

## 移動オブジェクトのクラスタリング手法に関する一提案

羅 勇<sup>†</sup> 天 笠 俊 之<sup>†</sup> 波 多 野 賢 治<sup>†</sup>  
宮 崎 純<sup>†</sup> 植 村 俊 亮<sup>†</sup>

モバイル端末の普及とともに、多人数が無線ネットワークを介して情報を交換する環境が一般になりつつある。モバイル端末はその性能に制約があるため、効率的な運用を行うことが重要である。特にモバイル端末の移動を考慮に入れた効率的な Ad-Hoc ネットワーク環境を構築するためのクラスタリングは効率のよい情報交換には欠くことのできない要素技術である。本研究では、Ad-Hoc ネットワークにおいて、移動オブジェクトの移動ベクトルを考慮したクラスタリング手法を提案する。これにより、移動オブジェクト間のネットワークポロジをより長く維持することができるようになる。

### A Clustering Technique for Moving Objects

YONG LUO,<sup>†</sup> TOSHIYUKI AMAGASA,<sup>†</sup> KENJI HATANO,<sup>†</sup> JUN MIYAZAKI<sup>†</sup>  
and SHUNSUKE UEMURA<sup>†</sup>

With the spread of moving objects (NotePC, cellular-phone, and all), the environment where many people exchange information through a fully radio network is becoming general. Since the moving objects have constraint in the performance, they are efficiently used as information terminals is being important. Particularly, since the mobile terminals change their positions with movement of users, clustering technique for building efficient Ad-Hoc network environment, which takes their movements into consideration, is necessary. In this paper, we propose a clustering technique based on the movement of the moving objects in a radio network. Thus, the network topology between moving objects can be maintained for a long time.

#### 1. はじめに

近年、無線通信技術が著しい発展を見せている。また、ノートパソコン、PDA や携帯電話に代表される情報端末が小型化・高機能化しており、ワイヤレスネットワークを介して情報を得る環境も整いつつある。モバイル端末には CPU 処理能力、メモリ容量、電源などといった機能的な制約、及びワイヤレスネットワークの通信能力などのインフラ環境の制約があるが、移動しながら通信できるという特徴を利用して、移動オブジェクトへの情報配信、移動オブジェクトによる Ad-Hoc ネットワーク、移動オブジェクトに対する LBS(Location-Based Services) などの応用が注目されている。

今後このような移動オブジェクトの数は増え続けることが予想されるため、基地局による移動オブジェクトの集中管理、移動オブジェクトと基

地局の一対一の通信などにパフォーマンスの問題が出てくることが予想される。この問題を克服する技術の一つとして Ad-Hoc ネットワークが注目されており、その構築のために、移動端末のクラスタリング手法が要素技術として重要である。モバイル端末の移動を考慮したクラスタリング手法を使えば、ある基準を用いて複数の移動オブジェクトを幾つかのグループに分けることができ、個々の移動オブジェクトを直接管理するよりも計算コストや通信コストを抑えることができる。これは移動オブジェクトにとっては電源消費量を節約することにもつながる。例えば、実際に複数の移動オブジェクトをクラスタリングし、それぞれの外部通信路を仮想的な一つの通信路に見立てて、互いに仮想通信路を共有することで、通信効率を高める研究がある<sup>1),2)</sup>。また、ルーティングのスループット向上および通信トラフィック低減を図るための Ad-Hoc ネットワークの研究<sup>3)-6)</sup>にもクラスタリング手法が取り入れられている。しかし、これらの研究では、モバイル端末の移動を考慮していない。

<sup>†</sup> 奈良先端科学技術大学院大学 情報科学研究科  
Graduate School of Information Science, Nara Institute of Science  
and Technology

移動オブジェクトをクラスタリングする際には、その移動特性を考慮することが重要である。類似した方向に類似した速度で移動しているオブジェクトをグループ化することができれば、グループの構成をより長く保つことが期待できる。例えば、戦場や遊園地などの環境における Ad-Hoc ネットワークを考えると、安定した情報の共有を実現するため、共有する移動オブジェクトの間のトポロジをできるだけ安定して保つことが要求される。

