

Title	アドホックネットワークにおけるクラスタリングアルゴリズムに関する研究
Author(s)	石川, 英智
Citation	
Issue Date	2003-03
Type	Thesis or Dissertation
Text version	author
URL	<a href="http://hdl.handle.net/10119/1660">http://hdl.handle.net/10119/1660</a>
Rights	
Description	Supervisor:中野 浩嗣, 情報科学研究科, 修士

修 士 論 文

アドホックネットワークにおける  
クラスタリングアルゴリズムに関する研究

北陸先端科学技術大学院大学  
情報科学研究科情報処理学専攻

石川 英智

2003年3月

# 修士論文

## アドホックネットワークにおける クラスタリングアルゴリズムに関する研究

指導教官 中野浩嗣 助教授

審査委員主査 中野浩嗣 助教授  
審査委員 浅野哲夫 教授  
審査委員 平石邦彦 助教授

北陸先端科学技術大学院大学  
情報科学研究科情報処理学専攻

110010 石川 英智

提出年月: 2003 年 2 月

## 概要

アドホックネットワークでは、ネットワークを複数のクラスタに分割し、各クラスタをその代表ノード（クラスタヘッドと呼ぶ）で管理することが行われる。

クラスタに分割する作業をクラスタリングと呼ぶが、本研究では効率良いクラスタリング法を提案する。基本的なアイデアは、クラスタヘッドにスレーブノードを付加することであり、従来からのDCA(Distributed Clustering Algorithm)と比較して、クラスタリングに要する step 数を削減することに成功した。

# 目次

第1章	はじめに	1
第2章	アドホックネットワークとクラスタリング	3
2.1	アドホックネットワーク	3
2.2	クラスタリングに関する定義	4
2.2.1	DCA(Distributed Clustering Algorithm)	5
2.2.2	2クラスタヘッドアルゴリズム	5
2.3	クラスタヘッド	6
第3章	クラスタリングアルゴリズム	8
3.1	ID法(最小ID法)	8
3.2	DCA(Distributed Clustering Algorithm)	10
3.3	2クラスタヘッドアルゴリズム	11
第4章	コリジョンがランダムに発生する場合のクラスタリングアルゴリズム	14
4.1	コリジョン	14
4.2	クラスタリングアルゴリズム	17
第5章	シミュレーション結果・考察	19
5.1	シミュレーションの概要と構成	19
5.2	結果1:コリジョンを考慮しない場合	20
5.3	結果2:コリジョンがランダムに発生するとき	27
第6章	まとめ	35

# 第1章 はじめに

近年、情報通信技術の進歩により移動端末が無線通信を利用して様々なネットワークを構築することが可能になった。それに伴い、無線で接続できる端末(パソコン, PDA, 携帯電話など)のみで構成される「自立分散型無線ネットワーク」について、様々な研究がなされている。[1]~[10]

現在の移動体通信システムでは、基地局やアクセスポイントを介在させ固定網をバックボーンとしているものが主となる。そのため、そのようなシステムにおいて、例えば大規模な災害が発生し基地局が機能なくなると、被災地への通信手段を失うことになる。また、海外などでインフラの整備がされていない広大な場所での通信は、大きなコストがかかってしまう。これに対してインフラを必要としないアドホックネットワークは、上記のような状況でも安価にネットワークを構築することができ、限られた域内での簡易なネットワークの構築手段として有効である。

ネットワークにおいて階層的な構成を得るのは有線ネットワークでも重要な分野であり、同様にアドホックネットワークにおいても重要である。本稿では、アドホックネットワークでのクラスタリングについて考える。

クラスタを構成するには、ノードはその通信範囲内のノードと相互に情報を交換する。そしてその隣接ノード群の中から一定の条件の基にクラスタ内の情報を管理する機能を持つノードを選ぶ。このノードをクラスタヘッドと呼ぶ。クラスタ内のクラスタヘッドに隣接するノードをオーディナリノードと呼び、フィールド上のノードは必ずクラスタに含まれ、全てがどちらかのタイプのノードになる。クラスタを構成することにより、無線チャネルの帯域幅の空間再利用ができ、各クラスタで帯域幅が共有できる。また、ノードの移動・参入・退出などのネットワークトポロジーの変化に柔軟に対応ができる。更に、経路制御情報を少なくすることができる等の利点がある。

アドホックネットワークではノードからノードヘデータを中継することでエンド ツーエンドの通信を行うマルチホップ通信を行う。そのため、ノードの移動に伴うネットワークトポロジーの変化で交換される制御情報はできるだけ少ない方がよい。また、ノードは限られた処理能力と記憶容量と電力量しか備えていないため、ノードに対する負荷はできるだけ小さくしなければならない。このような点を踏まえた上でクラスタリングアルゴリズムを考える。

クラスタリングは、以下のようなケースで行われる。

1. クラスタリングされていないネットワークに対して行う。
2. 既にクラスタリングされているネットワークのノードの移動、参入、退出等によるトポロジーの変化に対応するために行う。

本稿では、特にクラスタ構成されていない状態のネットワークに対して行われるクラスタのセットアップについて検討する。この状態における一般的な仮定は、クラスタリングしている間ノードは動かないというものである。クラスタリングアルゴリズムは DCA ( Distributed Clustering Algorithm ) をベースに最小 ID 法を用いる。ID 法とはクラスタヘッドを決定する際にノード ID ( 識別子 ) を用いる方法である。直接通信が可能なノードの中で ID の大きさが最も小さいノードがクラスタヘッドになるというものである。なお、ID は  $[0, 1)$  のランダムな実数を割り当てる。

更に、通信状態の仮定として

1. ノード間の通信は確実に行われる。
2. ノード間の通信に衝突(コリジョン)がランダムに発生する。

という2つのケースについて考える。従来のクラスタリングにおいては、クラスタを構成する際のノード間の通信は、アクセス制御が適切に行われており、その情報は正しく受信ができるという仮定のもとで行われていた。今回は、さらに擬似的にコリジョンを与え、それがどの様にクラスタリングに影響を与えるかをシミュレーションしてみることにした。評価の方法として、フィールド上の全ノードのクラスタリングが終了するまでの step 数に注目した。

本稿では、クラスタリングアルゴリズムとして、従来のクラスタヘッドとオーディナリノードで構成される1クラスタヘッドのアルゴリズムに対し、マスタクラスタヘッドとスレーブクラスタヘッドの「2種類のクラスタヘッド」とオーディナリノードからなる「2クラスタヘッドアルゴリズム」を提案する。クラスタヘッドをマスタノードとスレーブノードとすることにより、クラスタ内での階層化が深くなり、クラスタリングアルゴリズムの再帰的な1ターンにおいて、少なくとも1 step 増えることになるが、最終的な総 step 数は改善されることをシミュレーションで示す。

本稿の構成は以下の通りである。第2章ではアドホックネットワークのモデルとクラスタについて定義する。第3章ではノード間通信が確実に行われる状態での DCA と2クラスタヘッドアルゴリズムを示す。第4章ではノード間通信に衝突(コリジョン)がランダムに発生する状態での DCA と2クラスタヘッドアルゴリズムを示し、第5章で DCA と2クラスタヘッドアルゴリズムのシミュレーションによる実験結果を示す。最後に第6章においてまとめを述べる。

# 第2章 アドホックネットワークとクラスタリング

## 2.1 アドホックネットワーク

アドホックネットワークは、有線ケーブルも基地局も持たないインフラストラクチャー不要のネットワークである。そのため、ノード（端末）は移動することができ、どのような場所でも利用することができる。一方、ノードの処理能力、記憶容量、電力源などで制限がある。ノードはそれぞれ通信範囲が限られ、直接通信できないノードに対しては、その間にいるノードが情報を転送することによって通信を可能とする。ノードが集まれば自律的にネットワークを構成し機能するため、柔軟性が高く、設置が容易なネットワーク構築が可能になる。アドホックネットワークの応用例として以下のようなケースが考えられる。

- 災害時、戦時下など  
インフラが利用不可能な状況下。インフラが破壊されている状況や、インフラが整備できない状況において、移動体ノードを持ち込むだけでネットワークを構築することができる。
- 短期的な利用  
コンサート会場やスポーツスタジアムなどのように、一時的に多くの人々が一定の範囲内に集まるような場合、必要な通信量に併せてネットワークを構築することができる。
- トポロジーの変化が頻繁な状況  
タクシーネットワークなど、頻繁なノードの移動に伴うネットワークトポロジーの変化に柔軟に対応することができる。

アドホックネットワークにおけるノードは固有のID（識別子）を持ち、通信はブロードキャストによって行われる。各ノードは通信機能に加えデータの中継機能も有する。アドホックネットワークではノードが自由に移動できるため、ネットワークトポロジーは時間とともに変化する。しかし、基地局が存在しないため、ノード自身がネットワークを把握しなければならない。物理的な上限もあり、ノード数が増えるにつれ、個々のノードが全ネットワークを把握するのは非常に難しくなる。必要最小限の情報でデータがやり取りできるようにするため、有効な方法としてクラスタを構成する。クラスタとは、複数のノードが

らなり、ネットワークを分割したグループを言う。クラスタを構成することによりネットワークを階層化することができる。階層化によってネットワーク内の情報を効果的にやり取りすることが可能になる。

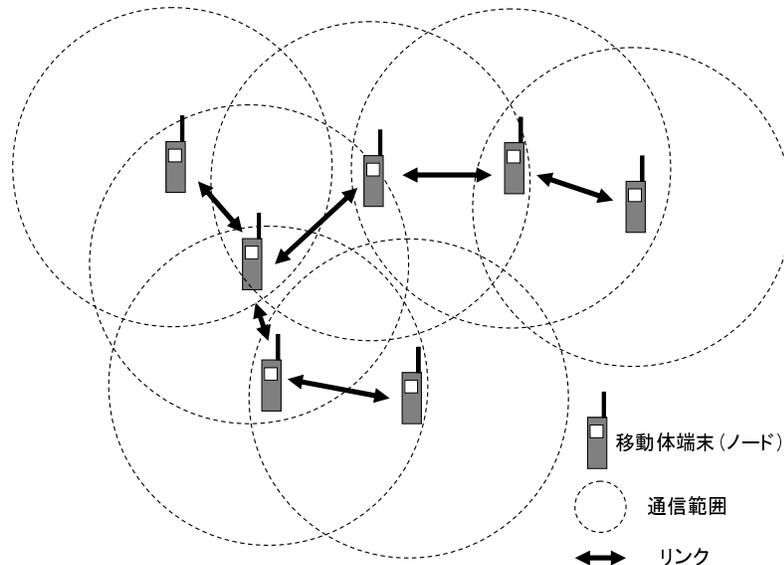


図 2.1: アドホックネットワークのモデル

## 2.2 クラスタリングに関する定義

ネットワークをいくつかのクラスタに分割することをクラスタリングという。ノードは通信範囲内に存在する全てのノード（以下、隣接ノードという）と情報の交換をして、クラスタ内における役割を決める。クラスタ内のノードの位置情報や最適経路の選択等の調停機能を果たすノードをクラスタヘッドという。クラスタはクラスタヘッドとオーディナリノードから構成される。クラスタに分割することにより、同じクラスタに属するノード間で時分割を行ったり、クラスタ間で通信資源を再利用することができるという利点がある。クラスタリングアルゴリズムは色々な方法があるが、本稿ではDCA (Distributed Clustering Algorithm)を使う。また、今回提案するスレーブクラスタヘッドを加えた2クラスタヘッドアルゴリズムはDCAをベースにしている。以下にクラスタリングにおける定義をする。

## 2.2.1 DCA(Distributed Clustering Algorithm)

クラスタは、1つのクラスタヘッドとクラスタヘッドに隣接するオーディナリノードで構成される。次章で示されるクラスタリングアルゴリズムを実行するときは下記の定義を満たすようにする。

### 定義

- クラスタヘッド同士は隣接しない。
- オーディナリノードは必ずクラスタヘッドに隣接する。

## 2.2.2 2クラスタヘッドアルゴリズム

本稿で提案するクラスタリングは、クラスタをマスタクラスタヘッドとスレーブクラスタヘッドという「2種類のクラスタヘッド」とオーディナリノードで構成する。DCAが1つのクラスタヘッドでクラスタを構成するで、以降、このアルゴリズムを2クラスタヘッドアルゴリズムと呼ぶ。クラスタヘッドが階層化され、スレーブクラスタヘッドが複数存在することにより、少ない通信半径で大きなクラスタサイズを得ることができクラスタ数を減らすことができる。また、ノードの数が多くなると、オーディナリノードは複数のスレーブクラスタヘッドに接することが多くなるので、通信経路に障害が起こったときに、再経路探索を行うことが容易になると考えられる。2クラスタヘッドアルゴリズムを実行するには以下の定義を満たすようにする。

