

## 位置情報を用いた P2P 型ネットワークの分散生成アルゴリズム の提案と評価

大西 真晶<sup>†</sup> 源元 祐太<sup>†</sup> 加藤 宏章<sup>†</sup> 上島 紳一<sup>†</sup>

<sup>†</sup> 関西大学大学院総合情報学研究科  
〒 569-1095 大阪府高槻市霊仙寺町 2-1-1  
E-mail: †fa4d001@edu.kansai-u.ac.jp

**あらまし** 近年、ノードの位置情報を用いた P2P パラダイムが盛んである。本稿では、著者らが提案している位置情報を用いた P2P ネットワークの分散生成アルゴリズムの概略を述べ、形成過程についてシミュレーションにより議論する。提案アルゴリズムでは、各ノードは、他のノードに関する位置情報をもとに、計算幾何で知られるドロネー三角形分割を自律的に行い、局所的なドロネーネットワークを生成する。同時にノードが協調的に相互に位置情報を交換することにより、大域的なネットワークをベースネットワーク上に重畳的に構築することができる。生成された P2P ドロネーネットワークを用いて、ノード間の通信が行われる。提案アルゴリズムは、逐次添加法を分散的環境に拡張して各ノードが自律分散的にネットワークを構成するアルゴリズムとなっている。提案アルゴリズムは次の利点を持つ。つまり、(1) 不必要な遠隔ノードを経由せずに地理的に局所的に存在するノード間で直接的に通信が行える、(2) ネットワークを常時更新続けることでノード数に関するスケーラビリティが得られる、(3) 地図や地理空間などの 2 次元空間を段階的に構造化することができ、スーパーノードが地理的な領域を管理し、領域検索を可能とすることが期待できる、(4) 小さなサイズの経路表を構成して地理的ルーティング手法を構成できる、などの利点を得られる。本手法を用いることにより、ノードの地理的位置関係に基づいて、位置指向情報システムを開発することが容易になるものと考えられる。

**キーワード** P2P, オーバーレイネットワーク, 分散協調, 計算幾何

## Distributed Generation of Location-based P2P Network and its Geometrical Features

Masaaki OHNISHI<sup>†</sup>, Yuta MINAMOTO<sup>†</sup>, Hiroaki KATO<sup>†</sup>, and Shinichi UESHIMA<sup>†</sup>

<sup>†</sup> Graduate School of Informatics, Kansai University,  
Ryozenjicho 2-1-1, Takatsuki-city, OSAKA, 569-1095 Japan  
E-mail: †fa4d001@edu.kansai-u.ac.jp

**Abstract** This paper discusses an incremental algorithm for generating geometric networks in P2P settings, and explores the algorithm according to the movement of each node and interaction between other nodes through numerical simulation. It employs a mathematical construct of Delaunay network as a topology of overlaid network and assumes practical situations of P2P network that (1) nodes are autonomous and connected to the basic network, and (2) each node has only local scope of neighboring nodes. Nodes work in parallel and new behaviors for both data delegation and notification to neighbors have been introduced for Delaunay triangulation from local scopes and updations among nodes in P2P settings. Owing to these, every node uses Delaunay triangulation autonomously by applying its knowledge of other nodes and cooperatively exchanging information among nodes using the base network, which generates and refines overlaid networks. Furthermore, we investigate the characteristics of the network constructed according to our method, from behaviors of both the single node and parallel node, as each nodes work in parallel and construct the Delaunay Triangle autonomously.

**Key words** P2P, Overlaid network, Distributive cooperation, Computational Geometry

## 1. はじめに

近年、スケーラビリティに優れた P2P 情報システムに関して多くの研究が行われている。特に、ワイヤレス通信システム、センサーネットワーク、地理情報システムなどの幅広い分野において位置情報を利用した研究が進められている [1]~[4]。これらでは、位置情報が空間内の情報通信路の決定、機器の監視範囲などの管理領域を決定などに有効に用いられている。

