

# 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

大西 真 晶<sup>†</sup> 源 元 佑 太<sup>†</sup> 江 口 隆 之<sup>†</sup>  
加 藤 宏 章<sup>††</sup> 西 出 亮<sup>†</sup> 上 島 紳 一<sup>††</sup>

本稿では、スケーラブルなネットワーク基盤として、計算幾何の分野で知られるドロネー図をトポロジとして持つ P2P ドロネーネットワークとその自律分散生成アルゴリズムを提案する。ここでは、まず本 P2P ドロネーネットワークの特徴について述べ、次に生成アルゴリズムについて述べる。提案アルゴリズムは、各ノードの局所的な動きから、ノード間の幾何学的な位置関係を利用して接続関係を更新し続けるアルゴリズムであり、さらにノードが相互に情報交換することでネットワークを構成できる特徴を持つ。また、ノードが幾何学的退化状態にある場合も動作できる。本アルゴリズムにより、与えられた 2 つの P2P ドロネーネットワークを融合することも可能であり、P2P パラダイムの持つスケーラビリティを活かしながら、システムの対象空間を段階的に拡張できる。提案アルゴリズムでは、ノードの 3 つの操作を定義している。すなわち、局所ドロネー化操作と三角化通知操作が、局所的なドロネー図を自律分散的に生成し、委譲操作により、ノード間でノード情報の情報交換を行う。最後に、数値シミュレーションにより、P2P ドロネーネットワークの形成過程を、ノードへの負荷、P2P ドロネーネットワークへの収束ステップ数、各ノードの次数の変化、ネットワーク負荷などから検証し、提案アルゴリズムの有効性を確認する。また、本アルゴリズムの適用性についても議論する。

## Autonomous and Distributive Generation Algorithm of Delaunay Network for P2P Model Utilizing Node Location

MASAAKI OHNISHI,<sup>†</sup> YUTA MINAMOTO,<sup>†</sup> TAKAYUKI EGUCHI,<sup>†</sup>  
HIROAKI KATO,<sup>††</sup> RYO NISHIDE<sup>†</sup> and SHINICHI UESHIMA<sup>††</sup>

This paper proposes a P2P Delaunay network whose topology is a Delaunay diagram well-known in computational geometry as a scalable network infrastructure for spatial data management. We first discuss its features as a P2P network, and propose an algorithm to construct the network autonomously and distributively in P2P settings. In the proposed algorithm, nodes update their connection defined by node adjacency with respect to geometric location and generate local Delaunay networks of neighboring nodes, while they exchange node-location information to generate a network. The algorithm also works for the case nodes locate in geometrical degeneracy. Furthermore, the algorithm can also be applied to merging two independent P2P Delaunay networks. Owing to the algorithm, we can manage large target spaces using the P2P paradigm, and furthermore extend the target space incrementally by utilizing scalability that the P2P paradigm possesses. By numerical simulations, the authors examine the construction process of P2P Delaunay networks in terms of loads of nodes, time-steps consumed for constructing P2P Delaunay networks, degree of each node, and networkload cost. The applicability of the proposed algorithm for P2P models is also discussed.

### 1. はじめに

近年、空間情報管理のための構造型 P2P ネットワークの研究がさかんである<sup>1),2)</sup>。これらでは空間を分割

してノードに割り当てて管理する方式がとられており、空間の分割方法に依存したトポロジを構成している。

こうして構造化された P2P ネットワークには、(i) 問合せにフラッディングを用いないためにスケーラビリティを期待でき、大量のノードからなるネットワークを構成できる、(ii) 全体空間を部分空間に分割し、その内部の情報の管理をそれぞれ固有のノードに割り当てる手法により、ネットワーク全体において情報と、

<sup>†</sup> 関西大学大学院総合情報学研究科  
Graduate School of Informatics, Kansai University

<sup>††</sup> 関西大学総合情報学部  
Faculty of Informatics, Kansai University

その管理ノードを1対1に決定できる、など空間情報を用いるアプリケーションや、センサネットワークに適した特徴がある。

著者らも、スケーラブルな空間情報管理基盤の構築を目指し、幾何学的モデルを用いたP2Pネットワークをノードの自律的な組織化によって構築するアルゴリズムを提案している<sup>3),4)</sup>。この手法では、P2Pネットワークのトポロジとしてドロネー図の導入を試みている。ドロネー図は平面上に与えられた点(母点という)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。ドロネー図の母点を計算主体、辺を通信路と見なせば空間情報を処理するのに有効なP2Pネットワーク計算基盤となるものと考えられる<sup>5),6)</sup>。グラフがドロネー図の接続関係を満たしたネットワークをP2Pドロネーネットワークと呼んでいる<sup>3)</sup>。

しかし、従来のドロネー図作成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作成法であるため、P2P環境において適用することが困難である。それに対して、提案アルゴリズムは各母点が自律的に局所的なドロネー図を生成し、全体をドロネー図とする分散型のアルゴリズムである。すなわち、各ノードが自律的に他ノードの情報から周囲をドロネー三角形分割し、局所的なP2Pドロネーネットワークを構成する。さらに各ノードが相互にノード情報を交換することで局所ネットワークが融合し、全体がP2Pドロネーネットワークとなる。

また、一般のドロネー図作成法では、母点の位置関係が特殊である幾何学的退化のときにアルゴリズムが正常に動作しないことがある<sup>7)</sup>。P2P環境に適用するとき、母点はノード、辺は接続を意味するため、特殊な位置の母点に対しても辺が構成されなければならない。提案アルゴリズムはこの幾何学的退化にも対応する。

本稿では、提案アルゴリズムの詳細を述べ、数値シミュレーションによりノードの動きならびにP2Pドロネーネットワークの形成過程を評価する。また、実際的な応用を考慮してノードの配置に偏りがある場合についても検討する。

2章ではドロネー図の特徴について述べ、3章で提案アルゴリズムの基本構造と具体的アルゴリズム、そして特徴について述べる。4章でP2Pドロネーネットワークの形成過程をシミュレーションにより評価し、5章で関連研究と比較する。

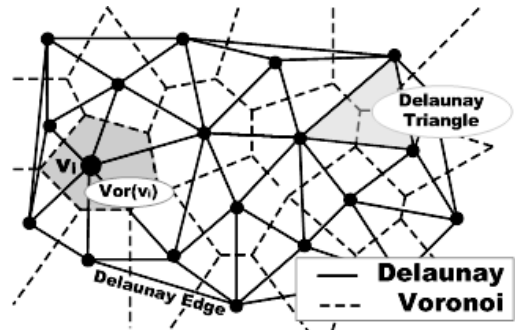


図1 ドロネー図とボロノイ図

Fig. 1 Delaunay diagram and Voronoi diagram.

## 2. ドロネー図の特徴

本章では、ドロネー図の特徴と提案アルゴリズムに關係する定理についてまとめる。図1にドロネー図とボロノイ図を示す。ここで、 $v_i$ などの黒丸は母点であり、コンピュータなどの計算主体を表す。そして、破線がボロノイ図、実線がドロネー図である。

一般に、 $P = \{v_1, v_2, \dots, v_N\}$ を母点の集合とすると、ボロノイ図は、平面を $N$ 個(母点数)の領域に分割したものであり、 $v_i \in P$ に対するボロノイ領域(破線で囲まれた領域)は、 $P$ の他の点よりも $v_i$ が最も近い平面上のすべての点を含む(式(1))。

$$Vor(v_i) = \{x | d(v_i, x) < d(v_j, x), \text{ for } j \neq i\} \quad (1)$$

ここで $d$ はユークリッド距離である。

点 $v_i$ のボロノイ領域を $Vor(v_i)$ と表し、 $P$ に対するボロノイ図を $Vor(P)$ と表す。2つのボロノイ領域が1辺を共有するとき、2つの母点を辺で結んだグラフを平面に埋め込んだものをドロネー図、あるいはドロネー三角形分割と呼び、 $Del(P)$ と表す。

ドロネー図が次の特徴を持つことはよく知られている<sup>8)</sup>。

- (i) ボロノイ図は母点で平面を分割し、ドロネー図は領域を三角形分割して被覆する。
- (ii) ドロネー図の母点の次数の期待値は6以下となる。
- (iii) 平面上の点集合のドロネー図は平面グラフである。すなわち、どの2辺も交差しないように平面に埋め込まれている。
- (iv) 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とどまる。
- (v) ボロノイ図とドロネー図はノード数に関して線形時間で相互変換が可能である。
- (vi) 平面上の任意の2母点の片方から他方へ、方向を基にドロネー辺をたどることで到達できる。

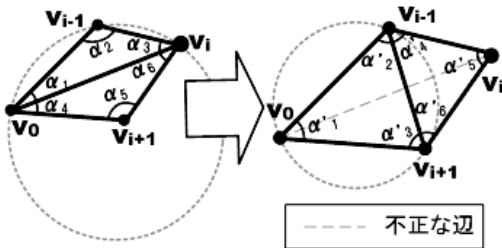


図 2 不正な辺  $\overline{v_0v_i}$  の辺フリップ ( $\alpha_i, \alpha'_i$  を内角とする)  
 Fig.2 Flipping illegal edge  $\overline{v_0v_i}$  ( $\alpha_i, \alpha'_i$ : interior angles).