この考え方に基づき、我々は移動オブジェクトの移動ベクトルを考慮したクラスタリング手法を提案する。提案手法において、各移動オブジェクトはできるだけ似たような移動ベクトルを持つ自分の近接移動オブジェクトでクラスタを構成する。この結果クラスタ内においてネットワークトポロジは比較的安定となることが期待される。

本稿の構成は以下の通りである。2章では、関連研究として、Ad-Hoc ネットワークで使われる主なクラスタリング手法を説明する。3章では我々の提案手法の基本概念を述べる。4章では提案手法である移動ベクトルを考慮したクラスタリングについて述べる。5章はまとめである。

## 2. 関連研究

本章では、Ad-Hoc ネットワークで使われる主なクラスタリング手法を簡単に説明する。

Ad-Hoc ネットワークに関する研究の多くはルーティング方式に関係するものである。ルーティング方式は二つに分類することができる。一つは定期的にメッセージ交換を行ない常にネットワークのトポロジを把握し、通信時にそのトポロジ情報に従ったルーティングを行なう方法である。もう一つは必要な通信の時だけブロードキャストにより経路情報を得る方法である。前者は経路情報維持に大きなトラフィックコストが発生するに対して、後者では通信時にブロードキャストによるトラフィックコストが生じる。周知の通り、モバイル端末には電源及び処理能力といった制限があり、トラフィックを低減させることができれば、モバイル端末の省電力及びスループットを高めることに貢献できる。

トラフィックを低減させるもう一つの手段としてクラスタリング手法がある。基本的なアイデアは移動オブジェクトからクラスタ代表 (cluster-head) の集合を選び、それぞれのクラスタ代表に

近接するクラスタ代表以外の移動オブジェクトをクラスタのメンバとしてクラスタを生成する。ルーティングの時は処理要求をユニキャストでクラスタ代表に委ねる。クラスタ間ではクラスタ代表間で直接情報交換を行なうか、あるいは二つのクラスタ間に介在するゲートウェイとして振舞う移動オブジェクトを介して行なう。

ここで用いられるクラスタリング手法は、大きく (1) Highest-Degree 法<sup>7)</sup>, (2) Lowest-ID 法<sup>4)</sup>, (3) Node-Weight 法<sup>5),6)</sup> の三種類に分けることができる。これらの違いは、クラスタ代表の選択アルゴリズムにある。Highest-Degree 法は最も近接している移動オブジェクトを多く持つ移動オブジェクトをクラスタ代表にし、近接移動オブジェクトをクラスタメンバとする。Lowest-ID 法は最も小さい ID 識別子を持つ移動オブジェクトをクラスタ代表とする。Node-Weight 法は移動オブジェクトの特徴を考慮して、大きな重みを持つ移動オブジェクトをクラスタ代表とする。ここで用いられる特徴量は、移動オブジェクトの速度や電力消費であってそれらの特徴を数値化して用いている。

しかしながら、これらの方法はどれもオブジェクトの移動ベクトルを考慮していない。例えばクラスタ代表を上の方法で選んだとしても、クラスタのメンバとの移動方向によっては、短時間で通信が途絶してしまう可能性がある。Node-Weight 法では速度を重みとして考えるが、単純に速度の値を重みにしているだけであって近接オブジェクトとの相対移動方向などは考慮していない。すなわち相対移動関係を考えずにクラスタを構成しても、移動の激しい環境においてはクラスタのトポロジを安定に維持することは難しい。

本稿では、移動ベクトルを考慮したクラスタ代表の選抜方法と、それを中心としたクラスタリング手法を提案している。従来の手法と大きく違う点は、隣接移動オブジェクトの相対移動を考慮したクラスタ代表の選び方にある。つまりクラスタ内のトポロジの相対的な安定を目指すことを目的としている。

## 3. 提案手法のポリシー

移動オブジェクトは互いの通信を介してグラフ構造のネットワークを構成するので、ここでは移動オブジェクトをノードと呼ぶことにする。安定したクラスタを得るためには、そのクラスタ内の