### 定義

- マスタクラスタヘッド同士は隣接しない。
- スレーブクラスタヘッドは必ずマスタクラスタヘッドに隣接する。  
(スレーブクラスタヘッド同士は隣接しても良い。)
- オーディナリノードは必ずスレーブクラスタヘッドに隣接する。

## 2.3 クラスタヘッド

一定の大きさのネットワークフィールドにおいてクラスタリングを行うとき、クラスタヘッドの数はノード数に比例する傾向にある。しかし、一定の数に達するとそれ以上増えることはなくなってしまふ。これは、フィールドにおける最大のクラスタヘッド数は通信半径によって決まるためである。クラスタリングの定義より「クラスタヘッド同士は隣接しない」ため、クラスタヘッドの最大数は「フィールドに設置することのできる通信半径の半分の大きさの円の最大数」に関係している。クラスタヘッド数が最大になるとき、クラスタヘッドは円と円の間隙に位置することになる。(ただし、全ての間隙ではない)

図 2.2 は部分的にはあるが、クラスタヘッド数が最大限になるようにノードを配置している図である。なお、図では通信範囲に対してクラスタヘッドが隣接しているように見えてしまふが、これはクラスタヘッドを拡大して描いているためである。

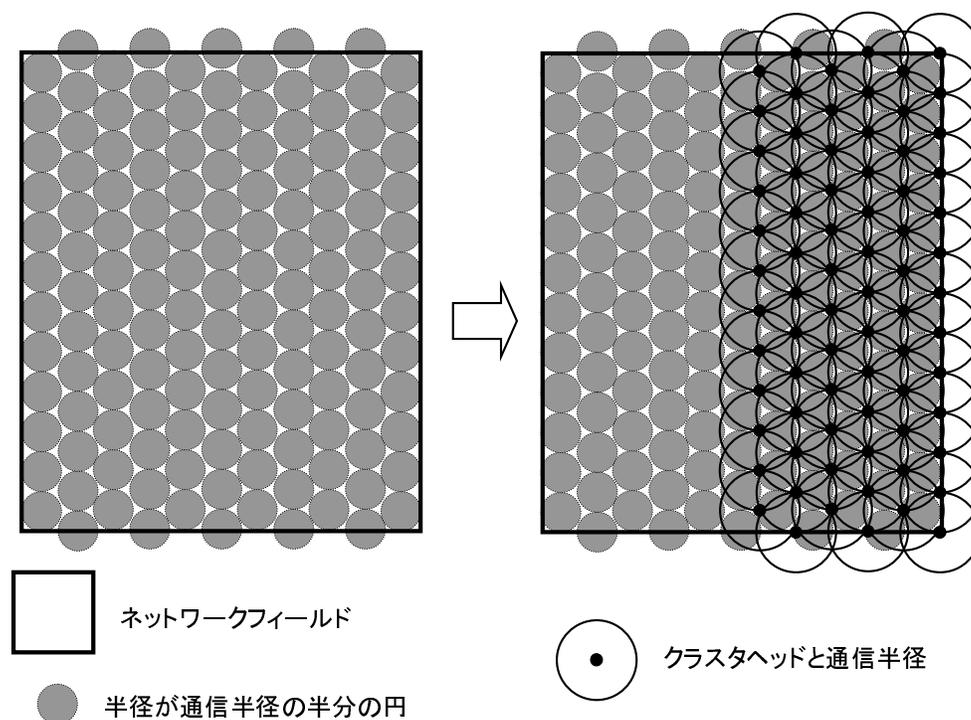


図 2.2: クラスタヘッドの数が最大になる配置

さらに、最小ID法DCAを用いて、本稿のシミュレーションと同様、 $1000 \times 1000$ のフィールドに、ノードの総数を1000,5000,10000の3通り配置し、通信半径は10刻みに10~100まで10通りについて、100回試行を行ったときの平均クラスタヘッド数の比較を示す。図2.3より、最初はノード数に比例していたクラスタヘッド数が、通信半径が大きくなると、ノード数は10倍になるにもかかわらず、ほぼ同数に収束していくことがわかる。

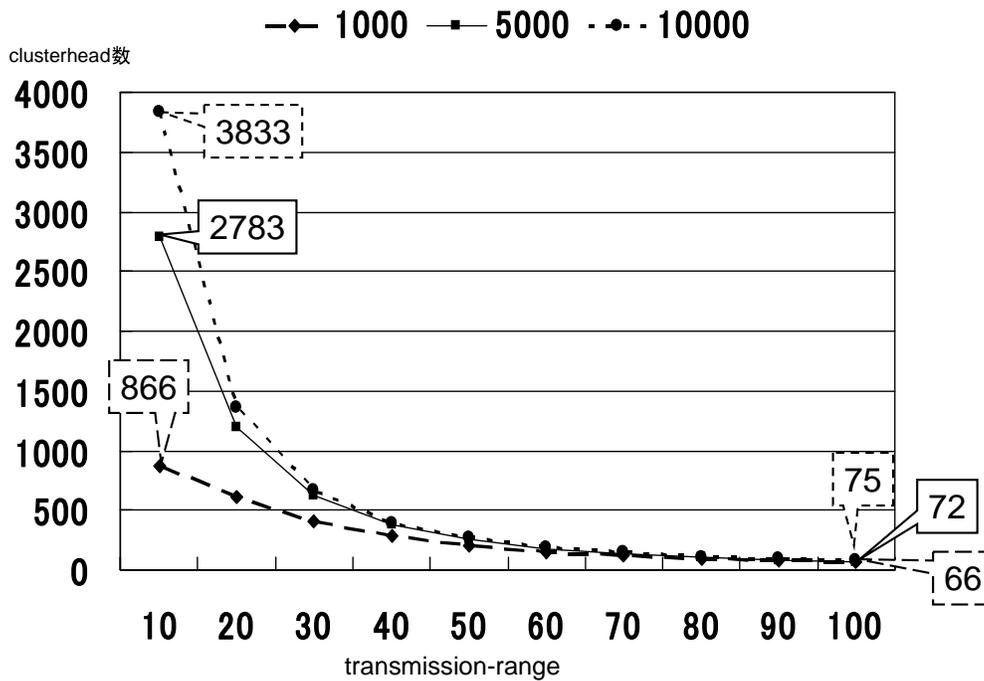


図 2.3: ノード数と通信半径ごとの平均クラスタヘッド数

# 第3章 クラスタリングアルゴリズム

## 3.1 ID法(最小ID法)

クラスタリングを行うときは、ある一定の基準の優先度によってクラスタヘッドを決定する。既存の優先度の算出方法としては

- ID法
- 最大次数法
- 重み付き最大次数法

などが提案されている。ID法は各ノードに割り当てられる固有のIDの大小により優先度を割り当てる方法である。最大次数法はノードの次数(隣接するノードの数)によってクラスタを構成する方法である。もし、ノードの次数が同じ場合は、IDの大小によって優先度を決定する。この方法は多くのノードに隣接するノードがクラスタヘッドになるため、ネットワーク全体で、より少ないクラスタヘッド数でクラスタリングをすることができる。クラスタヘッド数が少ないということはクラスタ数が少ないということになるので、ネットワーク全体に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利点がある。重み付き最大次数法は、最大次数法を改良した方法である。最大次数法は次数の高いノードからクラスタヘッドを決定するため、クラスタヘッド数を少なく押さえることができる。しかし、ネットワークトポロジーによっては、次数の低いノードが余ってしまい、その分だけクラスタ数が多くなってしまう場合がある。このような欠点を補うため、次数の低いノードに多く隣接しているノードがクラスタヘッドになるように、隣接ノードの次数の逆数の和を優先度にする方法である [1][2][3]。

本稿ではネットワーク上の各ノードに割り当てられる固有の識別子(ID)を基準に優先度を計算する。IDは $[0,1)$ の実数を割り当てる。ノードは隣接ノードのIDと自分のIDを比較して、最も小さいIDを持つノードがクラスタヘッドに選ばれるものとする。例として図3.1、図3.2を示す。なお、図においての数値はノードIDを表している。

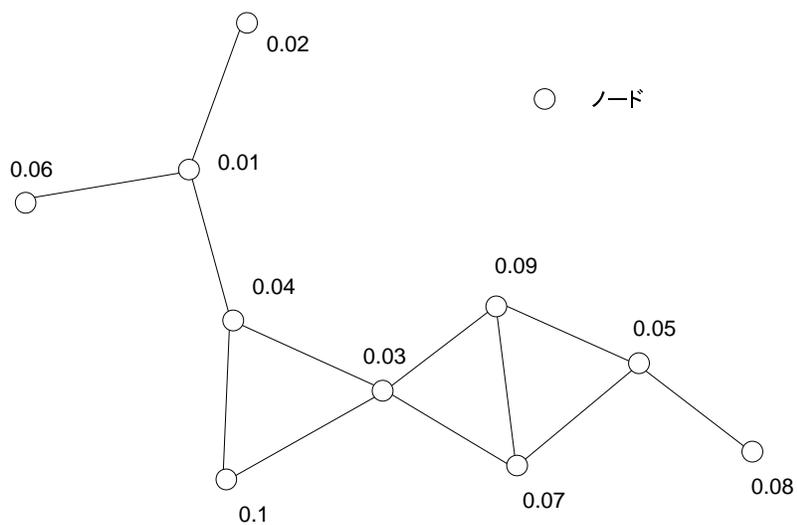


図 3.1: ネットワークポロジ

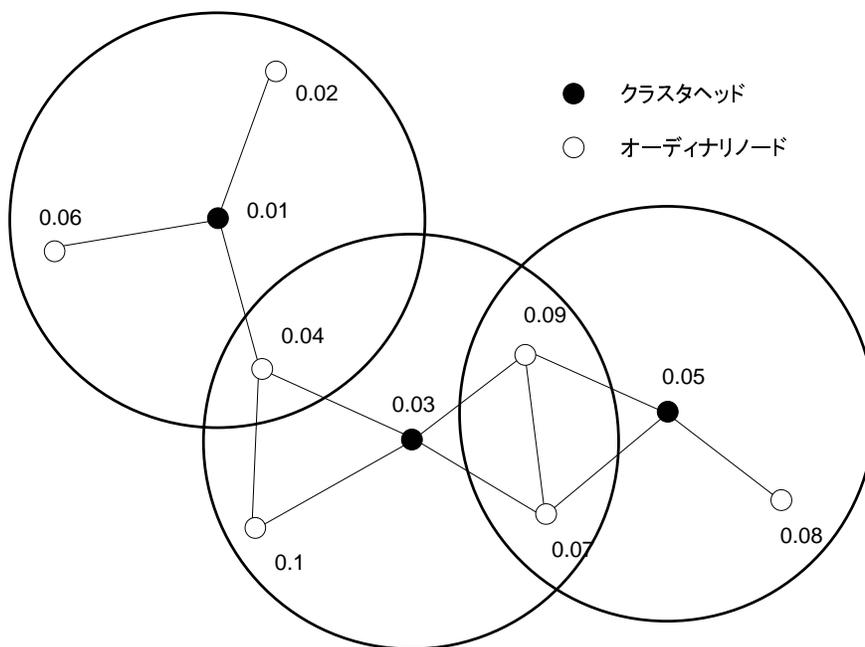


図 3.2: 最小ID 法によるクラスタリング

## 3.2 DCA(Distributed Clustering Algorithm)

DCA ( Distributed Clustering Algorithm) は, クラスタリングのセットアップのための一般的なアルゴリズムである. これは, 局所的な情報によりクラスタリングを行う分散型のクラスタ構成法で, 以下のような操作上の仮定がされている.

- 任意のノードが送信したメッセージは, 有限時間 ( step) 以内に, すべての隣接するノードによって正しく受信される.
- ネットワークトポロジはアルゴリズムを実行している間, 変化しない.

