

## 無線センサネットワークにおける 省電力クラスタリング方式の提案と評価

豊田慎之介<sup>†1</sup> 佐藤文明<sup>†2</sup>

現在、無線技術を応用した無線センサネットワークが注目されている。センサネットワークにおいて、省電力を考慮したルーティング方式は重要であり、盛んに研究されている。LEACH は、センサネットワークに適した代表的なクラスタリング方式である。しかし、LEACH ではノードの消費電力が偏りやすいという問題がある。HEED はノードの電力の情報を利用することで、LEACH を改良したクラスタリングアルゴリズムである。しかし、HEED は隣接ノードの数を考慮しないため、クラスタヘッドの出現位置が非効率になりやすい。HIT はクラスタ内でマルチホップ通信を用いることにより、通信距離を抑えることで、省電力化を実現している。しかし、HIT ではノードの電力を考慮していないため、消費電力が偏りやすい。本論文では、クラスタヘッドを選ぶ際に隣接ノード集合の重複度と残存電力を評価することで、ノードの残存電力の偏りを抑えたクラスタリング方式を提案する。また、シミュレーションによりノードの電力と領域カバー率の点で、従来方式との比較を行い、提案方式の有効性を検証する。

### Energy-Effective Clustering Algorithm in Wireless Sensor Networks

SHINNOSUKE TOYODA<sup>†1</sup> FUMIAKI SATO<sup>†2</sup>

Nowadays, there has been a growing interest in wireless sensor networks (WSN). One of the major issues in WSN is developing an energy-efficient routing protocol. LEACH is very energy-efficient routing protocol based on the clustering of the sensor nodes. However, energy consumption of nodes tends to become unbalanced in LEACH. HEED improves the LEACH clustering algorithm by using information of residual electric power of nodes. Although HEED provides better performance than LEACH, it does not consider the number of adjacent nodes. Therefore, the cluster head does not efficiently cover the nodes in HEED. HIT is based on the small transmission range and multi-hop communication. Though HIT has improved the performance dramatically, unbalance of the electric power consumption ins remained. In this paper, we propose energy-effective clustering algorithm considering adjacent nodes and residual electric power. In addition, we inspect effectiveness of our method by comparing our method with the traditional method by the simulation.

#### 1. はじめに

無線センサネットワークは搭載するセンサの種類や、ネットワークの規模などにより幅広い応用が可能で、様々な分野から期待されている。センサ端末の小型化・低価格化、更にはエネルギーを有効に活用するスマートグリッドへの関心の高まりから、近年特に注目が集まっている。

一方で、センサ端末の電源は主に電池に依存するため、有限である。さらに、外部から電源を供給することも困難な場合が多く、電池を交換するのにもコストがかかってしまう。このため、特に大規模な無線センサネットワークの運用には、省電力に考慮したルーティング方式は重要であり、盛んに研究が行われている[1][2][3][4][5]。

センサネットワークの代表的なルーティング方式に LEACH[1]がある。LEACH はクラスタリングを用いたルーティング方式であるが、ノードがクラスタヘッドになる際に残存電力が考慮されていないため、負荷が偏りやすいという問題がある。この問題を解消するために、HEED[2]が提案されている。HEED はノードの消費電力の情報を利用

することで、消費電力の偏りが改善されている。しかし、LEACH, HEED とともにマルチホップ通信でないため、データ送信の消費電力が大きく、ネットワーク全体の寿命が短くなってしまいやすい。マルチホップ通信を用いた HIT[3]が提案されている。しかし、HIT はノードの電力の情報が考慮されていないため、LEACH 同様に消費電力の偏りが生じやすい。

そこで、本稿では隣接ノード集合と残存電力を考慮した省電力ルーティング方式を提案する。提案手法では、隣接ノード集合と残存電力の情報から各ノードに対し評価値を計算し、評価値の高いノードを中継ノードとして採用することで、省電力化、及び領域カバー率の長時間維持を実現した。また、提案手法ではスリープを導入することで、領域カバー率を大きく損なうことなく、センサネットワークの長寿命化を図った。本研究では PEGASIS 等で用いられている位置情報は用いていない。また、RPL(IPv6 Routing Protocol for Low Power and Lossy Networks)[6]等で用いられているようなリンクの信頼度の情報は用いていない。また、本稿では提案手法の有効性をシミュレーションにより評価する。

<sup>†1</sup> 東邦大学大学院理学研究科  
Toho University, Graduate School of Science

<sup>†2</sup> 東邦大学理学部情報科学科  
Department of Information Science, Faculty of Science, Toho University

## 2. 従来研究

### 2.1 LEACH

センサネットワークに適した代表的なルーティングプロトコルとして LEACH(Low-Energy Adaptive Clustering Hierarchy) [1]が挙げられる。LEACH はクラスタリングを用いたルーティング方式で、ノードからクラスタヘッド、クラスタヘッドからシンクまで、それぞれ1ホップでデータを送信する。

LEACH の最大の特徴は、負荷の集中するクラスタヘッドを一定期間ごとに交代することである。これにより、負荷を分散させ、ネットワーク全体の寿命をのばすことができる。ここで、各ノードからシンクへデータを1回送信する周期をサイクル、クラスタヘッドを交代してから次のクラスタヘッドを交代するまでのサイクル周期をラウンドと呼ぶ。

LEACH では、前提として、すべてのノードがシンクと通信可能であり、各ノードが1ラウンド目にクラスタヘッドに立候補する基本確率  $P$  を予め知っている必要がある。ラウンドが変わると、ノード  $n$  は自身がクラスタヘッドに立候補するかどうかを判定する。このとき、 $0 \sim 1$  の乱数を作り、以下の式(2.1)の計算結果がその数以下であった場合、 $n$  はクラスタヘッドに立候補する。

$$T(n) = \begin{cases} \frac{P}{1 - P * (r \bmod \frac{1}{P})} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases} \quad (2.1)$$