さらに次の外接円に関する定理と補題が知られている。これらは、本稿 3 章のドロネー図を作成する提案アルゴリズム動作原理として用いられている<sup>8)</sup>。

定理 1 (外接円条件). 平面上の有限点集合  $P$  とし、 $\tau$  を  $P$  の三角形分割とする。  $\tau$  が  $P$  のドロネー三角形分割であるための必要十分条件は、 $\tau$  の任意の三角形の外接円がその内部に  $P$  の点を含まないことである。

ドロネー図上で隣接する 3 点  $v_0, v_i, v_j$  の外接円を  $C(v_0, v_i, v_j)$  とする。

補題 1. ユークリッド平面上の点  $v_0$  と 3 つの点  $v_{i-1}, v_i, v_{i+1}$  を考える。このとき、4 点が同一円周上にないものと仮定する。3 点  $v_0, v_{i-1}, v_i$  の外接円  $C(v_0, v_{i-1}, v_i)$  が残りの 1 点  $v_{i+1}$  を内部に含むための必要十分条件は、3 点  $v_0, v_i, v_{i+1}$  の外接円  $C(v_0, v_i, v_{i+1})$  が残りの 1 点  $v_{i-1}$  を含むことである。

定義 1. 共有辺を持つ 2 つの三角形において、辺  $\overline{v_0v_i}$  を削除し、辺  $\overline{v_{i-1}v_{i+1}}$  を挿入することを辺フリップ、または単にフリップと呼ぶ (図 2)。

定義 2. 共有辺を持つ 2 つの三角形において、ある辺が不正であるとは、その辺を辺フリップすることによって、2 つの三角形の 6 つの内角のうち最小角度を局所的に増大させるときをいう (図 2)。

$$\min_{1 \leq i \leq 6} \alpha_i < \min_{1 \leq i \leq 6} \alpha'_i \quad (2)$$

不正な辺はドロネー図を構成しない辺 (正当でない辺) を意味する。本稿で議論する平面上のノード間の経路決定では、不正な辺は隣の辺となす角が小さく、方向が偏った経路を与えることになる。また、凸四角形では 2 本の対角線の一方が不正でもう片方が正当である。つまり、次の補題が成り立つ。

補題 2. 4 点  $v_0, v_{i-1}, v_i, v_{i+1}$  が凸四角形をなし、同一円周上にはないとき、辺  $\overline{v_0v_i}$  と辺  $\overline{v_{i-1}v_{i+1}}$  のいずれか一方が不正な辺である。

一方、ドロネー図における幾何学的退化は、

(i) 3 点が同一直線上に存在する場合、

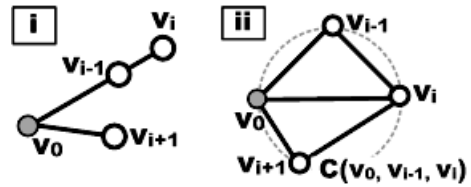


図 3 ドロネー図の幾何学的退化 : (i) 3 点が同一直線上に存在する場合、(ii) 4 点が同一円周上に存在する場合

Fig.3 Degeneracy of Delaunay diagram: (i) 3 vertices lie on a line, (ii) 4 vertices lie on a common circle.

(ii) 4 点が同一円周上に存在する場合、に生じる (図 3)。同図 (i) では、 $v_0, v_{i-1}, v_i$  が同一直線上に存在しており、同図 (ii) では、 $C(v_0, v_{i-1}, v_i)$  上に  $v_{i+1}$  が存在している。このとき、ドロネー図の作成アルゴリズムは一意に動作することができないため、これに対応する必要がある<sup>7)</sup>。

3 章の構成アルゴリズムは、上記の幾何学的特徴をもとに、P2P ノードがドロネー図を自律分散生成するアルゴリズムとなっている。

### 3. 提案アルゴリズム

#### 3.1 ドロネー図と P2P ノードの関係

提案アルゴリズムを実行する P2P ノードは以下の特徴を備えるものとする。

- ユークリッド平面上に位置座標を持つ、一意な識別子を備えた計算主体であり、つねに稼動状態にある。
- 基盤ネットワーク上のオーバーレイ型に P2P ドロネーネットワークを自律分散的に生成・更新する。
- ノード間の通信は隣人ノードを通してマルチホップ通信形態により行う。

次節でアルゴリズムの基本構造について述べる。

#### 3.2 アルゴリズムの基本構造

前提: 各ノードの持つ情報の交換を可能とするため、ここでは連結グラフとして P2P ネットワークを仮定する。つまり、任意の 2 ノード間に少なくとも 1 本の経路が存在すると仮定する。

目標: 連結グラフの各ノードが、フラッディングを用いずに直接接続されたノードとの情報交換を繰り返しることによりドロネー図を作成する。

各ノードが他の隣人ノードの情報を持ち、その時点で、そのノードと双方向の接続関係を持つ。全ノード間において、接続関係を持つノードの情報を互いに保持しているならば、ネットワークは全体として無向グラフとして連結性を持つ。しかし、各ノードは全体グラフの連結性については知りえない。

全ノードが著者らの提案する処理を繰り返し実行す

### Node.constructP2PDelaunayNetwork()

1. List *NIL* //ノード情報リスト
2. List *NeighborNIL* //隣人ノード情報リスト
3. List *NonNeighborNIL* //非隣人ノード情報リスト
4. while(truc)
  5. clearAll *NonNeighborNIL*
  6. addAll *NeighborNIL* to *NIL*
  7. clearAll *NeighborNIL*
  8. sortAsClockwise *NIL*
  9. localDelaunayTriangulation  
(*NIL*, *NeighborNIL*, *NonNeighborNIL*)
10. delegation(*NeighborNIL*, *NonNeighborNIL*)
11. notificationOfTriangulation(*NeighborNIL*)

図4 提案アルゴリズム

Fig. 4 The proposed algorithm.

る．これにより全ノードが，ネットワークの連結性を維持したまま，並列的に辺の繋ぎ換えを実行して全体のドロネー図を作成する．以下に各ノードが実行する提案アルゴリズムの基本的な構成についてまとめる．  
[アルゴリズムの構成要素]

図4は各ノードが自律的に実行する提案アルゴリズムの全体である．これは以下に示す3つの処理要素から構成される．

- (i) 局所ドロネー化
- (ii) 委譲
- (iii) 三角化通知

局所ドロネー化(図4内9行目)は，保持しているノード情報から自身の周囲に局所的に正しいドロネー図を作成し，その隣人関係に基づいて通信経路を確立する処理である．そのとき，隣人関係にないノード情報は自身にとっては不要であるが，その情報を必要とするノードが他に存在する．委譲(図4内10行目)は，その不要なノード情報を，必要とするノードへ届ける処理である．また，局所ドロネー図上で接続関係にあるノードどうしが互いに存在を知り，通信可能であるという保証はない．三角化通知(図4内11行目)は，その隣人ノード間の接続を促すために隣人ノードの存在を通知する処理である．

これら一連の処理を全ノードが繰り返し実行し，ドロネー図状の接続関係を持つネットワークを構成する．次節において，利用するデータ・コンテナの定義とこの3つの処理について詳細な説明を行う．

### 3.3 アルゴリズムの詳細

#### [データ構造]

連結リスト構造のデータ・コンテナを3つ定義する．

- ノード情報リスト
- 隣人ノード情報リスト
- 非隣人ノード情報リスト

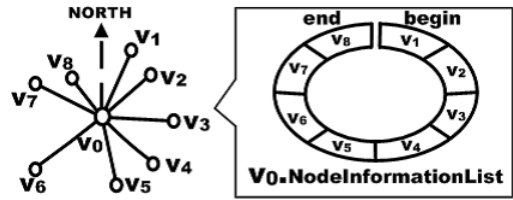


図5 ノード情報リストの構造

Fig. 5 Data structure of NodeInformationList.

