

バッテリー残量と接続可能ノード数をもとにしたクラスターヘッドの決定によるノード稼働時間の均一化

筒井 優貴¹ 大沢 恭平² 串田 高幸¹

概要：農場では温度や湿度の情報を取得するために IoT デバイスが利用されている。IoT デバイスは電源に接続されて稼働しているデバイスとバッテリーに接続されて稼働しているデバイスが存在する。マルチホップネットワークでは IoT デバイスがデータの中継を行う。データの中継を行うデバイスはデータの中継を行わないデバイスと比較してバッテリーの消費量が多い。課題は、中継を行うデバイスのバッテリーが枯渇すると中継を必要とするデバイスが孤立する点である。本稿では、バッテリー残量と接続可能ノード数をもとにクラスターヘッドを決定する手法を提案する。評価実験では提案方式と LEACH のノードの稼働時間を測定した。提案方式で w_b を 0.8, w_s を 0.2 に設定することで、最も稼働時間が短いノードの稼働時間は 6585[s]、最も稼働時間が長いノードの稼働時間は 7235[s] となり、その差は 650[s] となった。各ノードの稼働時間の平均は 6902[s] であった。一方、LEACH は最も稼働時間が短いノードの稼働時間は 6275[s]、最も稼働時間が長いノードの稼働時間は 7290[s] となり、その差は 1015[s] となった。各ノードの稼働時間の平均は、6842[s] であった。よって、提案方式は LEACH と比較して各ノードの平均稼働時間がこれにより、提案手法を使用することで、LEACH と比較して、平均稼働時間が約 1[%] 延長した。提案方式の最も稼働時間が長いノードと最も稼働時間が短いノードの稼働時間の差は、LEACH の約 64[%] になった。

1. はじめに

背景

Internet of Things(IoT) はヘルスケア、スマートシティ、工業、輸送で利用されている [1, 2]。農業では、土壌の状態や大気の状態を知るために IoT デバイスが利用されている [3]。農場のデータを収集する際には、マルチホップネットワークが使用されている [4]。マルチホップネットワークとはセンサーを搭載した小型で軽量の無線ノードの分散ネットワークである [5]。マルチホップネットワークのノードは、温度や湿度の収集、処理、他のノードとの通信を行うことができる [6, 7]。マルチホップネットワークのノードは、処理能力、メモリ、エネルギー資源が制限されている [8]。センサーのエネルギーが枯渇すると、センサーは動作することができなくなる [9]。センサーノードは、配置される環境によっては、バッテリーの交換や充電は困難となる [10, 11]。そのため、センサーノードは長期間稼働することが求められる [12]。

マルチホップネットワークでノードをクラスタリングすると、スケラビリティの向上、エネルギー効率の向上、ルーティング遅延の低減ができる [13]。各クラスターでは、クラスターヘッドと呼ばれるノードが、クラスター内のクラスターメンバーと呼ばれるノードからデータの収集と送信を行う [14, 15]。クラスターヘッドは、クラスターメンバーからデータを収集し、転送する役割があるため、クラスターメンバーと比較して多くの電力を消費する [16]。

課題

課題は、クラスターヘッドのバッテリー残量が枯渇することによって、孤立ノードが発生することである。図 1 に課題の概要を示す。クラスターメンバーとクラスターヘッドは温度データを取得している。クラスターメンバーはクラスターヘッドを介してサーバーにデータを送信している。クラスターヘッドは、クラスターメンバーと比較してバッテリーを多く消費する。そのため、クラスターメンバーよりも先にクラスターヘッドのバッテリーが枯渇する。クラスターヘッドのバッテリーが枯渇すると、クラスターメンバーはバッテリーが残っていてもデータをサーバーに送信することが不可能になる。