ここで、 $r$  は現在のラウンド数、 $G$  は過去  $1/P$  ラウンド以内にクラスタヘッドになっていないノードの集合を表す。つまり、すべてのノードは  $1/P$  ラウンド以内にクラスタヘッドに立候補することになる。クラスタヘッドに立候補したノードは自分の周囲のノードにクラスタヘッド広告をブロードキャストし、クラスタヘッドに立候補したことを知らせる。クラスタヘッドに立候補しなかったノード（非クラスタヘッドノード）は、一定時間クラスタヘッド広告の受信待ちを行う。クラスタヘッド広告を受け取ったノードはクラスタヘッド広告を受信したノードのリストに送信してきたノードを加え、受信電波強度を記憶する。受信待ちを終えると、非クラスタヘッドノードは、リストの中から受信電波強度の最も強かったクラスタヘッドを選び、参加要求を送信する。クラスタヘッド広告を受信したノードが1つもなかった場合は、クラスタには所属せずに、直接シンクへデータを送信する。一方、クラスタヘッド広告を送信し終えたノードは、参加要求待ちを行う。すべての参加要求を受信したら、クラスタヘッドはクラスタメンバの TDMA 送信スケジュールを作成し、メンバに送信する。メンバは、送信スケジュールを受け取ると、クラスタヘッドが交代するまで自分の送信順を記憶しておく。ここまでの

処理をラウンドが変わるごとに実行する。

スケジュールが決まったあとは、メンバはスケジュール通りの順番にセンサデータをクラスタヘッドに送信し、クラスタヘッドはすべてのメンバからデータを受け取ってからシンクへ圧縮したデータを送信する。メンバからクラスタヘッドへのデータ送信時の送信電力は、相手に到達可能な必要最小限の電力を用いる。クラスタヘッドからシンクへの通信も同様である。これが LEACH における1ラウンド1サイクルの流れである。

LEACH には問題点もいくつか存在する。代表的な問題点として、ノードの電力消費が偏りやすいというものがある。これは、ノードがクラスタヘッドに立候補するかどうかを単純に確率とクラスタヘッドになった回数で判定しているため、最終的にどのノードも同じ頻度でクラスタヘッドになっていることが原因である。データ送信時の消費電力は送信距離に依存するため、シンクから離れたノードと近くにあるノードで同じ回数送信を行うと、離れたノードの電力のほうが早く枯渇してしまうのである。また、クラスタメンバを一切持たないクラスタヘッドが出現したり、どのノードもクラスタヘッドに立候補しないラウンドが存在したりするなど無駄が多いことも原因であると考えられる。

### 2.2 HEED

LEACH を改良したクラスタリング方式も提案されている。その一つに HEED(Hybrid, Energy-Efficient Distributed clustering)[2]がある。HEED では、各ノードがクラスタヘッドに立候補する確率を、初期電力  $E_{max}$  とその時点の残存電力  $E_{residual}$  の比とすることで、より残存電力の大きいノードがクラスタヘッドに立候補しやすくしている。また、HEED ではクラスタヘッドの状態に、暫定クラスタヘッド状態を示す *tentative\_CH* とクラスタヘッド決定状態を示す *final\_CH* の二種類を設定している。*final\_CH* を知らせるメッセージをブロードキャストしたノードはそのラウンドでクラスタヘッドを担うことが決定づけられる。一方、*tentative\_CH* をブロードキャストしたノードは、後から自分よりも全体の通信コストを低くするノードがクラスタヘッド立候補をブロードキャストしてきた場合に、自身の立候補を破棄して他のクラスタへ所属することができるのである。

HEED では、ノードがクラスタヘッドに立候補する確率  $CH_{prob}$  を以下の式(2.2)で計算する。

$$CH_{prob} = \max \left( C_{prob} * \frac{E_{residual}}{E_{max}}, p_{min} \right) \quad (2.2)$$

ここで、 $C_{prob}$  は予め定められたクラスタヘッドの割合、 $E_{residual}$  をノードの残存電力、 $E_{max}$  をノードの初期電力、 $p_{min}$  は  $E_{max}$  に反比例して決められる  $CH_{prob}$  の最小値をそれぞれ表している。

各ノードは式(2.2)で  $CH_{prob}$  を計算した後に、以下を繰り返す。

- 自身のものを含め、クラスタヘッド立候補を1つ以上受信している場合
  - 通信コストが最小になるノードを自身のクラスタヘッドに選ぶ
  - 選んだノードが自分自身であった場合
    - ◇  $CH_{prob}$  が1のとき
      - $final\_CH$  メッセージをブロードキャストする
    - ◇  $CH_{prob}$  が1未満のとき
      - $tentative\_CH$  メッセージをブロードキャストする
- クラスタヘッド立候補メッセージを1つも受信していない場合
  - $CH_{prob}$  が1のとき
    - ◇  $final\_CH$  メッセージをブロードキャストする
  - $CH_{prob}$  が1未満のとき
    - ◇  $0 \sim 1$  の乱数  $\leq CH_{prob}$  のとき
      - $tentative\_CH$  メッセージをブロードキャストする
- $CH_{prob}$  が1のとき
  - 繰り返しを終了する
- $CH_{prob}$  が1未満のとき
  - $CH_{prob} \leftarrow \min(CH_{prob} * 2, 1)$
  - 最初から繰り返す

繰り返しの中でノードが  $final\_CH$  をブロードキャストしなかった場合、クラスタヘッド立候補メッセージを受信したノードから通信コストが最小のノードを自身のクラスタヘッドに選び、参加要求を送信する。また、クラスタヘッド立候補メッセージを受信したノードがなかった場合、 $final\_CH$  メッセージをブロードキャストし、クラスタヘッドとなる。

### 2.3 HIT

クラスタ内でマルチホップ通信を用いることで電波干渉を抑え、できるだけ多くの通信を同時並行的に行う HIT (Hybrid Indirect Transmissions) [3] がある。HIT は以下の7つのフェーズから成る。

#### (1) クラスタヘッド選出フェーズ

このフェーズでは、1つあるいは複数のクラスタヘッドが選出される。各クラスタはクラスタヘッドを1つ持つ。クラスタヘッドが1つの場合、ノードIDの順序によってクラスタヘッドの役割をローテーションさせる。

#### (2) クラスタヘッド広告フェーズ

このフェーズでは、クラスタヘッドが自身のノード情報

をクラスタヘッド広告としてブロードキャストする。クラスタヘッドでないノード  $i$  がクラスタヘッド広告を受信すると、クラスタヘッドとの距離を計算する。そして、自身に最も近いクラスタに所属する。 $i$  は所属するクラスタヘッド  $H$  までの距離を  $d(H, i)$  として保持する。