著者らも、スケーラブルな仮想共有空間を幾何的な P2P ネットワーク上への構築を目指して、ノードの自律的な組織化によるオーバーレイネットワークの構築アルゴリズムを提案している [5]。この手法は、P2P 環境においてネットワークトポロジーとしてドロネー図を導入している。ドロネー図は、平面上に与えられた各母点を用いて一定の条件を満たすように平面を三角形分割した図であり、平面上の任意の点を最も近い母点に属する領域に分割するボロノイ図とは双対関係にある。ドロネー図は、(i) 母点数が増えても各母点の次数が大きくなる、(ii) 平面上の任意の 2 点間を結ぶ経路が存在する、(iii) 母点の移動や削除による同図の更新の影響がその点の近傍に留まる、などの特徴がある [6]。ドロネー図の母点をサーバレスネットワーク上の計算機資源とし、エッジを通信路とみなせば空間における有効なネットワークとなるものと考えられる [1], [2]。しかし、従来法はすべての母点の位置が既知であることを前提とした大域的なネットワークの構成法を与えているために P2P 環境において適用しにくい。

提案アルゴリズムは、各ピアが自律的な局所ネットワークを生成し、全体のピアが協調して大域的なネットワークを構成するような分散協調型のアルゴリズムとなっている。即ち、各ノードは、他のノードに関する情報をもとに、ドロネー三角形分割を自律的に行い、局所的なドロネーネットワークを生成する。同時にノードが協調的に相互に情報を交換することにより、大域的なネットワークをベースネットワーク上に重畳的に構築することができる特徴を持つ。本稿では、提案アルゴリズムの概略を述べ、シミュレーションにより各ノードの動きの解析ならびにドロネーネットワークの形成過程の解析を行う。特にネットワーク形成における各ノードの負荷をドロネー三角形分割におけるフリップ回数ならびに情報のノード間の転送回数の観点から考察し、実際に母点の増加した場合のネットワーク形成についてネットワーク初期状態、種々の形成パラメータに対して検討する。また、実際の応用を考慮してノードの位置的な配置に偏りがある場合についても検討する。その結果、提案アルゴリズムがノード数  $10^5$  のオーダーに対して働くことについて確認する。

## 2. P2P ドロネーネットワーク

### 2.1 ノードと動き

平面上に位置するコンピュータノードの集合を考える。ノードは常に独立に稼働状態にあり、自身の位置情報を持つものとする。また、隣人として 1 ホップ先のノード ID とその位置情報を持つものと仮定する。ノード間の通信は隣人を通して行

われ、遠隔ノード間についてはマルチホップ通信形態により通信が行われるものとする。この P2P ネットワークトポロジーとしてコンピュータノードを母点、ノード同士の接続関係を辺とみなしたグラフを考えることができる。このグラフがドロネー図の接続関係を満たすネットワークを P2P ドロネーネットワークと呼ぶ。このネットワークはノードの位置に基づいたネットワークである。

3. で述べる構成アルゴリズムによって、ノードは隣人間で位置情報を交換して隣人表を更新し、ベースネットワークの上のオーバーレイ型に P2P ドロネーネットワークを生成・更新できる。また、ノードの ON/OFF によるネットワークへの新規参加、脱退、に対しても局所的な情報交換によって隣人表が更新され、トポロジーが更新される。この P2P ドロネーネットワーク構成法により、ノードの数の増加に対しても同様の方法でトポロジーを形成できる特徴を持つ。

### 2.2 特徴

ノードの位置に基づく P2P ドロネーネットワークを構成することで以下の特徴が得られる。まず、ネットワークは、**[局所性 1]** ある時点毎の接続関係にあるノード間における情報交換のみでネットワークを構築する為、ノード数に依存せず構築可能である。

**[局所性 2]** 位置的に近いノード同士が接続関係にあるため、近距離のノードの協調動作の際にネットワーク全体に対する負荷が低い。

**[均一性]** 平面空間上の位置的な隣接に基づくネットワークであるため、ノードがある一定の範囲に集中している場合も、各ノードの隣人数はほとんど変化しない。反面、ノードが集中している空間では、実距離が少し離れたノードに対する通信を行うだけでも、経由ノード数が増加する事が考えられる。このような場合、LRC(Long Range Contact)が必要となる [3]。

