

アドホックネットワークにおけるクラスタ構成法

谷口 博人^{†*} 井上美智子[†] 増澤 利光^{††} 藤原 秀雄[†]

Clustering Algorithms in Ad Hoc Networks

Hirohito TANIGUCHI^{†*}, Michiko INOUE[†], Toshimitsu MASUZAWA^{††},
and Hideo FUJIWARA[†]

あらまし 本論文では、移動端末だけからなる分散移動システムであるアドホックネットワーク上でのクラスタ構成法を考察する。クラスタ構成法とは、ネットワーク上の全ノードをクラスタヘッドとそれと直接通信可能なノードであるクラスタメンバからなるクラスタに分割することである。移動端末は、計算能力、通信能力などの点でパフォーマンスが低いいため、移動端末にかかる負荷が小さい手法が望まれる。分散システムの問題として、端末の移動や、トポロジーの変化に伴うオーバーヘッドを考慮しなければならない。更に、無線チャネルの帯域幅の空間再利用の観点などから、クラスタ構成をすることによって、階層構造を構築する利点がある。その際、ネットワークで交換する情報量を少なくするためクラスタヘッドを少なくすることや、管理情報の受け渡しを少なくするためクラスタヘッドの変更数を少なくすることが望まれる。本論文では、アドホックネットワーク上にクラスタを構成するクラスタ構成法及び、移動端末の移動などによりトポロジーが変化した場合に対応するクラスタ再構成法を提案する。提案するクラスタ構成法は、トポロジーグラフが密な場合を除き、従来手法に比べてクラスタ数が少ないこと、また、提案するクラスタ再構成法は、従来手法に比べ、クラスタ数が少なく、またクラスタヘッドの変更数が少ないことをシミュレーション実験で示す。

キーワード 分散アルゴリズム, アドホックネットワーク, モバイルコンピューティング, クラスタ構成法

1. ま え が き

近年の移動計算機の普及に伴い、移動端末が基地局を介してネットワークに接続するセルラシステムに加えて、移動端末間で直接通信を行うことのできる分散システムも利用されている。本論文では、移動端末だけからなる分散システムであるアドホックネットワークに関して、ネットワーク上の端末をクラスタに分割するクラスタ構成について考える。クラスタ構成は、ネットワーク上の多数の端末を階層的に効率良く管理するため、また、周波数等の資源を効率良く割り当てるために用いられ、クラスタ構成法に関して多くの研究がなされている [1]~[6]。

アドホックネットワークは、移動端末だけからなるネットワークで、各端末は、地理的あるいは論理的な

通信可能範囲をもつ。互いに通信可能な2端末間は無線チャネルを介して通信を行うことができ、このとき、2端末は隣接すると呼ぶ。クラスタとは端末の集合で、クラスタを管理するクラスタヘッドとクラスタヘッドに隣接するクラスタメンバからなる。ネットワーク上の全端末をクラスタで被覆するようにクラスタヘッドを選択することをクラスタ構成と呼ぶ。ただし、クラスタヘッド間の干渉を避けるために、どの二つのクラスタヘッドも隣接しないことを条件とする。

クラスタ構成は以下の条件を満たすクラスタヘッド集合 H を求める問題と考えることができる。

- 支配性: 各端末は H に属する、または、 H 中の端末と隣接する。

- 独立性: H に属するどの2端末も隣接しない。クラスタヘッドに選択された端末は、クラスタ内及びクラスタ間の接続関係等の情報の維持管理といった負荷を伴うので、クラスタヘッド数は小さい方が望ましい。

既存のクラスタ構成法として、最大ID法 [3]、最大次数法 [4] が提案されている。これらの手法では、端末と無線チャネルからなるトポロジーグラフに対して、

[†] 奈良先端科学技術大学院大学情報科学研究科, 生駒市
Graduate School of Information of Science, Nara Institute of
Science and Technology, 8916-5 Takayama-cho, Ikoma-shi,
630-0101 Japan

^{††} 大阪大学大学院基礎工学研究科, 豊中市
Graduate School of Engineering Science, Osaka University,
1-3 Machikaneyama-cho, Toyonaka-shi, 560-8531 Japan

* 現在, NTTドコモ関西株式会社

各端末に対してクラスタヘッドになる優先度を割り当て、最大の優先度をもつ端末をクラスタヘッドに選択し、隣接端末とともにグラフから削除するという処理を繰り返す。最大 ID 法 [3] は、各端末に固有の優先度を用いる手法でトポロジを考慮しないためクラスタヘッド数を小さくできない。これに対し、最大次数法はトポロジを考慮してクラスタヘッド数を抑えることを目的としている。しかし、トポロジによってはクラスタヘッド数が小さくならない場合もある。本論文では、最大次数法でクラスタヘッド数が小さくならない場合も考慮する新たな優先度を利用した重み付き最大次数法を提案する。シミュレーション実験によって、トポロジグラフが密な場合を除き、提案手法が従来法に比べてより少ないクラスタヘッドを選択することを示す。