¹ 東京工科大学コンピュータサイエンス学部
〒 192-0982 東京都八王子市片倉町 1404-1

² 東京工科大学大学院バイオ・情報メディア研究科コンピュータサイエンス専攻
〒 192-0982 東京都八王子市片倉町 1404-1

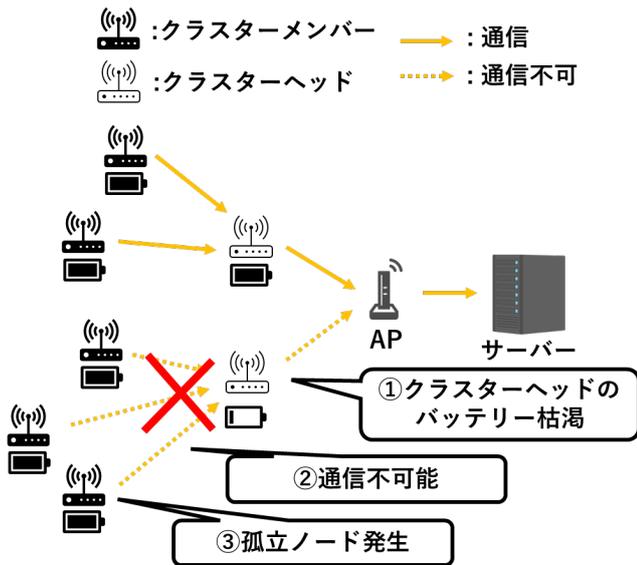


図 1 クラスタヘッドのバッテリーが枯渇した状態

各章の概要

第 2 章では関連研究について記述する。第 3 章では提案方式、ユースケース・シナリオについて記述する。第 4 章では実装について記述する。第 5 章では評価実験について記述する。第 6 章では議論について記述する。第 7 章では本稿のまとめを記述する。

2. 関連研究

消費電力の均一化を目的としてクラスタヘッドをランダムに選択する LEACH (Low Energy Adaptive Clustering Hierarchy) というアルゴリズムが存在する [17]。しかし、LEACH はクラスタヘッドがランダムに選択されるため、クラスタヘッドの選出に偏りが発生する場合がある。クラスタヘッドの選出に偏りが発生すると、ノードの消費電力を完全に均一にすることはできない。その結果、一部のノードに負荷が集中し、バッテリーが早期に枯渇する [18]。

HEED はノードの残エネルギーと通信距離をもとにクラスタヘッドを決定するプロトコルである。このプロトコルは、シングルホップネットワークを前提としているため、マルチホップネットワークで利用することができない [19]。

エネルギー消費効率予測をもとにクラスタリングとルーティングを行うアルゴリズムが存在する [20]。このプロトコルは、ノード間の距離やノードのバッテリー残量をもとにクラスタリングを行い、通信時の消費電力をもとにルーティングを行う。しかし、このプロトコルは実験環境がシミュレーションであり、現実での実験が行われていない。

