレイであるため、経路長を短く保ちつつ様々な指標を導入する拡張性や経路表サイズの動的な設定などの特徴を継承する。

実験により、提案手法がノード ID に偏りがある場合にも経路長を短く保ってルーティングを行うことを確認した。また、ノード ID に偏りがなく一様に分布する場合に、距離に基づいて経路表を構築する手法 (4.1.1 項) と同程度のルーティング性能を持つことを確認した。

今後の課題として、提案手法のルーティング効率について論理的、数学的な説明を与えることや、より良いルーティング効率を達成する新たな手法の考案がある。また、十分なルーティング効率を達成する経路表を得るための改良回数について、一般のノード数や経路表サイズでどのようなかを考察する必要がある。今後、FRT 由来の特徴を活かし、位置情報を用いる際の様々な制約に対応したアルゴリズムを実装し、その評価を行っていく。

謝辞 本研究は JSPS 科研費 25700008, 26540161 の助成を受けたものです。

参考文献

- [1] Shu, Y., Ooi, B.C., Tan, K.-L. and Zhou, A.: Supporting Multi-Dimensional Range Queries in Peer-to-Peer Systems, *Proc. IEEE P2P 2005*, pp.173-180, IEEE (2005).
- [2] Schütt, T., Schintke, F. and Reinefeld, A.: A Structured Overlay for Multi-dimensional Range Queries, *Proc. Euro-Par 2007*, pp.503-513, Springer (2007).
- [3] 大西真晶, 源元佑太, 江口隆之, 加藤宏章, 西出 亮, 上島紳一: ノード位置を用いた P2P モデルのためのドロネー図の自律分散生成アルゴリズム, 情報処理学会論文誌データベース (TOD), Vol.47, No.SIG4 (TOD29), pp.51-64 (2006).
- [4] Nagao, H. and Shudo, K.: Flexible Routing Tables: Designing Routing Algorithms for Overlays Based on a Total Order on a Routing Table Set, *Proc. IEEE P2P 2011*, pp.72-81, IEEE (2011).
- [5] Aspnes, J. and Shah, G.: Skip Graphs, *ACM Trans. Algorithms*, Vol.3, No.4, pp.1-25 (2007).
- [6] Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Shenker, S.: A Scalable Content-Addressable Network, *Proc. SIGCOMM 2001*, pp.161-172, ACM (2001).
- [7] 大西真晶, 坪井新治, 平山雅夫, 江口隆之, 上島紳一: P2P ドロネーネットワークにおける遠隔接続経路の自律分散生成法, 情報処理学会論文誌データベース (TOD), Vol.48, No.SIG11 (TOD34), pp.190-214 (2007).
- [8] Ando, Y., Nagao, H., Miyao, T. and Shudo, K.: FRT-2-Chord: A DHT Supporting Seamless Transition between On-hop and Multi-hop Lookups with Symmetric Routing Table, *Proc. ICOIN 2014*, pp.170-175, IEEE (2014).
- [9] Miyao, T., Nagao, H. and Shudo, K.: A Method for Designing Proximity-aware Routing Algorithms for Structured Overlays, *Proc. IEEE ISCC 2013*, pp.508-514, IEEE (2013).
- [10] Miyao, T., Nagao, H. and Shudo, K.: A Structured Overlay for Non-uniform Node Identifier Distribution Based on Flexible Routing Tables, *Proc. IEEE ISCC 2014*, IEEE (2014).
- [11] Shudo, K.: Overlay Weaver (2006), available from <http://overlayweaver.sourceforge.net>.
- [12] Shudo, K., Tanaka, Y. and Sekiguchi, S.: Overlay Weaver: An Overlay Construction Toolkit, *Computer Communications*, Vol.31, No.2, pp.402-412 (2008).
- [13] Kleinberg, J.: The Small-World Phenomenon: An Algorithm Perspective, *Proc. STOC 2000*, pp.163-170, ACM (2000).



北條 真史 (学生会員)

2014 年東京工業大学理学部情報科学科卒業。現在、同大学大学院情報理工学研究科数理・計算科学専攻修士課程在学中。分散システムに興味を持つ。



長尾 洋也 (学生会員)

2010 年東京工業大学理学部情報科学科卒業。2012 年同大学大学院情報理工学研究科数理・計算科学専攻修士課程修了。現在、同博士課程在学中。自律分散アルゴリズムによるグラフ構築に興味を持つ。



宮尾 武裕

2011 年東京工業大学理学部情報科学科卒業。2013 年同大学大学院情報理工学研究科数理・計算科学専攻修士課程修了。分散システム, P2P ネットワークに興味を持つ。



首藤 一幸 (正会員)

1996 年早稲田大学理工学部情報科学科卒業。1998 年同大学助手。2001 年同大学大学院理工学研究科情報科学専攻博士後期課程修了。同年産業技術総合研究所研究員。2006 年ウタゴエ (株) 取締役最高技術責任者。2008 年 12 月より東京工業大学准教授。2009 年 5 月より IPA 未踏 IT 人材発掘・育成事業プロジェクトマネージャを兼任。博士 (情報科学)。分散システム, プログラミング言語処理系, 情報セキュリティ等に興味を持つ。SACIS2006 最優秀論文賞。IPA 未踏ソフトウェア創造事業スーパークリエイター認定。情報処理学会平成 18 年度論文賞。情報処理学会平成 19 年度山下記念研究賞。平成 21 年度船井学術賞。平成 24 年度科学技術分野の文部科学大臣表彰若手科学者賞。情報処理学会 2012 年度長尾真記念特別賞。電子情報通信学会, 日本ソフトウェア科学会, ACM, IEEE, IEEE CS, IEEE ComSoc 各会員。