図4内1行目において，ノード情報リストを定義する．これには図5に示すように，自身の保持するノード情報を真北から時計回りに格納する．ノード情報リスト内のノード情報に対して，局所ドロネー化の処理を実行する．図4内2行目で定義される隣人ノード情報リストには，局所ドロネー化により，自身とドロネー隣接すると判定されたノード情報(隣人ノード情報)を格納する．各ノードは，自身の隣人ノード情報リスト内に情報が格納されているノードと接続関係を構成する．図4内3行目で定義される非隣人ノード情報リストには，局所ドロネー化により，隣人ノードではないと判定されたノード情報(非隣人ノード情報)が格納される．隣人ノード情報リスト，非隣人ノード情報リストは図4内5~8行目において，ノード情報リストは次に述べる局所ドロネー化が終了した時点(図6内12行目)において初期化される．

#### [局所ドロネー化]

図4内9行目で処理が行われる局所ドロネー化は，保持するノード情報から周囲に局所的なドロネー図を作成し，それに基づいて通信経路を確立する処理である(図6)．時計回りにソートされたノード情報リストから順にノード情報を取り出し，外接円条件により，ドロネー隣接するノードとしないノードに判別する．これらはそれぞれ，隣人ノード情報リスト，非隣人ノード情報リストに格納される．そして，隣人ノードと接続関係を構成し，非隣人ノード情報は後で述べる委譲の処理に渡される．

図6に局所ドロネー化の手順を示す．また，この中で呼び出される部分処理の関連は図7のとおりである．以下，図を用いながら局所ドロネー化の部分処理について述べる．

#### ・外接円判定

ノード情報リストから自身を中心とした隣り合う2つの三角形を順に取り出し外接円条件により辺フリップを行うか判定する．判定の結果，辺フリップが行われなければ，さらに次の三角形との判定に移行する．この処理は図6内の5~8行目にあたる．

図8内(i)では，2つの三角形の共有辺  $v_0v_{i+1}$  は不正

```

Node.LocalDelaunayTriangulation(List NIL,
    List NeighborNIL, List NonNeighborNIL)
1. for i = 0 to NIL.end()
2.   if endCondition()
3.     break
4.   set v0info, v1info, v2info, v3info from NIL[ i ]
5.   if isInCircumcircle
        (Δ(v0info, v1info, v2info), v3info)
6.     add v2info to NonNeighborNIL
7.     remove v2info from NIL
8.     i = i - 1
9.   else //外接円条件, または例外条件に該当した場合
10.    i = i + 1
11.  addAll NIL to NeighborNIL
12.  clearAll NIL
    
```

図 6 局所ドロネー化アルゴリズム

Fig. 6 Local Delaunay Triangulation Algorithm.

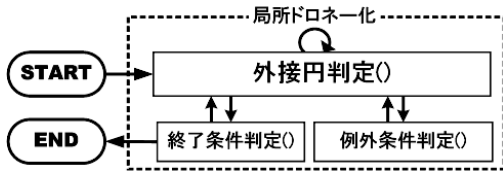


図 7 局所ドロネー化：内部処理の関連  
Fig. 7 Component relations in LDT.

でない」と判定されている。そのため次の、 $\Delta v_0 v_{i+1} v_{i+2}$  と  $\Delta v_0 v_{i+2} v_{i+3}$  の共有辺  $\overline{v_0 v_{i+2}}$  の判定に進んでいる。

一方、辺フリップが行われた場合には、その辺を自身とともに構成するノードの情報を非隣人ノード情報リストに移動する。そして、次に判定する三角形は順序を遡って取り出す。これは隣り合う三角形について、新しい三角形の影響を再検査する必要があるためである。ただし、ノード情報リストの初めて取り出した三角形の比較において辺フリップが行われた場合には、遡らずに再び同じ三角形との比較を行う。

図 8 内 (ii) では、 $\Delta v_0 v_i v_{i+1}$  と  $\Delta v_0 v_{i+1} v_{i+2}$  の共有辺  $\overline{v_0 v_{i+1}}$  が不正であると判定されている。そのため、三角形を 1 つ遡り  $\Delta v_0 v_{i-1} v_i$  と  $\Delta v_0 v_i v_{i+2}$  の判定を行っている。

これらの規定に従い判定を繰り返す。図 9 は外接円判定が進行している様子を表している。丸で囲まれたノード情報  $v_6$  まで処理が進行している。また、灰色のノード情報  $v_4$  は非隣人ノードであると判定されたため、非隣人ノード情報リストに移されている。

これらの処理により、局所ドロネー化が以降で述べる終了条件により終了するときには、非隣人ノード情報リストに非隣人ノード情報が格納され、ノード情報リストには隣人ノード情報のみが残る。これらの処理については後で述べる。

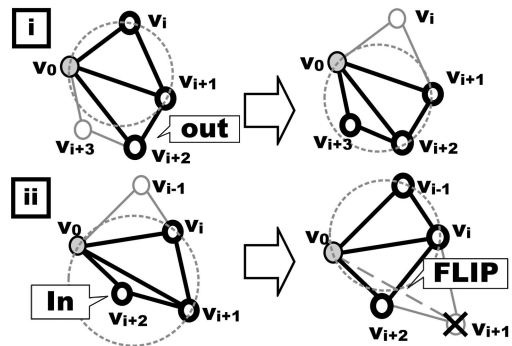


図 8 外接円判定：(i) 辺フリップしない場合、(ii) 辺フリップする場合

Fig. 8 Judge circum circle criterion: (i) proceed to next  $v_0 v_{i+2}$  without EdgeFlipping, (ii) flip edge  $v_0 v_{i+1}$  to  $v_i v_{i+2}$ .

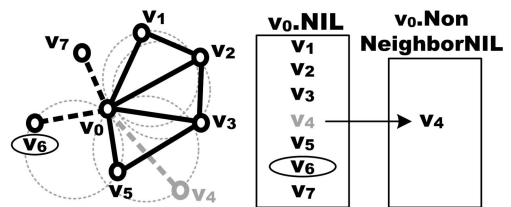


図 9 局所ドロネー化：外接円判定が進行する様子

Fig. 9 In LDT,  $v_4$  is judged as a non-Delaunay neighbor, and moved from  $v_0.NIL$  to  $v_0.NonNeighborNIL$ .

・例外条件判定

局所ドロネー化の内部に、外接円判定の際に発生する例外条件に対応するための処理を用意する。これらの処理は、図 6 内では省略しているが、5~8 行目に加えられるものである。

- 例外条件 1：隣り合う三角形が取り出せない場合
- 例外条件 2：同一直線上に自身を含め 3 ノード以上が存在する場合（幾何学的退化）
- 例外条件 3：隣り合う三角形の 4 頂点ノードが同一円周上に存在する場合（幾何学的退化）

例外条件 1 は  $\Delta v_0 v_i v_{i+1}$  と  $\Delta v_0 v_{i+1} v_{i+2}$  を取り出した際に、 $v_0, v_i, v_{i+1}, v_{i+2}$  からなる四角形が凸でない場合に該当する。2 章で述べたように、凸四角形をなしていない場合には辺フリップを行うことができない。そのため、これに該当する場合、判定を行わずに次の三角形の処理へ移行する。

例外条件 2 に該当した場合は、2 章で述べた幾何学的退化 (i) が生じている。このときは、自身から近いノードとのみ接続関係を構成する。例外条件 3 は、幾何学的退化 (ii) に該当する場合である。このときは、近傍ノード間でノード識別子を利用して一意な接続関係を構成する。つまり、 $v_0, v_i, v_{i+1}, v_{i+2}$  が同一円

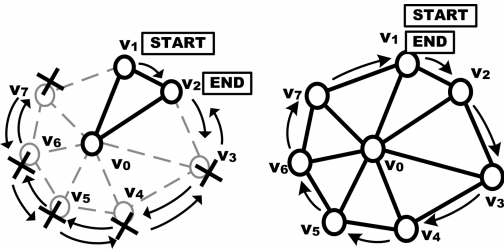


図 10 終了条件判定：終了条件 1 (左) と終了条件 2 (右)  
Fig.10 Termination conditions 1, 2.