アドホックネットワークでは、端末の移動に伴うトポロジ変化への対応も考える必要がある。既にクラスタが構成されているネットワークにおいて、トポロジの変化によって支配性、または独立性が成立しなくなった場合に、クラスタを再構成することを考える。クラスタ再構成では、クラスタヘッド数だけでなく、新たにクラスタヘッドになる、クラスタヘッドでなくなるといったクラスタヘッド変更数も少ない方が望ましい。これは、クラスタヘッドの変更によって、クラスタヘッドが維持管理する情報の交換が必要になるためである。

クラスタ再構成法としては、既存のクラスタ構成法を適用することも考えられる。この場合、クラスタヘッド数は小さく抑えられるがクラスタヘッド変更数は大きくなってしまふ。クラスタヘッド変更数を小さくする手法として、LCC (Least Cluster Change) 法が提案されている [5]。LCC 法では、トポロジが変化した場合に、支配性、または独立性が成り立たない部分だけで局所的にクラスタを再構成する手法である。LCC 法は、クラスタヘッド変更数を抑えることに主眼がおかれ、クラスタヘッド数が大きくなる場合もある。本論文では、LCC 法でのクラスタ再構成法を改良した改良 LCC 法を提案する。提案手法では、クラスタヘッド変更数だけでなくクラスタヘッド数を抑えることも考慮している。シミュレーションによる比較では、提案手法が LCC 法よりクラスタヘッド数を小さく抑えることを示す。また、トポロジ変化とクラスタ再構成を繰り返した場合、結果的に改良 LCC 法の方がクラスタヘッド変更数が小さくなるという結果

を示す。

2. 諸 定 義

2.1 アドホックネットワーク

アドホックネットワーク (以下、ネットワークと呼ぶ) は、有線チャネルやそれを經由する基地局をもたず、移動ノードだけで構成される。各ノードは無線が有効な通信範囲内のノードとだけ通信できる。ネットワークのトポロジをグラフ $G = (V, E)$ で表す。 $V = \{p_1, p_2, \dots, p_n\}$ はノードの集合、 E は無線チャネルの集合である。各ノード p_i は固有の識別子 $id(i)$ をもつ。ノード p_i とノード p_j がそれぞれの通信範囲内に位置し、互いにメッセージを送受信できるなら、無線チャネル (p_i, p_j) が存在し $((p_i, p_j) \in E)$ 、 p_i と p_j は隣接しているという。

2.2 クラスタ構成

クラスタは、ノードの部分集合で、一つのクラスタヘッドとその全隣接ノードからなる。

[定義 1] (クラスタ) ノード p_i をクラスタヘッドとするクラスタとは以下のノード集合 C である。

$$C = \{p_i\} \cup \{p_j | (p_j, p_i) \in E\}$$

クラスタヘッドに隣接するノードをクラスタメンバと呼ぶ。 □

クラスタ構成とは、全ノードを被覆するようにクラスタヘッド集合を選択することである。ただし、クラスタヘッド間の干渉を避けるため、隣接する 2 ノードをとともクラスタヘッドには選択できないとする。

[定義 2] (クラスタ構成) クラスタ構成とは、以下の条件を満たすようにクラスタヘッド集合 H を選択することである。

- 支配性: 各ノードは H に属する、または、隣接ノードが H に属する。
- 独立性: H に属するどの 2 ノードも隣接しない。 □

図 1 にクラスタ構成の例を示す。

また、本論文ではアドホックネットワークを構成するノードの移動によりトポロジが変化したときに、クラスタを再構成する問題を考える。

[定義 3] (クラスタ再構成) グラフ G において、ネットワークのクラスタ構成がなされているとする。ここで、グラフが G から G' に変化したとき、 G' に対するクラスタ構成を行うことをクラスタ再構成と呼ぶ。 □

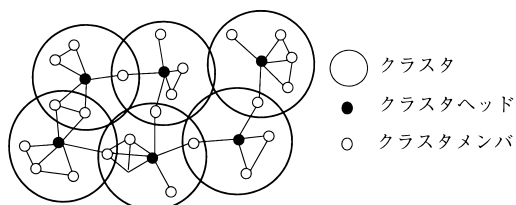


図1 クラスタ構成
Fig. 1 Clustering.

クラスタ構成, クラスタ再構成の評価尺度として, 以下の二つを用いる.

- クラスタ数 (クラスタ構成, 再構成)
- クラスタヘッドの変更数 (クラスタ再構成)

グラフ G のクラスタヘッドの集合を H , グラフ G' でクラスタ再構成したときのクラスタヘッドの集合を H' とする. このとき, クラスタヘッドの変更数は,

$$|H - H'| + |H' - H|$$

と定義する.