各ノードがなるべく相対的に静止した関係を保つ方が良い。言い換えればクラスタ内の各ノードがすべて同じ方向に同じ速度で移動していれば相対的にはお互い静止状態にあり、ノード間のトポロジ関係は長く保たれる。図1が示しているように、三つのクラスタ  $C_1 = \{A\}$ ,  $C_2 = \{B, C, D, F\}$ ,  $C_3 = \{E, G\}$  があり、クラスタ内のノードはほぼ同じ移動ベクトルを取っている。時間が経つにつれクラスタ間のリンクが短い期間で切れたとしても、クラスタ内のトポロジは相対的に静止しているのでその関係を長く維持できることが分かる。

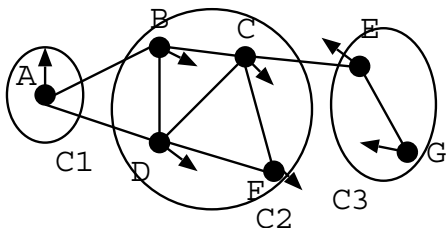


図1 移動ベクトルを考慮したクラスタリング

以上の考え方に基づき、クラスタ内のトポロジ関係を長く保つ方法でクラスタリングを行なえば、安定度の高いクラスタを得られると考えられる。現実的に広域な地域に大量のノードが分布する場合、それぞれのクラスタ内のノードがまったく静止することはありえない。よって目標としては、各クラスタの安定度ができるだけ高くなるようにクラスタ代表を選んでクラスタリングを行なう。

ここで使われる安定度  $\Upsilon$  をクラスタ代表の移動ベクトルと各クラスタメンバーの移動ベクトルの距離の分散を使って表す。

$$\Upsilon = \frac{1}{N} \sum_{i=1}^N |Sim(\vec{v}_h, \vec{v}_{m_i}) - \overline{Sim}_{avg}|^2 \quad (1)$$

ここで  $N$  はメンバーの数を表し、 $\vec{v}_h$  はクラスタ代表の移動ベクトルを表す。 $\vec{v}_{m_i}$  は、クラスタの各メンバーの移動ベクトルである。 $Sim(\vec{v}_h, \vec{v}_{m_i})$  は、クラスタ代表の移動ベクトルと各メンバーの移動ベクトルの類似度を表す。 $\overline{Sim}_{avg}$  は、各類似度の平均である。

$$\overline{Sim}_{avg} = \frac{1}{N} \sum_{i=1}^N Sim(\vec{v}_h, \vec{v}_{m_i}) \quad (2)$$

すなわちクラスタ代表の移動ベクトルとメンバーの移動ベクトル間の距離の分散が小さければ、そのクラスタはクラスタ代表と似た移動ベクトルを持つメンバーで構成されることを意味する。 $Sim(\vec{v}_h, \vec{v}_{m_i})$  は、式(3)を用いて求められる。

$$Sim(\vec{v}_h, \vec{v}_{m_i}) = \frac{1}{\sqrt{|\vec{v}_h - \vec{v}_{m_i}| + 1}} \quad (3)$$

この式では二つのベクトル間の距離が小さいほどベクトル間の類似度が高いことを意味する。二つのベクトルが一致する時その距離が0となり類似度が最大値1を取る。よって、クラスタ代表の移動ベクトルとメンバーの移動ベクトルが類似していればその距離が小さい。

我々の提案手法の基本は、クラスタ代表を中心にして、その移動ベクトルがメンバーとの移動ベクトルの距離による類似の分散が小さければ、そのクラスタ代表がより多くの類似したベクトルを持つメンバーを含むことにある。

#### 4. 移動ベクトルを考慮したクラスタリング

3章では、どのようなクラスタを生成するかについて述べた。クラスタのトポロジができるだけ安定するようにするには、移動ベクトルを考慮したクラスタ代表の選び方が重要である。これは安定度の求め方がクラスタ代表を基準にしているからである。

我々の提案手法は、クラスタのセットアップとその維持の二つのステップに分けられる。本章ではまずクラスタリングを行なう前準備を述べ、次にクラスタのセットアップ、クラスタの維持について順に説明を行なう。