#### (3) クラスタセットアップフェーズ

このフェーズでは、クラスタが形成され、上流、下流の関係が生まれる。まず、クラスタヘッドでないノード  $i$  は Member メッセージをブロードキャストする。Member メッセージには  $i$  のクラスタヘッドのノードID、及び  $i$  からクラスタヘッドまでの距離が格納される。Member メッセージを交換することにより、各ノード  $i$  は他のすべてのノード  $j$  について、 $i$  から  $j$  までの距離  $d(i, j)$ 、及び  $j$  からクラスタヘッド  $H$  までの距離  $d(j, H)$  を保持する。これらの情報を用いて、ノード  $i$  は、以下の2つの条件を満たすノードのうち、 $i$  に最も近いノードを  $i$  の上流ノード  $u_i$  として選出する。

- $d(u_i, H) < d(i, H)$
- $d(i, u_i) < d(u_i, H)$

これらの条件は、 $u_i$  が  $i$  よりもクラスタヘッドに近く、また  $i$  がクラスタヘッドと直接通信するよりも  $u_i$  と通信するほうが、コストが小さいことを示している。

#### (4) ルートセットアップフェーズ

各ノードがフェーズ3で上流ノードを決定した後、すべてのノードは、上流ノードまでの推定距離を含んだ Upstream メッセージをブロードキャストする。これにより、すべてのノードが、他のすべてのノード  $j$  の上流ノード  $u_j$ 、及びノード  $j$  から  $j$  の上流ノード  $u_j$  までの距離  $d(j, u_j)$  をそれぞれ知ることが可能になる。これらの情報を元に、各ノードはノード  $j$  の下流ノード集合  $DOWN(j)$  を計算することができる。

#### (5) ブロッキングセット作成フェーズ

このフェーズでは、各ノードはフェーズ4で収集した情報を利用し、下流ノードのためのブロッキングセットを作成する。ノード  $i$  は必要十分条件、

- $d(i, u_i) > d(i, u_j)$

を満たすノード  $j$  をブロックする。このフェーズの最後に、すべてのノードは、ブロックした下流ノードのリストをブロードキャストする。

#### (6) TDMA スケジュール作成フェーズ

このフェーズでは、各ノードは衝突を回避しつつ、同時通信数が最大になるように TDMA スケジュールを作成する。

#### (7) データ伝送フェーズ

このフェーズでは、各ノードは環境のセンシングを行い、センシングしたデータを TDMA スケジュールに従い、上流ノードに転送する。

## 2.4 その他の従来研究

この他にも、クラスタではなく最も近いノード同士を結ぶチェーンを構成する PEGASIS[4], クラスタ内でこのチェーンを構成する TPC[5]など, センサネットワークに適した様々な省電力ルーティングプロトコルが提案されている。これらは, 位置情報を用いているため, 本研究の比較対象としていない。

## 3. 提案手法

本稿では, 隣接ノード集合と残存電力を考慮したクラスタリング方式を提案する。提案手法では, クラスタヘッドを選出する際に, 隣接ノード集合と残存電力の情報から各ノードに対し評価値を計算し, 評価値の高いノードを中継ノードとして採用することで, 省電力化, 及び領域カバー率の長時間維持を図った。ここで, 領域カバー率とは, センサが観測領域をどの程度カバーしているかの割合のことである。また, 提案手法では, ノード密度が高い場合, データ収集を休止(スリープ)することで, 領域カバー率を大きく損なうことなく, センサネットワークの長時間稼働を図った。以下, 詳細なアルゴリズムについて述べる。

### 3.1 Hello メッセージの交換

ラウンドの開始時に, シンクは全ノードに対して Hello メッセージ交換要求をブロードキャストする。このとき, シンクはこのラウンドで各ノードが用いる電波到達距離  $R$  を指定する。はじめは狭い電波到達距離を利用し, 通信の電力消費を抑え, クラスタからあふれてしまうノードが発生すると, 段階的に電波到達距離を広げていく。

各ノードはこの Hello メッセージ交換要求を受信すると 1 回目の Hello メッセージをブロードキャストする。1 回目の Hello メッセージはシンクから指定された電波到達距離  $R$  とは関係なく, 領域全体にブロードキャストする。このメッセージには送信ノードの残存電力が含まれる。すべてのノードが 1 回目の Hello メッセージを送信し終わると, 各ノードは以下の情報を持つ。

- 各ノードの電力
- 隣接ノード集合

ここで, 隣接ノード集合は, Hello メッセージを受信したノードのうち, 距離が  $R$  以下のノードの集合である。

すべてのノードが 1 回目の Hello メッセージを送信し終わると, 各ノードは 2 回目の Hello メッセージを  $R$  内のノードにブロードキャストする。このときの送信電力は  $R$  内に到達可能な最小限の電力とする。2 回目の Hello メッセージには以下の情報が含まれる。

- 送信ノードの残存電力
- 送信ノードの隣接ノード集合

すべてのノードが 2 回目の Hello メッセージを送信し終わると, 各ノードは以下の情報を持つ。

- 各ノードの電力

- 隣接ノード集合
  - 各隣接ノードの隣接ノード集合
- 2 ホップ隣接ノード集合
- 未到達ノード集合

ここで, 2 ホップ隣接ノード集合とは, 1 ホップでは到達できないが 2 ホップで到達可能なノードの集合である。また, 未到達ノード集合とは, 2 ホップ隣接ノード集合に含まれるノードのうち, 隣接するクラスタヘッドを経由して 3 ホップ以内に到達不能なノードの集合である。

次に, 各ノードは代表ノードの選出を行う。残存電力が最も大きいノードが代表ノードに選出される。ここで, すべてのノードは, 各ノードの電力の情報を共有しているため, すべてのノード間で同じ代表ノードが選出される。