更に、位置を用いたネットワーク構成であるため、ノードの数が増加した場合にも同様の方法で以下の操作が可能になる。

**[指定位置の情報検索]** 指定位置に最も近いノードを探索する際、任意のノードから、その隣人の中で最も目的の点に近いノードに、探索パケットを転送する事を繰り返す事によって、その位置に最も近いノードに到達させることができる。

**[矩形領域の検索]** 平面上のある矩形の中に含まれるデータを取得する場合、検索パケットは、その矩形に向かって直線的に転送された後、矩形内では矩形に行き渡るようにパケットが転送される。矩形内のノードの管理領域は、ドロネー隣接するノード同士の領域で連続しておりその領域外のいかなるノードを経由する必要が無い。

**[ノードの管理領域の明示]** 双対関係にあるドロネー図における近傍ノードの位置から、自身のボロノイ領域を算出する事ができる為、このボロノイ領域を利用して様々な目的に利用する事が考えられる。例えば、このボロノイ領域が平面を完全に分割している事から、データ蓄積用の平面空間の各ノードの割り当てを部分を各ノードのボロノイ領域と定める事ができる。

### 3. 提案アルゴリズム

#### 3.1 基本的な考え方

本研究のP2Pドローネーネットワーク構築アルゴリズムは、ドローネー図の一般的な作図方法である逐次添加法の拡張である[5]。複数のノードが逐次添加法の部分的な動作を行うアルゴリズムであり、複数のノードがドローネー状態ネットワークとなっている時、全体として分散的に逐次添加法の動作を行う。各ノードは、位置と接続を確立する為のネットワークアドレスを持ち、いくつかのノードに関するそれらの情報のリスト(ノード情報リスト)を持つ。各ノードの動作は大きく分けて以下の三つを行う。

- 各ノードは、自身の知るノード情報のリスト(ノード情報リスト)を元に局所的なドローネー図を描き、自ノードとドローネー隣接するノードのリスト(隣接ノード情報リスト)とならないノードのリスト(非隣接ノード情報リスト)を作成する。
- 隣接ノード情報リストにあるノードに対しては接続を確立し、隣接ノード情報リストの中でそのノードとドローネー隣接する可能性のあるノードの情報を送信する。
- 非隣接ノード情報リストにあるノード情報を各ノード情報の位置に最も近い隣接ノード情報リストにあるノードに委譲する。

このアルゴリズムは分散的な逐次添加法であるが、局所的なドローネー図を描く処理によって、複数のノードの同時追加にも対応できる。また、平面上の位置付きノードと接続関係によって構成されるネットワークをグラフの点と辺で表す時、グラフの最小の構成要素である点と辺は、ノード数1とノード数2の小さなP2Pドローネーネットワークとなっている。この事から連結グラフであれば、各点、各辺という小さないくつものドローネーネットワークが個別に接続的に直接繋がっているノードを次々と取り込みながら成長する事になる。この時、ドローネーネットワークは、全ノードの位置の分布に対して唯一の形に定まる為、次第に融合して一つの大きなドローネーネットワークとなる。

#### 3.2 アルゴリズム

アルゴリズムの概略について述べる。各ノードは、他のノードから、情報を受信した時にそれをノード情報リストに追加し、以下の処理を行う。

[フリップによるドローネー三角形分割化] 各ノードは、図1に示すような局所的なドローネー図を描く。ドローネー図の一般的な描き方として、任意の三角形分割において隣合う三角形の頂点4つのうちから3つを選んで外接円を描き、その外接円の中に残り1点が入っている時、その4つの頂点を使って描ける2つの三角形を描く2通りの方法のうちもう一方の三角形に変更する(この操作をフリップと呼ぶ)。この操作を繰り返し全ての三角形の外接円に隣り合う三角形の残り一点が含まれない時、その三角形分割は、ドローネー三角形分割となっている。

[局所ドローネー図描画法] 各ノードが持つノード情報リストに含むノードとドローネー隣接していると仮定するとその仮定から一意に決まる三角形分割図を描ける。その後、ノード情報リスト内のノード情報を全て隣接ノード情報リスト内にコピーする。そして、三角形分割図において任意の三角形から時計回りの順