周上に存在するとき，ノード識別子の番号の若い順に 3 ノードがドロネー三角形を構成する．このことから，各ノードは隣り合う 2 つの三角形の共有辺を構成する自身と隣人ノードの識別子のどちらか一方でも，4 番目に若い識別子（つまり，最も値が大きい識別子）を持つノードであれば，辺フリップを行う．

例外条件 2，例外条件 3 の場合の処理（摂動処理）については，3.5 節において詳しく述べる．

・終了条件判定

外接円判定は，

(終了条件 1) 辺フリップが繰り返され，ノード情報リスト内のノード情報が 2 以下となった場合，  
(終了条件 2) すべての三角形について不正な辺を含まないことが検出された場合，  
に終了する．図 6 内 2~3 行目にあたる．

図 10 の左図は（終了条件 1）により終了する例である． $v_1$  から外接円判定を行い，辺フリップが繰り返されノード情報が非隣人ノード情報リストに移された結果，ノード情報リスト内のノード情報の数が 2 以下となっている．このとき，次に検査を行う 2 つの三角形を取り出せないため，処理を終了している．

図 10 の右図は（終了条件 2）により終了する例である．同様に外接円判定を進め，最後の  $\triangle v_0 v_7 v_1$  の検査が済み，次に判定する三角形が存在しないため，すべての三角形の検査が完了したとして，終了されている．また，最後に取り出す三角形において，前述の（例外条件 1）に該当して検査が行われなかった場合にも，検査済みとして外接円判定を終了する．

このとき，ノード情報リスト内にはドロネー隣接すると判定されたノード情報のみが残されている．ノード情報リストの内容を，すべて隣人ノード情報リストに移動する．この時点で，隣人ノード情報リスト内の隣人ノードと接続し，通信経路を確立する．これは図 6 内 11~12 行目にあたる．そして，局所ドロネー化処理を終了する．

```
Node.delegation(List NeighborNIL, List NonNeighborNIL)
1. for  $i = 0$  to NonNeighborNIL.end()
2. send NonNeighborNIL[  $i$  ] to NearestNeighborNode
```

図 11 委譲アルゴリズム

Fig.11 Delegation of Node Information Algorithm.

```
Node.notificationOfTriangulation(List NeighborNIL)
1. for  $i = 0$  to NeighborNIL.end();
2. set  $vainfo, vb$  from NeighborNIL[  $i$  ];
3. send NeighborNIL[  $i$  ] to getNode(NeighborNIL[  $i + 1$  ])
4. send NeighborNIL[  $i + 1$  ] to getNode(NeighborNIL[  $i$  ])
```

図 12 三角化通知アルゴリズム

Fig.12 Notification of Triangulation Algorithm.

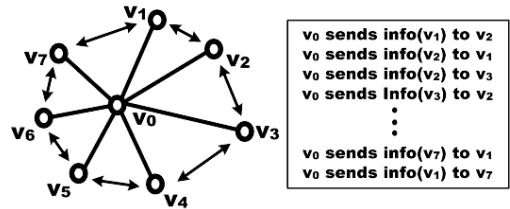


図 13 三角化通知

Fig.13 Notification of Triangulation

[ 委譲 ]

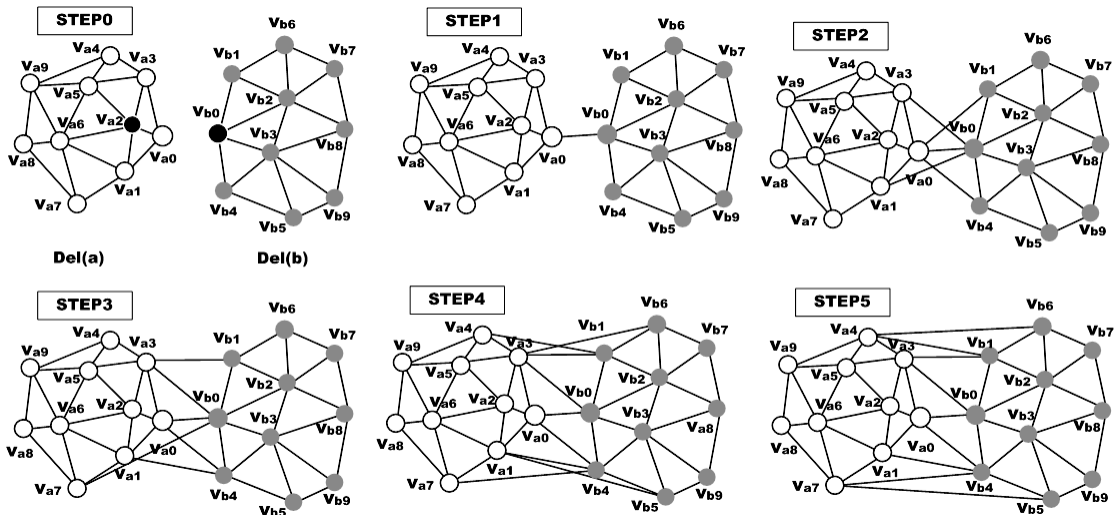
委譲とは，図 4 内 10 行目において実行される，非隣人ノードの情報を，その非隣人ノードに最も近い隣人ノードに渡す処理である．ノード情報は繰り返し委譲されることで，そのノードと接続関係を構成するノードまで最終的に到達する．委譲の手順を図 11 に示す．

局所的な接続関係を一部切断するとき，それによって全体ネットワークが分断されないように，切断以前の接続関係を保証する必要がある．そのため，各ノードは切断した非隣人ノードの情報を，自身と接続が確定している隣人ノードに対してのみ委譲する．これにより，自身と非隣人ノード間の直接の接続を断つても，隣人ノードを介することで，以前の接続関係を保証することができる．

[ 三角化通知 ]

三角化通知は，図 4 内 11 行目において，局所ドロネー化により得られた隣人ノードの情報を互いに三角形形状の接続関係を持つノードへ通知する処理である．この手順を図 12 に示す．図 13 では， $v_0$  が自身の隣人ノードに対して三角化通知を行っている．なお， $info(v_i)$  は  $v_i$  を指すノード情報を意味する． $v_0$  は， $v_1$  の情報を  $v_2$  に， $v_2$  の情報を  $v_1$  に通知し， $v_2$  の情報を  $v_3$  に， $v_3$  の情報を  $v_2$  に通知し，この処理を  $v_0$  の隣人ノードのすべてに対して行い互いの接続を促す．

これにより隣人ノード間で相互の存在が認識され，

図 14 ドロネー図  $Del(a)$ ,  $Del(b)$  の融合過程Fig. 14 Stepwise merging process of 2 distinct Delaunay diagrams  $Del(a)$ ,  $Del(b)$ .

接続関係が構成される。

### 3.4 P2P ドロネーネットワークの融合

提案アルゴリズムによって2つのP2Pドロネーネットワークを1つに融合させることができる。両方のネットワーク間で互いのノード情報を持ち合ったとき、委譲処理の繰返しによって両方のネットワーク中で最も近いノードどうしが接続される。さらに、局所ドロネー化と委譲、三角化通知を繰り返すことで次第に2つのネットワークは融合して1つになる。このように独立に構成された2つのP2Pドロネーネットワークを融合して大きなP2Pドロネーネットワークを構成することができる。これについて、例を示す。

#### [2つのドロネー図の融合例]

2つのP2Pドロネーネットワーク( $Del(a)$ ,  $Del(b)$ )を、連結することで1つのネットワークに融合する過程を、図14を用いて説明する。

**STEP0:**  $Del(a)$  内のノード  $v_{a2}$  に、 $Del(b)$  内のノード  $v_{b0}$  の情報を与える。 $v_{a2}$  は局所ドロネー化の結果、 $v_{b0}$  を非隣接と判定し、 $v_{a0}$  にノード情報を委譲する。

**STEP1:**  $v_{a0} - v_{b0}$  間に接続が構成される。この接続によって、 $Del(a)$  と  $Del(b)$  は連結され融合が始まる。 $v_{a0}$  は、 $v_{a1} - v_{b0}$  間と、 $v_{a3} - v_{b0}$  間に三角化通知を行う。 $v_{b0}$  も同様に、 $v_{a0} - v_{b1}$  間と、 $v_{a0} - v_{b4}$  間に三角化通知を行う。

**STEP2:**  $v_{a1} - v_{b0}$  間、 $v_{a0} - v_{b4}$  間、 $v_{a3} - v_{b0}$  間、 $v_{a0} - v_{b1}$  間に接続が構成される。そして、各ノードが新しく接続されたノード情報を、それぞれの隣人ノードに三角化通知を行う。

**STEP3:**  $v_{a1} - v_{b4}$  間、 $v_{a3} - v_{b1}$  間、 $v_{a7} - v_{b0}$  間に接続が構成され、 $v_{a1} - v_{b0}$  間、 $v_{a0} - v_{b1}$  間の接続がフリップされる。そして、各ノードが三角化通知を行う。

**STEP4:**  $v_{a1} - v_{b5}$  間、 $v_{a3} - v_{b6}$  間、 $v_{a4} - v_{b1}$  間、 $v_{a7} - v_{b4}$  間に接続が構成され、 $v_{a7} - v_{b0}$  間の接続がフリップされる。そして、三角化通知を行う。