分散型のクラスタ構成法は, 自分の優先度が隣接するノードの優先度より高ければクラスタヘッドになるというものである. 仮に, 自分より優先度が高い隣接ノードが存在しても, そのノードが既に他のクラスタのオーディナリノードになっていた場合は, 自分がクラスタヘッドになる. 分散型の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するため, クラスタヘッドを決めるための情報量が少なくすることができる. また, ネットワーク内の複数の場所で, 同時にクラスタリングを行うことができる. アルゴリズムは以下に示す.

### クラスタリングアルゴリズム

1. 各ノードは  $[0, 1)$  の乱数を生成し, ID とする.
2. 隣接ノードに自 ID 等を送信する. また, 隣接ノードから情報を受信する.
3. 隣接ノードの中で最小の ID を持つノードは, クラスタヘッドになることを送信してアルゴリズムを停止する.
4. クラスタヘッドに隣接する役割の決っていないノードは, オーディナリノードになりアルゴリズムを停止する.
5. 役割の決っていないノードは 2 に戻る.

### 3.3 2 クラスタヘッドアルゴリズム

クラスタリングの実行はより少ない step 数で行われることが望ましい。3.2 節で述べられた最小 ID 法の DCA では、1 ターンごとにクラスタヘッドとオーディナリノードが選択されていく。ノードの役割が決定すると、次の step からは、役割が決まっていないノードの隣接ノード集合からそのノードは除かれる。そこで、従来のクラスタヘッドをマスタクラスタヘッドとし、さらにその補助的な役割を行うスレーブクラスタヘッドを加え、クラスタヘッドを2つにすることによってクラスタリングする方法を提案する。

スレーブクラスタヘッドの役割は、従来の1つのクラスタヘッドが管理しているノードの情報や経路制御情報を階層構造を持って管理することができるというメリットがある。一方、クラスタリングを行うときにスレーブクラスタヘッドを選択するための step が、少なくとも1回以上必要である。しかし、1 ターン当り役割の決まるノード数は増える。

スレーブクラスタヘッドを加えたクラスタリング（以下 2 クラスタヘッドのクラスタリングという）における例として図 3.3, 図 3.4, 図 3.5 を示す。この図においても、ノードに付いている数値はノード ID である。図 3.4 においてクラスタは  $\{3, 6, 8\}$ ,  $\{1, 4, 5, 9, 10\}$ ,  $\{2, 7\}$  となる。なお、ノード ID  $\{7\}$  と  $\{9\}$  のノードは2つのスレーブクラスタヘッドの通信範囲に含まれているが、任意の条件により（例えば、より早く情報を受信した等）どちらかのクラスタに属する。2つ以上のマスタクラスタヘッドに隣接するスレーブクラスタヘッドや、2つ以上のスレーブクラスタヘッドに隣接しているオーディナリノードは、ルーティングを行う際にはデータを中継するゲートウェイノードの役割を果たす。

アルゴリズムは 3.2 節で述べた DCA をベースに、スレーブクラスタヘッドを選択するための step を加える。2 クラスタヘッドアルゴリズムを実行する際は、前章で述べられている定義を満たすようにする。

#### クラスタリングアルゴリズム（2 クラスタヘッド）

1. 各ノードは  $[0, 1)$  の乱数を生成し、ID とする。
2. 隣接ノードに自 ID 等を送信する。また、隣接ノードから情報を受信する。
3. 隣接ノードの中で最小の ID を持つノードはマスタクラスタヘッドになることを送信してアルゴリズムを停止する。
4. マスタクラスタヘッドに隣接する役割が決まっていないノードは、スレーブクラスタヘッドになることを送信してアルゴリズムを停止する。
5. スレーブクラスタヘッドに隣接する役割の決っていないノードは、オーディナリノードになりアルゴリズムを停止する。
6. 役割の決っていないノードは2に戻る。

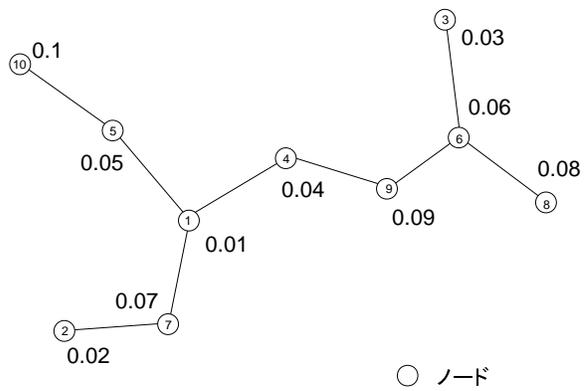


図 3.3: ネットワークトポロジ

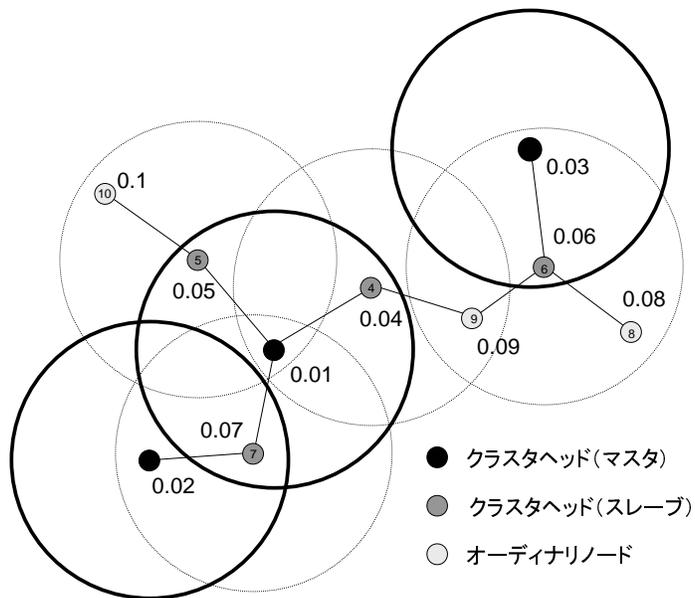


図 3.4: 2 クラスタヘッドのクラスタリング (途中)

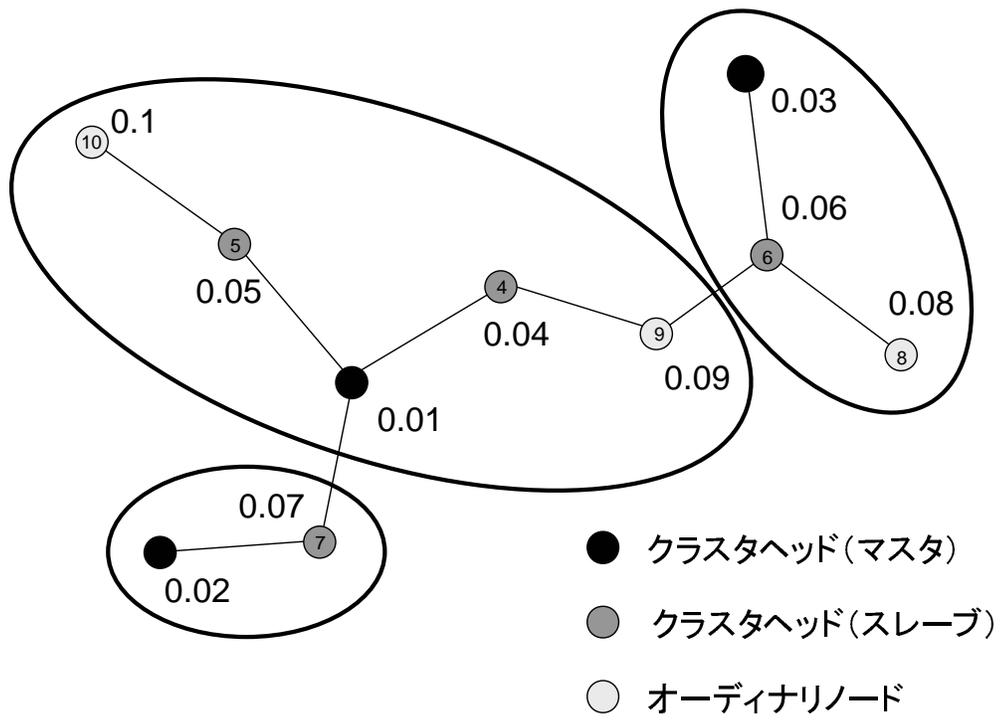


図 3.5: 2 クラスタヘッドのクラスタリング

# 第4章 コリジョンがランダムに発生する場合のクラスタリングアルゴリズム

## 4.1 コリジョン

コリジョンとはネットワークで生じるデータの衝突のことである。同じネットワーク上の異なるノードから同時にデータを送信したときに、そのデータ同士がぶつかってしまい、上手く受信できなくなってしまう。このような問題に対応するための送信の制御をアクセス制御という。無線LANにおけるアクセス制御では、CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) というアクセス制御方法が最も普及している。これは、キャリアを検出して使用中でないことを確認して、空いていたら送信する。一方、受信側は送信されたデータを「受信した」という応答を返す。送信側は、応答がない場合、再び送信しなおすという方法である。この方法は完全に衝突をなくす仕組みと言うよりは衝突が起きたときに、いかに対応するかという仕組みである。コリジョンが発生したときは、全部の応答を待ってから再送するため、有線ネットワークのアクセス制御方法であるCSMA/CDと比較するとパフォーマンスが低いという欠点もある。

さらに、通信範囲が限られている無線ネットワークでは、隠れ端末という問題もある。これは、互いに直接通信することのできないノードが、あるノードに同時に送信してしまいコリジョンを起こしてしまうという現象である(図4.1)。この問題に対してはRTS/CTS(Request To Send/Clear To Send) という方法がある。これは、通信を行う際に相手になるノードにあらかじめ「送信して良いか(RTS)」というメッセージを送り、「送信して良い(CTS)」という返信を受けてから通信を行う。これにより受信側の通信範囲内にいるノードは、RTSを送信したノードであれば通信を開始し、それ以外のノードは、現在、このノードは他のノードと通信中であることを認識することができる。

クラスタリングアルゴリズムを考慮する場合、多く場合、通信は確実に行われるという仮定がなされている。これはアドホックネットワークにおいてコリジョンを考慮していないか、アクセス制御が適切に行われているという仮定になるが、本稿では疑似コリジョンを与え、それがどのようにクラスタリングに影響を与えるかを検討してみることにした。疑似コリジョンにおける仮定は以下の通りである。

1. 無線チャネルとして 0 ~ 99 の値をランダムに与え、隣接ノードに同じチャネルがある場合、そのノード同士がコリジョンしている。
2. コリジョンが起きているノードからの情報は受け取ることができない。
3. クラスタは得ている情報だけで構成される。

さらに、どのくらいの割合でコリジョンを起こしているかの判別するために、以下の計算式により、ある step におけるコリジョン率を算出する。

$$\text{collision 率} = \frac{\text{同一 step におけるコリジョン数}}{\text{同一 step における総通信数}}$$

例として図 4.1 を示す。例では、RTS/CTS 等のアクセス制御は考慮しない。そのため、隠れ端末によるコリジョンは発生する。図は、ある step において、ネットワーク上の全ノードが一斉にブロードキャストし、その中で直接通信することができないノードが、偶然、同じチャネルで送信してしまうことによってコリジョンが発生したところである。