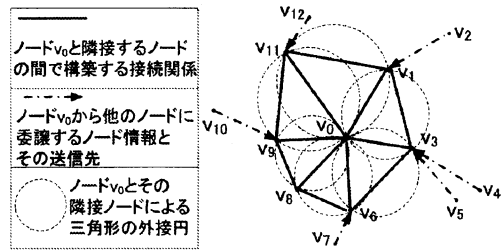


図1 ドローネー隣接と非隣接に基づくノード情報の分別と非隣接ノード情報の委譲先の決定

で隣接する三角形同士で外接円に他方の残り一点を含むかどうかを判定し、含まれる場合フリップ処理を行う。フリップを一度行う毎に一つのノード情報が隣接ノード情報リストから、非隣接ノード情報リストに移動する。

[接続の確立] 時計回りの処理が一巡し、局所的なドローネー図を描く事ができたら、隣接ノード情報リスト内のノードとその情報を使って接続を確立する。

[三角化、ノード情報の通知] 各三角形の頂点にあたるノードに他の頂点のノード情報を送信する。

[委譲] 非隣接ノード情報リスト内の各ノード情報は、隣接ノード情報のリスト内のノードの中で最も近いノードに委譲する。

#### 3.3 例

ノード数10のP2Pドローネーネットワークの構築例を示す。各ノードの保持するノード情報の遷移は表1、表2に表される。また、図2は、ネットワークの接続関係の遷移を視覚化したものである。これらの表の記述方法と前提について以下に述べる。まず”ノードID(x,y)”は、ノードの識別子と平面内の位置座標を意味する。”ノード情報リスト”には、通知された他のノードの情報を格納する。”非隣接ノード情報リスト”と”三角化通知リスト”内の、”Vi(Vj)”はノードViの情報がノードVjに通知されるということを示している。”三角化通知リスト”には隣接ノード情報リストの先頭から順に二つの連続するノード情報が格納され、選択された隣接ノード2つの情報を互いに通知する。

次にSTEP1のノードv1を例に挙げ解説する。v1のノード情報リスト内のノード情報(v2,v3,v9,v10)は、v1が保持しているノード情報である。v1がこれを処理した結果、v2,v9を隣接ノード情報リストに、v3,v10を非隣接ノード情報リストに分類する。非隣接ノード情報リスト内の非隣接ノードの情報は、最も近い隣接ノードに対して通知される。そのため、非隣接ノードv3の情報は隣接ノードv2に、非隣接ノードv10の情報は隣接ノードv9に通知される。また、v1を頂点に持つドローネー三角形はv1の隣接ノード情報リスト内の連続したノード2つと構成されることが予想される。そこで、 $\Delta v1v2v9$ を構成可能にするためにv9のノード情報をv2に、v2のノード情報をv9に通知する。これらの処理は非隣接ノード情報リストが空になり、適切なドローネー三角形が構成されるまで継続的に実行される。

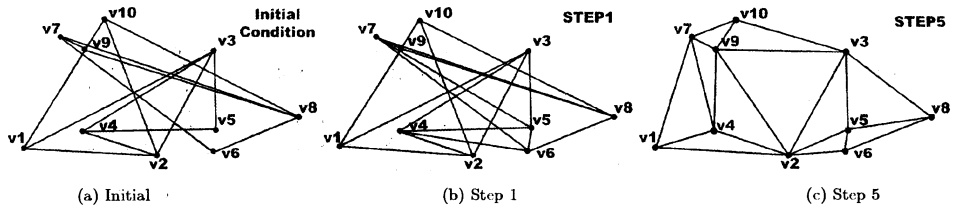


図 2 P2P ドロネーネットワーク構築過程図 (n=10)

表 1 各ノードが持つノード情報の遷移 (ステップ 1)