クラスタヘッドに選択されたノードは, クラスタ内及びクラスタ間の接続関係等の情報の維持管理といった負荷を伴う. また, アプリケーションレベルでは, クラスタヘッド間の論理リンクからなるクラスタヘッドアーキテクチャを考える. したがって, クラスタ数が少ないとクラスタヘッドアーキテクチャのネットワークサイズが小さくなるため, ネットワーク全体へ流れるトポロジー管理情報などを減少させることができる. したがって, クラスタ数の小さいクラスタ構成法, クラスタ再構成法が望ましい.

また, クラスタヘッドが変更されると, 古いクラスタヘッドがもっていた情報を新しいクラスタヘッドへ渡す, 若しくは新しいクラスタヘッドが新たにクラスタメンバの情報を収集する必要があり, クラスタヘッドの変更に伴い, オーバヘッドが生じる. したがって, クラスタヘッドの変更数が小さいクラスタ再構成法が望ましい.

本論文では, 与えられたトポロジーグラフに対して, クラスタ構成, またはクラスタ再構成を行うグラフアルゴリズムを考察する. 既存の多くのクラスタ構成法では, クラスタの構成中にトポロジーが変化するとといった頻繁なトポロジー変化は仮定していない [4], [6]. これは, トポロジー変化が頻繁に起こるネットワークでは, クラスタの維持にコストがかかってしまい, クラスタ構成をする利点が失われてしまうからである.

また, 本論文で考察する提案手法, 及び従来手法は, ネットワーク上のメッセージの伝送遅延のばらつき等に依存しないアルゴリズムであるため, クラスタ構成アルゴリズムの実行中にトポロジー変化が起きないという仮定のもとでは, 評価尺度であるクラスタ数, クラスタヘッドの変更数をグラフアルゴリズム上で評価することが可能である. 以上のような考察により, 本論文ではクラスタ構成法, 及びクラスタ再構成法をグラフアルゴリズムとして扱う. ただし, アルゴリズムの評価尺度, 後述するクラスタヘッドとなる優先度等はネットワークでの実現を考慮したものとなっている.

3. クラスタ構成法

3.1 集中型, 分散型クラスタ構成法

クラスタ構成法では一般に各ノードに対してクラスタヘッドになる優先度を与え, この優先度の高いノードから順にクラスタヘッドとして選択する. このとき, クラスタ構成法には集中型と分散型の2種類が考えられる.

集中型クラスタ構成法では, まずネットワーク全体で最も優先度の高いノードをクラスタヘッドに選択し, 選択したクラスタヘッドとそれに隣接するノードを削除する. 残ったノードに対して優先度を再計算し, 同じ手続きを繰り返す. これを, ノードがすべて削除されるまで繰り返す. 集中型クラスタ構成法は, 優先度を正確に反映することができるが, ネットワークの大域的な情報を必要とする. つまり, ネットワーク上で実現した場合, ネットワーク全体のノード情報を知る必要があるため, 通信量などネットワークへの負荷がかかってしまう. また, クラスタヘッドを一つずつ決定していくので, クラスタ構成にかかる時間も大きい.

分散型クラスタ構成法は, 集中型クラスタ構成法のようにネットワーク全体で優先度の比較を行うのではなく, あるノードの優先度がどの隣接ノードの優先度より高ければそのノードをクラスタヘッドとして選択する. そして, これらのクラスタヘッドとその隣接ノードを削除し, 残ったノードに対して優先度を再計算し, 同じ手続きを繰り返す.

分散型クラスタ構成法では, 隣接ノードの情報だけでクラスタヘッドを決定していく. したがって, ネットワーク上で実現した場合, 集中型クラスタ構成法に比べて, クラスタヘッド決定のために交換する情報量が少ない. また, ネットワークの異なる部分で, 同時にクラスタヘッドを決定できるので, 迅速にクラスタ

ヘッドを決定していくことが可能である。

図 2 に集中型クラスタ構成法を、図 3 に分散型クラスタ構成法を示す。このプログラムで用いている変数、関数、手続きは以下のとおりである。

- H : クラスタヘッド集合。
- S : どのクラスタにも含まれていないノードの集合。
- $G[S]$: S による G の誘導部分グラフ (induced subgraph)。
- $\Gamma(x)$: ノード x の隣接ノードの集合。
- $\Gamma(X)$: X に属するノードに隣接するノードの集合。つまり、 $\Gamma(X) = \cup_{x \in X} \Gamma(x)$ 。
- $max(S)$: ノード集合 S の中で優先度が最も高いノードを返す関数。
- $maximal(S)$: ノード集合 S の中で、優先度が極大 (S に属するどの隣接ノードよりも優先度が高い) ノード集合を返す関数。
- $compute\ priority(G)$: G の各ノードの優先度を計算する手続き。