**STEP5:**  $v_{a4} - v_{b6}$  間、 $v_{a7} - v_{b5}$  間に接続が構成され、 $v_{a1} - v_{b5}$  間、 $v_{a3} - v_{b6}$  間の接続がフリップされる。この段階で融合が完了する。

### 3.5 アルゴリズムの特徴と議論

提案アルゴリズムの特徴を以下に述べる。

#### [グラフの連結性と並列的な逐次添加法]

3.2節で述べたように、提案アルゴリズムの実行過程においては、グラフの連結性を前提としている。非隣人ノード情報を破棄する際には、すでに自身と辺で連結されている隣人ノードへ情報を委譲しておくことで、非隣人ノードへの経路を必ず確保する。この動作を行わずに非隣人ノード情報を破棄すれば、グラフの連結性が損なわれる可能性がある。この手法により、各ノードが並列的に局所ドロネー化を行っても、全体のグラフとしての連結性は保持される。

一方、ドロネー図の一般的な作図方法である逐次添加法<sup>3)</sup>では、新しく添加された母点の位置から近い母点を探し出す処理を繰り返す、新しい母点を含めたドロネー図を再構成する。この処理をP2P環境に適應させ並列に実行するために、提案アルゴリズムでは各ノードが自律的に新しいノードとのドロネー隣接性を判定しながら情報の委譲を繰り返す。グラフの連結性

を用いて、新しいノードの情報は最終的にドロネー隣接するノードにたどりつき、接続関係が確立される。グラフの連結性が保たれていない場合、ノード情報を適切な地点まで委譲することができず、P2P ドロネーネットワークを正しく構成することができない。

#### [初期状態の構成法]

提案アルゴリズムの前提である連結グラフを実現するには、全ノードがアクセス可能なリフレクタの役割をするノードを1つ存在させ、各ノードが以下のように手順を行う方法が考えられる。各ノードはそのノードにアクセスし、自身のIDと位置座標を登録する。同時に直前に登録したノードのIDと位置座標を取得する。これにより、順次登録したノードが数珠繋ぎに連結され、初期状態を構成できる。この構成法には、基盤としてインターネットが適しているものと考えられる。それにより、P2P ドロネーネットワークはIPのオーバーレイネットワークとなる。

#### [ノードの新規追加・削除]

P2P 環境ではノードの新規参加、離脱が頻繁に起こる。また、P2P ドロネーネットワークを空間情報システムの基盤として利用するときには、新たな空間の情報を持つノードが新規に参加することが、ネットワーク全体で管理する空間の拡張に相当する。これらの観点から、P2P ドロネーネットワークもノードの新規追加・削除に対応する必要がある。

P2P ドロネーネットワークに新規にノードを追加する場合、ネットワーク内の少なくとも1ノードに新規ノードの情報を与える。情報を与えるノードの位置は新規ノードの位置とは無関係でよい。ネットワーク内の全ノードが提案アルゴリズムを繰り返し実行することで、新規ノードの情報は、最初に情報を与えたノードを入り口として、ドロネー隣接するノードまで委譲されていく。そして、それらとドロネー図に基づいた接続関係を構成し、P2P ドロネーネットワークへの参加が完了する。

ここでも、初期状態のネットワークやノード追加前のP2P ドロネーネットワークが連結グラフを構成している点が重要な役割を果たしている。グラフの連結性が保証されることで、新規ノードの情報は、そのノードとドロネー隣接するノードへ必ず到達させることができる。

ノードの新規追加による影響が及ぶ範囲は、ネットワーク参加の入り口としたノードの位置が新規ノードの位置から近い場合には、その近傍に小さくとどまる。逆に遠い場合には、新規ノードの情報がドロネー隣接するノードに到達するまでに経由するノードに影響が

及ぶ。

一方、ノードの削除については、短時間に大量のノードが削除される場合や、局所的に少数のノードが集中して削除される場合に、グラフが分断される。これに対応するためには、提案アルゴリズムとは別途の仕組みが必要である。これについては今後の検討課題としたい。

#### [P2P 環境での幾何学的退化への対応]

2章で述べたようにドロネー図に対する幾何学的退化は、(i)、(ii)の2通りの場合に対応している。

##### (i) 3ノード以上が同一直線上に存在する場合

この場合は、処理主体のノードから見て最も近いノードとのみ接続関係を持つアルゴリズムを局所ドロネー化に組み込んでいる。

##### (ii) 4ノードが同一円周上の位置にある場合

円ボロノイ図の作成法<sup>7)</sup>を応用して、各ノードに厚みを持たせることでノードの位置を仮想的に少し移動する摂動処理を実行する。これにより、ドロネー図が作成可能になる。P2P 環境でこの摂動処理を適用するには、4つのノード間で摂動処理の対象を一致させる必要がある。ここでは、ネットワーク内で一意に定められたノード識別子を利用することを想定し、4ノード間で摂動させるノードを決定している。この処理に利用するノードの識別子は一意であるため、どのような状況においても、4ノード間で共通な摂動処理が可能である。

#### [非同期なノード情報の交換への対応]

提案アルゴリズムでは、ノードの3種類の処理のうち三角化通知、委譲によるノード情報の交換は同期的に行われることを想定しているが、ノード情報の到達ごとにこれらの処理を実行することで、非同期なノード情報の交換にも対応することができる。

また、ノードが局所ドロネー図を作成する段階では近傍ノードとの通信は行われない。このため、ある段階においては各ノードの作る局所ドロネー図が全体ドロネー図の一部として正しくない場合がある。しかし、前節で述べたように、提案アルゴリズムを繰り返し実行することで局所ドロネー図どうしは融合され、最終的に全体のドロネー図が正しく作成される。

接続関係の構成に用いる局所ドロネー図作成の段階では、近傍ノードと隣人表の共通化を行わないため、それらと同期をとる必要がない。このことからネットワークの構成、保守において、デッドロックが生じることはなく、ロック機構は用いられていない。

#### [ノード情報の最近傍ノードへの到達]

P2P ドロネーネットワークでは、幾何的な経路選択



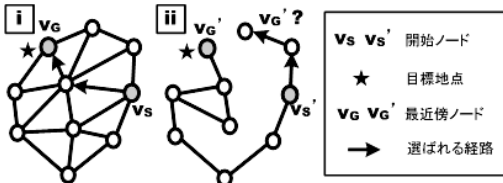


図 15 目標地点の最近傍ノードへの経路選択

Fig. 15 Route selection to the nearest node to a given target location.

が可能である．平面上の任意の目標地点に対して，始点ノードから出発し，より近いノードを順次たどれば，必ずその地点に最も近いノードに到達することができる．つまり，目標地点を含むボロノイ領域を管轄するノードへの経路を作成することができる．この特徴を利用すれば，目的の地点や領域へクエリを伝播させ，情報の問合せを行うことができる．しかし，ノードが提案アルゴリズムを実行せず，ネットワークが単に連結グラフである場合には，任意のノードにおいて目標地点により近い方向への経路が存在するとは限らないため，最も近いノードに到達できないことがある．

図 15 は  $(v_s, v_s')$  から，目標地点（印）の最近傍ノード  $(v_g, v_g')$  を目指した例である．(i) は P2P ドローネーネットワーク，(ii) はドローネー図ではない連結グラフのネットワークである．(i) では目標地点により近い隣人ノードを経由していくことで  $v_g$  に到達できる．一方，(ii) でも同様の経路選択を行っているが， $v_g'$  に到達することはできない．

P2P ドローネーネットワークでない (ii) のような形状のネットワークでは，最近傍ノードまでの経路に迂回路を選択しなければならない場合がある．これに対応するには，複雑な経路選択アルゴリズムを用いる必要がある．このことから，P2P ネットワークがドローネー図状のトポロジを備えることは効果的であるといえる．

#### 4. P2P ドローネーネットワーク形成過程の評価

本章では，提案アルゴリズムを用いた P2P ドローネーネットワークの形成過程をシミュレーションにより評価する．

##### 4.1 局所ドローネー化の評価

まず，ノード単体の動きを評価する．そのため，単一ノードにかかる局所ドローネー化の計算負荷を，ネットワークを切断した状態でみた．これはつまり，評価対象のノードが新たにノード情報を受信することがない状態である．

単一の評価対象ノードに，他の平面上に一樣に配置

したノードの情報を与えて，局所ドローネー化の負荷を評価する．与えるノード情報を 50,000 から 900,000 まで変化させ，外接円判定と辺フリップの回数，隣人ノードと非隣人ノードの数を計測した．