ノード ID(x,y)	ノード情報リスト	隣接ノード情報リスト	非隣接ノード情報リスト	三角化通知リスト
v1(124,100)	v2,v3,v9,v10	v2,v9	v3(v2),v10(v9)	v9(v2),v2(v9)
v2(422,92)	v1,v3,v4,v10	v1,v3,v4	v10(v3)	v4(v1),v4(v3),v1(v4),v3(v4)
v3(536,318)	v1,v2,v4,v5	v1,v2,v4,v5	[empty]	v1(v4),v4(v1),v4(v2),v5(v2),v2(v4),v2(v5)
v4(251,140)	v2,v3,v5,v6	v2,v3	v5(v2),v6(v2)	v2(v3),v3(v2)
v5(546,145)	v3,v4,v6,v7	v3,v4,v6,v7	[empty]	v3(v7),v6(v4),v4(v6),v4(v7),v7(v3),v7(v4)
v6(536,97)	v4,v5,v7,v8	v4,v5,v8	v7(v4)	v4(v5),v5(v4),v8(v5),v5(v8)
v7(202,349)	v5,v6,v8,v9	v6,v8,v9	v5(v6)	v6(v9),v8(v9),v9(v6),v9(v8)
v8(729,175)	v6,v7,v9,v10	v6,v10	v7(v10),v9(v10)	v6(v10),v10(v6)
v9(256,324)	v1,v7,v8,v10	v1,v7,v8,v10	[empty]	v1(v7),v1(v8),v7(v1),v7(v10),v8(v1),v8(v10),v10(v7),v10(v8)
v10(303,384)	v1,v2,v8,v9	v2,v8,v9	v1(v9)	v2(v8),v2(v9),v8(v2),v9(v2)

表 2 各ノードが持つノード情報の遷移 (ステップ 5)

ノード ID(x,y)	ノード情報リスト	隣接ノード情報リスト	非隣接ノード情報リスト	三角化通知リスト
v1(124,100)	v2,v4,v7	v2,v4,v7	[empty]	v2(v4),v4(v2),v4(v7),v7(v4)
v2(422,92)	v1,v3,v4,v5,v6,v9	v1,v3,v4,v5,v6,v9	[empty]	v1(v4),v4(v1),v4(v9),v3(v5),v3(v9),v6(v5),v9(v3),v9(v4)
v3(536,318)	v2,v5,v8,v9,v10	v2,v5,v8,v9,v10	[empty]	v2(v5),v2(v9),v5(v2),v5(v8),v8(v5),v9(v2),v9(v10),v10(v9)
v4(251,140)	v1,v2,v7,v9	v1,v2,v7,v9	[empty]	v1(v2),v1(v7),v2(v1),v2(v9),v7(v1),v7(v9),v9(v2),v9(v7)
v5(546,145)	v2,v3,v6,v8	v2,v3,v6,v8	[empty]	v2(v3),v2(v6),v3(v2),v3(v8),v6(v2),v6(v8),v8(v3),v8(v6)
v6(536,97)	v2,v5,v8	v2,v5,v8	[empty]	v2(v5),v5(v2),v5(v8),v8(v5)
v7(202,349)	v1,v4,v9,v10	v1,v4,v9,v10	[empty]	v1(v4),v4(v1),v4(v9),v9(v4),v9(v10),v10(v9)
v8(729,175)	v3,v5,v6	v3,v5,v6	[empty]	v3(v5),v5(v3),v5(v6),v6(v5)
v9(256,324)	v2,v3,v4,v7,v10,v1	v2,v3,v4,v7,v10	[empty]	v2(v3),v2(v4),v3(v2),v3(v10),v4(v2),v4(v7),v7(v4),v7(v10)
v10(303,384)	v3,v7,v9	v3,v7,v9	[empty]	v3(v9),v7(v9),v9(v3),v9(v7)

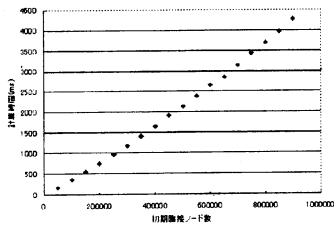


図 3 ノード情報数に対する  
ドロネー領域計算時間 (1ノード)

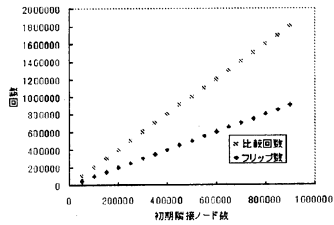


図 4 ノード情報数に対する隣接三角形  
の比較処理回数とフリップ回数 (1ノード)

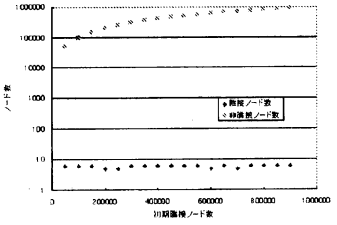


図 5 ノード情報数に対する隣接ノード数  
と非隣接ノード数 (1ノード)