3. 提案

本稿では、ノードのバッテリー残量を均一化し、ノードの稼働時間を延長することを目的とする。そのために、バツ

テリー残量と接続可能ノード数をもとにクラスタヘッドを決定する。提案方式の処理の流れを示したシーケンス図を図 2 に示す。

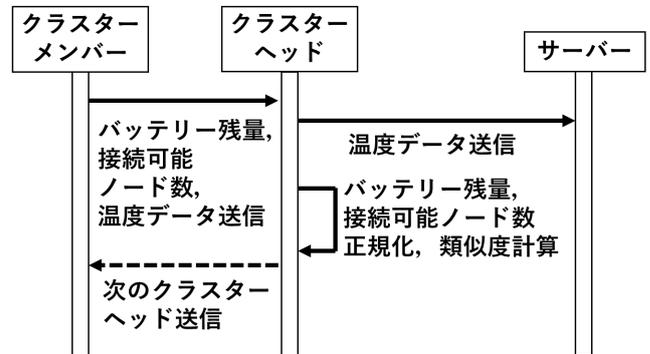


図 2 提案シーケンス図

提案方式を以下の 4 段階に分けて説明する。

- バッテリー残量, ノード数取得
- 正規化
- 各ノードの類似度計算
- クラスタヘッドの決定

3.1 バッテリー残量, 接続可能ノード数取得

クラスタヘッド候補ノードは、バッテリー残量と接続可能ノード数をクラスタヘッドに対して送信する。図 3 にバッテリー残量と接続可能ノード数を取得する様子を示す。

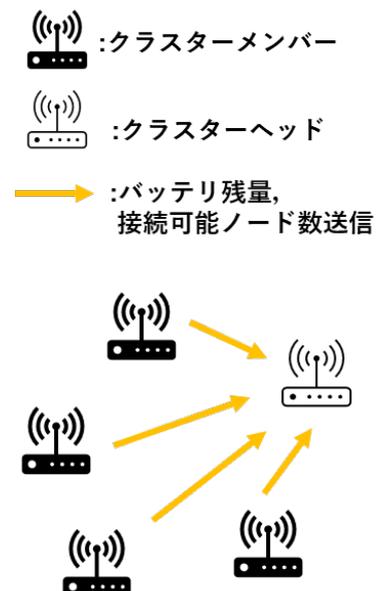


図 3 バッテリー残量, 接続可能ノード数を取得

3.2 正規化

各ノードから取得したバッテリー残量をクラスターヘッドは (1) の式で正規化する。

$$b_{normalized} = \frac{b_{collect} - b_{min}}{b_{max} - b_{min}} \quad (1)$$

$b_{collect}$ は各ノードから取得したバッテリー残量, b_{min} は全ノードのバッテリー残量の最小値, b_{max} は全ノードのバッテリー残量の最大値である。

各ノードから取得した接続可能ノード数をクラスターヘッドは (2) の式で正規化する。

$$n_{normalized} = \frac{n_{collect} - n_{min}}{n_{max} - n_{min}} \quad (2)$$

$n_{collect}$ は各ノードから取得した接続可能ノード数, n_{min} は全ノードの接続可能ノード数の最小値, n_{max} は全ノードの接続可能ノード数の最大値である。

3.3 各ノードの類似度計算

各ノードのバッテリー残量と接続可能ノード数とクラスターヘッドのバッテリー残量と接続可能ノード数の類似度を計算する。計算は式 (3) で行われる。

$$score = \sqrt{(w_b * (b_i - b_j))^2 + (w_n * (n_i - n_j))^2} \quad (3)$$

ここで, b_i は式 (1) で正規化された現在のクラスターヘッドのバッテリー残量, b_j は式 (1) で正規化されたクラスターヘッド候補ノードのバッテリー残量, n_i は式 (2) で正規化されたクラスターヘッドの接続可能ノード数, n_j は式 (2) で正規化されたクラスターヘッド候補ノードの接続可能ノード数である。 w_b , はバッテリー残量の重み, w_n は接続可能ノード数の重みである。

3.4 クラスターヘッドの決定

式 (3) でスコアが最も低いノードが次のクラスターヘッドとなる。次のクラスターヘッドが決定したら全てのクラスターメンバーに対して現在のクラスターヘッドが次のクラスターヘッドを通知する。図 4 に次のクラスターヘッドを通知する様子を示す。

ユースケース・シナリオ

本提案は、農場の状態の取得に使用されることを想定している。農場では、温度や湿度を取得するために、IoT デバイスが利用されている。図 5 にユースケースを示す。

図 5 は農場でデータを取得し、マルチホップでサーバーにデータを送信している。本稿の提案方式を使用することで、ノードのバッテリー消費量が均一化され、孤立ノードの発生を抑えることができる。