##### 4.1 前準備

###### 4.1.1 ノードの初期状態

ノードの全集合を  $M = \{m_1, m_2, \dots, m_n\}$  とする。各ノード  $m_i$  は、それぞれ独自のIDで識別される。各ノードが位置情報  $p = (x, y)$ 、移動ベクトル  $\vec{v} = (v_x, v_y)$  を自身で知ることができるものとする。またノードの通信範囲の半径を  $T_{range}$  で表す。よって、各ノードの初期状態は、表1のようになる。

表1 各ノードの初期状態値

ID	$p = (x, y)$	$\vec{v} = (v_x, v_y)$	$T_{range}$	$S = 0$
----	--------------	------------------------	-------------	---------

表1の  $S$ (Status) は、整数0, 1, 2を取ることができる。それぞれ未定 (Undecided)、クラスタ代

表 (Clusterhead), クラスタメンバ (ClusterMember) の三つの状態に対応する。クラスタリングを行なう前には Undecided を示す 0 の状態であるとする。

#### 4.1.2 ノードの近接リスト

各ノードは近接ノードの一覧をテーブルとして持つ。近接ノードリストの構造は表 2 のようになる。

表 2 の LINK\_STATUS には, 0 あるいは 1 の値を取る。0 はノードがその近接ノードと双方向リンクを維持していることを意味し, 1 は片方向リンクを意味する。すなわちその近接ノードの存在を知ることができるが, 自分からその近接ノードまでのリンクは存在しない状態を表す。片方向リンクが存在する可能性にはいくつかの理由が考えられる。一つは互いの通信範囲が異なる場合である。他には, 周囲環境による通信障害などで片方向リンクになってしまう可能性がある。

各ノードの近接ノードリストは, 次のように構築される。

- 各ノードが自分の初期状態情報を含めた HelloMessage をブロードキャストする。
- HelloMessage を受け取ったノードは, 受信した相手の初期状態情報を自分の近接リストに追加する。LINK\_STATUS を 1 にセットする。同時に自分の初期状態情報に相手の ID 情報を付加して, ReplyMessage をブロードキャストする。
- 各ノードが自分の ID を含んだ ReplyMessage を受信してから, 始めて自分と相手ノードの間に双方向リンクがあると認識し, 近接リストの中の相手の LINK\_STATUS を 0 にセットする。

#### 4.1.3 ノードの近接クラスタリスト

各ノードは自分が属するクラスタ及び近接クラスタのリストを持つ。提案手法ではメンバノードの場合複数のクラスタに属することができる。この時, 二つ以上のクラスタに属するのであれば, そのノードはゲートウェイノードとしての役割を果たすことができる。近接クラスタリストは表 3 のようになる。number\_of\_NeighborCluster はノードが属するクラスタの数を示している。

#### 4.2 クラスタの構築

本節では, クラスタ代表を選んでクラスタの構築を行なうアルゴリズムを述べる。まず任意の一つのノード  $A$  を選びそれがクラスタ代表とし

表 3 各ノードの近接クラスタリスト

NEIGHBOR_CLUSTER_ID	counter
ClusterHead 1	number_of_NeighborCluster
...	
ClusterHead m	

てふさわしいかどうかを, 前述の式 (1) で求められる安定度  $\Upsilon$  で判定する。すなわち安定度が高い (すなわち, 分散値が小さい) 場合にそのノードをクラスタ代表として選定し,  $A$  の近接ノードをメンバノードとするようにクラスタを生成する。

#### 4.2.1 クラスタ生成アルゴリズム

クラスタを生成するためのアルゴリズムは次の通りである。ここですべてのノードの初期状態では  $S = 0$ , すなわち未定 (Undecided) 状態からスタートする。

- (1) 各ノード  $m_i \in M$  の近接ノードを求める (求め方は 4.1.2 節を参照)。  $N[m_i]$  とする。
- (2)  $m_i$  の移動ベクトルを  $v_{m_i}^{\rightarrow}$  とし, 近接ノード  $n_j \in N[m_i]$  の移動ベクトルを  $v_{n_j}^{\rightarrow}$  とする。このとき  $m_i$  の近接ノードテーブル情報を使って  $m_i$  の  $\Upsilon(m_i)$  を式 (4) で求める。

$$\Upsilon(m_i) = \frac{1}{N} \sum_{i=1}^N |Sim(v_{m_i}^{\rightarrow}, v_{n_j}^{\rightarrow}) - \overline{Sim_{avg}}|^2 \quad (4)$$