与えるノード情報数を  $N$  とするとき，理論上，外接円判定の回数は  $2N$  となる．計測の結果，ノード数の増加にともない，それぞれの回数は理論どおり線形に増加することが確認できた．計算時間は，900,000 ノードを与えたとき，およそ 4.3 秒で終了した（CPU：Xeon 2.8 GHz，メモリ：2 GByte）．また，局所ドローネー化終了後の隣人ノード数は，与えたノードの数にかかわらず平均 6 以下となり，局所ドローネー図が 2 章で述べた (ii) の性質を持つことが確認できた．

##### 4.2 P2P ドローネーネットワーク形成過程の評価

次に，すべてのノードがネットワークに接続された状態で，P2P ドローネーネットワークを形成するときの，個々のノードとネットワークにかかる負荷を評価する．

初期状態を，各ノードに他ノードの情報を 2 つずつ与えた連結グラフの P2P ネットワークとする．ノード間はそれぞれの接続関係，位置関係とは独立にランダムに接続させた．各ノードは単位時間ごとに提案アルゴリズムを 1 回ずつ同期的に実行する．すべてのノードにおいて状態変化が生じなかった場合，アルゴリズムが収束したとしてシミュレーションを終了する．なお，ここで用いる情報交換回数とは，委譲と三角化通知の回数を足したものである．

##### ・ネットワーク構成法のスケーラビリティの評価

図 16 に，総ノード数 1,000 から 10,000 のときの，収束までに各ノードが行う辺フリップの回数の平均値と最大値を示す．同様に，図 17 に，情報交換回数の平均値と最大値を示す．総ノード数の増加とともに，平均値，最大値の増加率が小さくなっていくことが確認できる．これにより，提案アルゴリズムによるネットワーク構成法には，ノード数のスケーラビリティが確認できる．

##### ・個々のノードにかかる負荷の評価

図 18，図 19，図 20，図 21 に，辺フリップ回数，情報交換回数の平均値，分散を示す．最大値をとった後，単調に減少し収束に向かう．図 21 も同様の動きを示すが，ステップ 16～30 において分散が一時大きくなるのは，ネットワークが完成に近づくにつれて，三角化通知の回数が増加するためであると考えられる．最も多いとき，1 ステップあたり平均約 5.5 から 6.2 回程度の辺フリップと，約 16.2 から 17.8 回の情報交換が行われている．また，分散も平均値と同じく増加

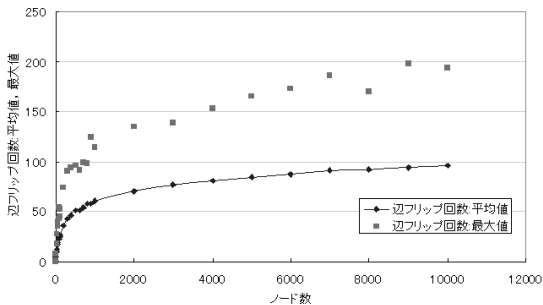


図 16 ノード数に対する辺フリップ回数の平均値, 最大値 (一様分布)

Fig. 16 Movement of average and maximum EdgeFlipping frequency to the number of nodes (uniform distribution).

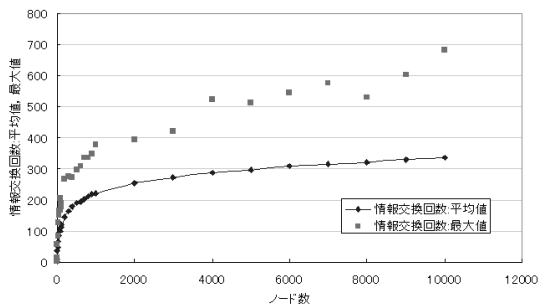


図 17 ノード数に対する情報交換回数の平均値, 最大値 (一様分布)

Fig. 17 Movement of average and maximum DataTransfer frequency to the number of nodes (uniform distribution).

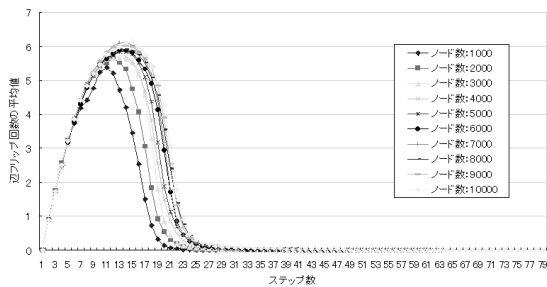


図 18 ステップごとの辺フリップ回数の平均値 (一様分布)

Fig. 18 Movement of average EdgeFlipping frequency to time step (uniform distribution).

減少しており, 全体的にみて極端に大きな負荷がかかるノードが多く存在しない.

・収束までのステップ数

図 18~図 21 から同様に, アルゴリズム収束までに要したステップ数が確認できる. これらによれば, ノード数が 1,000 で 30 ステップ, また, ノード数 10,000 に対して 81 ステップである. 総ノード数の増加に対して, 収束までに要するステップ数の増加率が下降して

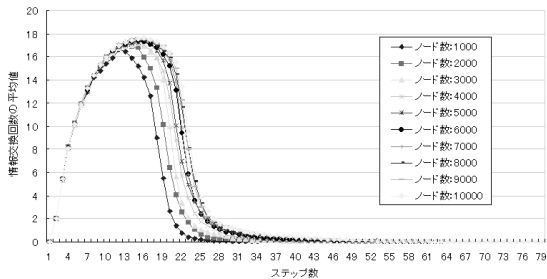


図 19 ステップごとの情報交換回数の平均値 (一様分布)

Fig. 19 Movement of average DataTransfer frequency to time step (uniform distribution).

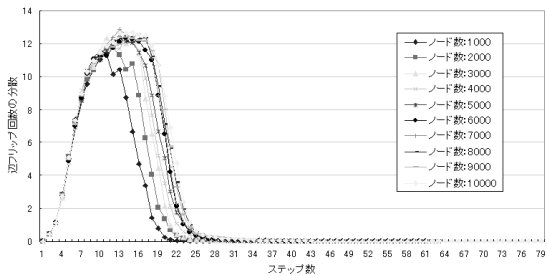


図 20 ステップごとの辺フリップ回数の分散 (一様分布)

Fig. 20 Movement of variance of EdgeFlipping frequency to time step (uniform distribution).

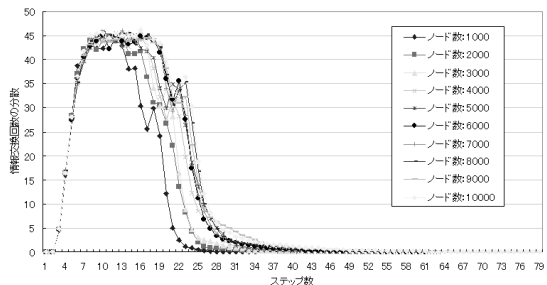


図 21 ステップごとの情報交換回数の分散 (一様分布)

Fig. 21 Movement of variance of DataTransfer frequency to time step (uniform distribution).

いる. この収束ステップ数は, 委譲によってノード情報が近傍ノードに到達するまでに要するステップ数に依存しているものと考えられる. 到達までに要するステップ数はネットワークの直径に依存している. ノードの配置密度が一定ならば, 直径は総ノード数  $N$  に対して  $\sqrt{N}$  に比例するため, このような結果になったものと考えられる.

また, このシミュレーションにより以下の点が同時に確認された.

- (i) 初期に与えるノード情報の数が, ノード間で差をつけた場合にも, 各ノードがノード情報を交換することで数ステップ後には, 保持するノード

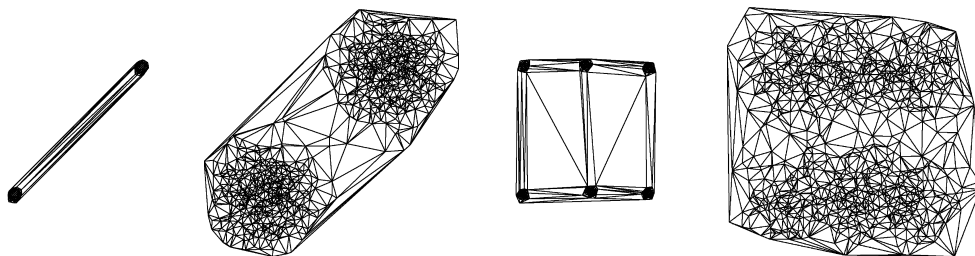


図 22 クラスタ分布 (左から, “クラスタ数: 2, 分散: 0.01”, “クラスタ数: 2, 分散: 0.1”, “クラスタ数: 6, 分散: 0.01”, “クラスタ数: 6, 分散: 0.1”)

Fig. 22 Sample clustered distributions ('2 clusters with variance 0.01', '2 clusters with variance 0.1', '6 clusters with variance 0.01', '6 clusters with variance 0.1', left to right).