3.2 最大 ID 法, 最大次数法

前節で述べたクラスタ構成法では、用いる優先度により、様々なクラスタ構成を行うことができる。本論

文では、ネットワークで効率良く実現することを考慮して、優先度としては、局所的に決定できるものに限る。本節では、既存の優先度の算出法として、最大 ID 法と最大次数法について述べる。

最大 ID 法は、ID の大きいノードほど高い優先度を割り当てる。つまり、各ノード p_i のグラフ G における優先度 $priority(G, p_i)$ を次式で与える。

$$priority(G, p_i) = id(i);$$

図 4 に最大 ID 法によるクラスタ構成の例を示す。図中の数字は、各ノードの ID を示す。

最大 ID 法では、各ノードの優先度はグラフに関係なく一定のため優先度の再計算を行う必要はない。また、集中型、分散型とも同じクラスタヘッドが選ばれる。しかし、トポロジーを考慮せずに優先度を決定するので、クラスタヘッドが多くなる。

トポロジーを考慮してクラスタヘッドを少なくするクラスタ構成法として、最大次数法が提案されている [4]。最大次数法では、次数の大きいノードから順にクラスタヘッドにすることにより、できるだけ多くのノードをクラスタに含んでいく。つまり、各ノード p_i のグラフ G における優先度 $priority(G, p_i)$ を次式で与える。

$$priority(G, p_i) = |\Gamma_G(p_i)|;$$

ここで、 $\Gamma_G(p_i)$ はグラフ G におけるノード p_i の隣接ノードを表す。また、ノードの優先度が同じ場合は、ID の大きいノードを優先させる。クラスタヘッドが決定されるにつれて、図 2、図 3 の $G[S]$ (S はどのクラスタにも含まれないノードの集合) は変化するので、最大次数法では $G[S]$ の変化に応じて、優先度を再計算しなければならない。また、集中型と分散型ではクラスタヘッドとなるノードや、クラスタ数が異なることがある。

図 5 に 最大次数法によるクラスタ構成の例を示す。

```
Centralized Clustering(G) {
  S = V;
  H = ∅;
  compute priority(G); /* 優先度の計算 */
  while (S ≠ ∅) {
    x = max(S); /* 優先度最大ノードを選択 */
    H = H ∪ {x};
    S = S - ({x} ∪ Γ(x));
    compute priority(G[S]); /* 優先度の再計算 */
  }
}
```

図 2 集中型クラスタ構成法
Fig. 2 Centralized clustering algorithm.

```
Distributed Clustering(G) {
  S = V;
  H = ∅;
  compute priority(G); /* 優先度の計算 */
  while (S ≠ ∅) {
    X = maximal(S); /* 優先度極大ノードを選択 */
    H = H ∪ X;
    S = S - (X ∪ Γ(X));
    compute priority(G[S]); /* 優先度の再計算 */
  }
}
```

図 3 分散型クラスタ構成法
Fig. 3 Distributed clustering algorithm.

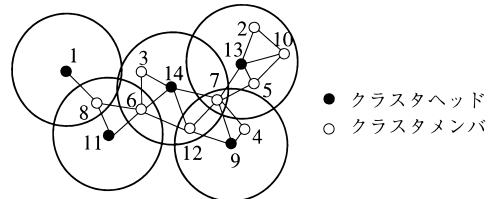


図 4 最大 ID 法
Fig. 4 Highest ID clustering algorithm.

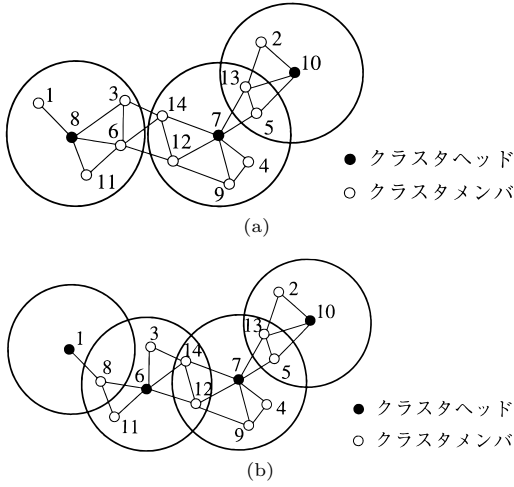


図5 最大度数法 (a) 集中型 (b) 分散型
 Fig. 5 Highest degree clustering algorithms.
 (a) Centralized algorithm. (b) Distributed algorithm.

(a) は集中型, (b) は分散型によるクラスタ構成である. この例では, 図 4 の同じネットワークに対して, 最大 ID 法に比べて, 最大度数法ではクラスタヘッド数を削減している.