コリジョン率を計算するときは全て単一方向の通信に分解して計算をする。図 4.1 では総通信数は①から⑤までの矢印の合計である。図におけるコリジョン率は以下の通りとなる。

$$\text{collision 率} = \frac{2}{2+2+4+3+3} \approx 0.14$$

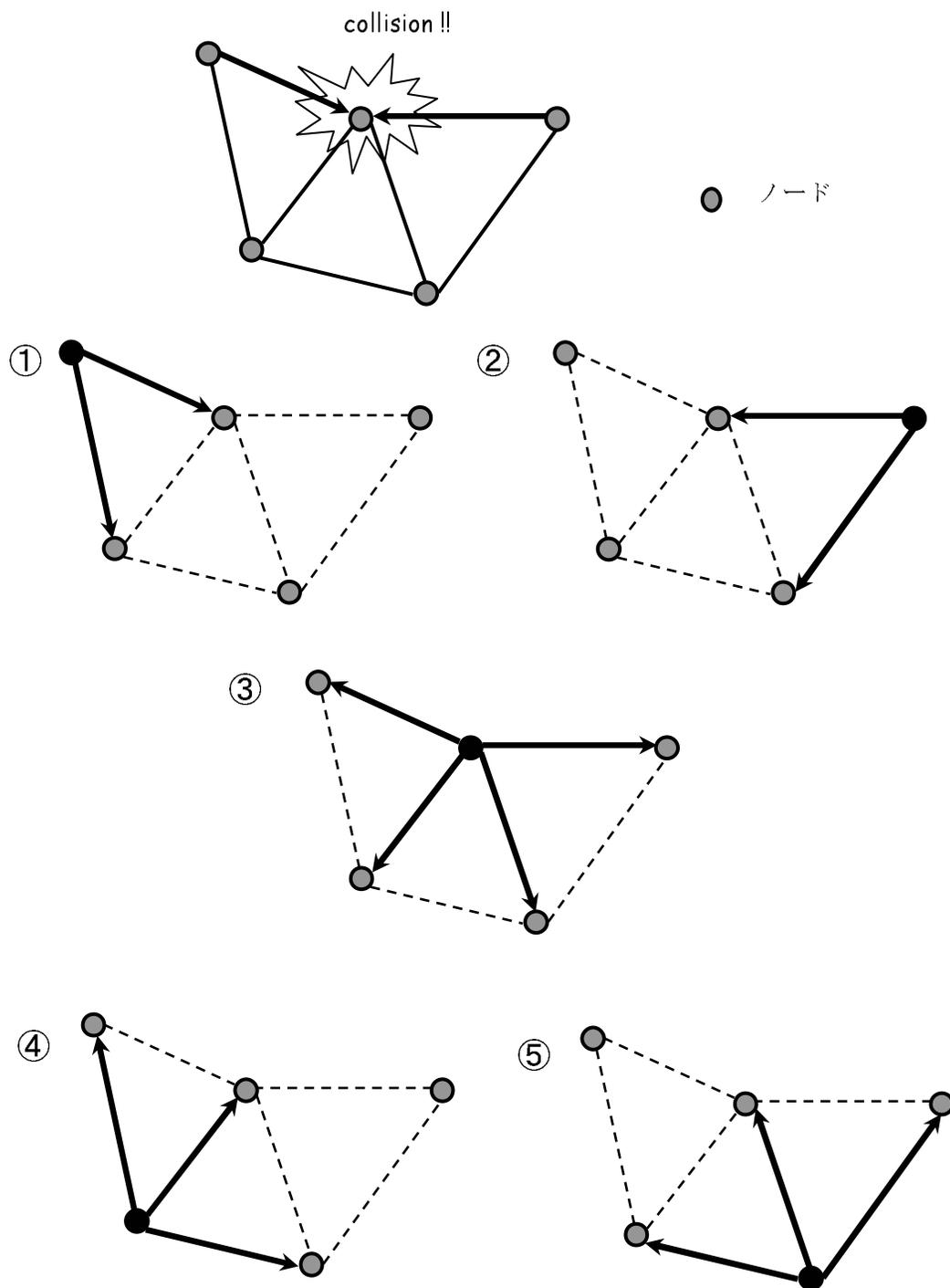


図 4.1: 同一 step におけるコリジョンと総通信数

## 4.2 クラスタリングアルゴリズム

コリジョンを考慮したクラスタリングアルゴリズムは以下の通りである。なお、本稿では、以降、2 クラスタヘッドアルゴリズムに対して、DCA(Distributed Clustering Algorithm) を1 クラスタヘッドアルゴリズムと呼ぶ。

### 1 クラスタヘッドアルゴリズム

1. 各ノードは  $[0, 1)$  の乱数を生成し、ID とする。
2. 隣接ノードに自 ID、状態等を送る。また、隣接ノードから情報を受信する。  
隣接ノードにクラスタヘッドが存在すれば、役割の決まっていないノードは、ID の大きさにかかわらずオーディナリノードになって停止する。
3. 役割の決まっていないノードは（受信することのできている）隣接ノードの中で自 ID が一番小さければクラスタヘッドになることを発信する。これを受信した役割の決まっていないノードは、オーディナリノードになりアルゴリズムを停止する。  
クラスタヘッドは、送信する step において、自分の通信範囲内でクラスタヘッド同士によるコリジョンが発生せず、かつ、他のクラスタヘッドからの送信を受信しなければアルゴリズムを停止する。
4. ‘3’ の発信をクラスタヘッドが受信したら（クラスタヘッド同士が隣接していることになるので）自 ID と比較して、自 ID が送られてきた ID と等しいか、より大きい時は、役割の決まっていないノードに戻り、そのことを発信する。また、自身の通信範囲内でクラスタヘッド同士のコリジョンが発生していれば役割の決まっていないノードに戻る（これらのノードをクラスタヘッド とする）  
クラスタヘッド に従属するオーディナリノードは役割の決まっていないノードに戻る。
5. 役割の決まっていないノードは ‘2’ に戻る。

## 2 クラスタヘッド アルゴリズム

1. 各ノードは  $[0, 1)$  の乱数を生成し, ID とする.
2. 隣接ノードに自 ID 等を発信する. また, 隣接ノードから情報を受信する.  
受信したノードは「my\_cid<sup>1</sup>」と送られてきた内容が異なっていたときは, 役割の決まっていないノードに戻る. (自分のクラスタヘッドが役割を変えていたが, コリジョンで現在までそのことを知らなかったとき)
3. 隣接ノードにマスタクラスタヘッドが存在すれば, 役割の決まっていないノードは, ID の大きさに関係なくスレーブクラスタヘッドになることを送信してアルゴリズムを停止する. これを受信した役割の決まっていないノードはオーディナリノードになりアルゴリズムを停止する.
4. 役割の決まっていないノードは, 受信した隣接ノードの中で自 ID が一番小さければマスタクラスタヘッドになることを送信する.  
マスタクラスタヘッドは, 送信する step において, 自分の通信範囲内でマスタクラスタヘッド同士によるコリジョンが発生せず, かつ, 他のクラスタヘッド (マスタ) からの送信を受信しなければアルゴリズムを停止する.
5. '4' を受信した役割の決まっていないノードはクラスタヘッド (スレーブ) になることを送信してアルゴリズムを停止する. これを受信した役割の決まっていないノードはオーディナリノードになりアルゴリズムを停止する.
6. '4' をマスタクラスタヘッドが受信したら (クラスタヘッド同士が隣接していることになるので), 送られてきた ID と自 ID を比較して, 自 ID が等しいか大きければ, 役割の決まっていないノードに戻ることを送信する. また, マスタクラスタヘッドの通信範囲内でマスタクラスタヘッド同士のコリジョンが発生したら役割の決まっていないノードに戻る. (これらをマスタクラスタヘッド とする)
7. '6' を受信したマスタクラスタヘッド に従属するクラスタヘッド (スレーブ) は 役割の決まっていないノードに戻ることを送信する. これを受信した に従属するオーディナリノードは役割の決まっていないノードに戻る.
8. 役割の決まっていないノードは '2' に戻る.

---

<sup>1</sup> ノードが認識しているクラスタ ID

## 第5章 シミュレーション結果・考察

本章では、第3章、第4章で述べたアルゴリズムを実際に計算機上でシミュレーションして、比較、評価してみる。まず、シミュレーションの概要と構成を述べ、その後シミュレーションの結果と考察を示す。

### 5.1 シミュレーションの概要と構成

本シミュレーションは、有限のフィールドに一定数のノードの初期配置においてクラスタリングが完了するまでの step 数を比較する。ネットワークのモデルは以下の通りである。

- フィールドサイズは  $1000 \times 1000$
- ノード数は 1000
- ノードはランダムに配置される。
- 通信範囲は 10 ~ 100 の範囲で 10 ごとに 10 通りについて試行を行う。
- 各ノードは通信範囲の中にいれば相互に通信可能とする。
- シミュレーションは各通信範囲ごとに 100 回づつ試行する。

このモデルは全てのシミュレーションに共通である。クラスタリングアルゴリズムは、1 クラスタヘッド ( 最小 ID 法 DCA ) と 2 クラスタヘッド ( スレーブクラスタヘッドを加えたアルゴリズム ) を用いる。背景は、コリジョンを考慮しない ( ノード間の通信が確実に行える ) 場合と、コリジョンが発生する場合の 2 通りについて考える。なお、コリジョンがランダムに発生する場合については、4.1 節で述べられている擬似コリジョンの条件を満たしながら、クラスタリングを行う。実験結果は、それぞれのアルゴリズムにおいて、各通信範囲ごとに試行を 100 回行った結果、クラスタリングが完了するまでにかかる step 数の分布、平均 step 数の比較、ヒストグラムを示す。

## 5.2 結果 1 : コリジョンを考慮しない場合

結果について考察する。通信範囲が 10 ~ 30 の狭い範囲は、フィールド内で相互に通信ができるノードが極端に少ない状況である。若干、差はあるものの、平均 step 数は 1 クラスターヘッドアルゴリズムと 2 クラスターヘッドアルゴリズムで大きな差はない。step 数の分布で見ると表 5.1 の 1 クラスターヘッドアルゴリズムでは 4step~6step に多く分布している。このアルゴリズムでは、再帰的なアルゴリズムの 1 ターンにつき 2step あるので、およそ 2 ~ 3 ターンしていることになる。一方、表 5.2 の 2 クラスターヘッドアルゴリズムでは 3 ~ 6step に多く分布している。このアルゴリズムは 1 ターンにつき 3step ある。ゆえに、1 ~ 2 ターンでクラスタリングを行っていることになる。次に、通信範囲が大きくなるにつれ、一旦は増えるものの、2 クラスターヘッドアルゴリズムの step 数は減っていく。一方、1 クラスターヘッドアルゴリズムは徐々に step 数を積み重ねている。このことは図 5.2~ 図 5.11 までのヒストグラムからも読み取ることができる。これは通信範囲が大きくなると、2 クラスターヘッドアルゴリズムは 1 クラスターヘッドアルゴリズムと比べ、1 ターン当り、役割の決まるノードの数が多くなるためである。

表 5.1: シミュレーションにおける STEP 数 : 1 クラスターヘッド

step	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	average	
range10	0	1	0	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3.98
range20	0	0	0	70	0	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4.6
range30	0	0	0	6	0	88	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	6
range40	0	0	0	0	0	71	0	26	0	3	0	0	0	0	0	0	0	0	0	0	0	6.64
range50	0	0	0	0	0	61	0	38	0	1	0	0	0	0	0	0	0	0	0	0	0	6.8
range60	0	0	0	0	0	53	0	46	0	1	0	0	0	0	0	0	0	0	0	0	0	6.96
range70	0	0	0	0	0	42	0	55	0	2	0	1	0	0	0	0	0	0	0	0	0	7.24
range80	0	0	0	0	0	37	0	57	0	6	0	0	0	0	0	0	0	0	0	0	0	7.38
range90	0	0	0	0	0	31	0	62	0	6	0	1	0	0	0	0	0	0	0	0	0	7.54
range100	0	0	0	0	0	35	0	56	0	8	0	1	0	0	0	0	0	0	0	0	0	7.5

表 5.2: シミュレーションにおける STEP 数 : 2 クラスターヘッド

step	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	average	
range10	0	0	86	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3.28
range20	0	0	4	0	63	33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5.25
range30	0	0	0	0	13	84	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	5.93
range40	0	0	0	0	10	88	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	5.94
range50	0	0	1	0	16	82	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	5.83
range60	0	0	7	0	24	68	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	5.58
range70	0	0	16	0	26	58	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5.26
range80	0	0	30	0	17	51	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	4.98
range90	0	0	37	0	15	48	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4.74
range100	0	0	42	0	18	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4.56