代表ノードが決定すると, 代表ノードはクラスタヘッドの選出を開始する。

### 3.2 クラスタヘッドの選出

#### (1) クラスタヘッドの選出

提案手法におけるクラスタヘッドの選出方法は, 高密度アドホックネットワークにおける中継用ランドマークノードの選出方法[7]を参考にしている。

クラスタヘッド  $h$  は, 隣接ノード集合に含まれるすべてのノードに対して, 評価値  $v$  を計算する。ここで,  $h$  の隣接ノード  $n$  について, 評価値  $v_n$  は,  $h$  の隣接ノード集合と  $n$  の隣接ノード集合間のノード重複個数  $c_n$ ,  $n$  の残存電力  $e_n$ , 及び  $h$  のすべての隣接ノードの残存電力の平均値  $e_{ave}$  を用いて, 以下の式(3.1)で求められる。

$$v_n = c_n * \left( \frac{e_n}{e_{ave}} \right)^W \quad (3.1)$$

なお,  $W$  は残存電力の重みを示す定数である。つまり,  $h$  の未到達ノード集合と  $n$  の隣接ノード集合間のノード重複個数が多く, 残存電力が大きく, さらに, センシング範囲により多くのノードを持っているノードほど評価値が高くなる。 $h$  は評価値が最も大きかったノード  $n1$  を次のクラスタヘッドに選ぶ。このとき,  $h$  の未到達ノード集合から,  $n1$  の隣接ノード集合に含まれるノードを削除する。

さらに,  $h$  の 2 ホップ隣接集合のノード数に対する,  $h$  の 2 ホップ隣接集合と  $n1$  の隣接ノード集合間のノードの重複数の割合(以後 2 ホップカバー率という)がしきい値以上でなければ, さらに次のクラスタヘッドを選出する。 $h$  は  $n1$  を除いたすべての隣接ノード  $n$  の評価値  $v_n$  を再計算し, 最も評価値が高かったノード  $n2$  を次のクラスタヘッドとして選出する。その後, 再度 2 ホップカバー率を計算し, しきい値を超えていれば, 同様の処理を繰り返し, 2 ホップカバー率のしきい値を超えるまでクラスタヘッドを選ぶ。以下, クラスタヘッドが新たに選出したクラスタヘッドを子クラスタヘッド, 子クラスタヘッドを選出したクラスタヘッドを親クラスタヘッドと呼ぶ。

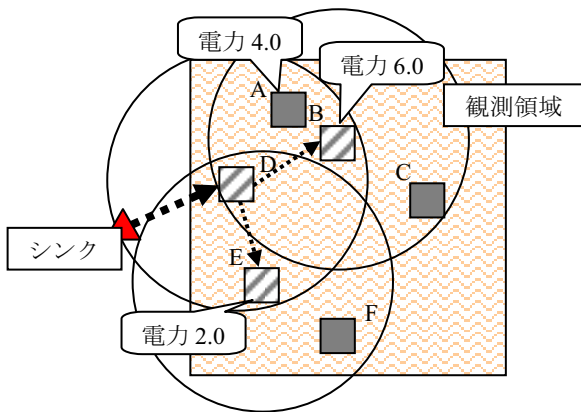


図 1 クラスタヘッド選出の様子

図 1 は、2 ホップカバー率のしきい値が 0.7、電力の重み  $W$  が 1 のときに、代表ノード D がクラスタヘッドを選ぶ様子を示している。

代表ノード D の隣接ノードリストに含まれているのは {ノード A, B, E} である。まず D は、{A, B, E} について、式(3.1)に従って評価値を計算する。ここで、隣接ノードの電力の平均値  $e_{ave}=4.0$  となる。よって、 $v_A=0$ ,  $v_B=1.5$ ,  $v_E=0.5$  となり、最大の評価値を持つノードは B である。従って、D は B を最初のクラスタヘッドに選出する。

次に代表ノード D は、2 ホップカバー率を求める。D の 2 ホップ隣接リストに含まれているのは、{ノード C, F} の 2 つである。このうち B の隣接ノードとの重複ノードは {C} のみなので、2 ホップカバー率は  $1/2=0.5$  となり、しきい値の 0.7 を下回っているので、D はさらにクラスタヘッドの選出を行う。ここで、D は既にクラスタヘッドに選んだ B を除く隣接ノードに対して評価値を再計算する。すると、 $e_{ave}=3.0$ ,  $v_A=0$ ,  $v_E=0.66$  となる。よって、D は E を次のクラスタヘッドに選ぶ。これによってノード F が 2 ホップで到達可能になるため、2 ホップカバー率は 1 となりしきい値の 0.7 を超えるので、D はクラスタヘッドの選出をここで終了する。

### (2) クラスタヘッド広告のブロードキャスト

クラスタヘッド  $h$  は 2 ホップカバー率のしきい値を超えるまで新たなクラスタヘッドを選出し終わると、クラスタヘッド広告を電波到達範囲内のノードにブロードキャストする。クラスタヘッド広告には以下の情報が含まれる。

- $h$  が新たに選出したクラスタヘッドのリスト
- 3 ホップチェック要求リスト
- 3 ホップチェック応答リスト

ここで、3 ホップチェック要求リストとは、クラスタヘッド広告を送信する段階での、 $h$  の未到達ノード集合である。3 ホップチェック応答リストとは、 $h$  の親クラスタヘッドの 3 ホップチェック要求リストに含まれるノードのうち、 $h$  が新たに選んだクラスタヘッドを含む、 $h$  の隣接クラスタヘッドを経由して  $h$  から 2 ホップで到達可能なノードのリストである。

$h$  のクラスタヘッド広告を受信した  $h$  の隣接ノード  $n$  は、自身の隣接するクラスタヘッドの情報を保持するリストである、隣接クラスタヘッドリストに  $h$  を加える。また、 $n$  がクラスタヘッドに選ばれていた場合、 $n$  は  $h$  を親クラスタヘッドとしてクラスタヘッドになり、(1)の処理に従ってクラスタヘッドの選出を行う。また、 $n$  がすでにクラスタヘッドであった場合、3 ホップチェック応答リストを送信ノードに送信する。

### (3) 3 ホップチェック応答の受信処理

クラスタヘッド  $h$  は、3 ホップチェック要求リストを送信後一定時間、3 ホップチェック応答の受信待ちを行う。この期間中に 3 ホップチェック応答を受け取ると、 $h$  は未到達ノード集合から 3 ホップチェック応答リストに含まれているノードを削除する。受信待ちが終了しても未到達ノード集合が空になっていなかった場合、 $h$  の隣接ノード集合に含まれるノードの評価値を式(3.1)に基づいて再計算する。その中で評価値が最も高いノード  $n$  を、新たにクラスタヘッドに選び、 $n$  に対してクラスタヘッド要請メッセージと共に 3 ホップチェック要求リストを送信する。