4. 実装

IoT デバイスとして ESP32 を使用する。MicroPython

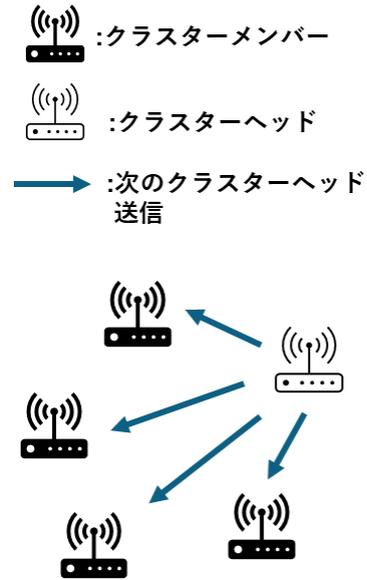


図 4 次のクラスターヘッドを通知

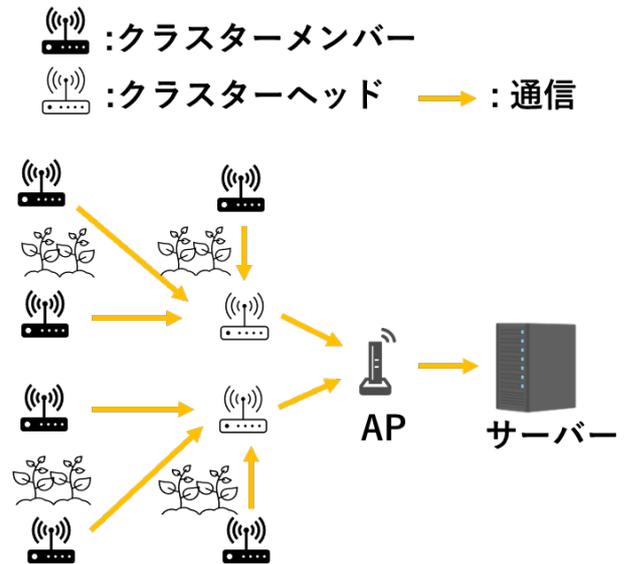


図 5 ユースケース

で実装する。電流計として INA219 を使用する。通信方式にはソケット通信を使用する。図 6 はソフトウェア概要図である。クラスターメンバーはバッテリー残量と接続可能台数をクラスターヘッドに送信する。クラスターヘッドは送信されたバッテリー残量と接続可能ノード数をもとに次のクラスターヘッドを決定する。次のクラスターヘッドが決定されたら現在のクラスターヘッドはクラスターメンバーに対して次のクラスターヘッドを通知する。

5. 評価実験

各ノードのバッテリー残量が枯渇するまでの時間を計測する。式 (3) の w_b , w_n の値を 0.1 から 0.9 の範囲で 0.1 ずつ変動させ、各重みでのバッテリー残量が枯渇するまでの時間を計測した。また、LEACH でクラスターヘッドを動

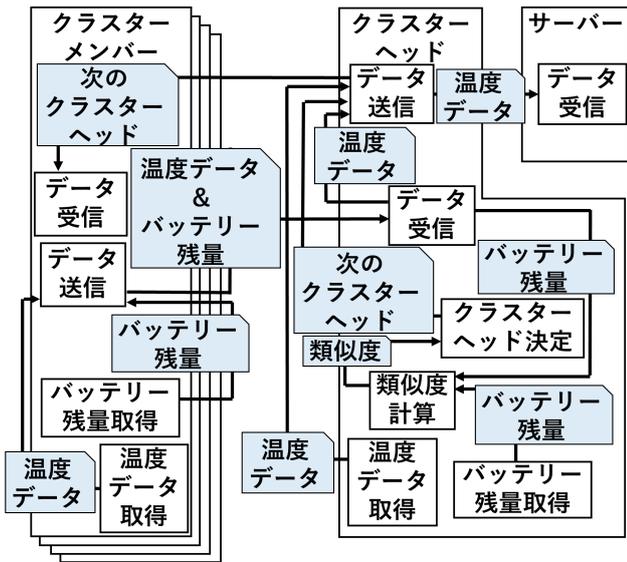


図 6 ソフトウェア概要図

的に変更した場合のバッテリー残量が枯渇するまでの時間の計測も行った。

実験環境

IoT デバイスとして 5 台の ESP32, 電流計として INA219 を使用して実験を行う。バッテリー容量は 200[mAh] である。提案ソフトウェアと LEACH の実装は MicroPython で行った。

実験結果と分析

図 7 は w_b を 0.1, w_n を 0.9 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである。最も稼働時間が短かったのは ID5 のノードで 4965[s], 最も稼働時間が長かったのは ID2 のノードで 7715[s] であった。全ノードの稼働時間の平均は 6990[s] であった。

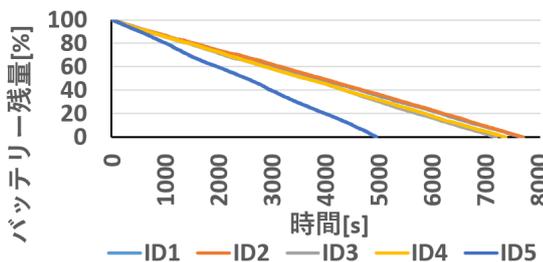


図 7 $w_b=0.1, w_n=0.9$ のノード稼働時間

図 8 は w_b を 0.2, w_n を 0.8 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである。最も稼働時間が短かったのは ID5 のノードで 5740[s], 最も稼働時間が長かったのは ID2 のノードで 7245[s] であった。全ノードの稼働時間の平均は 6682[s] であった。