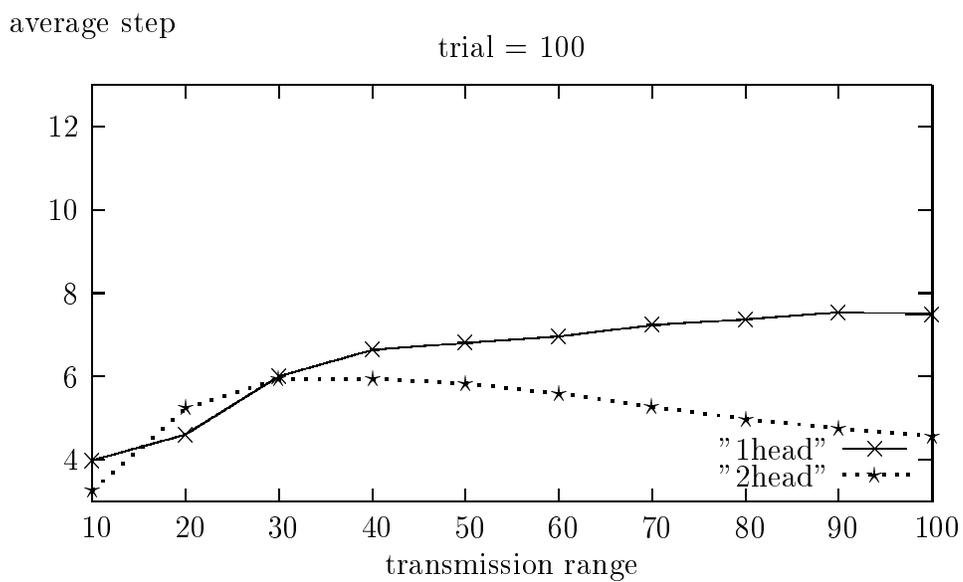


図 5.1: 通信半径におけるの平均STEP数の比較

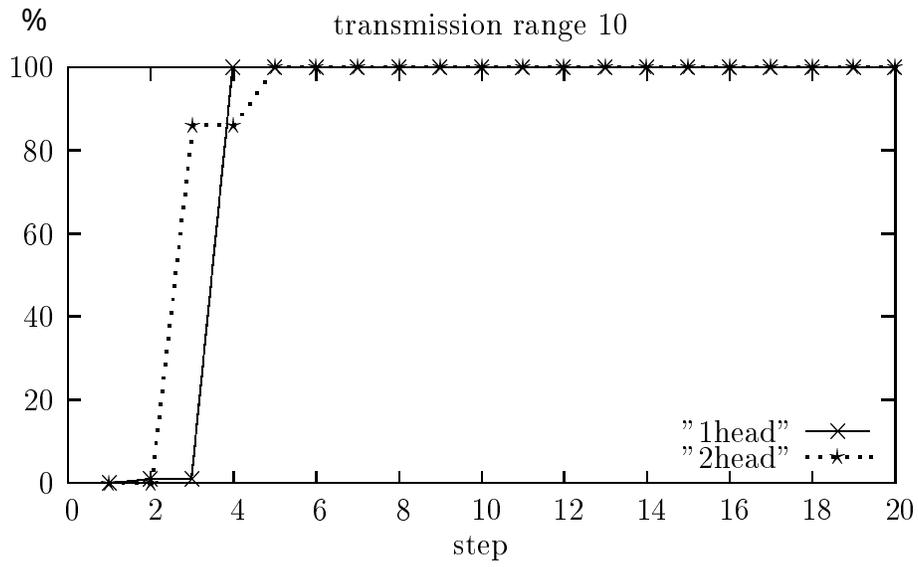


図 5.2: 通信半径ごとのSTEP数の比較

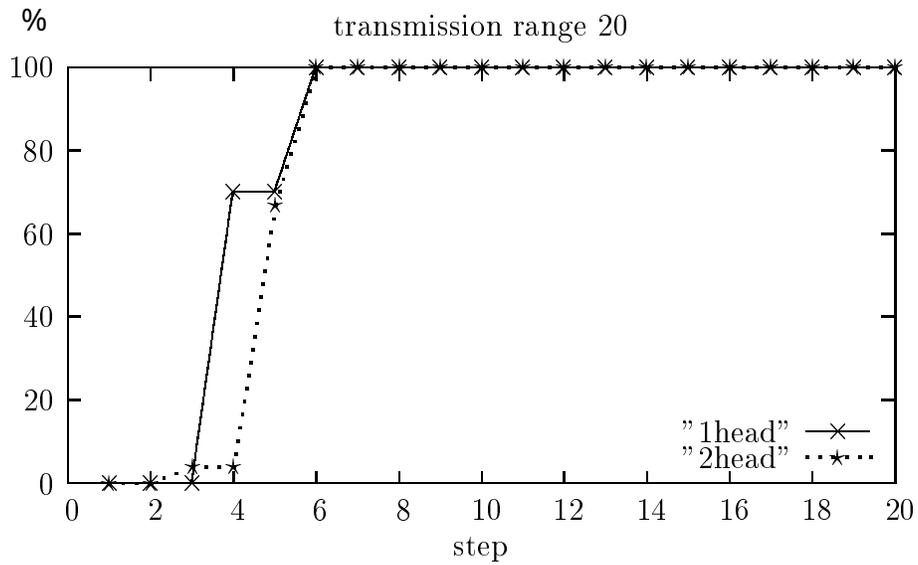


図 5.3: 通信半径ごとのSTEP数の比較

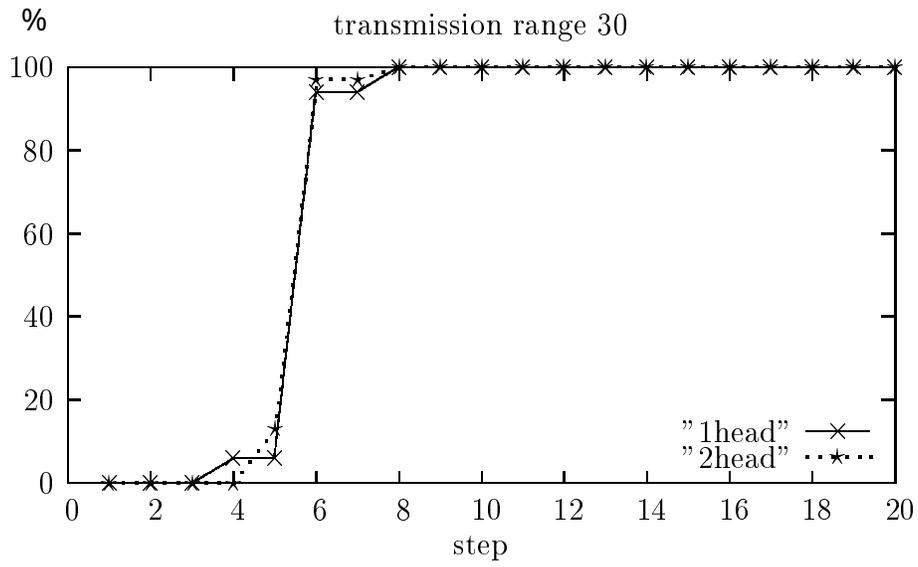


図 5.4: 通信半径ごとのSTEP数の比較

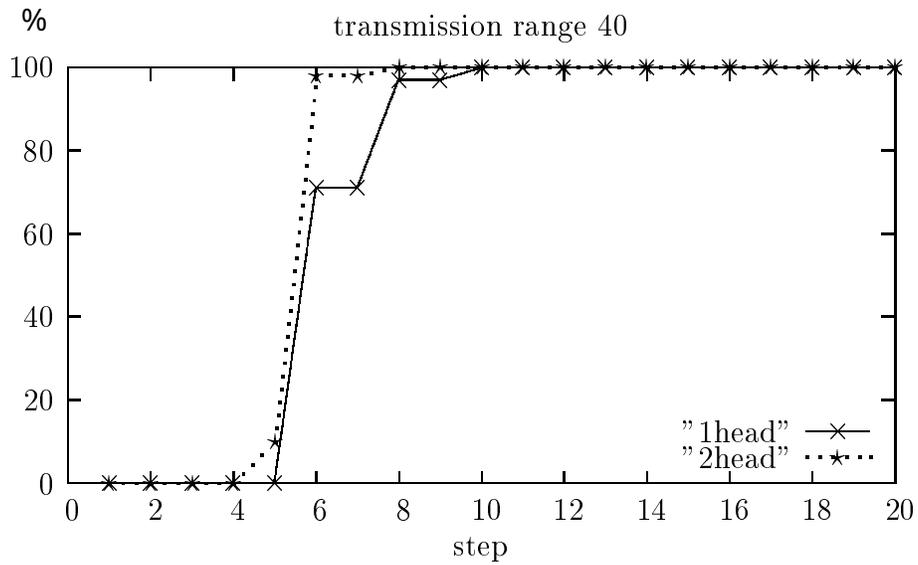


図 5.5: 通信半径ごとのSTEP数の比較

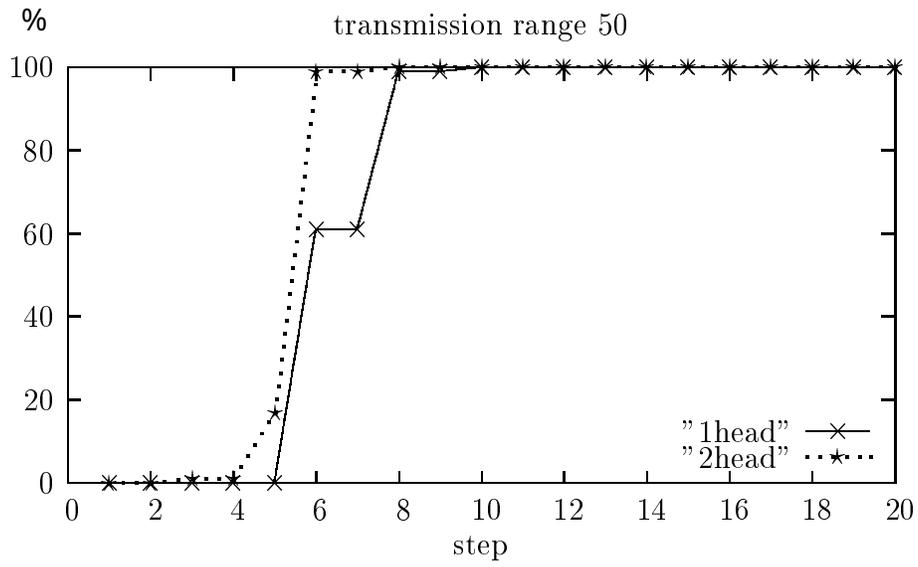


図 5.6: 通信半径ごとのSTEP数の比較

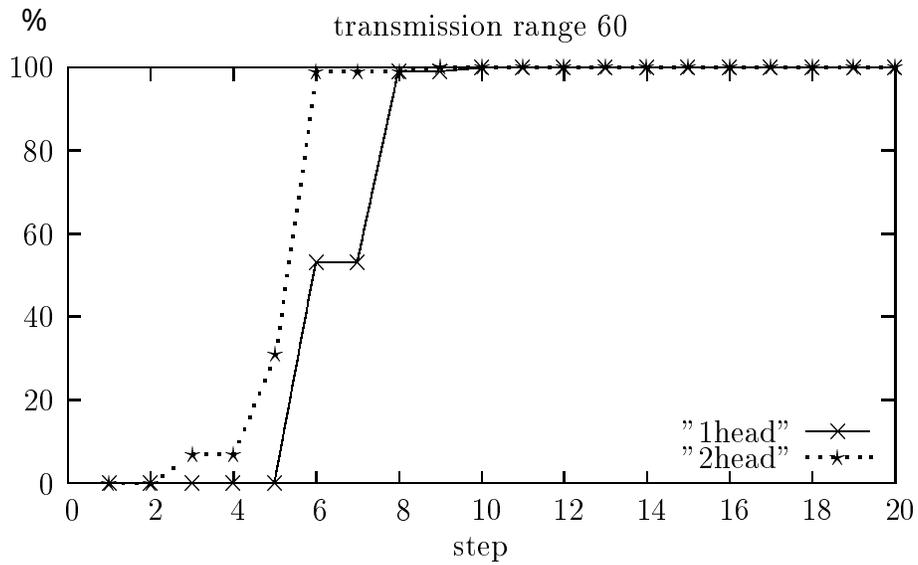


図 5.7: 通信半径ごとのSTEP数の比較

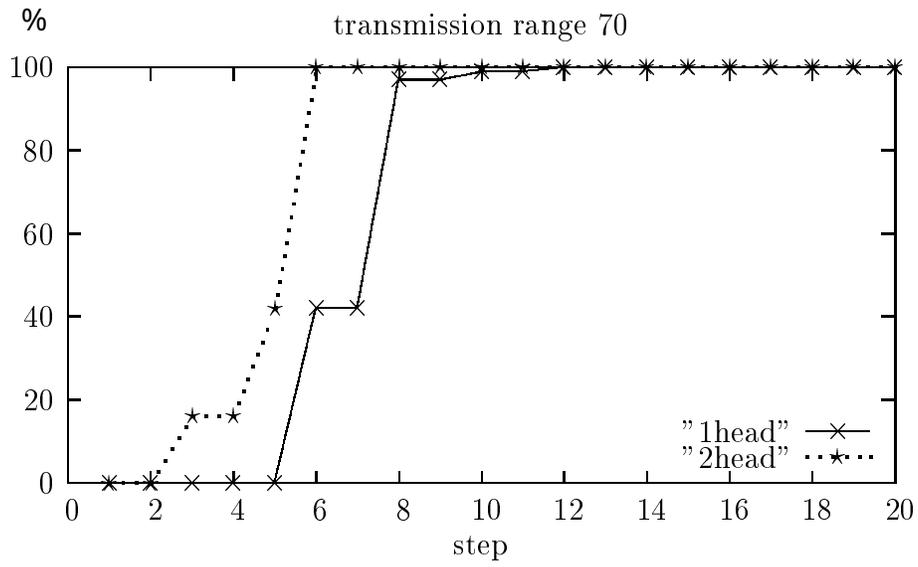


図 5.8: 通信半径ごとのSTEP数の比較

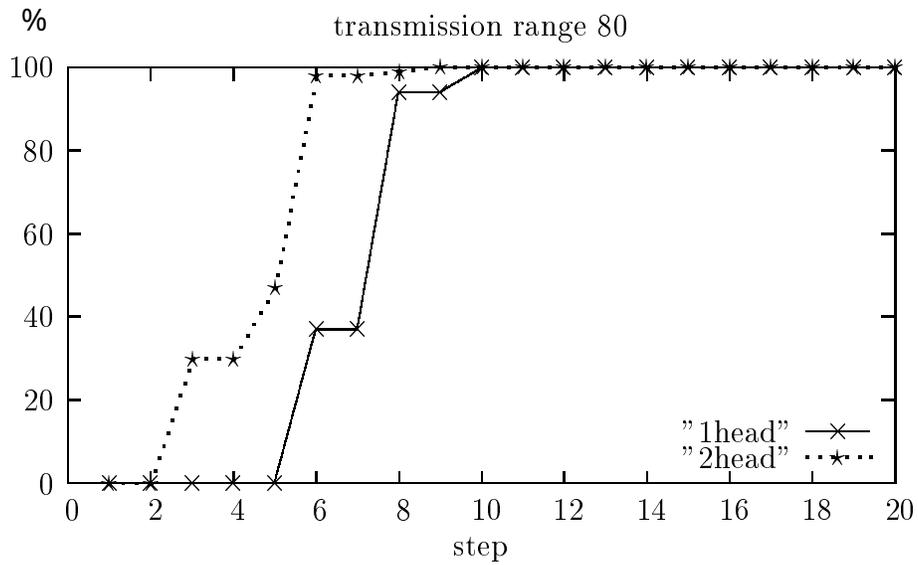


図 5.9: 通信半径ごとのSTEP数の比較

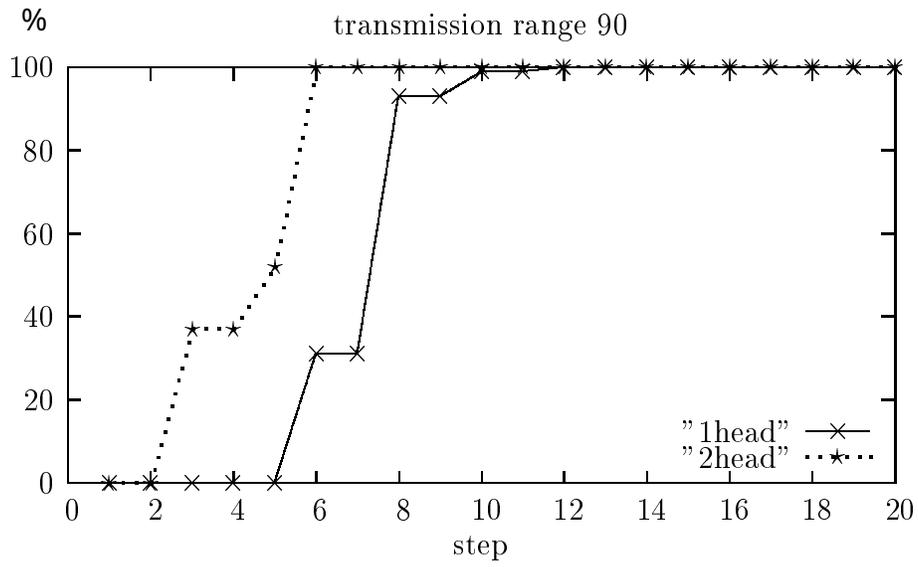


図 5.10: 通信半径ごとのSTEP数の比較

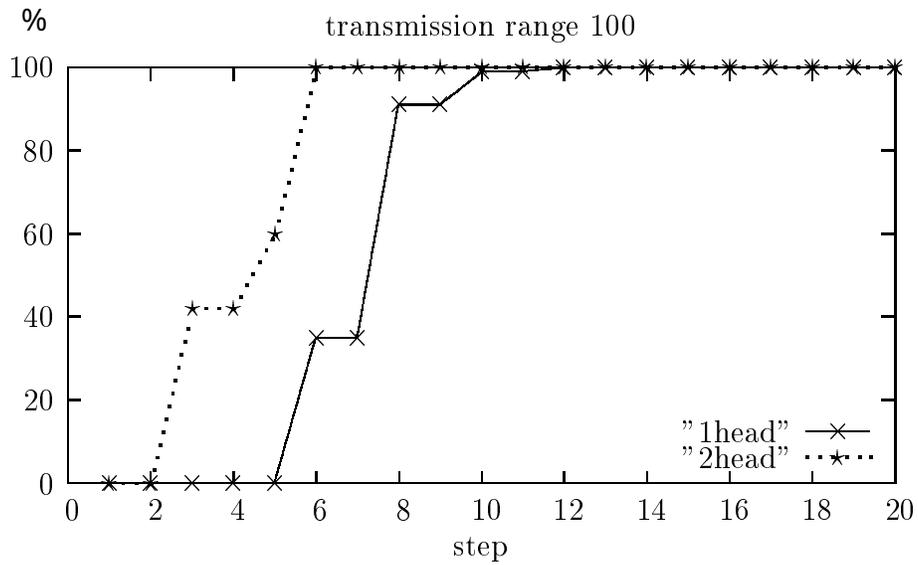


図 5.11: 通信半径ごとのSTEP数の比較

### 5.3 結果2：コリジョンがランダムに発生するとき

5.2節では、ノード間の通信が確実にできるという仮定のもとで、シミュレーションを行ってきた。しかし、無線通信において、ノードから送られてくるデータの衝突（コリジョン）に対して、どのように対応するかは重要な問題である。現実的な対応としては現在のところ、CSMA/CA や RTS/CTS といった方策が講じられている。しかし、今回のシミュレーションでは、4.1節で述べた擬似コリジョンのもとに、4.2節で示されているアルゴリズムを実行する。そのため、隠れ端末によるコリジョンは起きてしまう。なお、コリジョンがどの程度発生しているかの目安として、4.1節で示される計算式でコリジョン率を表す。コリジョン率は同一 step におけるコリジョン数と総通信数の比になる。表 5.3 は、試行 100 回に行ったときの第 1 step における平均のコリジョン率である。第 1 step なので、コリジョンの発生条件は 1 クラスタヘッドアルゴリズムと 2 クラスタヘッドアルゴリズムにおいて全く同一である。通信範囲が狭いと直接通信を行えるノードの数が少なくなるのでコリジョン率は低く、通信範囲が大きくなるとコリジョン率は高くなる。

表 5.3: 第 1 step における平均 collision 率

range	10	20	30	40	50	60	70	80	90	100
collision 率	0.56	1.89	3.61	5.62	8.16	11.1	14.4	18.0	22.0	26.1

シミュレーション結果について考察する。コリジョンのないケースと比較すると、step 数の分布で表 5.1 では 2step ごとに、表 5.2 は 3step ごとに step 数が分布している。しかし、表 5.4 と表 5.5 では、コリジョンを原因とする、役割を取り消すための step が加わるため、step 数の分布が以前ほど規則的ではなく、クラスタリングを完了するまでの step 数も増加している。図 5.12 において 1 クラスタヘッドアルゴリズムと 2 クラスタヘッドアルゴリズムを比較すると、前節の結果と同様、通信範囲が狭い範囲（10～40）では平均の step 数がほぼ同じである。また、通信範囲が大きくなるにつれ 2 つのアルゴリズムの step 数の差は大きくなる。しかし、前節の結果に比べ、その差があまり大きくなる。そこで、表 5.3 を見ると、通信範囲が大きくなるほどコリジョン率が高くなっている。また、表 5.6 と表 5.7 は、試行 100 回における、平均のノードの役割の分布を表している。