$n$  はこれを受け取ると、(1)の手順でクラスタヘッドの選出を行い、 $h$  に対する 3 ホップチェック応答リストを含めたクラスタヘッド広告をブロードキャストする。これを受け取ってもまだ未到達ノード集合が空にならない場合は、未到達ノード集合が空になるまで、これらの処理を繰り返す。

### (4) 終了報告の送信

クラスタヘッド  $h$  は、未到達ノード集合が空になり、新たにクラスタヘッドを選出できなかった場合、親クラスタヘッドにクラスタヘッド選出の終了報告を送信する。 $h$  の親クラスタヘッドは、 $h$  を含むすべての子クラスタヘッドから終了報告を受け、未到達ノード集合も空になったら、さらに親クラスタヘッドへ終了報告を送信する。このようにして、終了報告を転送していき、シンクが終了報告を受信すると、ネットワークのすべてのノードがいずれかのクラスタヘッドに所属できることになる。よって、これをもってクラスタヘッドの選出を終了する。

クラスタヘッドの選出が終了すると、シンクはクラスタ所属要求をすべてのノードに対してブロードキャストする。これを受信したノードは、隣接するクラスタヘッドのうち、クラスタヘッド広告の受信電波強度が最も強かったクラスタヘッドを所属するクラスタヘッドとして、参加要求を送信する。これにより、すべてのノードがいずれかのクラスタヘッドへ所属することになる。このとき、クラスタに所属できないノードがあった場合、電波到達範囲を広げ、再度クラスタリングの作業を行う。

クラスタヘッドは、クラスタ所属要求を受信してから一定時間後に、クラスタメンバのデータ送信スケジュールを作成し、クラスタメンバにブロードキャストする。クラス

タメンバはスケジュールを受け取ると、ラウンドが変わるまで自身の送信順を保持する。

提案手法の経路構築におけるメッセージのやり取りの様子を図 2 に示す。なお、CH<sub>1</sub> は代表ノード、CH<sub>2</sub> は CH<sub>1</sub> が選んだクラスタヘッド、CH<sub>3</sub> は CH<sub>2</sub> が選んだクラスタヘッドである。また、CH<sub>X</sub> は CH<sub>1</sub> が他クラスタヘッドからの 3 ホップチェック応答を受け、未到達ノードをカバーするために新たに選出したクラスタヘッドである。CH<sub>Y</sub> は CH<sub>X</sub> が選んだクラスタヘッドである。

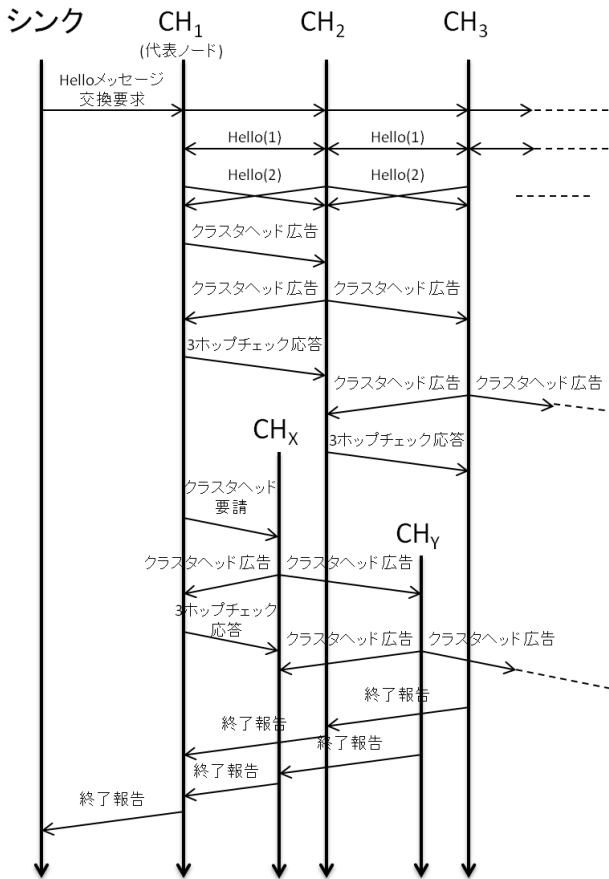


図 2 経路構築におけるメッセージのやり取りの様子

### 3.3 センサデータの収集

センサデータを収集するタイミングは、シンクがデータの送信要求をブロードキャストすることで各ノードに伝える。

クラスタヘッドでないノードは、このメッセージを受け取ると、送信スケジュールに従いクラスタヘッドにセンシングデータを送信する。

一方、クラスタヘッドは、すべてのクラスタメンバ及び子クラスタヘッドからセンシングデータを受け取った後に、それらと自身のデータを圧縮して、親クラスタヘッドへ送信する。親クラスタヘッドも同様にして、すべてのクラスタメンバ、子クラスタヘッドからのデータの受信を完了してから親ノードへデータを転送する。

このようにして、ネットワークの末端からすべてのノ

ードが収集したセンサデータをシンクに収集する。シンクがすべてのデータを収集し終わると、1 サイクルが完了する。これを一定サイクル繰り返し、次のラウンドへ移る。

### 3.4 ノードのスリープ

クラスタヘッドでないノードがシンクからデータ送信要求を受信した際、各ラウンドの最初のサイクルであった場合、受信ノードはこのラウンドをスリープするかの判断を行う。このとき、受信ノード  $n$  は以下の 2 つの条件を満たすとラウンドが変わるまでスリープに移行する。

- $n$  のセンシング範囲内の他ノードが  $k$  個以上存在 ( $k > 0$ )
- $n$  のセンシング範囲内に存在するすべての他ノードと比較して、 $n$  の残存電力が最小

## 4. 評価

評価は、Java 言語により LEACH, HEED, HIT, 提案方式のシミュレーションプログラムを作成し、シミュレーションにより行う。シミュレーションは、砂漠や汚染地帯など人が立ち入ることが困難な観測領域にセンサを空中から散布する場合を想定し行う。また、シンクは観測領域の外にあるものとする。図 3 に想定する環境の概略図を示す。