図 9 は w_b を 0.3, w_n を 0.7 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである。最も

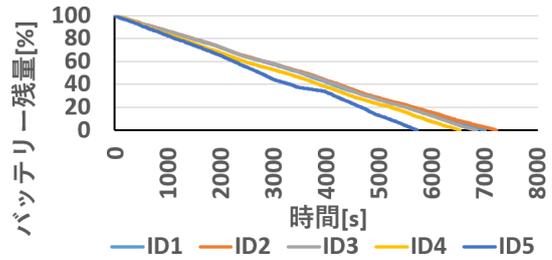


図 8 $w_b=0.2, w_n=0.8$ のノード稼働時間

稼働時間が短かったのは ID5 のノードで 6230[s], 最も稼働時間が長かったのは ID2 のノードで 7050[s] であった。全ノードの稼働時間の平均は 6692[s] であった。

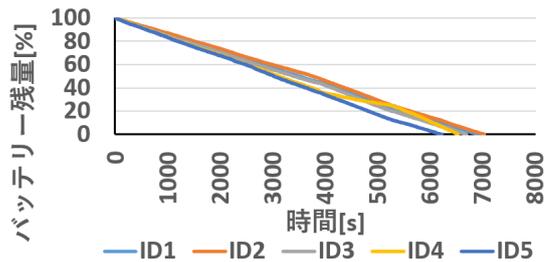


図 9 $w_b=0.3, w_n=0.7$ のノード稼働時間

図 10 は w_b を 0.4, w_n を 0.6 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである。最も稼働時間が短かったのは ID5 のノードで 5910[s], 最も稼働時間が長かったのは ID1 のノードで 7300[s] であった。全ノードの稼働時間の平均は 6747[s] であった。

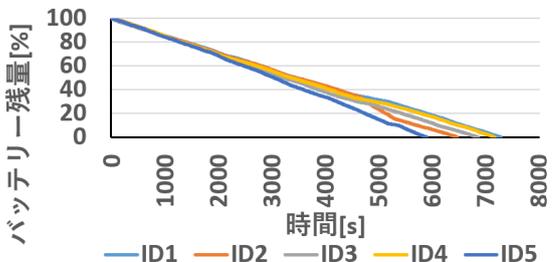


図 10 $w_b=0.4, w_n=0.6$ のノード稼働時間

図 11 は w_b を 0.5, w_n を 0.5 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである。最も稼働時間が短かったのは ID1 のノードで 6085[s], 最も稼働時間が長かったのは ID5 のノードで 8025[s] であった。全ノードの稼働時間の平均は 7231[s] であった。

図 12 は w_b を 0.6, w_n を 0.4 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである。最も稼働時間が短かったのは ID4 のノードで 6190[s], 最も稼働時間が長かったのは ID1 のノードで 7705[s] であった。全ノードの稼働時間の平均は 6918[s] であった。

図 13 は w_b を 0.7, w_n を 0.3 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである。最も

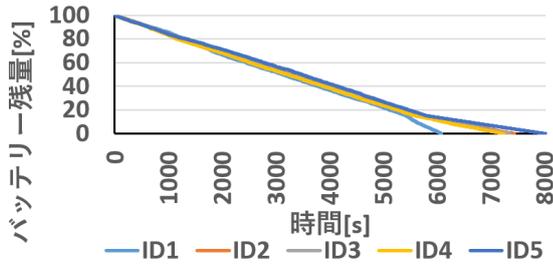


図 11 $w_b=0.5, w_n=0.5$ のノード稼働時間

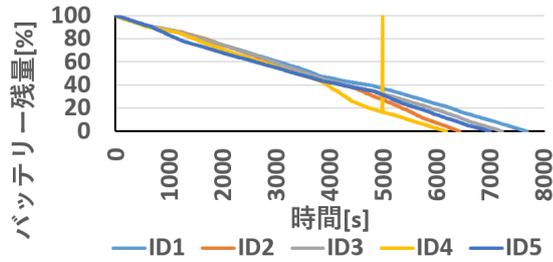


図 12 $w_b=0.6, w_n=0.4$ のノード稼働時間

稼働時間が短かったのは ID5 のノードで 6370[s], 最も稼働時間が長かったのは ID2 のノードで 7170[s] であった. 全ノードの稼働時間の平均は 6794[s] であった.