この表が示すように 1 クラスタヘッドアルゴリズムにおけるクラスタヘッドと、2 クラスタヘッドアルゴリズムにおけるマスタクラスタヘッドの数を比べると、通信範囲の大きいところではマスタクラスタヘッドの数は 1 クラスタヘッドの数の半分近くになっている。これは 2 クラスタヘッドアルゴリズムの方がクラスタ内での階層が深いため、コリジョンによりノードの役割の取消し（特にマスタクラスタヘッド）で受ける影響が大きく、その結果、通信範囲の大きいところでも、アルゴリズムの 1 ターン当りに役割の決まるノードの数が、“コリジョンが無いとき” に比べ、少なくなってしまうためと思われる。

表 5.4: シミュレーションにおけるSTEP数：1 クラスタヘッド

step	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	average
range10	0	0	0	27	1	29	40	1	2	0	0	0	0	0	0	0	0	0	0	0	5.93
range20	0	0	0	2	0	8	0	41	10	39	0	0	0	0	0	0	0	0	0	0	7.74
range30	0	0	0	0	0	2	2	12	65	1	11	4	1	2	0	0	0	0	0	0	9.27
range40	0	0	0	0	0	0	0	2	44	2	29	8	9	5	0	1	0	0	0	0	10.5
range50	0	0	0	0	0	0	0	3	25	4	30	10	11	13	2	2	0	0	0	0	11.26
range60	0	0	0	0	0	0	0	3	12	3	34	9	22	12	1	3	1	0	0	0	11.78
range70	0	0	0	0	0	0	0	1	10	1	34	5	28	12	4	3	1	1	0	0	12.17
range80	0	0	0	0	0	0	0	2	9	5	23	8	20	20	2	6	3	2	0	0	12.49
range90	0	0	0	0	0	0	0	2	4	1	28	7	30	20	6	1	0	1	0	0	12.48
range100	0	0	0	0	0	0	0	0	6	1	24	7	30	15	7	6	2	2	0	0	12.83

表 5.5: シミュレーションにおけるSTEP数：2 クラスタヘッド

step	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	average
range10	0	0	21	0	25	35	13	6	0	0	0	0	0	0	0	0	0	0	0	0	5.37
range20	0	0	0	0	2	2	14	42	1	4	3	0	0	0	0	0	0	0	0	0	8.34
range30	0	0	0	0	0	2	12	11	47	1	22	3	1	0	1	0	0	0	0	0	9.23
range40	0	0	0	0	0	0	3	8	39	2	29	10	7	2	0	1	0	0	0	0	10.14
range50	0	0	0	0	0	4	6	6	22	0	39	18	5	0	0	0	0	0	0	0	10.22
range60	0	0	0	0	0	3	4	5	26	3	33	16	7	3	0	0	0	0	0	0	10.38
range70	0	0	0	0	0	3	4	7	25	4	31	15	9	2	0	0	0	0	0	0	10.33
range80	0	0	0	0	1	2	7	7	22	6	23	22	6	4	0	0	0	0	0	0	10.31
range90	0	0	0	0	2	2	6	2	17	9	34	19	2	5	1	1	0	0	0	0	10.52
range100	0	0	0	0	1	5	8	3	21	5	26	20	7	2	0	1	1	0	0	0	10.32

表 5.6: 平均のノードの役割の分布：1 クラスタヘッド

range	10	20	30	40	50	60	70	80	90	100
クラスタヘッド	868.73	616.71	414.98	286.31	206.45	153.99	120.75	96.52	78.85	65.93
オーディナリノード	131.27	383.29	585.02	713.69	793.55	846.01	879.25	903.48	921.15	934.07