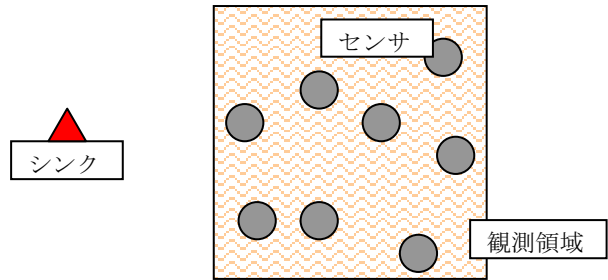


図 3 シミュレーション想定環境

今回のシミュレーションでは、MAC 層以下は詳細に作り込んでおらず、衝突によるパケットロス、再送は考慮していない。Qualnet 等のシミュレーションによる評価は今後実装していく。

### 4.1 シミュレーション概要

シミュレーションの想定環境として、観測領域を 100m \* 100m とし、観測領域の北西端から南に 50m、西に 75m の位置にシンクを配置する。ノードは観測領域内のランダムな座標に配置する。

また、送受信における電力の消費モデルは[1]と同じものを用いた。 $k$  bits を  $d$  m 離れたノードへ送信するときの消費電力  $E_T$  は式(4.1)で、 $k$  bits を受信するときの消費電力  $E_R$  は式(4.2)で与えられる。

$$E_T = E_{elec}k + \epsilon_{amp}kd^2 \quad (4.1)$$

$$E_R = E_{elec}k \quad (4.2)$$

ここで、 $E_{elec}$  は 1 bit を送受信する際に消費する電力、 $\epsilon_{amp}$  は送信時にかかる電力を示す。

また、このシミュレーションでは、電波の干渉、衝突は考慮していない。

評価は、各アルゴリズムで、すべてのノードが電力を使い果たすまでシミュレーションを行い、サイクルごとのノード数、及び領域カバー率の推移を比較して行う。

領域カバー率については、以下のように定義する。

各センサは半径 10m の範囲をカバー可能であるとし、また観測領域には縦横 1m 間隔で格子点があるとす。この点がセンサによってどの程度カバーされているかの比率を領域カバー率と定義する。

#### 4.2 シミュレーションパラメータ

シミュレーションに用いる共通パラメータを以下の表 1 に示す。

表 1 共通シミュレーションパラメータ

パラメータ	値
$E_{elec}$	50 nJ / bit
$\epsilon_{amp}$	100 pJ / bit / m <sup>2</sup>
初期電力	4 J
制御メッセージ	500 bits
データサイズ	2000 bits

LEACH のシミュレーションパラメータとして、1 ラウンドあたりのサイクル数とノードがクラスタヘッドに立候補する基本確率  $P$  を定める必要がある。ノード数 50, 100, 200 それぞれの場合について各パラメータを変化させてシミュレーションを行い、最適値を求めた。表 2 に LEACH のシミュレーションパラメータを示す。

表 2 LEACH のシミュレーションパラメータ

パラメータ	ノード 50	ノード 100	ノード 200
1 ラウンドのサイクル数	100	100	100
基本確率 $P$	0.1	0.075	0.05

HEED のシミュレーションパラメータとしては、1 ラウンドあたりのサイクル数、ノードがクラスタヘッドに立候補する基準確率  $C_{prob}$  を、及び  $CH_{prob}$  の最小値  $P_{min}$  をそれぞれ定める必要がある。LEACH 同様のシミュレーションを行い、最適値を求めた。HEED のシミュレーションパラメータを表 3 に示す。

表 3 HEED のシミュレーションパラメータ

パラメータ	ノード 50	ノード 100	ノード 200
1 ラウンドのサイクル数	100	100	50
基本確率 $C_{prob}$	0.025	0.025	0.025
$CH_{prob}$ の最小値 $P_{min}$	0.0025	0.0025	0.0025

HIT のシミュレーションパラメータとして、1 ラウンドあたりのサイクル数を定める必要がある。LEACH, HEED

同様のシミュレーションの結果に従い定めた HIT のシミュレーションパラメータを示す。

表 4 HIT のシミュレーションパラメータ

パラメータ	ノード 50	ノード 100	ノード 200
1 ラウンドのサイクル数	700	1000	1000

最後に提案方式のシミュレーションパラメータとしては、1 ラウンドあたりのサイクル数、2 ホップカバー率のしきい値、そして残存電力の重み  $W$  定める必要がある。従来手法同様にシミュレーションを行った結果、の結果を最適値としてシミュレーションパラメータに採用した。

表 5 提案手法のシミュレーションパラメータ

パラメータ	ノード 50	ノード 100	ノード 200
1 ラウンドのサイクル数	1000	1000	1000
2 ホップカバー率のしきい値	0.5	0.5	0.5
電力の重み $W$	3	3	2

また、提案手法において電波の到達範囲を最小 (25m) と最大 (領域全体) のみに固定した場合と、20m, 30m, 40m, 領域全体の 4 段階の電波到達範囲を設けた場合との比較を行った。シミュレーションの結果を図 4, 図 5 に示す。

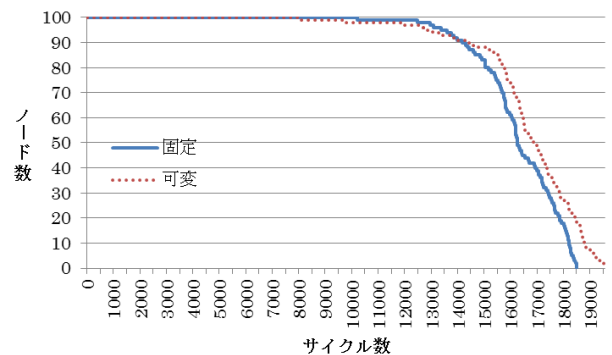


図 4 サイクル数ごとのノード数の推移

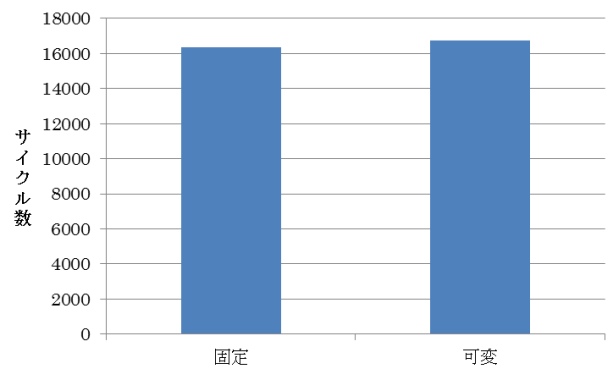


図 5 ノードの平均生存時間

シミュレーションの結果を見ると、電波到達範囲を固定し

た場合よりも、可変の場合のほうが、パフォーマンスが向上していることがわかる。よって、提案手法では、電波到達範囲は 20m, 30m, 40m, 領域全体の 4 段階を使用した。