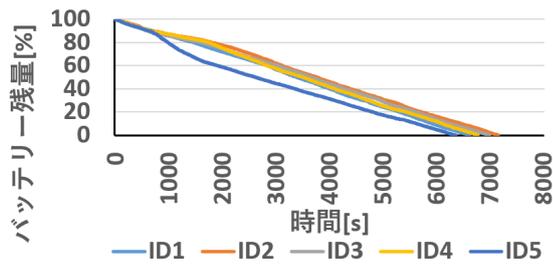


図 13 $w_b=0.7, w_n=0.3$ のノード稼働時間

図 14 は w_b を 0.8, w_n を 0.2 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである. 最も稼働時間が短かったのは ID5 のノードで 6585[s], 最も稼働時間が長かったのは ID2 のノードで 7235[s] であった. 全ノードの稼働時間の平均は 6902[s] であった.

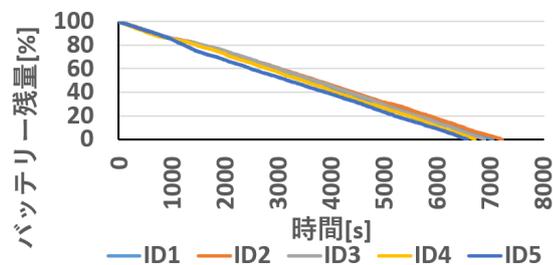


図 14 $w_b=0.8, w_n=0.2$ のノード稼働時間

図 15 は w_b を 0.9, w_n を 0.1 に設定した場合の各ノードの稼働時間とバッテリー残量を示したグラフである. 最も稼働時間が短かったのは ID3 のノードで 6390[s], 最も稼働時間が長かったのは ID1 のノードで 7295[s] であった.

全ノードの稼働時間の平均は 6749[s] であった.

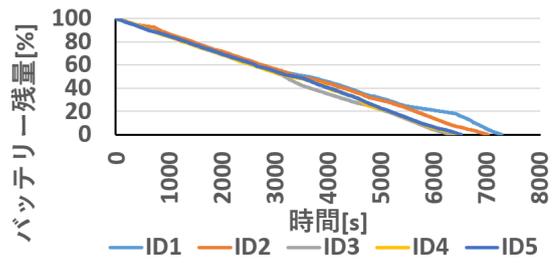


図 15 $w_b=0.9, w_n=0.1$ のノード稼働時間

図 16 は LEACH で動的にクラスターヘッドを選択した場合の各ノードの稼働時間とバッテリー残量を示したグラフである. 最も稼働時間が短かったのは ID4 のノードで 6275[s], 最も稼働時間が長かったのは ID2 のノードで 7290[s] であった. 全ノードの稼働時間の平均は 6842[s] であった.

