

# 柔軟な経路表に基づく二次元平面上的構造化オーバーレイ

北條 真史<sup>1,a)</sup> 長尾 洋也<sup>1,b)</sup> 宮尾 武裕<sup>1,c)</sup> 首藤 一幸<sup>1,d)</sup>

**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 に偏りがある場合でもホップ数の小さい経路を導くことが求められる。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。

## A Structured Overlay on a Two-Dimensional Space based on Flexible Routing Tables

MASASHI HOJO<sup>1,a)</sup> HIROYA NAGAO<sup>1,b)</sup> TAKEHIRO MIYAO<sup>1,c)</sup> KAZUYUKI SHUDO<sup>1,d)</sup>

### 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する場合などが該当する。

位置情報を扱うことを考えたとき、二次元の ID 空間を採用することで実空間との対応を簡単に取ることができる。これ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的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。しかし、これまでに提案されてきた手法では、経路表サイズの変更、ルーティングに際して新たな指標を採り入れる拡張が難しいといった問題点がある。

そこで我々は、これらを解決した二次元平面上的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。その上で、メッ

<sup>1</sup> 東京工業大学  
Tokyo Institute of Technology  
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

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もつ. 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挙げられる. 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, 経路表に全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

本節では, まず二次元の ID 空間を用いる構造化オーバレイのアルゴリズムを幾つか説明する. 次に, 二次元 ID 空間上の構造化オーバ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 二次元平面上的の構造化オーバレイ

#### 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もつ構造化オーバレイである. 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, 二次元平面上的のオーバ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点の担当ノードに到達することができる.

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. 図 1 は二次元平面上的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 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.

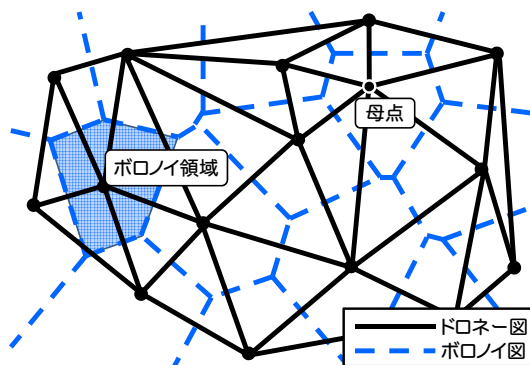


図 1 二次元平面上のドロネー図とボロノイ図

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

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 は全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、後述の利点で挙げる様々な柔軟性を備える。

### 2.3.1 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 つの操作を伴う。

- **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  $L$  を越えたときに経路表エントリを 1 つ削除する操作である。2.3.2 節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  $L$  を越えるまでは学習操作が繰り返される。 $L$  を越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。

### 2.3.2 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、後述する 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。

### 2.3.3 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、例えば以下が挙げられる。

- 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。
- 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) [8] 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。

### 2.3.4 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、これまでに一次元の環を ID 空間とする FRT-Chord [4] や FRT-2-Chord [9], 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<sup>#</sup> 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上で，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 から直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，様々な指標を導入する拡張性を持つ．例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 節にて述べた 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．

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) |

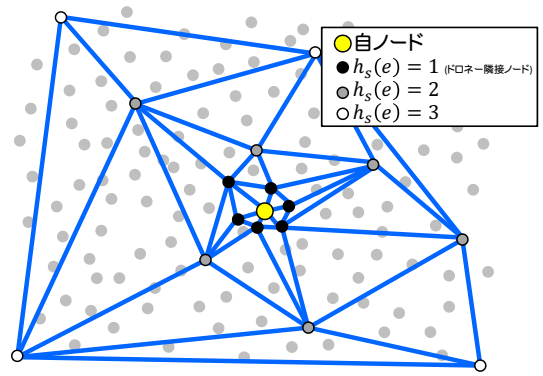


図 2 二次元平面上の経路表エントリ

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

##### 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．ノードの分布に偏りがある場合でもショートカットリンクを形成すべきノードを適切に決定するために，提案手法では P2P ドロネーネットワークにおけるホップ数を考える．

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ため，提案手法では経路表内の各エントリ  $e$  に対してホップ数を指標とした順位を与える．この順位を定義するために，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， $s$ ， $e$  間のグラフ距離である．

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多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。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  $\leq_{dic}$ )**