3.3 重み付き最大度数法

最大度数法は, 最大 ID 法に比べてクラスタヘッドの数を少なく抑えることができる. しかし, 例えば図 6 のグラフに最大度数法を適用すると, 図 6(a) のようになる. このクラスタ構成では, 最初にノード p_8 をクラスタヘッドに選択し, 度数の小さいノード p_1, p_3, p_5, p_6 が残るため, ノード数が二つだけのクラスタが多数できてしまう. このため, 全体のクラスタヘッド数が多くなってしまふ. そこで, このような最大度数法の欠点を解決するために, 重み付き最大度数法を提案する.

提案する優先度は度数の低いノードを, より多く隣接ノードとしてもつノードに高い優先度を与える. そのため, 隣接ノードの度数の逆数の和を優先度とする. つまり, 各ノード p_i のグラフ G における優先度 $priority(G, p_i)$ を次式で与える.

$$priority(G, p_i) = \sum_{j \in \Gamma_G(p_i)} 1/|\Gamma_G(p_j)|;$$

重み付き最大度数法を図 6 に適用すると, その優先度は図 6(b) のようになり, 図 6(c) のようにクラスタが構成され, 最大度数法に比べ, クラスタ数を少なく

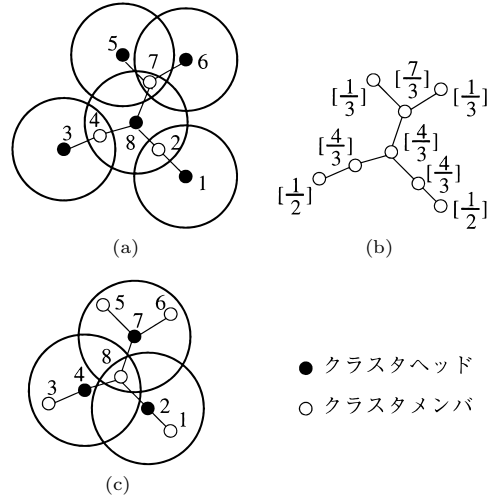


図6 (a) 最大度数法 (b) 重み付き最大度数法による優先度 (c) 重み付き最大度数法
 Fig. 6 (a) Highest degree clustering algorithm.
 (b) Priority of weighted highest degree.
 (c) Weighted highest degree clustering algorithm.

することができる.

4. クラスタ再構成法

4.1 LCC 法 (Least Cluster Change clustering algorithm)

アドホックネットワークでは, ノードの移動によるネットワークトポロジーの変化を考慮しなければいけない. トポロジー変化に応じてクラスタヘッドを変更しなければならないが, クラスタヘッドの変更が生じると, クラスタヘッドのもつ管理情報を更新するために, 大きなオーバーヘッドを生じる. そこで, クラスタ再構成アルゴリズムではクラスタヘッドの変更数をできるだけ少なくすることが望まれる.

トポロジーの変化により, どのクラスタヘッドにも隣接しないノードが生じ支配性が成り立たない場合, クラスタヘッドが隣接し独立性が成り立たない場合に, クラスタを再構成する必要がある. このとき, 前節のクラスタ構成法を適用し (その時点でのクラスタヘッドを考慮せずに) 一からクラスタを構成することも可能であるが, クラスタヘッドの変更数が大きくなってしまふ.

そこで, クラスタヘッドの変更を抑えるクラスタ再構成法として, LCC 法 [5] が提案されている. LCC 法では, 次の二つの場合のみクラスタヘッドの変更を局

所的に行う。

(1) 二つのクラスタヘッドが隣接した場合、優先度の低い方のクラスタヘッドがクラスタヘッドの役割をやめる。このために、どのクラスタヘッドにも隣接しないノードが生じた場合には、クラスタ構成法により、それらの中からクラスタヘッドを選ぶ。

(2) どのクラスタヘッドとも隣接しなくなったノードが生じた場合、クラスタ構成法により、それらの中からクラスタヘッドを選ぶ。

図7を用いて、この動作を説明する。優先度は簡単のため、最大ID法を用いる。ノード移動により、クラスタヘッド p_1, p_2 が隣接したとする(図7(b))。このとき、優先度の低いノード p_1 がクラスタヘッドの役割を放棄する(図7(c))。図7(c)において、どのクラスタヘッドにも隣接しないノードは p_3 だけであるので、 p_3 が新たにクラスタヘッドとなる。

LCC法を図8に示す。ここでは、以下の変数、関数を用いている。

- H : トポロジー変化前のクラスタヘッド集合。
- G' : トポロジー変化後のグラフ。

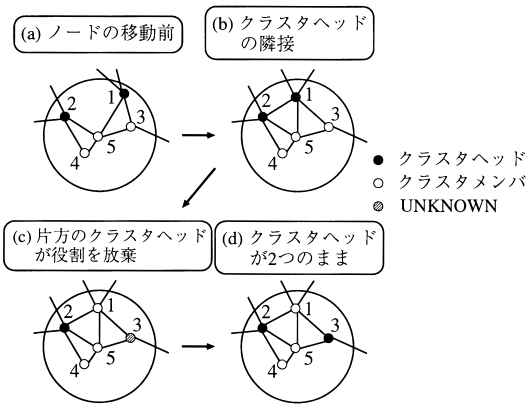


図7 LCC法によるクラスタ再構成の例
Fig.7 An example of re-clustering by LCC algorithm.

```
LCC Clustering {
  compute priority( $G'$ );
   $X = maximal(H)$ ;
  /*  $G'$  における  $H$  の極大独立頂点集合 */
   $G'' = G' [V - (X \cup \Gamma(X))]$ ;
   $H' = X \cup Clustering(G'')$ ;
}
```

図8 LCC法
Fig.8 LCC algorithm.