ここで  $N$  は  $N[m_i]$  の要素の数である。また  $Sim(v_{m_i}^{\rightarrow}, v_{n_j}^{\rightarrow})$  は  $m_i$  の移動ベクトルと各近接ノードの移動ベクトルの類似度を表す。 $\overline{Sim_{avg}}$  は各類似度の平均である。

$$\overline{Sim_{avg}} = \frac{1}{N} \sum_{i=1}^N Sim(v_{m_i}^{\rightarrow}, v_{n_j}^{\rightarrow}) \quad (5)$$

式 3 同様に,

$$Sim(v_{m_i}^{\rightarrow}, v_{n_j}^{\rightarrow}) = \frac{1}{\sqrt{|v_{m_i}^{\rightarrow} - v_{n_j}^{\rightarrow}| + 1}} \quad (6)$$

すべての  $m_i \in M$  に関して  $\Upsilon(m_i)$  を求める。このステップの目的は, クラスタ代表を選定するための  $\Upsilon(m_i)$  値を算出することである。小さい  $\Upsilon(m_i)$  値を持つのであれば, ノード  $m_i$  の移動ベクトルと類似する近接ノードが多いことを意味する。

- (3) 次に各ノードは, 自分の  $\Upsilon(m_i)$  値を含んだ CHC (ClusterHead Candidate) message をブ

表 2 近接ノードデータベース

NEIGHBOR_ID	LINK_STATUS	位置情報	移動ベクトル	通信範囲	ノード状態
neighbor 1	0 / 1	$p_1 = (x, y)$	$v_1^i = (v_x, v_y)$	$T_{range_1}$	$S_1 = 0$
neighbor 2	0 / 1	$p_2 = (x, y)$	$v_2^i = (v_x, v_y)$	$T_{range_2}$	$S_2 = 0$
...	...	...	...	...	...
neighbor n	0 / 1	$p_n = (x, y)$	$v_n^i = (v_x, v_y)$	$T_{range_n}$	$S_n = 0$

ロードキャストする。

- (4) 有限時間内にノード  $m_k \in M$  が自分の  $\Upsilon(m_k)$  より小さい  $\Upsilon(m_i)$  値を含んだ CHC message を受信しなければ、ノード  $m_k$  は、自分をクラスタ代表として宣言し、自分の状態を  $S = 1$ (クラスタ代表) とセットする。さらに近接ノードに向け、 $CH(m_k)$ (クラスタ代表) message をブロードキャストする。一方、近接ノードは  $CH(m_k)$  message を受信すると自分の状態を  $S = 2$ (ClusterMember) にセットする。以降はメンバノードとして振舞うことになる。結果として  $m_k$  をクラスタ代表としてその近接ノードをメンバとするクラスタが得られる。
- (5) すべてのノードがクラスタ代表か、メンバノードかのどちらに属する状態に達した時点でクラスタの生成が完了する。

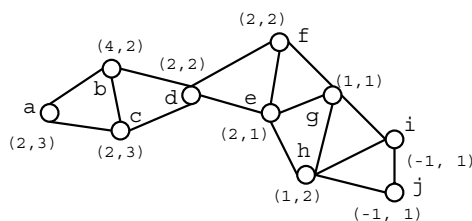


図 2 移動ベクトルを持つノードリンク

表 4 各ノードの Can 値

NODE_ID	移動ベクトル	$\Upsilon$
a	(2, 3)	0.0897
b	(4, 2)	4.0308
c	(2, 3)	0.0687
d	(2, 2)	0.0536
e	(2, 1)	3.5014
f	(2, 2)	0.0607
g	(1, 1)	0.0012
h	(1, 2)	0.0017
i	(-1, 1)	0.0780
j	(-1, 1)	0.0897

#### 4.2.2 生成されたクラスタの性質

生成されたクラスタには次のような性質がある。

- 任意の二つのクラスタ代表は近接しない。あるノード A がクラスタ代表になった時点で、その近接ノードはメンバノードになる。よって二つのクラスタ代表が近接することはない。
- クラスタ間のオーバーラップを許す。例えば、あるノード B がクラスタ  $C_1$  のメンバであるとする。もし B の近接に他のノード C がクラスタ代表(クラスタ  $C_1$  のクラスタ代表と近接しない)であると宣言したとすると、B は同時に C をクラスタ代表とするクラスタに属することができる。