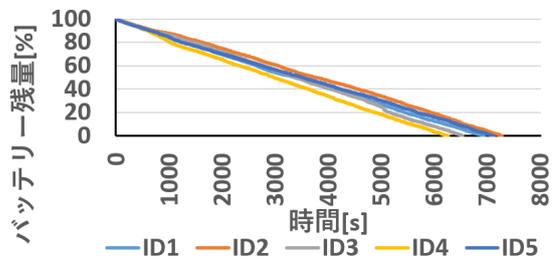


図 16 LEACH

表 1 は各条件でのノード稼働時間の平均値, 最大値, 最小値をまとめたものである. 表の赤いセルは各条件のうち最も値が低かったものであり, 表の水色のセルは最も値が高かったものである. 平均値が最も大きくなったのは w_b, w_n をそれぞれ 0.5 に設定した場合で 7231[s] であった. 平均値が最も小さくなったのは w_b を 0.2, w_n を 0.8 に設定した場合で 6682[s] であった. 最大値が最も大きくなったのは w_b, w_n をそれぞれ 0.5 に設定した場合で 8025[s] であった. 最大値が最も小さくなったのは w_b を 0.3, w_n を 0.7 に設定した場合で 7050[s] であった. 最小値が最も大きくなったのは w_b を 0.8, w_n を 0.2 に設定した場合で 6585[s] であった. 最小値が最も小さくなったのは w_b を 0.1, w_n を 0.9 に設定した場合で 4965[s] であった. 最大値と最小値の差が最も大きくなったのは w_b を 0.1, w_n を 0.9 に設定した場合で 2750[s] であった. 最大値と最小値の差が最も小さくなったのは w_b を 0.8, w_n を 0.2 に設定した場合で 650[s] であった. これらの結果から, w_b の値が大きいと各ノードの稼働時間の差が小さくなり, w_n の値が大きいと各ノードの稼働時間の差が大きくなることがわかった. また, 一つのノードのバッテリーを集中的に使用したほうが, 全ノードの稼働時間の平均は長くなることが分かった. 本稿の目的は, 各ノードのバッテリー消費量を均一化するこ

とであるため、 w_b を 0.8、 w_n を 0.2 に設定することが最も適している。

表 1 各条件下でのノード稼働時間の平均, 最大, 最小

条件	平均 [s]	最大 [s]	最小 [s]
$w_b=0.1, w_n=0.9$	6990	7715	4965
$w_b=0.2, w_n=0.8$	6682	7245	5740
$w_b=0.3, w_n=0.7$	6692	7050	6230
$w_b=0.4, w_n=0.6$	6747	7300	5910
$w_b=0.5, w_n=0.5$	7231	8025	6085
$w_b=0.6, w_n=0.4$	6918	7705	6190
$w_b=0.7, w_n=0.3$	6794	7170	6370
$w_b=0.8, w_n=0.2$	6902	7235	6585
$w_b=0.9, w_n=0.1$	6749	7295	6390
LEACH	6842	7290	6275

6. 議論

本稿では、バッテリー残量と接続可能ノード数をもとにクラスターヘッドを動的に変更することで、各ノードのバッテリー消費を均一化する手法を提案した。しかし、条件によっては、バッテリー消費量を均一化することが適切であるとは限らない。例えば、クラスターメンバーが 1 台のクラスターヘッドとクラスターメンバーが 10 台のクラスターヘッドが存在した場合、クラスターメンバーが 10 台存在するクラスターヘッドのほうが重要である。そのため、クラスターメンバーが少ないノードのバッテリーを優先的に使用することで、クラスターメンバーが多いノードのバッテリーをより長く使用できるようにすることで、より多くのデータをより長い時間取得することができるようになる。

7. おわりに

本稿では、バッテリー残量と接続可能ノード数をもとにクラスターヘッドを動的に変更する手法を提案した。課題は、クラスターヘッドのバッテリー残量が枯渇することによって孤立ノードが発生することである。評価実験では提案方式と LEACH のノードの稼働時間を測定した。提案方式で w_b を 0.8、 w_n を 0.2 に設定することで、最も稼働時間が短いノードの稼働時間は 6585[s]、最も稼働時間が長いノードの稼働時間は 7235[s] となり、その差は 650[s] となった。各ノードの稼働時間の平均は 6902[s] であった。一方、LEACH は最も稼働時間が短いノードの稼働時間は 6275[s]、最も稼働時間が長いノードの稼働時間は 7290[s] となり、その差は 1015[s] となった。各ノードの稼働時間の平均は、6842[s] であった。よって、提案方式は LEACH と比較して各ノードの平均稼働時間がこれにより、提案手法を使用することで、LEACH と比較して、平均稼働時間が約 1[%] 延長した。提案方式の最も稼働時間が長いノード