- $H' : G'$ でのクラスタヘッドの集合。
- $Clustering(G'')$: クラスタ構成法により、 G'' のクラスタヘッド集合を返す関数。

4.2 改良LCC法

LCC法においては、ノードの移動により二つのクラスタヘッドが隣接した場合は、一方のクラスタヘッドはそのままクラスタヘッドとなり、クラスタヘッドの役割をやめるノードを一つに抑えている。しかし、クラスタヘッドを続けるノードがトポロジー変化後のグラフにおいても、高い優先度をもつとは限らない。このため、もとのクラスタヘッドをクラスタヘッドとして残すことで、新たにクラスタヘッドとなるノード数が増えることも考えられる。

そこで本論文では、改良LCC法を提案する。改良LCC法ではクラスタヘッドが隣接した場合、いったん両方のクラスタヘッドが役割を放棄する。その後、どのクラスタヘッドにも隣接していないノードの集合に対しクラスタ構成法を用いて、クラスタを再構成する。このことにより、トポロジー変化後のグラフにおいて優先度の高いノードをクラスタヘッドとして選択することができ、LCC法に比べてクラスタヘッド数を減らすことができると考えられる。

具体例を図9に示す。クラスタヘッドが隣接する(b)までは、図7と同じである。改良LCC法では、クラスタヘッド p_1, p_2 が両方いったんクラスタヘッドの役割をやめる(図9(c))。そして、どのクラスタヘッドにも隣接しないノード集合 p_1, p_2, p_3, p_4, p_5 で優先度を再計算すると、ノード p_5 がクラスタヘッドになる

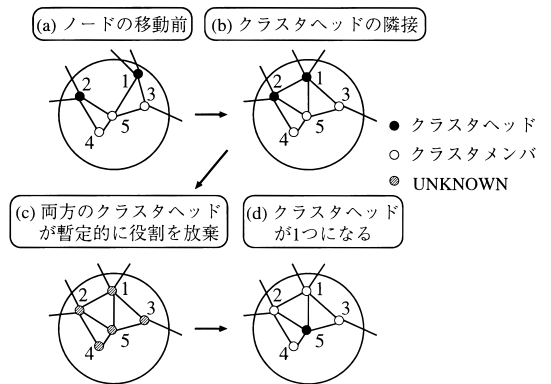


図9 改良LCC法によるクラスタ再構成の例
Fig.9 An example of re-clustering by improved LCC algorithm.

```

Improved LCC Clustering {
  compute priority(G');
  X = {p | p ∈ H, ΓG'[H](p) = ∅};
  /* G'[H] における孤立ノード集合 */
  G'' = G'[V - (X ∪ Γ(X))];
  Clustering(G'');
}
    
```

図 10 改良 LCC 法
Fig. 10 Improved LCC algorithm.

(図 9(d)). この例では, LCC 法に比べ, 改良 LCC 法ではクラスタヘッドが一つ減っている. 改良 LCC 法を図 10 に示す. 用いている変数, 関数は LCC 法(図 8)と同じである.

5. 実験結果, 考察

5.1 クラスタ構成法の比較

本節では, 既存の最大 ID 法と最大次数法, 本論文で提案した重み付き最大次数法について, 集中型, 分散型クラスタ構成法のそれぞれに関するシミュレーション結果を示す.

シミュレーションは, ノード数は 100, 200, 300 の三つの場合に対して行った. いずれの場合もノードを一樣乱数によって 1000×1000 のフィールド上に配置する. また, 各ノードの通信範囲はそのノードを中心とする円とし, 半径 50, 100, 150, 200, 250, 300 の場合について行った.

表 1 に集中型クラスタ構成法を実行したときのクラスタヘッド数を, 表 2 に分散型クラスタ構成法を実行したときのクラスタヘッド数を示す. それぞれ 100 回初期配置をして, クラスタ構成をしたときの平均値を示している.