#### 4.2.3 セットアップの例

図 2 のようなトポロジ構成のノードを用いて、クラスタリングのセットアップの例を示す。各ノードの (Integer, Integer) は、移動ベクトルを示す。

よって、ノード  $\{a, b, \dots, j\}$  の  $\Upsilon$  値は式 (4), (6) を用いて、計算を行なう、表 4 になる。

表 4 から、ノード g の  $\Upsilon(g) = 0.0012$  が最も小さいので、g がクラスタ代表となり、 $\{e, f, h, i\}$

をメンバノードとしてクラスタ  $C(g)$  を形成する。e, f, h, i は、メンバノードになったので、その  $\Upsilon$  値を考慮から除外する。次にノード d の  $\Upsilon(d) = 0.0536$  が小さいので、d がクラスタ代表となり、メンバノード  $\{b, c, e, f\}$  を持つクラスタ  $C(d)$  を形成する。同様な手順で、クラスタ  $C(a) = \{b, c\}$ ,  $C(j) = \{h, i\}$  を得る。結果は、図 3 に示される。図 3 の中に点線で囲まれた部分に対応するノードは、ゲートウェイノードとなる。

よって、図 2 のようなリンク構造を成すノードは、図 4 のようなクラスタに分けることができる。

#### 4.3 クラスタの維持

クラスタの生成が完了するとクラスタの維持過程に入る。各ノードの位置及び移動ベクトルが随時に変化する、新しいノードが領域内に現れる、リンク切れにより消えるなど、全体としてのネットワーク構成は常に変化している。よってクラスタを効率よく維持することが重要である。クラスタの維持は大きく新しいノードが現れた時の処理と既存クラスタの維持に分けるこ

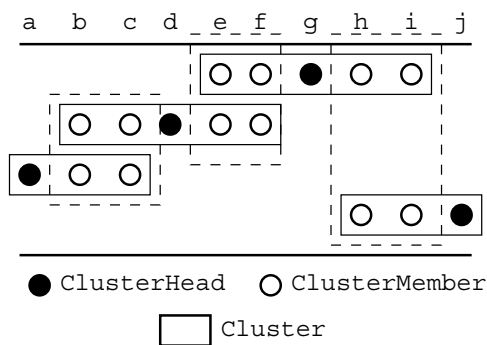


図3 クラスタの形成

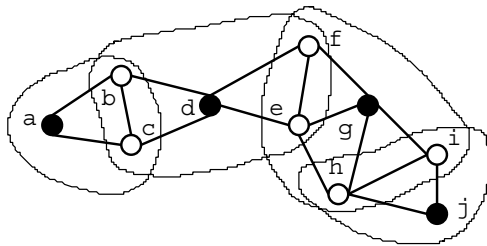


図4 クラスタリングの結果

とができる。

#### 4.3.1 新しいノードの加入

新しいノードが領域内に現れたら次の手順で最適なクラスタを探す。

- $m_{new}$  は、まず自分の近接ノードを求める。
- 近接ノードの中にクラスタ代表があるかどうかを確認する。クラスタ代表があれば、そのクラスタ代表のクラスタにメンバノードとして加入する。その時、クラスタ代表との移動ベクトルの距離が大きくても、加入しなければならない。そうでなければ4.2.2節のクラスタの性質から外れてしまう。もし近傍ノードの中に複数のクラスタ代表が存在する場合、その中から最も移動ベクトルの距離が小さいクラスタ代表を選んでそのクラスタに加入する。
- 近接ノードの中にクラスタ代表がなければ、 $m_{new}$  は、自分でクラスタ代表を宣言し、その近接ノードをメンバノードとして、新しいクラスタを作る。

#### 4.3.2 既存クラスタの維持

既存クラスタの維持は新しいノードの加入とは状況異なる。クラスタ代表およびメンバは互いの状態を定期的なブロードキャストメッセージで確認しているものとする。あるメンバがク