と最も稼働時間が短いノードの稼働時間の差は、LEACH の約 64[%] になった。

参考文献

- [1] Haras, M. and Skotnicki, T.: Thermoelectricity for IoT—A review, *Nano Energy*, Vol. 54, pp. 461–476 (2018).
- [2] Gokhale, P., Bhat, O. and Bhat, S.: Introduction to IOT, *International Advanced Research Journal in Science, Engineering and Technology*, Vol. 5, No. 1, pp. 41–44 (2018).
- [3] Muangprathub, J., Boonnarn, N., Kajornkasirat, S., Lekbangpong, N., Wanichsombat, A. and Nillaor, P.: IoT and agriculture data analysis for smart farm, *Computers and electronics in agriculture*, Vol. 156, pp. 467–474 (2019).
- [4] Martinelli, M., Ioriatti, L., Viani, F., Benedetti, M. and Massa, A.: A WSN-based solution for precision farm purposes, *2009 IEEE International Geoscience and Remote Sensing Symposium*, Vol. 5, IEEE, pp. V–469 (2009).
- [5] Anisi, M. H., Abdul-Salaam, G. and Abdullah, A. H.: A survey of wireless sensor network approaches and their energy consumption for monitoring farm fields in precision agriculture, *Precision Agriculture*, Vol. 16, pp. 216–238 (2015).
- [6] Zhu, H., Li, S., Zheng, L. and Yang, L.: Modeling and validation on path loss of WSN in pig breeding farm, *Transactions of the Chinese Society of Agricultural Engineering*, Vol. 33, No. 2, pp. 205–212 (2017).
- [7] Sharma, S., Bansal, R. K. and Bansal, S.: Issues and challenges in wireless sensor networks, *2013 international conference on machine intelligence and research advancement*, IEEE, pp. 58–62 (2013).
- [8] Wang, Y., Attebury, G. and Ramamurthy, B.: A survey of security issues in wireless sensor networks (2006).
- [9] Seah, W. K., Eu, Z. A. and Tan, H.-P.: Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)-Survey and challenges, *2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology*, Ieee, pp. 1–5 (2009).
- [10] Rezaei, Z. and Mobinejad, S.: Energy saving in wireless sensor networks, *International Journal of Computer Science and Engineering Survey*, Vol. 3, No. 1, p. 23 (2012).
- [11] Chan, L., Gomez Chavez, K., Rudolph, H. and Hourani, A.: Hierarchical routing protocols for wireless sensor network: A compressive survey, *Wireless Networks*, Vol. 26, pp. 3291–3314 (2020).
- [12] Kausar, A. Z., Reza, A. W., Saleh, M. U. and Ramiah, H.: Energizing wireless sensor networks by energy harvesting systems: Scopes, challenges and approaches, *Renewable and Sustainable Energy Reviews*, Vol. 38, pp. 973–989 (2014).
- [13] Afsar, M. M. and Tayarani-N, M.-H.: Clustering in sensor networks: A literature survey, *Journal of Network and Computer applications*, Vol. 46, pp. 198–226 (2014).
- [14] Wohwe Sambo, D., Yenke, B. O., Förster, A. and Dayang, P.: Optimized clustering algorithms for large wireless sensor networks: A review, *Sensors*, Vol. 19, No. 2, p. 322 (2019).
- [15] Sasikumar, P. and Khara, S.: K-means clustering in wireless sensor networks, *2012 Fourth international conference on computational intelligence and communica-*

- tion networks*, IEEE, pp. 140–144 (2012).
- [16] Izadi, D., Abawajy, J. and Ghanavati, S.: An alternative clustering scheme in WSN, *IEEE sensors journal*, Vol. 15, No. 7, pp. 4148–4155 (2015).
 - [17] Omari, M. and Laroui, S.: Simulation, comparison and analysis of wireless sensor networks protocols: LEACH, LEACH-C, LEACH-1R, and HEED, *2015 4th International Conference on Electrical Engineering (ICEE)*, IEEE, pp. 1–5 (2015).
 - [18] Song, L., Song, Q., Ye, J. and Chen, Y.: A hierarchical topology control algorithm for WSN, considering node residual energy and lightening cluster head burden based on affinity propagation, *Sensors*, Vol. 19, No. 13, p. 2925 (2019).
 - [19] Younis, O. and Fahmy, S.: HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks, *IEEE Transactions on mobile computing*, Vol. 3, No. 4, pp. 366–379 (2004).
 - [20] Zhang, D.-g., Wang, X., Song, X.-d., Zhang, T. and Zhu, Y.-n.: A new clustering routing method based on PECE for WSN, *EURASIP Journal on Wireless Communications and Networking*, Vol. 2015, pp. 1–13 (2015).