表 5.7: 平均のノードの役割の分布：2 クラスタヘッド

range	10	20	30	40	50	60	70	80	90	100
マスタクラスタヘッド	864.14	581.6	349.74	215.44	139.66	97.37	71.94	55.42	43.96	35.62
スレーブクラスタヘッド	130.96	374.28	548.33	638.62	675.21	690.28	697.07	695.24	691.29	686.13
オーディナリノード	4.9	44.12	101.93	145.94	185.13	212.35	230.99	249.34	264.75	278.25

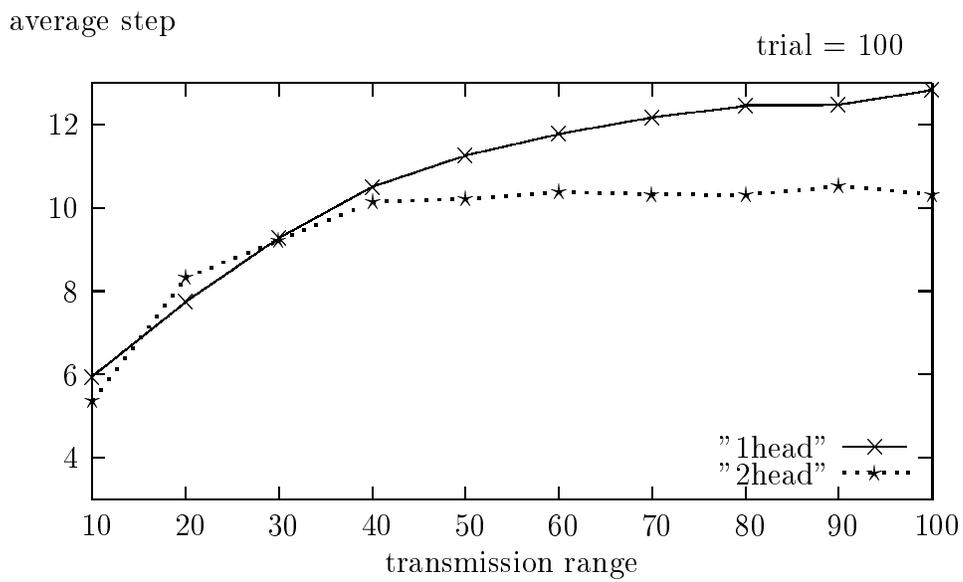


図 5.12: 通信半径におけるの平均STEP数の比較

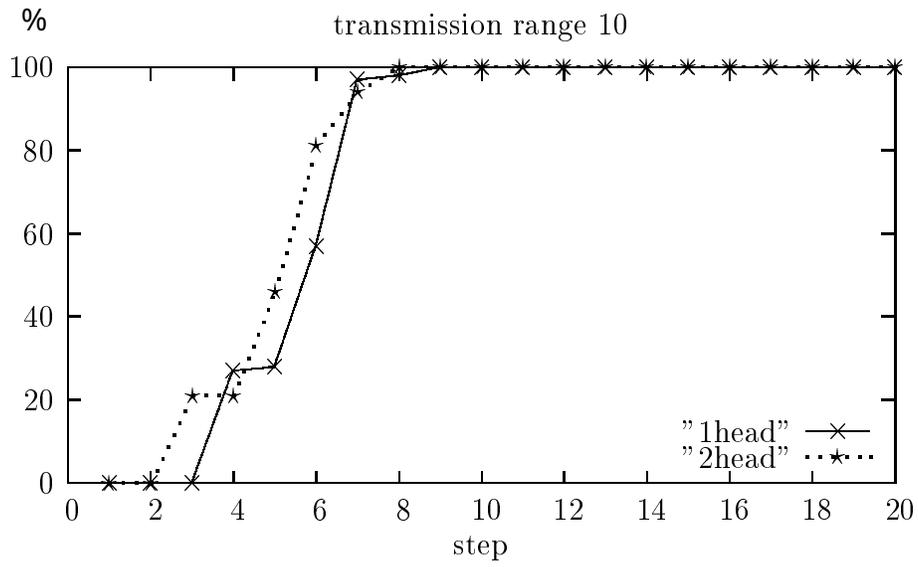


図 5.13: 通信半径ごとのSTEP数の比較

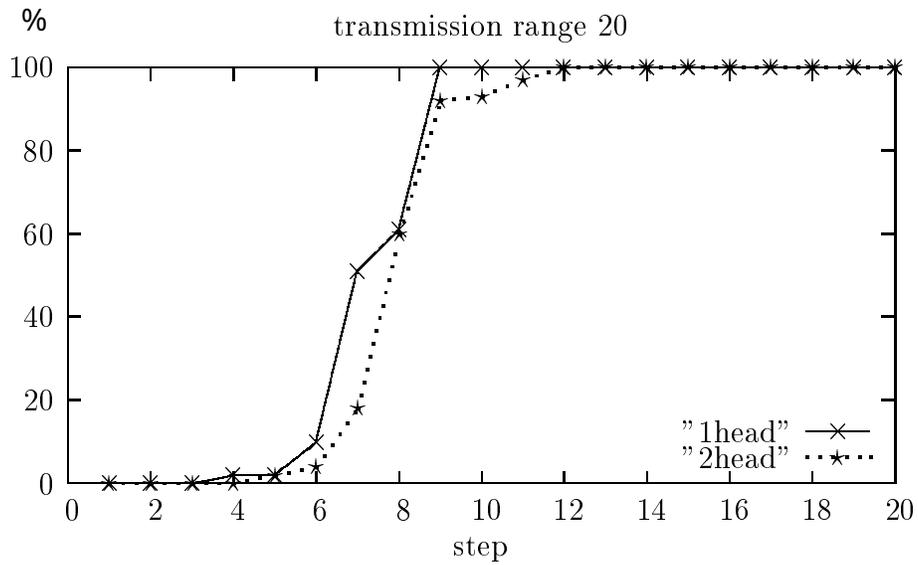


図 5.14: 通信半径ごとのSTEP数の比較

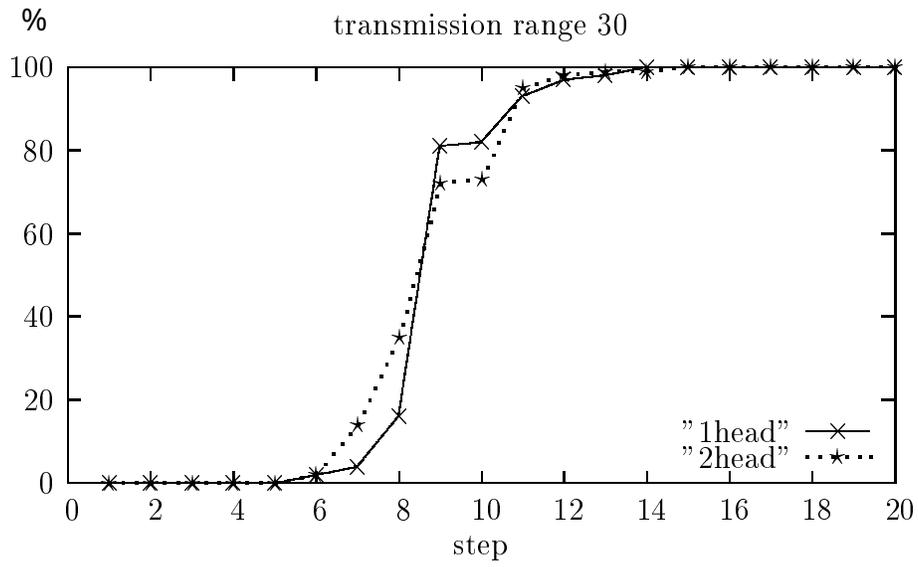


図 5.15: 通信半径ごとのSTEP数の比較

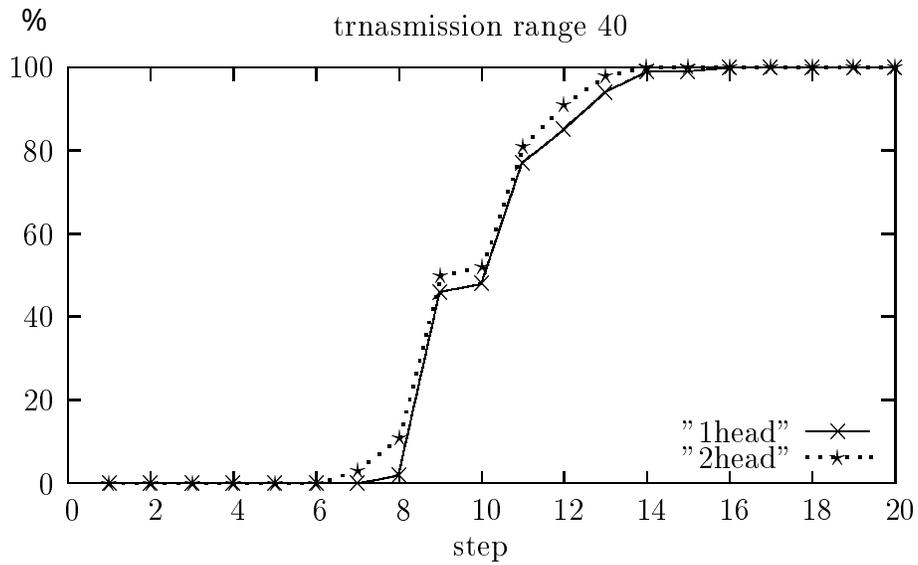


図 5.16: 通信半径ごとのSTEP数の比較

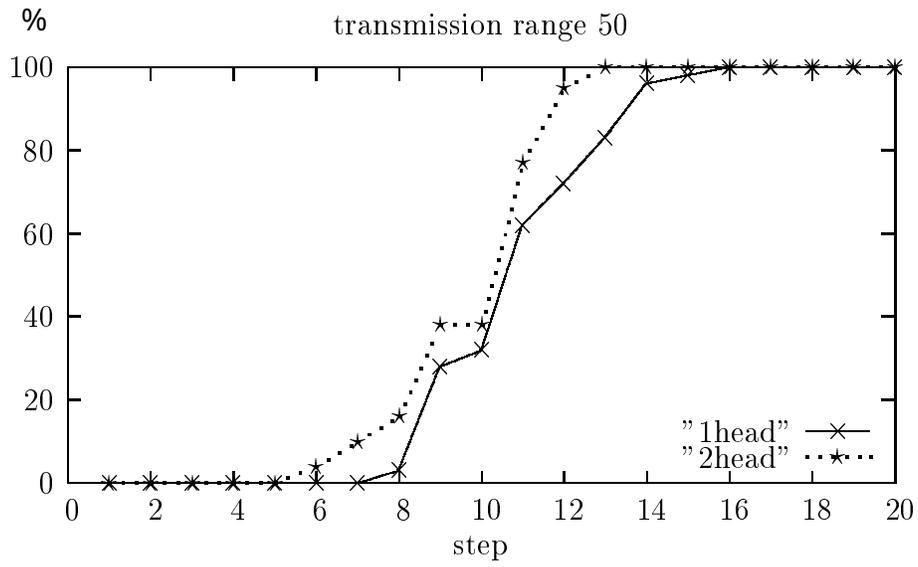


図 5.17: 通信半径ごとのSTEP数の比較

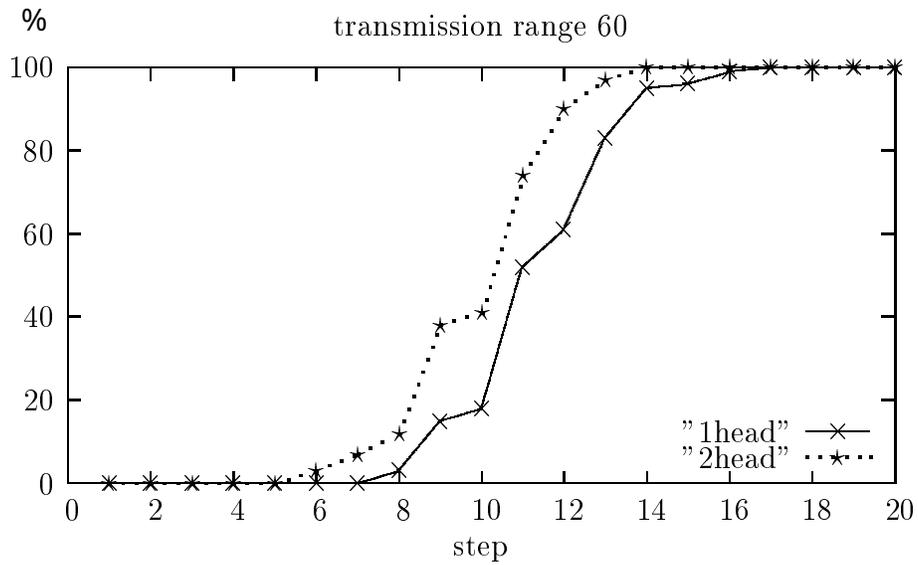


図 5.18: 通信半径ごとのSTEP数の比較

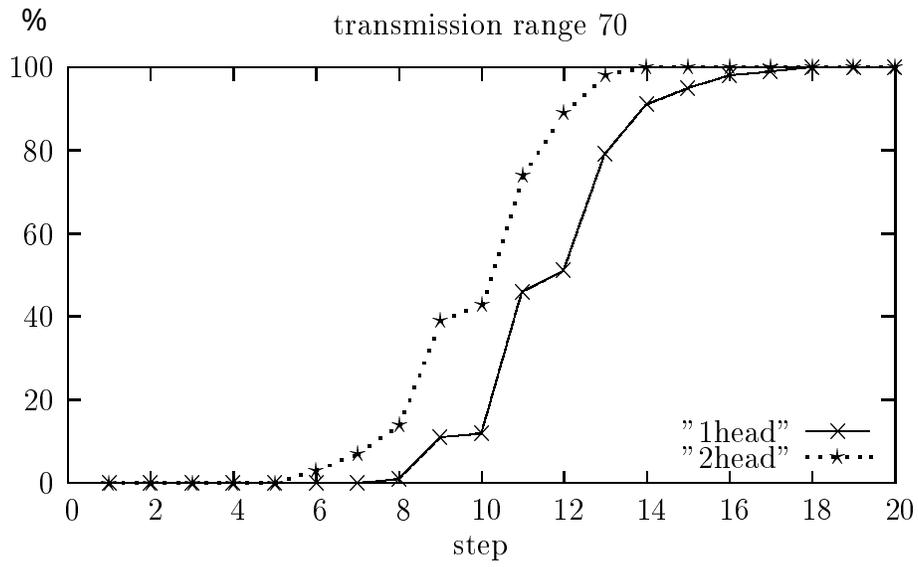


図 5.19: 通信半径ごとのSTEP数の比較

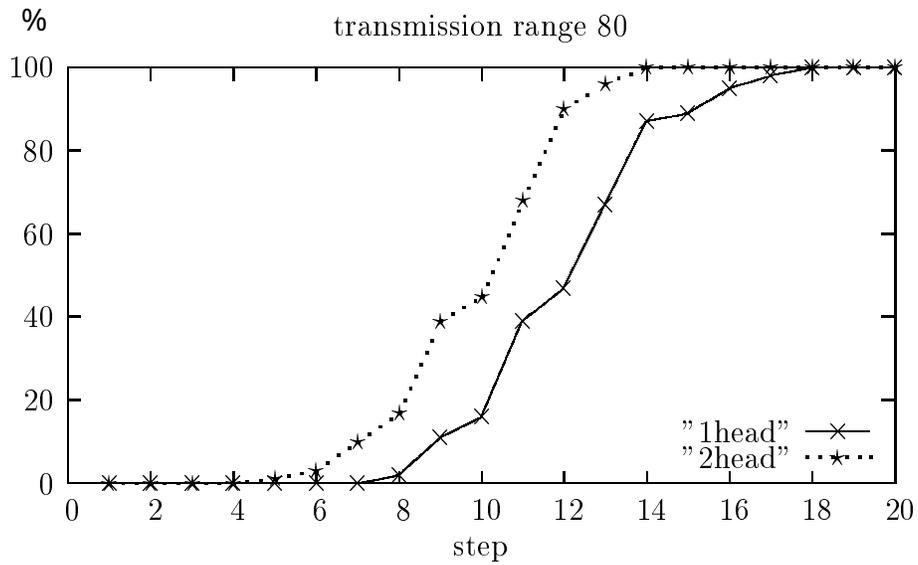


図 5.20: 通信半径ごとのSTEP数の比較

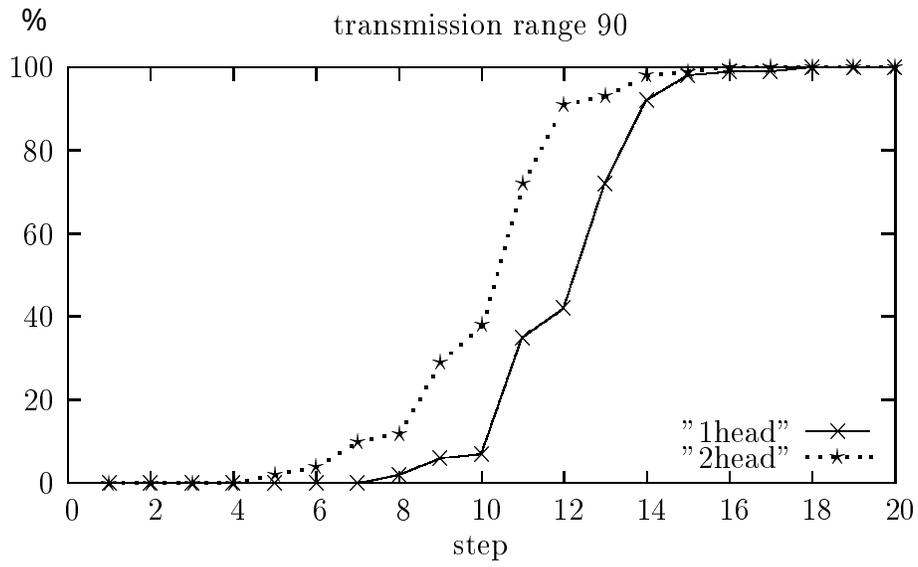


図 5.21: 通信半径ごとのSTEP数の比較

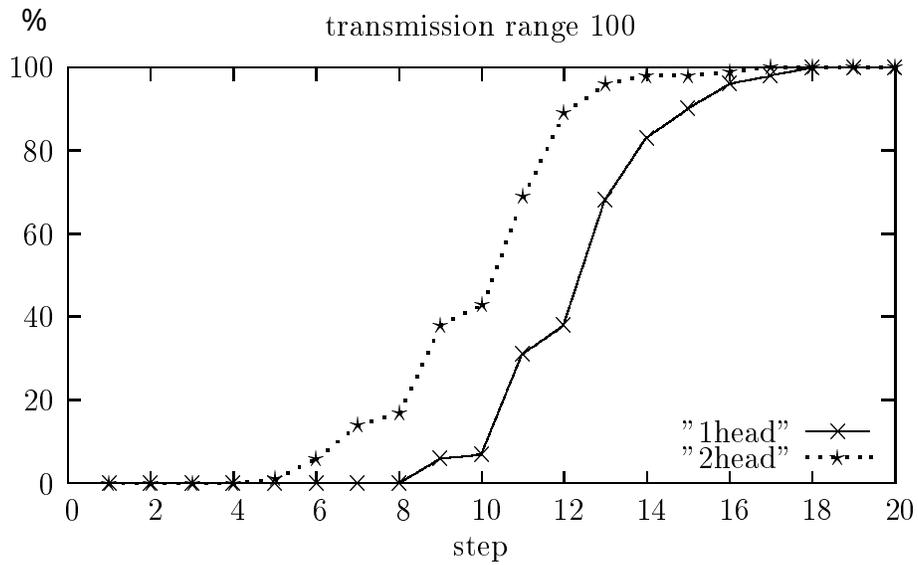


図 5.22: 通信半径ごとのSTEP数の比較

## 第6章 まとめ

本論文では、アドホックネットワーク上のクラスタリングにおけるアルゴリズムである DCA ( Distributed Clustering Algorithm) をベースに、クラスタヘッドをマスタクラスタヘッドとスレーブクラスタヘッドという「2種類のクラスタヘッド」と、それに隣接するオーディナリノードからなる「2クラスタヘッドアルゴリズム」を提案した。

クラスタリングを行う際には、クラスタヘッドを決めるための優先度を何にするかを決めなければならないが、種々ある既存の選択方法のなかで、ノードが持つ ID をもとに、ID の最も小さいものを優先する最小 ID 法を用いることとした。なお、ID は  $[0, 1)$  のランダムな値を割り当てる。さらに、クラスタリングアルゴリズムを操作する仮定において、情報が常に正しく受け取られるケースと、コリジョンによって情報のいくつかが欠落してしまう可能性があるケースについて評価してみた。評価は特にクラスタリングが完了するまでの step 数に注目して行った。