### 4.3 評価・考察

上記のシミュレーションパラメータを用いて、提案方式と LEACH, HEED, HIT との比較を行い、提案方式の性能評価を行った。

ノード数が 50, 100, 200 のとき、すべてのノードが電力を使い果たすまでの、サイクル数ごとのノード数、及びカバー率の推移でそれぞれ比較する。ノード数 50, 100, 200 での、サイクル数ごとのノード数の推移の結果をそれぞれ図 6, 図 7, 図 8 に示す。

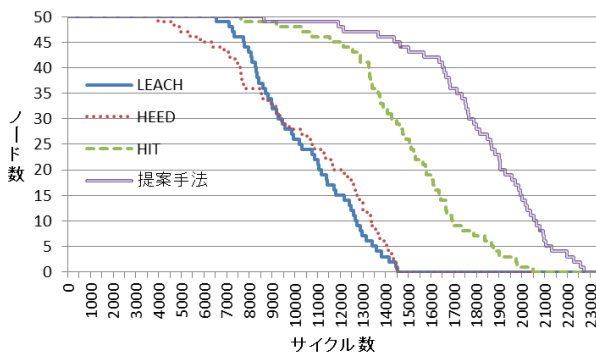


図 6 ノード数の推移 (ノード数 50)

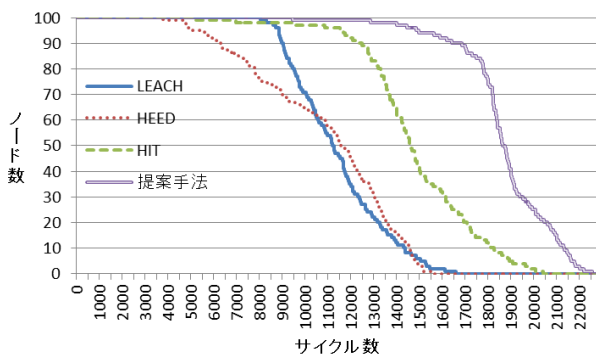


図 7 ノード数の推移 (ノード数 100)

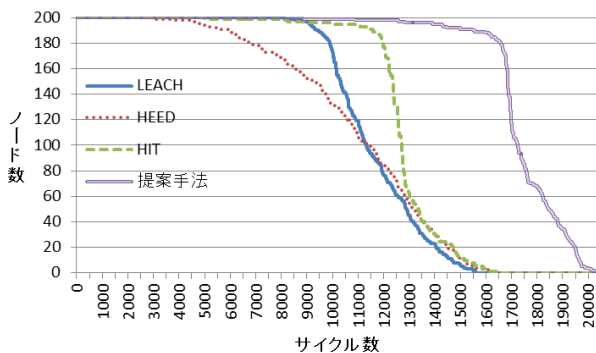


図 8 ノード数の推移 (ノード数 200)

これらの結果を見ると、提案手法はいずれのノード数においても、従来手法よりもパフォーマンスが向上していることがわかる。特に、ノード数 200 の場合に、提案手法で

はノード数の減少が急激であることがわかる。このことから、ノードの電力をバランスよく消費することができたと考えられる。また、それぞれのノード数における各方式のノードの平均生存時間の比較を図 9 に示す。

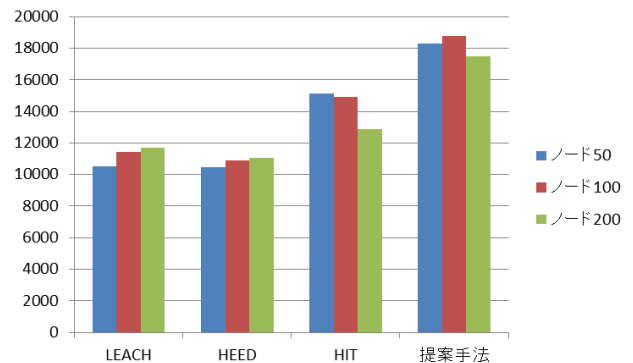


図 9 ノードの平均生存時間

この結果を見ると、提案手法はいずれのノード数においても、ノードの平均生存時間が最大になっていることがわかる。一方で、LEACH, HEED がノード数 200 の場合に最もパフォーマンスが良くなっているのに対して、HIT 及び提案方式ではノード数 200 の場合が最も悪くなっている。これは、HIT、及び提案手法では多数のノードが全体に対してブロードキャストするため、ノード数が増えると全体の受信電力のコストも増えるためと考えられる。また、提案手法ではノード数 100 のときに最もノードの平均生存時間が長くなっている。これは、提案手法のスリープ方法では、ノード密度が一定以上になると、スリープに移行できるノードの割合が減ってしまうためだと考えられる。

次に、ノード数 50, 100, 200 のときの、領域カバー率の推移の結果をそれぞれ図 10, 図 11, 図 12 に示す。

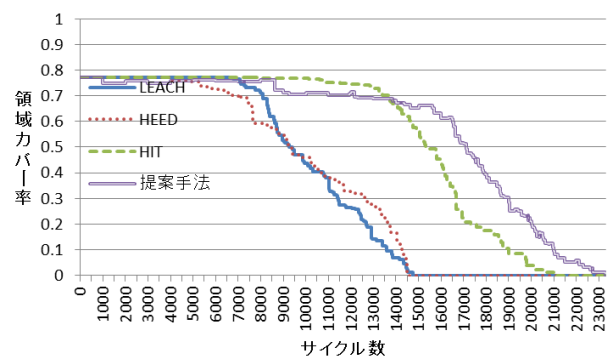


図 10 領域カバー率の推移 (ノード数 50)