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

図 3 を用いて、提案手法の提案手法の経路表間順序関係  $\leq_{ID}$  の考え方について説明する。図 3 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  $h_s(e) = 3$ ,  $h_e(s) = 4$  となっており、経路表は経路表内の順位に関しては  $r_s(e) = 13$ ,  $r_e(s) = 24$  となっている。このとき  $f(e) = |r_s(e) - r_e(s)| = 11$  と計算される。ここで計算した誤差が意味するものは、 $s$ - $e$  間の空間内でそれぞれの経路表に載っているエントリ数に差があるということである。すなわち、 $s$  に比べて  $e$  は  $s$ - $e$  間の空間内により多くのショートカットリンクを形成しているため、 $r_s(e) < r_e(s)$  となっている。提案手法では、自ノードと経路表エントリでの順位差を小さいものを優れた経路表と定義しているため、ここで取り上げたような経路表エントリ  $e$  は経路表の改良が繰り返される過程で削除される。

## 4. 評価

提案手法のルーティング性能を検証するために、オーバー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。実験環境は以下の通りである。

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

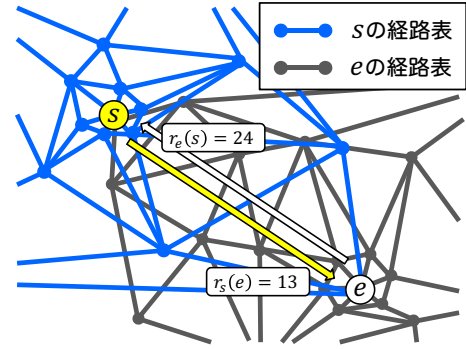


図 3 経路表エントリの順位差

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

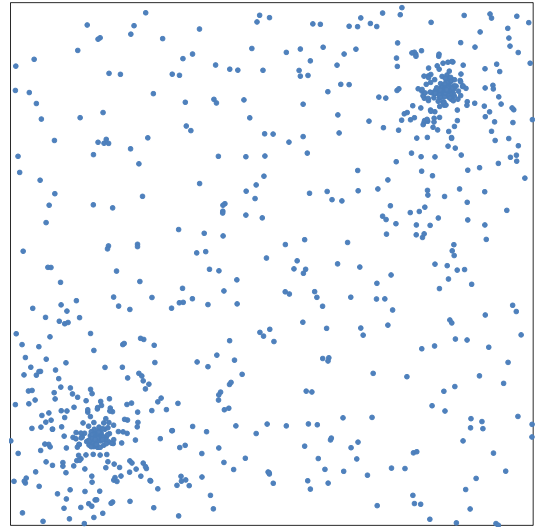


図 4 Zipf 分布 ( $\alpha = 0.7$ ) を用いた際のノード配置

Fig. 4 Node's layout at Zipf distribution ( $\alpha = 0.7$ )

### • 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様に分布させたもの
- (2) 全ノードのうち 2 割を平面上に様に分布させた上で、平面上の離れた 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、一様分布、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。

実験では探索先 ID がノード ID と同様の分布に従う探索を繰り返すことでネットワークを構築し、探索時の経路長を計測した。

### 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

提案手法の他に、比較対象として 2 種類のアルゴリズムを用意した。いずれも提案手法と同様に学習操作と厳選操

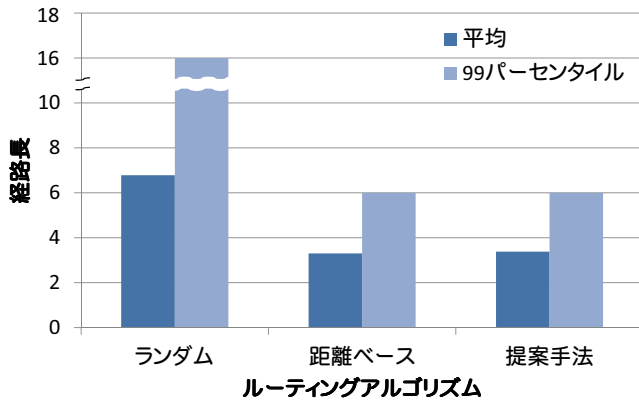


図 5 一様分布における経路長

Fig. 5 Route lengths at uniform distribution

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全く用いず、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様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。

$$f(e_i) = C \frac{1}{d(s, e_i)} \quad (5)$$

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、ノードが一様に分布する状況下で効率よ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、これの他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、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 から図 7 ままで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