## 4. ネットワーク形成過程の解析

### 4.1 前提

本節では、提案アルゴリズムに基づくノードの振舞いについて定量的に解析し、P2P ドロネーネットワークの形成過程について議論する。すべてのシミュレーションに共通して以下の前提を設ける。

- ①各ノードはユークリッド平面上に存在し、自身の位置

情報ならびに一意の ID を保持する。

- ②各ノードは初期状態として異なる他のノード情報 (位置と ID) を保持している。

②は、ネットワークに新規参加するノードも、ping 発行などの方法によりネットワーク内の少なくとも 2 つのノードを知っていることを仮定している。これによりまったくネットワークに参加する可能性のないノードは議論から除外している。実装においては、初期状態においてノードの接続関係が連結グラフ

となるように、ノードの ID の数字順に次とその次のノードの情報保持する。つまり環状の接続関係を作る。ID と位置に関連性はなく、隣接ノードの位置的な関係は独立である。

提案アルゴリズムでは、個々のノードが並列に動作し、P2P ドローネネットワークを構築するため、単一ノードの局所的な振舞いと全ノードによる大域的な振舞いの両面より考察を進める。また、全ノードの振舞いに関しては、ノードの初期位置分布の偏りによる振舞いの差異についても検討する。

#### 4.2 単一ノードの動き

平面上に  $N$  個のノードを平面上に一樣に配置する。各ノードは、毎時点でドローネ三角化処理を行って自身と隣接すべきノードを選択する。ここはノード数の増加に対する処理の負荷を見るため、 $N$  を 50,000 から 900,000 まで変化させて以下の項目についてデータを取得した。

- 単一ノードの処理時間シミュレーション (図 3)
- 比較回数ならびにフリップ回数 (図 4)
- 各ノードの隣接ノード数と非隣接ノード数 (図 5)

図 3 は、このグラフは提案アルゴリズムの実行時間がノード情報数の変化に対して比例して増加することを示している。

また、隣接三角形の比較回数およびフリップ回数がノード情報数に対して比例することを示している。比較回数は三角形の外接円とその三角形に隣接する三角形の頂点位置の比較を表している。比較回数はほぼ  $O(2N)$  となり  $N$  の増加に対してアルゴリズム内の主要な処理の実行回数は比例的な増加程度に収まると言える。

図 5 からは、ノード数  $N$  の増加に対しても隣接ノード情報数はほぼ一定であり、非隣接ノード情報数はノード情報数に対して比例することが解る。ノード情報数に関わらず、隣接ノード情報数がほぼ一定になる、つまり  $N$  が増加しても特定のノードに接続の集中がおこらないことがわかる。

#### 4.3 複数ノードの動き

各ノードが単一で 3.2 で述べた振舞いをする時、相互接続された複数ノードの振舞いに関してみる。正方形平面上に  $N$  個のノードを一樣に分布した配置形態を用い、すべてのノードで単一ノードの提案アルゴリズムを並列に実行する。更に、ノードの平面上の位置に偏りがあった場合を想定し、 $N$  個のノードを、2 から 6 個の中心を持つクラスタとして配置し、同様にすべてのノードで単一ノードの提案アルゴリズムを並列に実行する。図 6 は、今回のシミュレーションで使用したノード分布の例である。全てノード 1000 個を平面上に分布させドローネネットワークを構成した図である。最も左の図がクラスタ数 2 で、各クラスタの中心点からのその属するノードすべてまでの距離についての分散が 0.01 に収まっている図である。同様に左から二番目の図がクラスタ数 2 で、分散が 0.1 に収まっている図であり、右から二番目の図がクラスタ数 6 で、分散が 0.01 に収まっている。最も右の図がクラスタ数 6 で、分散が 0.1 に収まっている。

$N$  を 3 から 10,000、クラスタを 2 から 6、クラスタの分散を 0.01, 0.5, 0.1 の範囲で変化させた。ノードが協調して動作する本アルゴリズムの負荷が、各ノードに分散しているかを確認