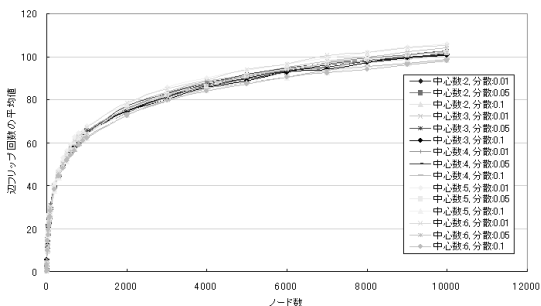


図 23 ノード数に対する辺フリップ回数の平均値 (クラスタ分布)  
Fig. 23 Movement of average EdgeFlipping frequency to the number of nodes (clustered distribution).

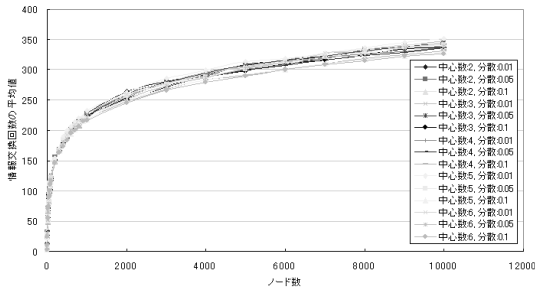


図 24 ノード数に対する情報交換回数の平均値 (クラスタ分布)  
Fig. 24 Movement of average DataTransfer frequency to the number of nodes (clustered distribution).

ド情報の数はほぼ等しくなった。

- (ii) 初期に与えるノード情報数が大きい場合は, 小さい場合より早く P2P ドロネーネットワークが構築された. しかし, その情報数にかかわらず, 隣人ノード数の平均もアルゴリズムの 1 回目の実行で 6 以下まで下がり, その付近で推移した.

### 4.3 ノードの位置分布の影響の評価

ノードの位置分布に偏りを持たせた状態で, P2P ドロネーネットワーク形成時の個々のノードとネットワークにかかる負荷を評価する. ここでは各ノードの配置をクラスタ分布により与え, クラスタ中心数と分散を変化させる (図 22).

図 23, 図 24 は, 総ノード数を変化させた場合の辺フリップ回数と情報交換回数の平均値を示している. 一様分布のとき (図 16, 図 17) と同様に, 総ノード数の増加に対して増加率は減少する. また, 値も一様分布の場合と比較して大きな差はない. このことから, 提案アルゴリズムによる構成法は, ノードの分布が偏った場合にも, 総ノード数に対してスケラビリティを持つことが確認できた.

次に, 総ノード数が 5,000 ノードのときのノードの

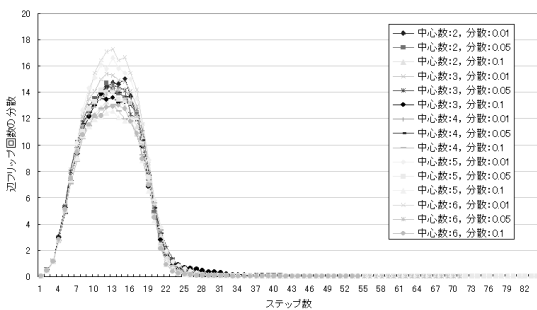


図 25 5,000 ノードにおける, ステップごとの辺フリップ回数の分散 (クラスタ分布)  
Fig. 25 Movement of variance of EdgeFlipping frequency to time step (uniform distribution, 5,000 nodes).

各分布状況における辺フリップ回数と情報交換回数の分散を図 25, 図 26 に示す. この結果から, ノードの位置分布が偏るときは, 一様分布の場合 (図 20, 図 21) と比較して分散が大きくなるのが分かる. また, クラスタ数が小さい場合に収束ステップ数が大きくなる傾向がみられた. クラスタ数が小さいとき, ノードは直線状に配置される. そのためネットワークの直径が大きくなり, ノード情報の委譲に多くのステップ数を要するためであると考えられる.

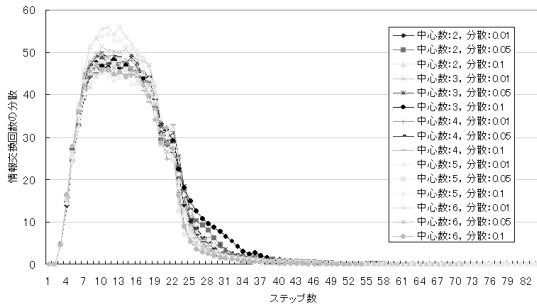


図 26 5,000 ノードにおける，ステップごとの情報交換回数の分散（クラスタ分布）

Fig. 26 Movement of variance of DataTransfer frequency to time step (uniform distribution, 5,000 nodes).

[注 1] 提案アルゴリズムにより生成された P2P ドロネーネットワークに対して，検証プログラムを別に作成し，ドロネー図の必要十分条件である外接円条件を満たすこと，ならびにオイラーの定理<sup>8)</sup>における母点数，辺数，面数の関係を満たすことを確認している。

[注 2] 各ノードでの提案アルゴリズムの 1 ステップに要する時間は，ノードの削除に備えて隣接ノードと情報交換する時間と直接には関係ない。ノード間の情報交換は同期的，非同期的のいずれでもよい。しかし，ノードの削除への対応は P2P 環境において重要な問題であるので今後の検討課題としたい。

[注 3] P2P ドロネーネットワーク構築に要するステップ数を減少させるために，P2P ネットワーク構築段階でもネットワークの直径を小さくする接続を構成し，委譲の際に近傍ノードではなく，遠方のノードに委譲する方法が考えられる。

## 5. 関連研究と提案アルゴリズムの利用

### 5.1 関連研究

P2P ネットワークにおいてノード繋ぎ換え手法が研究されている。片山ら<sup>9)</sup>は非構造型で対称型フラッディングを用いた P2P ネットワークにおいて，問合せ時の TTL 値を低く抑えることを目的としてノード間のパケット転送履歴に基づく接続の繋ぎ換えを行っている。また，Shrestha ら<sup>10)</sup>は，ID 空間での近傍を定義し，近傍ノード間での検索木の分散構成手法を提案している。そしてその P2P ネットワークトポロジ上で，問合せの TTL 値に応じて複数の問合せプロトコルによる成否を評価している。

これらの非構造型 P2P ネットワークに対して，本手法では，トラフィックが増加するフラッディングの使用を前提としていない。隣人ノード間での転送を繰り返すことで確実に目的ノードにたどりつく特徴を持

つ構造型 P2P ネットワークとして構成している特徴を持つ。

一方，構造型 P2P ネットワークにおいて位置依存情報を取り扱うモデルとして，分散ハッシュ法 (DHT) を位置座標系への拡張した GHT がある<sup>1)</sup>。GHT では，各ノードは地理的座標系の部分領域となる 2 次元担当領域を持ち，位置に基づくデータは，その名前がハッシュ法により座標系へ写像されて格納されるノードを決定する。このように格納位置を抽象化することで，データ中心型 (data centric) 格納を可能としてノードの故障や移動への対応した耐障害性を生み出し，大規模センサネットワークにおけるデータ散布などへの応用を可能にしている。

また，Tanin ら<sup>2),11)</sup>は，与えられた 3D ユークリッド空間を 8 分木 (Octree) として再帰的に分割し，DHT を用いて得られた部分空間をノードへ格納している。実際に 3D 空間として CG 空間を用いて空間内のオブジェクトをノードへ格納する手法を実装している。また，8 分木の基準レベルに対して，検索時のノード負荷などについて評価している。

上記 2 つの手法は，DHT を用いて位置に依存した情報やオブジェクトの格納しており，ノード数が増加しても構造型の特質が保持される有効な手法であるが，固定された空間を対象としており，空間の拡張を考慮する場合には適用しにくい。また，ノードの追加・削除も可能であるが，参加・離脱の頻繁なネットワークには利用しにくい。

これに対して，提案アルゴリズムは，ネットワーク構成法がボトムアップ的なアルゴリズムになっており，連結グラフである限りドロネー構造に変化し続けるため，空間が拡張される場合にも適用できる。このアルゴリズムを各ノードが実装することで，ノードの自律分散的な動作により，対象とする空間を自在に広げることが可能になる。

一方，ノードの空間内の配置や空間の幾何学的分割に基づく P2P モデルも提案されている。Steiner ら<sup>12)</sup>は，3 次元ユークリッド空間における分散ドロネー三角化アルゴリズムを提案している。同手法も辺フリップを用いてドロネー図を局所的に変化させていくアルゴリズムであるが，従来のドロネー図生成アルゴリズムの拡張であるため，隣人ノード間で繋ぎ換えの処理の際にロックを行って同期をとる必要があると考えられる。また，具体的なアルゴリズムの実現手法をはじめ，母点間のデータの授受などのデータ構造，定量的な解析，幾何学的な退化への対応についての詳細は記述されていない。