集中型クラスタ構成法のクラスタヘッド数を比較すると, ほとんどの場合で重み付き最大次数法が他の手法より少ないクラスタヘッド数になっている. 特に通信半径が小さいときほど差は大きくなっている. また, ノード数が多く, 通信半径が大きくなったときに, 最大次数法の方がクラスタヘッドの数が少なくなっている. この場合のグラフはノードの平均次数が高くなっていて, 次数の低いノードがあまり存在していないため, 重み付き最大次数法の効果が現れないと考えられる. 最大 ID 法は例外なくクラスタヘッド数は多くなっている. これは, トポロジーに関係なく優先度を決定しているためである.

分散型クラスタ構成法のクラスタヘッド数を見る

表 1 集中型クラスタ構成法のクラスタヘッド数
Table 1 The number of clusterheads by centralized clustering algorithms.

全ノード数	通信半径	重み付き 最大次数法	最大次数法	最大 ID 法
100	50	69.80	69.94	72.58
	100	33.30	34.86	40.62
	150	18.43	20.00	24.24
	200	12.27	13.26	16.02
	250	9.29	9.90	11.38
200	50	101.52	103.01	113.47
	100	38.63	41.77	50.71
	150	21.44	23.20	27.76
	200	13.83	14.88	17.71
	250	11.01	11.12	12.51
300	50	117.71	121.03	138.79
	100	41.92	45.41	55.84
	150	23.17	24.76	29.49
	200	15.06	15.52	18.51
	250	12.08	11.82	12.82
	300	8.60	8.27	9.54

表 2 分散型クラスタ構成法のクラスタヘッド数
Table 2 The number of clusterheads by distributed clustering algorithms.

全ノード数	通信半径	重み付き 最大次数法	最大次数法	最大 ID 法
100	50	69.81	69.94	72.58
	100	33.71	35.01	40.62
	150	19.43	20.15	24.24
	200	12.89	13.47	16.02
	250	9.45	9.96	11.38
200	50	101.74	103.24	113.47
	100	40.56	42.14	50.71
	150	22.47	23.25	27.76
	200	14.28	15.02	17.71
	250	10.98	11.18	12.51
300	50	118.62	121.68	138.79
	100	44.29	45.81	55.84
	150	24.08	24.81	29.49
	200	15.84	15.58	18.51
	250	12.00	11.99	12.82
	300	9.41	8.44	9.54

と, 集中型クラスタ構成法によって決定されたクラスタヘッド数より多くなってはいるが, それほど値は変わっていない. 更に, 集中型クラスタ構成法と同様に, トポロジーグラフが密な場合を除いて, 重み付き最大次数法が他の手法より少ないクラスタヘッド数になっている.

5.2 LCC 法と改良 LCC 法の比較

本節では, LCC 法 [5] と, 4. で提案した改良 LCC

法に関するシミュレーション結果を示す。

シミュレーションは、ノード数は 100, 200, 300 の三つの場合に対して行った。いずれの場合もノードの通信半径は 200 とした。シミュレーションの方法は、5.1 と同じ方法で、まず初期配置を決めた後にクラスタ構成法を実行する。その後、全ノードは確率 1/2 で移動するノードと移動しないノードに分け、移動するノードは確率 1/4 で上下左右方向に動くものとし、LCC 法、改良 LCC 法に従って、クラスタ再構成を行った。クラスタ再構成法でサブルーチンとして用いるクラスタ構成法には、集中型重み付き最大次数法を採用した。ノード数 200 の場合のクラスタヘッドの変化を図 11 に示す。グラフでは、クラスタヘッド数を 1000 単位時間ごとに 10000 単位時間まで表した。1000 単位時間ごとでのクラスタヘッド数は、その時点の 100 単位時間前からの平均をとっている。更に、10000 単位時間のシミュレーションを異なる初期配置に対して、10 回試行し、その平均をとっている。表 3 は、クラスタヘッドの変更数を示す。変更数は、10 回試行の平均をとっている。

LCC 法では、移動回数の増加に伴い、クラスタヘッド数は初期値から上がって 16 程度で安定している。これに対して、改良 LCC 法では、移動回数が増加してもクラスタヘッド数はほとんど増加しない。また、改

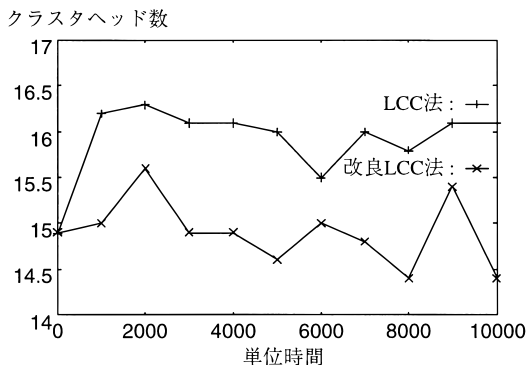


図 11 クラスタヘッド数 (ノード数: 200)

Fig. 11 The number of clusterheads (the number of node: 200).

表 3 クラスタヘッド変更数

Table 3 The number of changes of clusterheads.