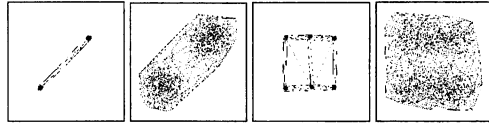


図 6 ノードのクラスタ状分布

するため以下の項目についてデータを取得した。

- 1 回のシミュレーションの間の各ノードのフリップ回数の合計のうち最大の値を取得した。 $N$  の数の増加に対して、本アルゴリズムの主要な処理による負荷が最も大きいノードについて定量的に評価する。
- 1 回のシミュレーションの間の各ノードのフリップ回数の合計の分散を取得した。本アルゴリズムは逐次添加法の分散処理という側面を持っており、その主要な処理であるフリップ処理が全ノードに分散して実行されているかを確認する。
- 1 回のシミュレーションの終了後の各ノードの隣接ノード情報の数の分散について計測する。隣接ノード情報の数の偏り具合について確認する。

表 3、図 7 は、シミュレーション間の各ノードのフリップ回数の合計値の中の最大のものについての結果である。図 7 は、横軸に  $N$ 、縦軸にフリップ回数の合計値の最大値を取ったグラフである。図 7 では、クラスタ数 2 でその分散が 0.01 の分布の時最大値は  $N$  の増加にともない増加しているが、その増加率はノード数が大きくなるにつれて減少している事が伺える。また、一樣分布か、それに近い分布では最大値はほぼ横這いに推移している。この違いは委譲の際経由していくノードが前者の場合、ノードの分布が直線状に分布している為、経由するノードが多くなる事、直線状の配置の中心に位置しているノードは多くのノード情報の通り道になる為と考えられる。一方、後者は、情報の委譲の直線ルートが重なっておらず負荷が分散されるものと思われる。表 3 でも同様の傾向が見られ、クラスタ数が小さい程、また、各クラスタの分散が小さいほど最大値が大きくなっている。これらの事から、 $N$  が増加した場合に飛びぬけて処理量が大きくなるノードができないものと考えられる。

表 4、図 8 は、シミュレーションの間の各ノード毎のフリップ回数を合計したものの分散のグラフである。図 8 は、横軸に  $N$ 、縦軸に分散を取ったグラフである。表 4、図 8 共に各クラスタの分散が低い時はクラスタ数が多い時ほど、分散が大きくなっている。逆に、各クラスタの分散が高い時はクラスタ数が少ない時ほど、分散が大きくなっている。各クラスタの分布の分散が 0.1 の場合は、ほぼ一樣分布に近い値であり、クラスタ数による差は余りない。値も 400 前後に収まっており  $20^2$  程度であるから、処理は分散している。一方、分散が 0.01 の場合も、クラスタ数による差は余り無く分散も最大で 780.0 であり、 $28^2$  程度であるから処理は分散している。また、 $N$  の増加に従ってどの場合の分散も増加しているがその増加率は、次第に減少している。よって、 $N$  が増加した場合でも、処理が全体的に分散するものと考えられる。図 9 は、シミュレーション実行後の各ノードの隣接ノード情報数の分散のグラフである。横軸に  $N$ 、

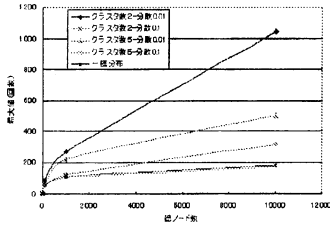


図7 ノードの分布状況とノード数毎のフリップ回数の最大値

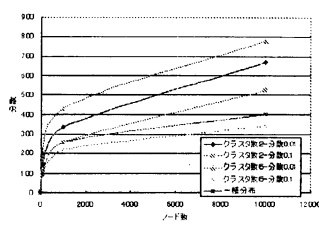


図8 ノードの分布状況とノード数毎のフリップ回数の分散

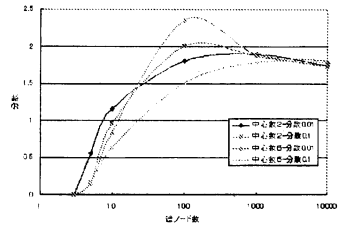


図9 ノードの分布とノード数毎の隣接ノード数の分散

表3 ノードの分布状況に対するノード毎のフリップ回数の最大値

クラスタ数	2			3			4			5			6			一様分布	
分散	0.01	0.05	0.1	0.01	0.05	0.1	0.01	0.05	0.1	0.01	0.05	0.1	0.01	0.05	0.1	-	
ノード数	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	5	2	1	1	2	1	1	1	1	1	1	1	1	1	2	1	2
	10	8	8	8	9	9	7	8	8	8	8	7	6	7	6	7	6
	100	91	80	53	62	59	59	64	61	51	67	53	55	57	61	62	58
	1000	270	258	125	276	172	130	184	159	123	203	173	120	223	142	107	111
	10000	1045	607	317	633	500	285	420	341	191	501	282	187	502	262	173	183