また、ドロネー図を用いた P2P ネットワークとして、Li ら<sup>5)</sup> は、Bluetooth のような通信範囲を持つ短距離無線通信機器のルーティング経路生成法を提案している。各ノードは、下位のプロトコルを用いて無線通信範囲内のノード情報を収集し、各ノードの近傍における部分的なドロネーネットワークを重ねて用いる。また、対象とするノード数が小さいため、分散的な生成法は与えているわけではなく、P2P 環境に適用しにくい。また、生成されるドロネー図は部分的なものである。

Araujo ら<sup>13)</sup> は、地理的な位置に基づく資源の探索を可能にするドロネーネットワークである GeoPeer を提案している。さらに、ネットワークの長い半径を考慮した、Long Range Contact (LRC) を構成しルーティング性能の向上を試みている。

また、Hu ら<sup>14)</sup> は、スケーラブルな P2P 型仮想環境の実現を目指し、ポロノイ図、ドロネー図をとりあげて幾何学的な観点からドロネー図を用いた P2P ネットワークについて述べている。しかし、具体的な構成法は示されていない。

## 5.2 提案アルゴリズムの利用

提案アルゴリズムは、ノードを計算主体と見做し、自律分散動作を与えているため、本稿で述べている P2P 環境における幾何学的なネットワーク構成法以外にも、GRID ノード間のネットワーク形成、自律オブジェクト、移動体、自走ロボットなどでの利用、また、実空間、仮想空間での利用、なども期待できる。

また、提案アルゴリズムの実機での実装面では、P2P システム、無線ネットワークノード、センサネットワークノードなどにおいて、数十から数百ノードが実装しやすいものと想定されるが、数値シミュレーションにより、提案アルゴリズムは  $10^5$  のオーダーでの稼働を確認しているため、上記の分野での母点数の大きい対象にも適用できるものと考えられる。

## 6. おわりに

本稿では、スケーラブルな空間情報処理基盤として P2P ドロネーネットワークを提案し、これを自律分散的に構成するアルゴリズムを提案し、その特徴について議論した。シミュレーションでは、提案アルゴリズムにより P2P ドロネーネットワークが構築されることの数値的な確認と、その際のノード数の増加に対する P2P ドロネーネットワーク完成までのステップ数の変化の確認、ノードの負荷とネットワークの負荷、などを確認した。また、ノードの位置に偏りがある場合についても検証した。今後の課題として、本稿でと

りあげられなかったノードの離脱、移動に対する検討、ならびに情報管理基盤としての P2P アプリケーションの実装に関する検討などがあげられる。

謝辞 貴重なご指摘を賜った査読者に衷心より感謝の意を表す。本研究の一部は、平成 15-19 年度文部科学省私立大学学術研究高度化事業オープン・リサーチ・センター整備事業（知識ネットワーク社会創造のための人的・情報環境の構築に関する研究）ならびに学術フロンティア推進事業（合意形成のための認知的・数理的情報処理システムの構築）の支援を受けた。また、財団法人 C&C 振興財団と原財団による助成を受けた。

## 参 考 文 献

- 1) Ratnasamy, S., Karp, B., Yin, L., Yu, F., Estrin, D., Govindan, R. and Shenker, S.: GHT: A Geographic Hash Table for Data-Centric Storage, *Proc. 1st ACM international workshop on Wireless sensor networks and applications (WSNA '02)*, pp.78-87 (2002).
- 2) Tanin, E., Harwood, A. and Samet, H.: A Distributed Quadtree Index for Peer-to-Peer Settings, *ICDE2005 (International Conference on Data Engineering)*, pp.254-255 (2005).
- 3) Ohnishi, M., Nishide, R. and Ueshima, S.: Incremental Construction of Delaunay Overlaid Network for Virtual Collaborative Space, *Proc. 3rd Conference on Creating, Connecting and Collaborating through Computing (C5)*, pp.77-84, IEEE CS Press (2005).
- 4) 大西真晶, 源元祐太, 加藤宏章, 上島紳一: 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, IEICE Technical Report (DBWS 2005), Vol.105, No.171, pp.173-178 (2005).
- 5) Li, X.-Y., Stojmenovic, I. and Wang, Y.: Partial Delaunay Triangulation and Degree Limited Localized Bluetooth Scatternet Formation, *IEEE Trans. Parallel and Distributed Systems*, Vol.15, No.4, pp.350-361 (2004).
- 6) Stojmenovic, I., Ruhil, A.P. and Lobiyal, D.K.: Voronoi diagram and convex hull based geocasting and routing in wireless networks, *Proc. 8th IEEE International Symposium on Computers and Communication (ISCC'03)*, pp.51-56 (2003).
- 7) 杉原厚吉: 計算幾何工学, 培風館 (1994).
- 8) de Berg, M., van Kreveld, M., Overmars, M. and Schwarzkopf, O.: *Computational Geometry: Algorithms and Applications*, 2nd ed., Springer (2000).
- 9) 片山 肇, 中野宏一, 金子 雄, 春本 要, 西尾章治郎: 応答転送状況を用いた P2P ネットワーク

の繋ぎ換えアルゴリズムの評価, 日本データベース学会 Letters, Vol.4, No.1, pp.69-72 (2005).

- 10) Shrestha, S., Kobayashi, A., Yamaoka, K. and Sakai, Y.: An Efficient Content Location Algorithm for CDN based on Distributed Construction of Search Tree from Contents of Proximal Nodes, IEICE Technical Report (DBWS 2005), Vol.105, No.171, pp.161-166 (2005).
- 11) Tanin, E., Harwood, A., Samet, H., Nutanong, S. and Truong, M.T.: A Serverless 3D World, *Proc. 12th annual ACM international workshop on Geographic information systems*, pp.157-165 (2004).
- 12) Steiner, M. and Biersack, E.: A fully distributed peer to peer structure based on 3D Delaunay Triangulation, *Algotel — Septiemes Rencontres Francophones sur les Aspects Algorithmiques des Telecommunications* (2005).
- 13) Araujo, F. and Rodrigues, L.: GeoPeer: A Location-Aware Peer-to-Peer System, *Proc. 3rd IEEE International Symposium on Network Computing and Applications (NCA'04)*, pp.39-46 (2004).
- 14) Hu, S.-Y. and Liao, G.-M.: Scalable Peer-to-Peer Networked Virtual Environment, *Proc. ACM SIGCOMM 2004 Workshops on NetGames '04*, pp.129-133 (2004).

(平成 17 年 9 月 21 日受付)

(平成 18 年 1 月 5 日採録)

(担当編集委員 原 隆浩)



大西 真晶 (学生会員)

昭和 53 年生。平成 14 年関西大学総合情報学部卒業。平成 16 年同大学院総合情報学研究科博士前期課程修了。現在, 博士課程後期課程在学中。仮想空間の高度利用, P2P ネットワークの研究に従事。電子情報通信学会, IEEE 各学生会員。

ネットワークの研究に従事。電子情報通信学会, IEEE 各学生会員。



源元 佑太

昭和 55 年生。平成 16 年関西大学総合情報学部卒業。同年同大学院博士前期課程へ進学。P2P システムに関する研究に従事。



江口 隆之

昭和 57 年生。平成 17 年関西大学総合情報学部卒業。同年同大学院博士前期課程へ進学。P2P システムに関する研究に従事。



加藤 宏章

昭和 60 年生。関西大学総合情報学部在学中。P2P システムに関する研究に従事。



西出 亮 (学生会員)

昭和 54 年生。平成 14 年関西大学総合情報学部卒業。平成 16 年同大学院総合情報学研究科博士前期課程修了。現在, 博士後期課程在学中。仮想空間の高度利用, e-ラーニングシステムの研究・開発に従事。ACM, IEEE 等の会員。

システムの研究・開発に従事。ACM, IEEE 等の会員。



上島 紳一 (正会員)

昭和 30 年生。昭和 53 年京都大学工学部数理工学科卒業。昭和 58 年同大学院工学研究科博士課程単位取得退学 (京都大学工学博士)。現在, 関西大学総合情報学部教授。マルチメディア情報システム, 柔軟な情報ベースに関する研究に従事。電子情報通信学会, ACM, IEEE 等の会員。

マルチメディア情報システム, 柔軟な情報ベースに関する研究に従事。電子情報通信学会, ACM, IEEE 等の会員。