有線のネットワークにおいて通信資源の有効活用の方法としてネットワークの階層化が行われている。無線端末だけからなるアドホックネットワークにおいても、同様に通信資源を有効利用するため、階層化構造を構築することは重要である。そのため、クラスタを構成して、ノードの限られた能力を最大限に活用できるようにしなければならない。従来 DCA では、1つのクラスタヘッドとそれに隣接する全てのオーディナリノードによって、ネットワークを複数のクラスタに分割していた。提案したアルゴリズムでは、クラスタヘッドを「マスタクラスタヘッド」と「スレーブクラスタヘッド」という2種類のクラスタヘッドにして階層化した。このことにより、クラスタリングの step 数が、少なくとも 1 step は多くなることになるが、総 step 数を減らすことができることをシミュレーションによって示した。特にコリジョンを考慮しない場合では、通信範囲が大きくなると step 数の差は大きなものになった。ノード間の通信にコリジョンがある場合は、コリジョンによって不十分な情報をもとに決定したノードの役割が、一旦クリアされ、再びクラスタリングを行うため、コリジョンを考慮しない場合のものと比較すると、その step 数の差は若干縮まってしまった。これは、2クラスタヘッドアルゴリズムが、クラスタ内において、DCA より深い階層構造を構築するため、ノード ( マスタクラスタヘッド ) が役割の取消をしたとき、再クラスタリングに加わるノードの数が多くなるからだと考えられる。

# 謝辞

本研究を進めるにあたり、適切なお指導とご助言を賜りました情報科学研究科 浅野 哲夫 教授, 中野 浩嗣 助教授, 小保方幸次助手, 元木光雄助手に深く感謝致します。

また, 有益な御討論, 並びに, ご助言を頂きました情報基礎学講座の皆様には感謝致します。

## 参考文献

- [1] A.Ephremides, J.E.Anthony, D.J.Barker, A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling, Proc.IEEE,75(1), pp.56-73, 1987.
- [2] M.Gerla, J.T.C.Tsai, Multicluster, mobile, multimedia radio network, ACM-Baltzer Journal of Wireless Networks,1(3), pp.225-265, 1995.
- [3] 谷口博人, 井上美智子, 増澤利光, 藤原秀雄, アドホックネットワークにおけるクラスタ構成法, 電気情報通信学会論文誌 D-I Vol.J84-D-I No.2 pp127-135,2.2001
- [4] S.Basagni, Distributed Clustering for Ad Hoc Network, Proc.International Symposium on Parallel Architectures, Algorithms, and Networks, pp.310-315, 1999.
- [5] C.R.Lin, M.Gerla, Adaptive Clustering for Mobile Wireless Networks, IEEE J.Selected Areas in Communication, vol15, no.7, pp1265-1275,1997
- [6] 山崎浩輔, 瀬崎薫, ネットワーク的側面から見た Location Aware サービスの現状と技術的課題, 生産研究 5 月号, 2001
- [7] Stephanie Lindsey, Cauligi S.Raghavendra Energy Efficient Broadcasting for Situation Awareness in Ad Hoc Networks Proc.2001 International Conference on Parallel Processing pp149-155, 2001
- [8] 呉屋健, アドホックネットワークにおけるクラスタ分割アルゴリズムに関する研究, 京都大学大学院, 2000.
- [9] 河口信夫 アドホックネットワークにおけるモバイルエージェントの応用ソフトウェア学会 MACC98, 1998.
- [10] 河口信夫 外山勝彦 稲垣康善 モバイルエージェントによるアドホックネットワークの構築ソフトウェア科学回 SPA '99, 1999.