ノード数	改良 LCC 法	LCC 法
100	562.9	646.4
200	718.0	820.3
300	757.8	948.7

良 LCC 法は LCC 法よりクラスタヘッドの変更数が少ないという結果が得られた。

以上のように、クラスタヘッド数、クラスタヘッドの変更数の両方において、改良 LCC 法は、LCC 法よりすぐれており、移動分散システムのクラスタ維持に適した手法といえる。

6. む す び

本論文では、移動端末だけからなるアドホックネットワークでのクラスタ構成法として重み付き最大次数法を提案した。更に、移動に伴いクラスタを再構成する手法として改良 LCC 法を提案した。

また本論文で提案した重み付き最大次数法と既存手法の最大次数法、最大 ID 法をクラスタヘッド数についてシミュレーションにより評価した。シミュレーション結果により、トポロジグラフが密な場合を除き、提案手法ではクラスタヘッド数が少なくなることが示された。更に、改良 LCC 法と LCC 法をクラスタヘッド数、クラスタヘッドの変更数についてシミュレーションにより評価した。シミュレーション結果により、提案手法ではクラスタヘッド数を少なく抑え、クラスタヘッド変更数においても少なくなることが示された。

本論文では、クラスタ構成法、クラスタ再構成法をグラフアルゴリズムとして考察を行い、クラスタ数、クラスタヘッドの変更数に関して評価を行った。提案手法をネットワークで実現した場合のクラスタ構成、及びクラスタ再構成にかかわる通信量の評価等は今後の課題である。

謝辞 本研究に関し、多くの貴重な意見を頂いた本学の太竹哲史助手及び情報論理学講座の皆様へ感謝します。本研究の一部は、文部省科学研究費補助金(特定領域研究 B-2 10205218)、及び、中部電力基礎技術研究所研究助成による。

文 献

- [1] S. Basagni, "Distributed clustering for ad hoc networks," Proc. International Symposium on Parallel Architectures, Algorithms, and Networks, pp.310-315, 1999.
- [2] S. Basagni, "A distributed algorithm for finding a maximal weighted independent set in wireless networks," Proc. IASTED International Conference, Parallel and Distributed Computing and Systems, pp.517-522, 1999.
- [3] A. Ephremides, J.E. Anthony, and D.J. Baker, "A design concept for reliable mobile radio networks with

frequency hopping signaling,” Proc. IEEE, vol.75, no.1, pp.56–73, 1987.

- [4] M. Gerla and J.T.C. Tsai, “Multicluster, mobile, multimedia radio network,” ACM-Baltzer J. Wireless Networks, vol.1, no.3, pp.255–265, 1995.
- [5] C.C. Chiang, H.K. Wu, W. Liu, and M. Gerla, “Routing in clustered multihop, mobile wireless networks with fading channel,” Proc. IEEE Singapore International Conference on Network, 1996.
- [6] C.R. Lin and M. Gerla, “Adaptive clustering for mobile wireless networks,” IEEE J. Selected Areas in Communications, vol.15, no.7, pp.1265–1275, 1997.

(平成12年4月20日受付, 8月11日再受付)



藤原 秀雄 (正員)

昭44 阪大・工・電子卒。昭46 同大学院博士後期課程了。阪大工学部助手, 明治大理工学部教授を経て, 現在奈良先端科学技術大学院大学情報科学研究科教授。昭56 ウォータールー大客員助教授。昭59 マツギル大客員準教授。論理設計, 高信頼設計, 設計自動化, テスト容易化設計, テスト生成, 並列処理, 計算複雑度に関する研究に従事。著書 “Logic Testing and Design for Testability” (The MIT Press) など。大川出版賞。工博。IEEE, 情報処理学会各会員, IEEE Fellow, IEEE Golden Core Member。



谷口 博人

平10 神戸大・工・電気電子卒。平12 奈良先端大博士前期課程了。同年 NTT ドコモ関西(株)に入社。



井上美智子 (正員)

昭62 阪大・基礎工・情報卒。平1 同大学院博士前期課程了。同年富士通研究所(株)入社。平7 阪大大学院博士後期課程了。奈良先端科学技術大学院大学情報科学研究科助手, 現在に至る。分散アルゴリズム, グラフ理論, テスト容易化設計, 高位合成の研究に従事。工博。IEEE, 情報処理学会, 人工知能学会各会員。



増澤 利光 (正員)

昭57 阪大・基礎工・情報卒。昭62 同大学院博士後期課程了。同年同大情報処理教育センター助手。同大基礎工助教授を経て, 平6 奈良先端科学技術大学院大学情報科学研究科助教授, 平12 大阪大学基礎工教授。平5 コーネル大客員準教授(文部省在外研究員)。分散アルゴリズム, 並列アルゴリズム, テスト容易化設計, テスト容易化高位合成に関する研究に従事。工博。ACM, IEEE, EATCS, 情報処理学会各会員。