表4 ノードの分布状況に対するノード毎のフリップ回数の分散

クラスタ数	2			3			4			5			6			一様分布	
分散	0.01	0.05	0.1	0.01	0.05	0.1	0.01	0.05	0.1	0.01	0.05	0.1	0.01	0.05	0.1	-	
ノード数	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	5	0.5	0.1	0.1	0.5	0.2	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.5	0.1	0.16
	10	3.0	3.0	3.4	3.4	2.3	2.6	2.0	2.0	3.0	2.6	1.8	1.6	1.0	2.0	1.0	1.9
	100	144.7	109.7	110.8	122.5	119.0	78.9	116.4	82.5	71.2	110.5	76.2	95.1	97.2	80.0	85.2	93.5
	1000	336.4	330.2	256.1	419.1	368.6	240.4	323.7	294.2	238.0	375.2	252.0	215.0	431.6	234.1	220.0	257.1
	10000	671.5	636.1	526.9	747.5	601.8	425.1	632.8	523.2	387.2	764.5	490.3	358.0	780.6	453.6	342.9	403.6

縦軸に隣接ノード情報数を取っている。図9からは、隣接ノード情報数の分散は一番高い場合でも2.5に届いておらず、更に隣接ノード情報数が整数値である事を考えれば非常に小さい。更にノード数の増加に従って分散の値は減少しており、P2Pドロネーネットワークの各ノードの他ノードとの接続数は、単数ノードの実験から6前後に集中する事がいえ、それがP2Pドロネーネットワークに所属する全てのノードに共通であると言える。

## 5. おわりに

本稿では、P2Pドロネーネットワークの形成過程についてシミュレーションにより議論した。本研究の結果、ネットワークに所属するノード数の増加によって、各ノードの処理が劇的に増加する様な事はなく、更にノード数が増加した場合も、本アルゴリズムの処理が各ノードにほぼ均等に分散して行われる事が確認できた。また、P2Pドロネーネットワークが構築された際に、接続数が極端に大きくなるようなノードは本シミュレーションでは見られなかった。

## 文献

[1] X.-Y.Li, I. Stojmenovic, Y. Wang, "Partial Delaunay Triangulation and Degree Limited Localized Bluetooth Scatternet Formation", IEEE Trans. Parallel and Distributed Systems, Vol.15 No.4, pp.350-361 (2004.4)

[2] I.Stojmenovic, A.Prakash Ruhil, D.K.Lobiyal, "Voronoi diagram and convex hull based geocasting and routing in wireless networks", Proceedings of the 8-th IEEE International Symposium on Computers and Communication (ISCC'03), pp.51-56 (2003)

[3] F.Araujo, L. Rodrigues, "GeoPeer: A Location-Aware Peer-to-Peer System", Proceedings of the 3-rd IEEE International Symposium on Network Computing and Applications(NCA'04), pp.39-46, (2004.9)

[4] S. Ratnasamy, B.Karp, L. Yin, F.Yu, D.Estrin, R.Govindan, S.Shenker, "GHT: A Geographic Hash Table for Data-Centric Storage", Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications (WSNA'02), pp.78-87 (2002.8)

[5] M. Ohnishi, R. Nishide, S. Ueshima, "Incremental Construction of Delaunay Overlaid Network for Virtual Collaborative Space", 3-rd Proc. Conference on Creating, Connecting and Collaborating through Computing (C5), pp.77-84(IEEE CS Press) (2005.1).

[6] M. de Berg, M., M. van Kreveld, M.Overmars, M., O. Schwarzkopf, "Computational Geometry : Algorithms and Applications", 2nd ed., Springer (2000)