ラスタ代表と通信が切断された場合、クラスタ代表はそのメンバがクラスタから離脱したと認識し、自分が管理しているメンバ情報更新する。一方クラスタから離脱したノードは、別の入るべきクラスタを探す。その時、前節の新しいノードの加入と同じ手順をとる。

まだ長い時間が経つと、クラスタ内においてはクラスタ代表がそのクラスタを代表するのにふさわしくなくなってしまう可能性がある。よってクラスタ内で再計算を行ない、クラスタ代表を再選択する。事前にある閾値  $T$  を設定して、もし  $\Upsilon(\text{clusterhead}) > T$  となったとき、そのクラスタはクラスタ代表と類似度の低いノードで形成していると判断し、そのクラスタを再クラスタリングする。

- クラスタ代表とそのメンバノードは自分の状態を未定にする。ここでもしそのメンバノードが他のクラスタにも属している場合にはそのメンバノードは ClusterMember のままである。
- 次にそのノードは4.2.1節の式(4)、(6)を用いて、それぞれの  $\Upsilon$  値を求める。
- その中から、小さい  $\Upsilon$  値を持つノードが新しくクラスタ代表となって新しいクラスタを形成する。この新しいクラスタに入れないノードは、次々と前節の新しいノードの加入と同じ手順をとり、入るべきクラスタを探す。

## 5. まとめ

本稿では、移動オブジェクトの移動ベクトルを考慮したクラスタリング手法の提案を行なった。我々はトポロジが激しく変化する移動オブジェクトの無線ネットワーク環境においてトポロジをできるだけ安定に維持するため、移動オブジェクトの移動ベクトルに着目したクラスタリングアルゴリズムを考案し、移動ベクトルに基づくクラスタ代表の選び方、クラスタ形成のアルゴリズムを記述した。

今後の課題は以下の通りである。提案手法ではクラスタ代表を中心にその近接移動オブジェクトをメンバとしてクラスタを形成する。そのため、直径  $1\text{-hop}$  のクラスタしか生成できない。理想的な状況としては、互いに類似した移動ベクトルを持つ移動オブジェクトがあれば、数 hop であっても、大きなサイズのクラスタにまとめた方がいいかも知れない。また本質的に、提案

した手法は Ad-Hoc ネットワークにおけるクラスタリングアルゴリズムと類似しているため、これらの手法との評価実験を行なう予定である。

謝辞 本研究の一部は、科学技術振興機構 (JST) の戦略的基礎研究推進事業 (CREST) 「横田クレスト」プログラムの支援によるものである。ここに記して謝意を表す。

#### 参 考 文 献

- 1) Mineno, H., Aono, M., Ohta, K., Ohta, K. and Mizuno, T.: Multiple paths protocol for a cluster type network, *International Journal of Communication Systems*, Vol. 12, pp. 391 – 403 (1999).
- 2) Wakikawa, R., Uehara, K. and Murai, J.: Multiple Network Interfaces Support by Policy-Based Routing on Mobile IPv6, *proc. of ICWN 2002*, pp. 9 – 17 (2002).
- 3) Jiang, M., Li, J. and Tay, Y.: Cluster based routing protocol (CBRP) functional specification, *Internet Draft, draft-ietf-manet-cbrp-sepc-00.txt* (1998).
- 4) Ephremides, A., Wieselthier, J. and Baker, D.: A design concept for reliable mobile radio networks with frequency hopping signaling, *Proceedings of the IEEE*, Vol. 75, No. 1, pp. 56 – 73 (1987).
- 5) Basagni, S.: Distributed Clustering for Ad Hoc Networks, *Proceeding of International Symposium on Paralled Architectures, Algorithms and Networks*, pp. 310 – 315 (1999).
- 6) Chatterjee, M., Das, S. K. and Turgut, D.: WCA: A Weighted Clustering Algorithm for Mobile Ad hoc Networks, *Journal of Cluster Computing (Special Issue on Mobile Ad hoc Networks)*, Vol. 5, pp. 193 – 204 (2002).
- 7) Parekh, A.: Selecting Routers in Ad-Hoc Wireless Networks, *In Proceedings of the SBT/IEEE International Telecommunications Symposium* (1994).