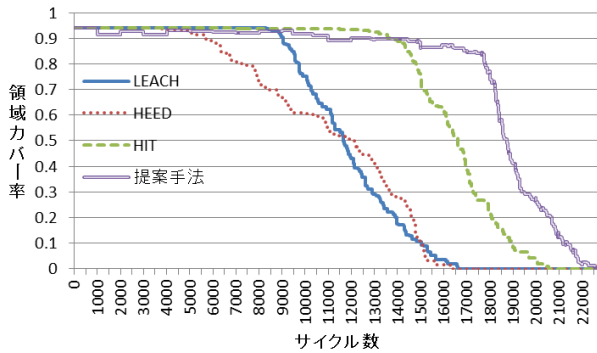


図 11 領域カバー率の推移 (ノード数 100)

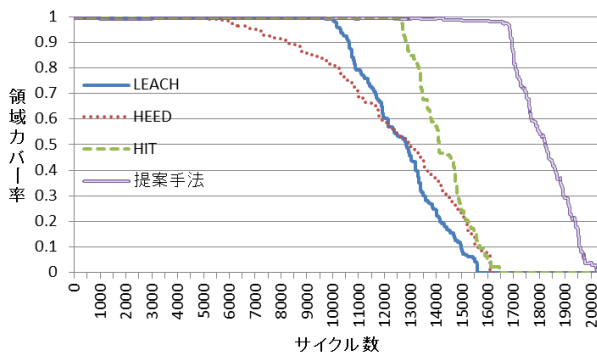


図 12 領域カバー率の推移 (ノード数 200)

提案手法では、スリープを導入しているため、生存ノードが多い状態では従来手法よりもカバー率が劣ってしまうことがわかった。特にこの傾向はノード数 50 の場合に強く見られた。一方で、ノード数 200 では生存ノード数が多い状態でもカバー率は従来手法とほぼ変わらない水準を維持することができた。これは、前述のとおりノード密度が一定以上になるとスリープに移行できるノードの割合が減少するためであると考えられる。このことから、提案手法は領域カバー率に関しては、ノード密度が高いほど有効であるといえる。

最後に、図 13 に各方式、各ノード数におけるノードの消費電力の制御メッセージが占める平均割合の比較を示す。

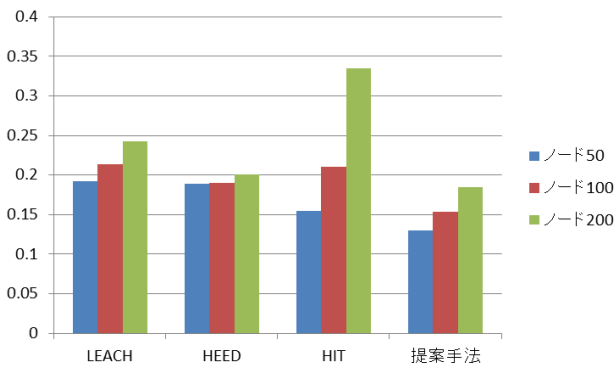


図 13 消費電力における制御メッセージの割合

提案手法は、従来手法と比較して、経路を構築するために必要な制御メッセージの数が多いが、それぞれ最良のパラメータを用いて比較すると、制御メッセージのオーバーヘ

ッドは従来手法と比較して最も小さくなっていることがわかる。

これらの結果から、提案手法は無線センサネットワークの省電力化において有効であるといえる。

## 5. おわりに

本稿では、センサネットワークにおける隣接ノード集合と残存電力に着目し、それらを用いた省電力クラスタリング方式を提案した。そして、提案方式に基づいたシミュレーションプログラムを作成し、従来方式との性能比較を行った。その結果、提案手法を用いたことにより、LEACH, HEED, HIT よりも電力をバランスよく利用することができることがわかった。ノードの平均生存時間は従来手法と比較してもっとも長くなることがわかった。

最後に今後の課題を述べる。

まず、今回のシミュレーションでは MAC 層以下は詳細に作り込んでおらず、電波の干渉等を想定していない。このため、Qualnet 等のシミュレータを用いて、より現実に近い環境での評価が必要である。

また、提案手法のスリープの判定方法では、ノードの密度が低いとノードの平均生存時間は長くなるが、領域カバー率は従来手法に大きく劣ってしまい、逆にノードの密度が高いと、領域カバー率は高い水準を維持できるが、ノードの平均生存時間が短くなってしまうことがわかった。このため、ノードの密度に応じてスリープさせるノードの数を変化させられるスリープ方式を導入する必要があると考える。今後は、上記のような課題の解決に向けて取り組んでいきたい。

**謝辞** 本研究の一部は日本学術振興会科学研究費基盤研究(c)(22500071)の助成を受けたものである。

## 参考文献

- 1) W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan: "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", Proceedings of the 33rd Hawaii International Conference on System Sciences, pp. 1-10, (2000).
- 2) O. Younis and S. Fahmy: "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks", IEEE Transaction on Mobile Computing, vol.3, No.4, pp. 366-379, (2004).
- 3) B. J. Culpepper, L. Dung and M. Moh: "Design and Analysis of Hybrid Indirect Transmissions (HIT) for Data Gathering in Wireless Micro Sensor Networks", ACM Mobile Computing and Communications Review, vol.3, pp.61-83, (2004).
- 4) S. Lindsey, C. Raghavenda and K. M. Sivalingam: "Data Gathering Algorithms in Sensor Networks Using Energy Metrics", IEEE Transactions on Parallel and Distributed Systems, vol.13, No.9, pp.924-935, (2002).
- 5) W. Choi, P. Shah and S. K. Das: "A framework for energy-saving data gathering using two-phase clustering in wireless sensor networks", Proceedings of the First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, pp.203-212, (2004).
- 6) T. Winter, Ed., P. Thubert, Ed., A. Brandt, J. Hui, R. Kelsey, P.

Levis, K. Pister, R. Struik, JP. Vasseur, R. Alexander, “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks(RPL)”, RFC 6552, (2012)

7) 牛島準一, 沖野正宗, 加藤聰彦, 伊藤秀一: “高密度アドホックネットワークにおける中継用ランドマークノードの選出方法の提案と評価”, 電子情報通信学会技術研究報告, pp. 93-96, (2003).