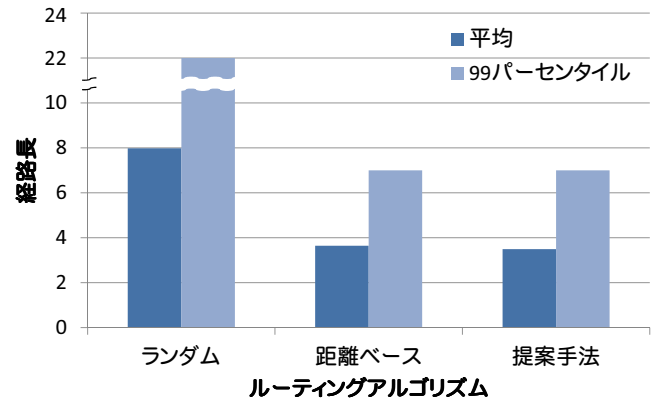


図 6 Zipf 分布 ( $\alpha = 0.7$ ) における経路長

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

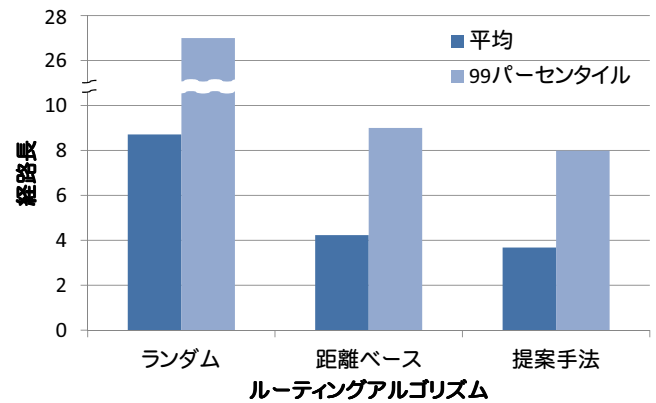


図 7 Zipf 分布 ( $\alpha = 1.0$ ) における経路長

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

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わかる。

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均経路長の増加は微小である。一様分布に対して 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、一様分布と 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わかる。また、一様分布では提案手法、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効果的に経路長を削減することがわかる。

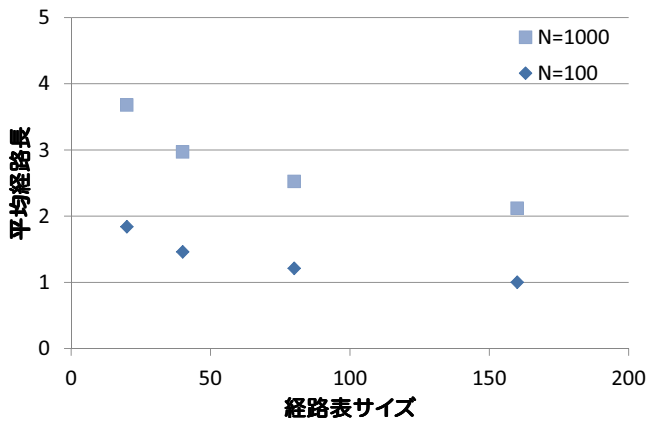


図 8 経路表サイズと平均経路長

Fig. 8 Routing table sizes and route lengths

#### 4.2 経路表サイズの変更

提案手法の FRT 由来の特徴のひと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、これに伴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。

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わかる。

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、経路表に総てのノードを載せ、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わかった。

#### 5. 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 節) と同程度

のルーティング性能をもつことを確認した。

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、よりよ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、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, 24650025 の助成を受けたものである。

#### 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*, IEEE, pp. 173–180 (2005).
- [2] Schütt, T., Schintke, F. and Reinefeld, A.: A Structured Overlay for Multi-dimensional Range Queries, *Proc. Euro-Par 2007*, Springer, pp. 503–513 (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ーネ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*, IEEE, pp. 72–81 (2011).
- [5] Aspnes, J. and Shah, G.: Skip Graphs, *ACM Transactions on 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*, ACM, pp. 161–172 (2001).
- [7] 大西真晶, 坪井新治, 平山雅夫, 江口隆之, 上島紳一: P2P ドローネ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] Miyao, T., Nagao, H. and Shudo, K.: A Method for Designing Proximity-aware Routing Algorithms for Structured Overlays, *Proc. IEEE ISCC 2013*, IEEE, pp. 508–514 (2013).
- [9] 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*, IEEE, pp. 170–175 (2014).
- [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) (to appear).
- [11] Shudo, K.: Overlay Weaver, <http://overlayweaver.sourceforge.net> (2006).
- [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*, ACM, pp. 163–170 (2000).